[Missing <summary> documentation for N:HttpServer]
<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConsoleLogWriter</td>
<td>This class writes to the console. It colors the output depending on the logprio and includes a 3-level stacktrace (in debug mode)</td>
</tr>
<tr>
<td>ContentType</td>
<td>Lists content type mime types.</td>
</tr>
<tr>
<td>ContentTypes</td>
<td>A list of content types</td>
</tr>
<tr>
<td>HttpClientContext</td>
<td>Contains a connection to a browser/client.</td>
</tr>
<tr>
<td>HttpFile</td>
<td>Container class for posted files</td>
</tr>
<tr>
<td>HttpForm</td>
<td>Container for posted form data</td>
</tr>
<tr>
<td>HttpHelper</td>
<td>Generic helper functions for Http</td>
</tr>
<tr>
<td>HttpInput</td>
<td>Contains some kind of input from the browser/client. can be QueryString, form data or any other request body content.</td>
</tr>
<tr>
<td>HttpInputItem</td>
<td>Represents a http input item. Each item can have multiple sub items, a sub item is made in a html form by using square brackets</td>
</tr>
<tr>
<td>HttpListener</td>
<td>HTTP Listener waits for HTTP connections and provide us with HttpListenerContexts using the RequestHandler delegate.</td>
</tr>
<tr>
<td>HttpParam</td>
<td>Returns item either from a form or a query string (checks them in that order)</td>
</tr>
<tr>
<td>HttpRequest</td>
<td>Contains serverside http request information.</td>
</tr>
<tr>
<td>HttpRequestParser</td>
<td>Parses a HTTP request directly from a stream Response that is sent back to the web browser / client. A response can be sent if different ways. The easiest one is to just fill the Body stream with content, everything else will then be taken care of by the framework. The default content-type is text/html, you should change it if you send anything else. The second and slightly more complex way is to send the response as parts. Start with sending the header using the SendHeaders method and then you</td>
</tr>
</tbody>
</table>
can send the body using `SendBody` method, but do not forget to set `ContentType` and `ContentLength` before doing so.

<table>
<thead>
<tr>
<th><strong>HttpServer</strong></th>
<th>A complete http server, you need to add a module to it to be able to handle incoming requests.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>Contains all HTTP Methods (according to the HTTP 1.1 specification)</td>
</tr>
<tr>
<td><strong>Methods</strong></td>
<td>See: <a href="http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html">http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html</a></td>
</tr>
<tr>
<td><strong>Obsolete.</strong></td>
<td>Contains all HTTP Methods (according to the HTTP 1.1 specification)</td>
</tr>
<tr>
<td><strong>RequestCookie</strong></td>
<td>cookie sent by the client/browser</td>
</tr>
<tr>
<td><strong>RequestCookies</strong></td>
<td>This class is created as a wrapper, since there are two different cookie types in .Net (Cookie and HttpCookie). The framework might switch class in the future and we dont want to have to replace all instances</td>
</tr>
<tr>
<td><strong>ResponseCookie</strong></td>
<td>cookie being sent back to the browser.</td>
</tr>
<tr>
<td><strong>ResponseCookies</strong></td>
<td>Cookies that should be set.</td>
</tr>
</tbody>
</table>
## Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpSession</td>
<td>Interface for sessions</td>
</tr>
<tr>
<td>HttpInputBase</td>
<td>Base class for request data containers</td>
</tr>
</tbody>
</table>
# Delegates

<table>
<thead>
<tr>
<th>Delegate</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClientDisconnectedHandler</td>
<td>Client have been disconnected. We don't want to let the server die due to exceptions thrown in worker threads. Therefore, we use this delegate to give you a change to handle uncaught exceptions.</td>
</tr>
<tr>
<td>ExceptionHandler</td>
<td>Invoked when a request has been completed.</td>
</tr>
<tr>
<td>HttpRequestParser::RequestCompletedHandler</td>
<td>Delegate used to find a realm/domain.</td>
</tr>
<tr>
<td>RealmHandler</td>
<td>Invoked when a client context has received a new HTTP request.</td>
</tr>
<tr>
<td>RequestReceivedHandler</td>
<td>A delegate that helps us keep track of errors in the system.</td>
</tr>
<tr>
<td>WriteLogHandler</td>
<td></td>
</tr>
</tbody>
</table>
## Enumerations

<table>
<thead>
<tr>
<th>Enumeration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectionType</td>
<td>Type of HTTP connection</td>
</tr>
<tr>
<td>HttpRequestParser::State</td>
<td>Current state in the parsing.</td>
</tr>
<tr>
<td>LogPrio</td>
<td>Priority for log entries</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Client have been disconnected.

Namespace:  [HttpServer]
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public delegate void ClientDisconnectedHandler(
    HttpClientContext client,
    SocketError error
)
```

**Visual Basic (Declaration)**

```vbnet
Public Delegate Sub ClientDisconnectedHandler ( _
    client As HttpClientContext, _
    error As SocketError _
)
```

**Visual C++**

```cpp
public delegate void ClientDisconnectedHandler(
    HttpClientContext client,
    SocketError error
)
```

**Parameters**

**client**
- Type: `HttpServer::::HttpClientContext`
- Client that was disconnected.

**error**
- Type: `System.Net.Sockets::::SocketError`
- Reason
See Also

HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ConnectionType Enumeration

Type of HTTP connection

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public enum ConnectionType

Visual Basic (Declaration)

Public Enumeration ConnectionType

Visual C++

public enum class ConnectionType
<table>
<thead>
<tr>
<th>Member name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close</td>
<td>Connection is closed after each request-response</td>
</tr>
<tr>
<td>KeepAlive</td>
<td>Connection is kept alive for X seconds (unless another request have been made)</td>
</tr>
</tbody>
</table>
See Also

HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ConsoleLogWriter Class

This class writes to the console. It colors the output depending on the logprio and includes a 3-level stacktrace (in debug mode)

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static class ConsoleLogWriter

Visual Basic (Declaration)

Public NotInheritable Class ConsoleLogWriter

Visual C++

public ref class ConsoleLogWriter abstract sealed
Inheritance Hierarchy

System..::..Object
HttpServer..::..ConsoleLogWriter
See Also

ConsoleLogWriter Members
HttpServer Namespace
HttpServer::<::WriteLogHandler

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ConsoleLogWriter** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🍰 <strong>GetColor</strong></td>
<td>Get color for the specified logprio</td>
</tr>
<tr>
<td>🍰 <strong>Logwriter</strong></td>
<td>Logwriters the specified source.</td>
</tr>
</tbody>
</table>
See Also

ConsoleLogWriter Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ConsoleLogWriter** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SetColor</strong></td>
<td>Get color for the specified logprio</td>
</tr>
<tr>
<td><strong>Logwriter</strong></td>
<td>Logwriters the specified source.</td>
</tr>
</tbody>
</table>
See Also

consoleLogWriter Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get color for the specified logprio

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static ConsoleColor GetColor(
   LogPrio prio
)

Visual Basic (Declaration)

Public Shared Function GetColor ( _
   prio As LogPrio _
) As ConsoleColor

Visual C++

public:
static ConsoleColor GetColor(
   LogPrio prio
)

Parameters

prio
   Type: HttpServer::LogPrio
   prio for the log entry

Return Value

A ConsoleColor for the prio
See Also

ConsoleLogWriter Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Logwriters the specified source.

**Namespace:**  [HttpServer](http://HttpServer)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static void Logwriter(
    Object source,
    LogPrio prio,
    string message
)

Visual Basic (Declaration)

Public Shared Sub Logwriter ( _
    source As Object, _
    prio As LogPrio, _
    message As String _
)

Visual C++

public:
    static void Logwriter(
        Object^ source, 
        LogPrio prio, 
        String^ message
    )

Parameters

source
    Type: System::Object
    object that wrote the logentry.

prio
    Type: HttpServer::LogPrio
    Priority of the log entry

message
    Type: System::String
The message.
See Also

ConsoleLogWriter Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Lists content type mime types.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ContentType

Visual Basic (Declaration)

Public Class ContentType

Visual C++

public ref class ContentType
Inheritance Hierarchy

System..::..Object
HttpServer..::..ContentType
See Also

ContentType Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ContentType` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ContentType</code></td>
<td>Initializes a new instance of the <code>ContentType</code> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>Html</td>
<td>text/haml</td>
</tr>
<tr>
<td>Text</td>
<td>text/plain</td>
</tr>
<tr>
<td>Xml</td>
<td>text/xml</td>
</tr>
</tbody>
</table>
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `ContentType` class

**Namespace:** [HttpServer](http://HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
C#

public ContentType()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
ContentType()
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ContentType** type exposes the following members.
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Htm</td>
<td>text/haml</td>
</tr>
<tr>
<td>Text</td>
<td>text/plain</td>
</tr>
<tr>
<td>Xml</td>
<td>text/xml</td>
</tr>
</tbody>
</table>
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ContentType...

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Html

Visual Basic (Declaration)

Public Const Html As String

Visual C++

public:
    literal String^ Html
See Also

ContentType Class
HttpResponse Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
text/plain

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Text

Visual Basic (Declaration)

Public Const Text As String

Visual C++

public:
    literal String^ Text
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ContentType....Xml Field

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public const string Xml
```

**Visual Basic (Declaration)**

```vbnet
Public Const Xml As String
```

**Visual C++**

```cpp
public:
    literal String^ Xml
```
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ContentType** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A list of content types

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ContentTypes : IEnumerable

Visual Basic (Declaration)

Public Class ContentTypes _
    Implements IEnumerable

Visual C++

public ref class ContentTypes : IEnumerable
Inheritance Hierarchy

System..::.Object
HttpServer..::.ContentTypes
See Also

ContentType Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ContentTypes` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ContentTypes</code></td>
<td>Initializes a new instance of the <code>ContentTypes</code> class</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contains</td>
<td>Searches for the specified type</td>
</tr>
<tr>
<td></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Equals</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>Get this first content type.</td>
</tr>
<tr>
<td>Item</td>
<td>Fetch a content type</td>
</tr>
</tbody>
</table>
See Also

**ContentTypes Class**

**HttpServer Namespace**

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
Initializes a new instance of the `ContentTypes` class

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public ContentTypes(
    string types
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New (   
    types As String _
)
```

Visual C++

```cpp
public:
    ContentTypes(
        String^ types
    )
```

Parameters

types

  Type: System::String

Semicolon separated content types.
See Also

SupportedContent Types Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ContentTypes` type exposes the following members.
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Contains</strong></td>
<td>Searches for the specified type</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and</td>
</tr>
<tr>
<td></td>
<td>perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Searches for the specified type

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Contains(
    string type
)

Visual Basic (Declaration)

Public Function Contains ( _
    type As String _
) As Boolean

Visual C++

public:
bool Contains(
    String type
)

Parameters

type
    Type: System::String
    Can also be a part of a type (searching for "xml" would return true for "application/xml").

Return Value

true if type was found.
See Also

ContentTypes Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Returns an enumerator that iterates through a collection.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IEnumerator GetEnumerator()

Visual Basic (Declaration)

Public Function GetEnumerator As IEnumerator

Visual C++

public:
virtual IEnumerator GetEnumerator() sealed

Return Value

An IEnumerator object that can be used to iterate through the collection.

Implements

IEnumerable::GetEnumerator()
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ContentTypes type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>Get this first content type.</td>
</tr>
<tr>
<td>Item</td>
<td>Fetch a content type</td>
</tr>
</tbody>
</table>
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get this first content type.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string First { get; }

Visual Basic (Declaration)

Public ReadOnly Property First As String

Visual C++

public:
property String^ First {
String^ get ();
}
See Also

- ContentTypes Class
- HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Fetch a content type

**Namespace:** [HttpServer](http://example.com/HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string this[
    string type
] { get; }

Visual Basic (Declaration)

Public ReadOnly Default Property Item ( _
    type As String _
) As String

Visual C++

public:
    property String^ default[String^ type] {
        String^ get (String^ type);
    }

Parameters

type

    Type: System:::String
    Part of type ("xml" would return "application/xml")

Return Value
See Also

ContentType Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
We don't want to let the server to die due to exceptions thrown in worker threads. Therefore we use this delegate to give you a change to handle uncaught exceptions.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public delegate void ExceptionHandler(
    Object source,
    Exception exception
)
```

Visual Basic (Declaration)

```vbnet
Public Delegate Sub ExceptionHandler (_
    source As Object, _
    exception As Exception _
)
```

Visual C++

```cpp
public delegate void ExceptionHandler(
    Object^ source, 
    Exception^ exception 
)
```

Parameters

source
Type: System::Object
Class that the exception was thrown in.

exception
Type: System::Exception
Exception
Remarks

Server will throw a InternalServerException in release version if you don't handle this delegate.
See Also

HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains a connection to a browser/client.

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public class HttpClientContext
```

**Visual Basic (Declaration)**

```vbnet
Public Class HttpClientContext
```

**Visual C++**

```cpp
public ref class HttpClientContext
```
Inheritance Hierarchy

System::Object
HttpServer::<HttpClientContext
See Also

HttpClientContext Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpClientContext** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpClientContext</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Respond</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Send</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>Write information to the log.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td><code>BufferSize</code></td>
<td>Buffersize determines how large the HTTP header can be.</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CurrentRequest</td>
<td>Current http request</td>
</tr>
<tr>
<td>Secured</td>
<td>Using SSL or other encryption method.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# □ Visual Basic □ Visual C++ □ Include Protected Members □ Include Inherited Members
C# Webserver
HttpClientContext Constructor
HttpClientContext Class See Also Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpClientContext(Boolean, Stream, RequestReceivedHandler)</code></td>
<td>Initializes a new instance of the <code>HttpClientContext</code> class.</td>
</tr>
<tr>
<td><code>HttpClientContext(Boolean, RequestReceivedHandler, ClientDisconnectedHandler, Stream, WriteLogHandler)</code></td>
<td>Initializes a new instance of the <code>HttpClientContext</code> class.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpClientContext Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

HttpClientContext Constructor (Boolean, Stream, RequestReceivedHandler)

HttpClientContext Class  See Also  Send Feedback

Initializes a new instance of the HttpClientContext class.

Namespace:  HttpServer  
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpClientContext(
    bool secured,
    Stream stream,
    RequestReceivedHandler requestHandler
)

Visual Basic (Declaration)

Public Sub New ( _
    secured As Boolean, _
    stream As Stream, _
    requestHandler As RequestReceivedHandler _
)

Visual C++

public:
HttpClientContext(
    bool secured,
    Stream^ stream,
    RequestReceivedHandler^ requestHandler
)

Parameters

secured
Type: System::Boolean
true if the connection is secured (SSL/TLS)

stream
Type: System.IO::Stream
Stream used for communication

requestHandler
Type: HttpServer::RequestReceivedHandler
delegate handling incoming requests.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.Net.Sockets.SocketException</code></td>
<td>If <code>beginreceive</code> fails</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpClientContext Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext Constructor (Boolean, RequestReceivedHandler, ClientDisconnectedHandler, Stream, WriteLogHandler)

HttpClientContext Class  See Also  Send Feedback

Initializes a new instance of the HttpClientContext class.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpClientContext(
    bool secured,
    RequestReceivedHandler requestHandler,
    ClientDisconnectedHandler disconnectHandler,
    Stream stream,
    WriteLogHandler writer
)

Visual Basic (Declaration)

Public Sub New ( _
    secured As Boolean, _
    requestHandler As RequestReceivedHandler, _
    disconnectHandler As ClientDisconnectedHandler, _
    stream As Stream, _
    writer As WriteLogHandler _
)

Visual C++

public:
HttpClientContext(
    bool secured,
    RequestReceivedHandler requestHandler,
    ClientDisconnectedHandler disconnectHandler,
    Stream stream,
    WriteLogHandler writer
)

Parameters

secured
    Type: System::::Boolean
true if the connection is secured (SSL/TLS)

requestHandler
Type: `HttpServer::RequestReceivedHandler`
delegate handling incoming requests.

disconnectHandler
Type: `HttpServer::ClientDisconnectedHandler`
delegate being called when a client disconnects

stream
Type: `System.IO::Stream`  
Stream used for communication

writer
Type: `HttpServer::WriteLogHandler`
delegate used to write log entries
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.Net.Sockets.SocketException</td>
<td>If beginreceive fails</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpClientContext Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpClientContext type exposes the following members.
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BufferSize</td>
<td>Buffersize determines how large the HTTP header can be.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Buffersize determines how large the HTTP header can be.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public static int BufferSize

Visual Basic (Declaration)
Public Shared BufferSize As Integer

Visual C++
public:
static int BufferSize
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpClientContext` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a> is equal to the current <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>. (Inherited from <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html">Type</a> of the current instance. (Inherited from <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>. (Inherited from <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>.)</td>
</tr>
<tr>
<td><strong>Respond</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Send</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/String.html">String</a> that represents the current <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>. (Inherited from <a href="https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html">Object</a>.)</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>Write information to the log.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

HttpClientContext...:..:Respond Method

HttpClientContext Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Respond(String)</td>
<td>Send a response.</td>
</tr>
<tr>
<td>Respond(String, HttpStatusCode, String)</td>
<td>Send a response.</td>
</tr>
<tr>
<td>Respond(String, HttpStatusCode, String, String)</td>
<td>Send a response.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpClientContext Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext...:::Respond Method (String)

HttpClientContext Class  See Also  Send Feedback

Send a response.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void Respond(
    string body
)
```

Visual Basic (Declaration)

```vbnet
Public Sub Respond ( _
    body As String _
)
```

Visual C++

```cpp
public:
void Respond(
    String^ body
)
```

Parameters

body

Type: `System::String`

# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentNullException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
Respond Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext...::Respond Method (String, HttpStatusCode, String)

Send a response.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public void Respond(
    string httpVersion,
    HttpStatusCode statusCode,
    string reason
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub Respond (
    httpVersion As String,
    statusCode As HttpStatusCode,
    reason As String
)
```

### Visual C++

```cpp
public:
    void Respond(
        String^ httpVersion,
        HttpStatusCode statusCode,
        String^ reason
    )
```

## Parameters

**httpVersion**
- Type: `System::String`
- Either HttpHelper.HTTP10 or HttpHelper.HTTP11

**statusCode**
- Type: `System.Net::HttpStatusCode`
- http status code

**reason**
- Type: `System::String`
reason for the status code.
See Also

HttpClientContext Class
Respond Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext...::Respond Method (String, HttpStatusCode, String, String)

Send a response.

Namespace: HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Respond(
    string httpVersion,
    HttpStatusCode statusCode,
    string reason,
    string body
)

Visual Basic (Declaration)

Public Sub Respond (_
    httpVersion As String, _
    statusCode As HttpStatusCode, _
    reason As String, _
    body As String _
)

Visual C++

public:
void Respond(
    String^ httpVersion,
    HttpStatusCode statusCode,
    String^ reason,
    String^ body
)

Parameters

httpVersion
Type: System::String
Either HttpHelper.HTTP10 or HttpHelper.HTTP11

statusCode
Type: System.Net::HttpStatusCode
http status code
reason
   Type: System::String
   reason for the status code.

body
   Type: System::String
   html body contents, can be null or empty.
# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentException</code></td>
<td>If <code>httpVersion</code> is invalid.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
Respond Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Send(array[Byte][][])</td>
<td>send a whole buffer</td>
</tr>
<tr>
<td>Send(array[Byte][][], Int32, Int32)</td>
<td>Send data using the stream</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpClientContext Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext...:::Send Method (array<Byte>[][])  

**HttpClientContext Class**  **See Also**  **Send Feedback**  

send a whole buffer

**Namespace:**  **HttpServer**  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Send(
    byte[] buffer
)

Visual Basic (Declaration)

Public Sub Send (_
    buffer As Byte() _
)

Visual C++

public:
void Send(
    array<unsigned char>^ buffer
)

Parameters

buffer
    Type: array< System::Byte >[]()[
    buffer to send
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System::ArgumentNullException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
Send Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext...:::Send Method (array<Byte>[][], Int32, Int32)

HttpClientContext Class  See Also  Send Feedback

Send data using the stream

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Send(
    byte[] buffer,
    int offset,
    int size
)

Visual Basic (Declaration)

Public Sub Send (~
    buffer As Byte(), ~
    offset As Integer, ~
    size As Integer ~
)

Visual C++

public:
void Send(~
    array<unsigned char>^ buffer,
    int offset,
    int size
)

Parameters

buffer
  Type: array< System::::Byte >[]
  Contains data to send

offset
  Type: System::::Int32
  Start position in buffer

size
  Type: System::::Int32
number of bytes to send
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System..::.ArgumentNullException</td>
<td></td>
</tr>
<tr>
<td>System..::.ArgumentOutOfRangeException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
Send Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Write information to the log.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void WriteLog(
    Object source,
    LogPrio prio,
    string message
)
```

Visual Basic (Declaration)

```vbnet
Public Sub WriteLog ( _
    source As Object, _
    prio As LogPrio, _
    message As String _
)
```

Visual C++

```cpp
public:
void WriteLog(
    Object^ source,
    LogPrio prio,
    String^ message
)
```

Parameters

source
Type: System::Object
object writing the message

prio
Type: HttpServer::LogPrio
importance of the log message

message
Type: System::String
the actual message
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpClientContext** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon] CurrentRequest</td>
<td>Current http request</td>
</tr>
<tr>
<td>![Icon] Secured</td>
<td>Using SSL or other encryption method.</td>
</tr>
</tbody>
</table>
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Current http request

**Namespace:** [HttpServer](https://example.com/HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpRequest CurrentRequest { get; }

Visual Basic (Declaration)

Public ReadOnly Property CurrentRequest As HttpRequest

Visual C++

public:
property HttpRequest^ CurrentRequest {
    HttpRequest^ get ();
}


See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpClientContext...:.Secured Property
HttpClientContext Class  See Also  Send Feedback

Using SSL or other encryption method.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public bool Secured { get; }
```

### Visual Basic (Declaration)

```
Public ReadOnly Property Secured As Boolean
```

### Visual C++

```cpp
public:
property bool Secured {
    bool get();
}
```
See Also

HttpClientContext Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpFile Class

Container class for posted files

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public class HttpFile : IDisposable
```

**Visual Basic (Declaration)**

```vbnet
Public Class HttpFile
    Implements IDisposable
```

**Visual C++**

```cpp
public ref class HttpFile : IDisposable
```
Inheritance Hierarchy

System..:::Object
HttpServer..:::HttpFile
See Also

HttpFile Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpFile** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpFile</td>
<td>Creates a container for a posted file</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose</td>
<td>Overloaded. Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td>Equals</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Destructor disposing the file (Overrides Object..::.Finalize().)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>

*(Inherited from Object.)*
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ContentType</strong></td>
<td>The type of file</td>
</tr>
<tr>
<td><strong>Filename</strong></td>
<td>The full file path</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>The name/id of the file</td>
</tr>
</tbody>
</table>
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpFile Constructor

HttpFile Class  See Also  Send Feedback

Creates a container for a posted file

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public HttpFile(
    string name,
    string filename,
    string contentType
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New (_
    name As String, _
    filename As String, _
    contentType As String _
)
```

### Visual C++

```cpp
public:
    HttpFile(
        String^ name,
        String^ filename,
        String^ contentType
    )
```

### Parameters

**name**
- **Type:** `System::::String`
- The identifier of the post field

**filename**
- **Type:** `System::::String`
- The file path

**contentType**
- **Type:** `System::::String`
The content type of the file
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentNullException</code></td>
<td>If any parameter is null or empty</td>
</tr>
</tbody>
</table>
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpFile` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose</td>
<td>Overloaded.</td>
</tr>
<tr>
<td></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>Equals</td>
<td>Destroys the file.</td>
</tr>
<tr>
<td></td>
<td>(Overrides <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.::Finalize())</td>
</tr>
<tr>
<td>Finalize</td>
<td>Destroys the file.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>

---
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpFile... Dispose Method

HttpFile Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose()()</td>
<td>Disposing interface, cleans up managed resources (the temporary file) and suppresses finalization</td>
</tr>
<tr>
<td>Dispose(Boolean)</td>
<td>Deletes the temporary file</td>
</tr>
</tbody>
</table>
See Also

HttpFile Class
HttpFile Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Disposing interface, cleans up managed resources (the temporary file) and suppresses finalization

**Namespace:**  [HttpServer](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Dispose()

Visual Basic (Declaration)

Public Sub Dispose

Visual C++

public:
virtual void Dispose() sealed

Implements

IDisposable:::Dispose()()
See Also

HttpFile Class
Dispose Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpFile.....Dispose Method (Boolean)

HttpFile Class  See Also  Send Feedback

Deletes the temporary file

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void Dispose(
    bool disposing
)

Visual Basic (Declaration)

Protected Sub Dispose ( _
    disposing As Boolean _
)

Visual C++

protected:
void Dispose(
    bool disposing
)

Parameters

disposing
    Type: System::Boolean
    True if manual dispose
See Also

HttpFile Class
Dispose Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpFile destructor disposing the file

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
protected override void Finalize()

Visual Basic (Declaration)
Protected Overrides Sub Finalize

Visual C++
protected:
virtual void Finalize() override
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpFile` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContentType</td>
<td>The type of file</td>
</tr>
<tr>
<td>Filename</td>
<td>The full file path</td>
</tr>
<tr>
<td>Name</td>
<td>The name/id of the file</td>
</tr>
</tbody>
</table>
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The type of file

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public string ContentType { get; set; }
```

**Visual Basic (Declaration)**

```vbnet
Public Property ContentType As String
```

**Visual C++**

```csharp
public:
property String^ ContentType {
    String^ get ();
    void set (String^ value);
}
```
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The full file path

**Namespace:**  [HttpServer](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Filename { get; set; }

Visual Basic (Declaration)

Public Property Filename As String

Visual C++

public:
property String^ Filename {
    String^ get ();
    void set (String^ value);
}
See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpFile Name Property

The name/id of the file

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Name { get; set; }

Visual Basic (Declaration)

Public Property Name As String

Visual C++

public:
property String^ Name {
    String^ get ();
    void set (String^ value);
}

See Also

HttpFile Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Container for posted form data

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpForm : HttpInput

Visual Basic (Declaration)

Public Class HttpForm
    Inherits HttpInput

Visual C++

public ref class HttpForm : public HttpInput
Inheritance Hierarchy

System...:::Object
   HttpServer...:::HttpInput
      HttpServer...:::HttpForm
See Also

HttpForm Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpForm** type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="#">HttpForm</a></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Add a new element. Form array elements are parsed and added in a correct hierarchy. (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td><strong>AddFile</strong></td>
<td>Adds a file to the collection of posted files.</td>
</tr>
<tr>
<td><strong>Clear</strong></td>
<td>Disposes all held HttpFile's and resets values (Overrides <a href="#">HttpInput..::.Clear()</a>.)</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Returns true if the class contains a HttpInput value with the corresponding name (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td><strong>ContainsFile</strong></td>
<td>Checks if the form contains a specified file</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Returns an enumerator that iterates through a collection. (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td><strong>GetFile</strong></td>
<td>Retrieves a file held by the form</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Outputs the instance representing all its values joined together (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
</tbody>
</table>
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ignoreChanges</td>
<td>Variable telling the class that it is non-initialized</td>
</tr>
<tr>
<td>Empty</td>
<td>(Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td>EmptyForm</td>
<td>Instance to help mark a non-initialized form</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Get a form item. (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td>Name</td>
<td>Form name as lower case. (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
</tbody>
</table>
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver

HttpForm Constructor

HttpContext, See Also, Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpForm(HttpInput)</td>
<td>Makes a deep copy of the input</td>
</tr>
<tr>
<td>HttpForm(String)</td>
<td>Initializes a form container with the specified</td>
</tr>
<tr>
<td></td>
<td>name</td>
</tr>
</tbody>
</table>
See Also

HttpForm Class
HttpForm Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpForm Constructor (HttpInput)

HttpForm Class  See Also  Send Feedback

Makes a deep copy of the input

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
# Syntax

**C#**

```csharp
public HttpForm(
    HttpInput input
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New ( _
    input As HttpInput _
)
```

**Visual C++**

```cpp
public:
HttpForm(
    HttpInput^ input
)
```

## Parameters

*input*

Type: `HttpServer...::HttpInput`

The input to copy
See Also

HttpForm Class
HttpForm Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpContext Constructor (String)

HttpForm Constructor (String)

HttpContext Class  See Also  Send Feedback

Initializes a form container with the specified name

Namespace:  HttpContext
Assembly:  HttpContext (in HttpContext.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpForm(
    string name
)

Visual Basic (Declaration)

Public Sub New (_
    name As String _
)

Visual C++

public:
HttpForm(
    String^ name
)

Parameters

name
  Type: System::String
  Name of the form
See Also

HttpForm Class
HttpForm Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpForm** type exposes the following members.
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ignoreChanges</td>
<td>Variable telling the class that it is non-initialized</td>
</tr>
<tr>
<td>Empty</td>
<td>(Inherited from <a href="#">HttpInput</a>)</td>
</tr>
<tr>
<td>EmptyForm</td>
<td>Instance to help mark a non-initialized form</td>
</tr>
</tbody>
</table>
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Instance to help mark a non-initialized form

**Namespace:** [HttpServer](#)  
**Assembly:** [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static readonly HttpForm EmptyForm

Visual Basic (Declaration)

Public Shared ReadOnly EmptyForm As HttpForm

Visual C++

public:
static initonly HttpForm^ EmptyForm
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpForm` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Add a new element. Form array elements are parsed and added in a correct hierarchy. (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td>AddFile</td>
<td>Adds a file to the collection of posted files</td>
</tr>
<tr>
<td>Clear</td>
<td>Disposes all held HttpFile's and resets values (Overrides <a href="#">HttpInput..::Clear()()</a>.)</td>
</tr>
<tr>
<td>Contains</td>
<td>Returns true if the class contains a HttpInput value with the corresponding name (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td>ContainsFile</td>
<td>Checks if the form contains a specified file</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetEnumerator</td>
<td>Returns an enumerator that iterates through a collection. (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
<tr>
<td>GetFile</td>
<td>Retrieves a file held by the form</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Outputs the instance representing all its values joined together (Inherited from <a href="#">HttpInput</a>.)</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpForm..:::AddFile Method

HttpForm Class  See Also  Send Feedback

Adds a file to the collection of posted files

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public void AddFile(
    HttpFile file
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub AddFile (_
    file As HttpFile _
)
```

### Visual C++

```cpp
public:
void AddFile(
    HttpFile^ file
)
```

## Parameters

file

Type: `HttpServer::::HttpFile`

The file to add
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::ArgumentException</code></td>
<td>If the file is already added</td>
</tr>
<tr>
<td><code>System:::ArgumentNullException</code></td>
<td>If file is null</td>
</tr>
<tr>
<td><code>System:::InvalidOperationException</code></td>
<td>If the instance is <code>HttpRequest.EmptyForm</code> which cannot be modified</td>
</tr>
</tbody>
</table>
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
C# Visual C++
C# Webserver

HttpForm...::.Clear Method

HttpForm Class  See Also  Send Feedback

Disposes all held HttpFile's and resets values

**Namespace:**  [HttpServer](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override void Clear()

Visual Basic (Declaration)

Public Overrides Sub Clear

Visual C++

public:
virtual void Clear() override
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ContainsFile Method

HttpForm Class  See Also  Send Feedback

Checks if the form contains a specified file

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public bool ContainsFile(
    string name
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function ContainsFile (_
    name As String _
) As Boolean
```

**Visual C++**

```cpp
public:
bool ContainsFile(
    String^ name
)
```

**Parameters**

**name**

Type: `System::String`
Field name of the file parameter

**Return Value**

True if the file exists
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.InvalidOperationException</code></td>
<td>If the instance is <code>HttpForm.EmptyForm</code> which cannot be modified</td>
</tr>
</tbody>
</table>
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver

HttpForm...::GetFile Method

HttpForm Class See Also Send Feedback

Retrieves a file held by the form

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpFile GetFile(
    string name
)
```

Visual Basic (Declaration)

```vbnet
Public Function GetFile (_
    name As String _
) As HttpFile
```

Visual C++

```cpp
public:
HttpFile^ GetFile(
    String^ name
)
```

Parameters

name

Type: System::String
The identifier of the file

Return Value

The requested file or null if the file was not found
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td>If name is null or empty</td>
</tr>
<tr>
<td>System:::InvalidOperationException</td>
<td>If the instance is HttpForm.EmptyForm which cannot be modified</td>
</tr>
</tbody>
</table>
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpForm** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| Item | Get a form item.  
(Inherited from [HttpInput](#).) |
| Name | Form name as lower case  
(Inherited from [HttpInput](#).) |
See Also

HttpForm Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generic helper functions for Http

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public static class HttpHelper
```

**Visual Basic (Declaration)**

```vbnet
Public NotInheritable Class HttpHelper
```

**Visual C++**

```cpp
public ref class HttpHelper abstract sealed
```
Inheritance Hierarchy

System:::Object
HttpServer:::HttpHelper
See Also

HttpHelper Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpHelper` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParseQueryString</td>
<td>Parses a querystring.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>✸ EmptyUri</td>
<td>An empty url</td>
</tr>
<tr>
<td>✸ HTTP10</td>
<td>Version string for HTTP v1.0</td>
</tr>
<tr>
<td>✸ HTTP11</td>
<td>Version string for HTTP v1.1</td>
</tr>
</tbody>
</table>
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpHelper** type exposes the following members.
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EmptyUri</td>
<td>An empty url</td>
</tr>
<tr>
<td>HTTP10</td>
<td>Version string for HTTP v1.0</td>
</tr>
<tr>
<td>HTTP11</td>
<td>Version string for HTTP v1.1</td>
</tr>
</tbody>
</table>
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver

HttpHelper...:.EmptyUri Field

HttpHelper Class  See Also  Send Feedback

An empty url

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public static readonly Uri EmptyUri
```

**Visual Basic (Declaration)**

```vbnet
Public Shared ReadOnly EmptyUri As Uri
```

**Visual C++**

```cpp
public:
    static initonly Uri^ EmptyUri
```
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Version string for HTTP v1.0

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string HTTP10

Visual Basic (Declaration)

Public Const HTTP10 As String

Visual C++

public:
literal String^ HTTP10
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpHelper...: HTTP11 Field

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)

Version string for HTTP v1.1
Syntax

C#

public const string HTTP11

Visual Basic (Declaration)

Public Const HTTP11 As String

Visual C++

public:
literal String^ HTTP11
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpHelper type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParseQueryString</td>
<td>Parses a querystring.</td>
</tr>
</tbody>
</table>
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpHelper..::.ParseQueryString Method

HttpHelper Class  See Also  Send Feedback

Parses a querystring.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static HttpInput ParseQueryString(
    string queryString
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function ParseQueryString (_
    queryString As String _
) As HttpInput
```

Visual C++

```cpp
public:
static HttpInput^ ParseQueryString(
    String^ queryString
)
```

Parameters

queryString

Type: System::String

Querystring (url decoded)

Return Value

A HttpInput object if successful; otherwise HttpInput.Empty
See Also

HttpHelper Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains some kind of input from the browser/client. can be QueryString, form data or any other request body content.

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpInput : HttpInputBase, IEnumerable<HttpInputItem>, IEnumerable

Visual Basic (Declaration)

Public Class HttpInput
    Implements HttpInputBase, IEnumerable(Of HttpInputItem), IEnumerable

Visual C++

public ref class HttpInput : HttpInputBase, IEnumerable<HttpInputItem^>, IEnumerable
Inheritance Hierarchy

System:::Object
HttpServer:::HttpInput
   HttpServer:::HttpForm
See Also

HttpInput Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpInput** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpInput</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Add a new element. Form array elements are parsed and added in a correct hierarchy.</td>
</tr>
<tr>
<td>Clear</td>
<td>Resets all data contained by class</td>
</tr>
<tr>
<td>Contains</td>
<td>Returns true if the class contains a HttpRequest value with the corresponding name</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ExtractOne</td>
<td>Extracts one parameter from an array</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetEnumerator</td>
<td>Returns an enumerator that iterates through a collection</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ParseItem</td>
<td>Parses an item and returns it. This function is primarily used to parse array items as in user[name]. Outputs the instance representing all its values joined together (Overrides Object::ToString()).</td>
</tr>
</tbody>
</table>
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>_ignoreChanges</code></td>
<td>Variable telling the class that it is non-initialized Empty</td>
</tr>
<tr>
<td><code>Empty</code></td>
<td>Representation of a non-initialized class instance</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Get a form item.</td>
</tr>
<tr>
<td>Name</td>
<td>Form name as lower case</td>
</tr>
</tbody>
</table>
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpInput(HttpInput)</td>
<td>Creates a deep copy of the HttpInput class</td>
</tr>
<tr>
<td>HttpInput(String)</td>
<td>Initializes a new instance of the HttpInput class.</td>
</tr>
<tr>
<td>HttpInput(String, Boolean)</td>
<td>Initializes a new instance of the HttpInput class.</td>
</tr>
</tbody>
</table>
See Also

HttpInput Class
HttpInput Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput Constructor (HttpInput)

HttpInput Class  See Also  Send Feedback

Creates a deep copy of the HttpInput class

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
protected HttpInput(httpInput input)
```

**Visual Basic (Declaration)**

```vbnet
Protected Sub New (_
    input As HttpInput _
)
```

**Visual C++**

```cpp
protected:
HttpInput(
    HttpInput^ input
)
```

**Parameters**

input

Type: HttpServer::HttpInput

The object to copy
Remarks

The function makes a deep copy of quite a lot which can be slow
See Also

HttpInput Class
HttpInput Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput Constructor (String)

HttpInput Class  See Also  Send Feedback

Initializes a new instance of the HttpInput class.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public HttpInput(
    string name
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    name As String _
)
```

**Visual C++**

```cpp
public:
HttpInput(
    String^ name
)
```

### Parameters

name

Type: `System::::String`

form name.
See Also

HttpInput Class
HttpInput Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput Constructor (String, Boolean)

HttpInput Class  See Also  Send Feedback

Initializes a new instance of the HttpInput class.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

protected IHttpInput(
    string name,
    bool ignoreChanges
)

**Visual Basic (Declaration)**

Protected Sub New (_
    name As String, _
    ignoreChanges As Boolean _
)

**Visual C++**

protected:
HttpInput(
    String^ name,
    bool ignoreChanges
)

**Parameters**

name
    Type: System:::String
    form name.

ignoreChanges
    Type: System:::Boolean
    if set to true all changes will be ignored.
Remarks

this constructor should only be used by Empty
See Also

HttpInput Class
HttpInput Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpInput Fields

HttpInput Class  See Also  Send Feedback

The HttpInput type exposes the following members.
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>_ignoreChanges</code></td>
<td>Variable telling the class that it is non-initialized Empty</td>
</tr>
<tr>
<td><code>Empty</code></td>
<td>Representation of a non-initialized class instance</td>
</tr>
</tbody>
</table>
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput...:_ignoreChanges Field

Variable telling the class that it is non-initialized Empty

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected readonly bool _ignoreChanges

Visual Basic (Declaration)

Protected ReadOnly _ignoreChanges As Boolean

Visual C++

protected:
initonly bool _ignoreChanges
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Representation of a non-initialized class instance

Namespace:  **HttpServer**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static readonly HttpInput Empty

Visual Basic (Declaration)

Public Shared ReadOnly Empty As HttpInput

Visual C++

public:
static initonly HttpInput^ Empty
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpInput` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Add a new element. Form array elements are parsed and added in a correct hierarchy.</td>
</tr>
<tr>
<td><strong>Clear</strong></td>
<td>Resets all data contained by class</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Returns true if the class contains a HttpInput value with the corresponding name</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>ExtractOne</strong></td>
<td>Extracts one parameter from an array</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>.</td>
</tr>
<tr>
<td><strong>ParseItem</strong></td>
<td>Parses an item and returns it. This function is primarily used to parse array items as in <code>user[name]</code>. Outputs the instance representing all its values joined together (Overides <strong>Object::::ToString()</strong>.)</td>
</tr>
</tbody>
</table>
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add a new element. Form array elements are parsed and added in a correct hierarchy.

**Namespace:** [HttpServer](http://example.com/HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public void Add(
    string name,
    string value
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Add (
    name As String,
    value As String
)
```

**Visual C++**

```cpp
public:
virtual void Add(
    String^ name,
    String^ value
) sealed
```

### Parameters

**name**

Type: `System::::String`

Name is converted to lower case.

**value**

Type: `System::::String`


### Implements

`HttpInputBase::::Add(String, String)`
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput..::.Clear Method

HttpInput Class  See Also  Send Feedback

Resets all data contained by class

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public virtual void Clear();

Visual Basic (Declaration)

Public Overridable Sub Clear

Visual C++

public:
virtual void Clear();
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput..:::Contains Method

HttpInput Class  See Also  Send Feedback

Returns true if the class contains a HttpInput value with the corresponding name

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Contains(
    string name
)

Visual Basic (Declaration)

Public Function Contains ( _
    name As String _
) As Boolean

Visual C++

public:
    virtual bool Contains(
        String^ name
    ) sealed

Parameters

name
    Type: System:::String
    The field/querystring name

Return Value

True if the value exists

Implements

HttpInputBase:::Contains(String)
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Extracts one parameter from an array

**Namespace:** [HttpServer](#)

**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string ExtractOne(
    string value
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function ExtractOne ( _
    value As String _
) As String
```

Visual C++

```cpp
public:
static String^ ExtractOne(
    String^ value
)
```

Parameters

value
Type: System::String
Containing the string array

Return Value

All but the first value
Examples

string test1 = ExtractOne("system[user][extension][id]"); string test2 = ExtractOne(test1); string test3 = ExtractOne(test2); // test1 = user[extension][id]
// test2 = extension[id] // test3 = id
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput..::.GetEnumerator Method

HttpInput Class  See Also  Send Feedback

Returns an enumerator that iterates through a collection.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public IEnumerator GetEnumerator()
```

**Visual Basic (Declaration)**

```vbnet
Public Function GetEnumerator As IEnumerator
```

**Visual C++**

```cpp
public:
virtual IEnumerator^ GetEnumerator() sealed
```

#### Return Value

An `IEnumerator` object that can be used to iterate through the collection.

#### Implements

`IEnumerable<>::GetEnumerator()`
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parses an item and returns it. This function is primarily used to parse array items as in user[name].

**Namespace:**  [HttpServer](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static HttpInputItem ParseItem(
    string name,
    string value
)

Visual Basic (Declaration)

Public Shared Function ParseItem ( _
    name As String, _
    value As String _
) As HttpInputItem

Visual C++

public:
static HttpInputItem^ ParseItem(
    String^ name,
    String^ value
)

Parameters

name
   Type: System:::String


value
   Type: System:::String


Return Value
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInput..::.ToString Method

HttpInput Class  See Also  Send Feedback

Outputs the instance representing all its values joined together

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override string ToString()

Visual Basic (Declaration)

Public Overrides Function ToString As String

Visual C++

public:
virtual String^ ToString() override

Return Value

See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpInput` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Get a form item.</td>
</tr>
<tr>
<td>Name</td>
<td>Form name as lower case</td>
</tr>
</tbody>
</table>
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get a form item.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpInputItem this[
    string name
] { get; }

Visual Basic (Declaration)

Public ReadOnly Default Property Item ( _
    name As String _
) As HttpInputItem

Visual C++

public:
virtual property HttpInputItem^ default[String^ name] {
    HttpInputItem^ get (String^ name) sealed;
}

Parameters

name
    Type: System:::String

Return Value

Returns HttpInputItem.Empty if item was not found.

Implements

HttpInputBase:::Item[[[String]]]
See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Form name as lower case

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

public `string` Name { get; set; }

**Visual Basic (Declaration)**

Public Property Name As `String`

**Visual C++**

public:
property `String^` Name {
    `String^` get ();
    void set (`String^` value);
}

See Also

HttpInput Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Base class for request data containers

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public interface IHttpInputBase : IEnumerable<HttpInputItem>, IEnumerable
```

#### Visual Basic (Declaration)

```vbnet
Public Interface IHttpInputBase
    Implements IEnumerable(Of HttpInputItem), IEnumerable
```

#### Visual C++

```cpp
public interface class IHttpInputBase : IEnumerable<HttpInputItem^>, IEnumerable
```
See Also

HttpInputBase Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpInputBase** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Adds a parameter mapped to the presented name</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Returns true if the container contains the requested parameter</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Returns a request parameter</td>
</tr>
</tbody>
</table>
See Also

HttpInputBase Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpInputBase** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Adds a parameter mapped to the presented name</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Returns true if the container contains the requested parameter</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
See Also

HttpInputBase Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputBase...:::Add Method

HttpInputBase Interface  See Also  Send Feedback

Adds a parameter mapped to the presented name

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

void Add(
    string name,
    string value
)

Visual Basic (Declaration)

Sub Add (_
    name As String, _
    value As String _
)

Visual C++

void Add(
    String^ name,
    String^ value
)

Parameters

name
    Type: System:::String
    The name to map the parameter to

value
    Type: System:::String
    The parameter value
See Also

HttpInputBase Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputBase...::Contains Method

Returns true if the container contains the requested parameter

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
bool Contains(
    string name
)
```

Visual Basic (Declaration)

```vbnet
Function Contains ( _
    name As String _
) As Boolean
```

Visual C++

```cpp
bool Contains(
    String^ name
)
```

Parameters

name

Type: System::String

Parameter id

Return Value

True if parameter exists
See Also

HttpInputBase Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpInputBase...::GetEnumerator Method

HttpInputBase Interface  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| GetEnumerator()()()   | Returns an enumerator that iterates through the collection.  
(Inherited from IEnumerable<Of <(HttpResponseItem)>>).)                                                                 |
| GetEnumerator()()()   | Returns an enumerator that iterates through a collection.  
(Inherited from IEnumerable.) |
See Also

HttpInputBase Interface
HttpInputBase Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpInputBase** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Returns a request parameter</td>
</tr>
</tbody>
</table>
See Also

HttpInputBase Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputBase...:::Item Property

**HttpInputBase Interface**  See Also  Send Feedback

Returns a request parameter

**Namespace:**  HttpServer

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
HttpInputItem this[string name] { get; }
```

**Visual Basic (Declaration)**

```vbnet
ReadOnly Default Property Item ( _
    name As String _
) As HttpInputItem
```

**Visual C++**

```cpp
property HttpInputItem^ default[String^ name] { 
    HttpInputItem^ get (String^ name);
}
```

### Parameters

**name**

Type: `System::String`

The name associated with the parameter

### Return Value
See Also

HttpInputBase Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
represents a http input item. Each item can have multiple sub items, a sub item is made in a html form by using square brackets

**Namespace:** [HttpServer](#)  
**Assembly:** `HttpContext` (in `HttpServer.dll`) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpInputItem : HttpInputBase,
    IEnumerable<HttpInputItem>, IEnumerable

Visual Basic (Declaration)

Public Class HttpInputItem
    Implements HttpInputBase, IEnumerable(Of HttpInputItem), IEnumerable

Visual C++

public ref class HttpInputItem : HttpInputBase,
    IEnumerable<HttpInputItem^>, IEnumerable
Remarks

All names in a form SHOULD be in lowercase.
Examples

// becomes: Console.WriteLine("Value: \{0\}", form["user"]["FirstName"].Value);
Inheritance Hierarchy

System..::..Object
HttpServer..::..HttpInputItem
See Also

HttpInputItem Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpInputItem` type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpInputItem</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Contains</td>
<td>checks if a subitem exists (and has a value).</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetEnumerator</td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>🌟</td>
<td><strong>Empty</strong> Representation of a non-initialized HttpInputItem</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Number of values</td>
</tr>
<tr>
<td>Item</td>
<td>Get a sub item</td>
</tr>
<tr>
<td>Name</td>
<td>Name of item (in lower case).</td>
</tr>
<tr>
<td>Value</td>
<td>Returns the first value, or null if no value exist.</td>
</tr>
<tr>
<td>Values</td>
<td>Returns the list with values. todo: Return a readonly collection</td>
</tr>
</tbody>
</table>
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver

HttpInputItem Constructor

HttpInputItem Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpInputItem(HttpInputItem)</code></td>
<td>Creates a deep copy of the item specified</td>
</tr>
<tr>
<td><code>HttpInputItem(String, String)</code></td>
<td>Initializes an input item setting its name/identifier and value</td>
</tr>
</tbody>
</table>
See Also

HttpInputItem Class
HttpInputItem Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem Constructor (HttpInputItem)

HttpInputItem Class  See Also  Send Feedback

Creates a deep copy of the item specified

**Namespace:**  **HttpServer**  
**Assembly:**  **HttpServer** (in **HttpServer.dll**)  **Version:**  **1.0.0.0** (1.0.0.0)
Syntax

C#

```csharp
public HttpInputItem(
    HttpInputItem item
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New ( _
    item As HttpInputItem _
)
```

Visual C++

```cpp
public:
HttpInputItem(
    HttpInputItem^ item
)
```

Parameters

item

Type: `HttpServer...::HttpInputItem`

The item to copy
Remarks

The function makes a deep copy of quite a lot which can be slow
See Also

HttpInputItem Class
HttpInputItem Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem Constructor (String, String)

Initializes an input item setting its name/identifier and value

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpInputItem(
    
    string name,
    string value
)

Visual Basic (Declaration)

Public Sub New ( _
    name As String, _
    value As String _
)

Visual C++

public:
HttpInputItem(
    String^ name,
    String^ value
)

Parameters

name
    Type: System::String
    Parameter name/id

value
    Type: System::String
    Parameter value
See Also

HttpInputItem Class
HttpInputItem Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpInputItem` type exposes the following members.
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🪀 Empty</td>
<td>Representation of a non-initialized HttpInputItem</td>
</tr>
</tbody>
</table>
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Representation of a non-initialized HttpInputItem

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static readonly HttpClient Empty

Visual Basic (Declaration)

Public Shared ReadOnly Empty As HttpClient

Visual C++

public:
static initonly HttpClient^ Empty
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpInputItem` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Checks if a subitem exists (and has a value).</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>

See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add(String)</td>
<td>Add another value to this item</td>
</tr>
<tr>
<td>Add(String, String)</td>
<td>Add a sub item</td>
</tr>
</tbody>
</table>
See Also

HttpRequestItem Class
HttpRequestItem Members
HttpRequestServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add another value to this item

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void Add(
    string value
)
```

Visual Basic (Declaration)

```vbnet
Public Sub Add (_
    value As String_
)
```

Visual C++

```cpp
public:
void Add(
    String^ value
)
```

Parameters

value

Type: System::String

See Also

HttpInputItem Class
Add Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpContext Add Method (String, String)

HttpContext Class  See Also  Send Feedback

Add a sub item

Namespace:  HttpContext
Assembly:  HttpContext (in HttpContext.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
class HttpInputItem
{
    public void Add(
        string name,
        string value
    )
}
```

### Visual Basic (Declaration)

```vbnet
Public Sub Add ( _
    name As String, _
    value As String _
)
```

### Visual C++

```cpp
public:
    virtual void Add(
        String^ name,
        String^ value
    ) sealed
```

## Parameters

**name**

Type: `System::String`

Can contain array formatting, the item is then parsed and added in multiple levels

**value**

Type: `System::String`


## Implements
HttpInputBase...::Add(String, String)
See Also

HttpInputItem Class
Add Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem...:::Contains Method

HttpInputItem Class  See Also  Send Feedback

checks if a subitem exists (and has a value).

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public bool Contains(
    string name
)
```

Visual Basic (Declaration)

```vbnet
Public Function Contains ( _
    name As String _
) As Boolean
```

Visual C++

```cpp
public:
    virtual bool Contains(
        String^ name
    ) sealed
```

Parameters

name

Type: System::String
name in lower case

Return Value

can be true if the subitem exists and has a value; otherwise false.

Implements

HttpInputBase::Contains(String)
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Returns an enumerator that iterates through a collection.

Namespace:  **HttpServer**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IEnumerator GetEnumerator()

Visual Basic (Declaration)

Public Function GetEnumerator As IEnumerable

Visual C++

public:
virtual IEnumerable^ GetEnumerator() sealed

Return Value

An IEnumerator object that can be used to iterate through the collection.

Implements

IEnumerable, ...::GetEnumerator()()
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpInputItem...::ToString Method
HttpInputItem Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| **ToString()** | Returns a formatted representation of the instance with the values of all contained parameters (Overrides `Object::ToString()`.)
| **ToString(String)** | Outputs the string in a formatted manner |
See Also

HttpInputItem Class
HttpInputItem Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem...::.ToString Method

Returns a formatted representation of the instance with the values of all contained parameters

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public override string ToString()
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function ToString As String
```

**Visual C++**

```cpp
public:
virtual String^ ToString() override
```

**Return Value**

See Also

HttpInputItem Class
ToString Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem...:::ToString Method (String)

Outputs the string in a formatted manner

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string ToString(string prefix)
```

Visual Basic (Declaration)

```vbnet
Public Function ToString(_
    prefix As String _
) As String
```

Visual C++

```cpp
public:
    String^ ToString(
        String^ prefix
    )
```

Parameters

prefix

Type: `System::String`
A prefix to append, used internally

Return Value

See Also

HttpInputItem Class
ToString Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpInputItem` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Count</strong></td>
<td>Number of values</td>
</tr>
<tr>
<td><strong>Item</strong></td>
<td>Get a sub item</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Name of item (in lower case)</td>
</tr>
<tr>
<td><strong>Value</strong></td>
<td>Returns the first value, or null if no value exist.</td>
</tr>
<tr>
<td><strong>Values</strong></td>
<td>Returns the list with values. todo: Return a readonly collection</td>
</tr>
</tbody>
</table>
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Number of values

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int Count { get; }

Visual Basic (Declaration)

Public ReadOnly Property Count As Integer

Visual C++

public:
property int Count {
    int get ();
}


See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get a sub item

**Namespace:** HttpServer

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public class HttpInputItem
{
    [String] this[string name]
    {
        get { };
    }
}
```

### Visual Basic (Declaration)

```vbnet
Public ReadOnly Default Property Item (name As String) As HttpInputItem
```

### Visual C++

```cpp
public:
    property HttpInputItem^ default[String^ name] { HttpInputItem^ get (String^ name); }
```

## Parameters

**name**

Type: `System::::String`

name in lower case.

## Return Value

HttpInputItem.Empty if no item was found.
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem...::Name Property

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Name { get; set; }

Visual Basic (Declaration)

Public Property Name As String

Visual C++

public:
property String^ Name {
    String^ get ();
    void set (String^ value);
}
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem...:::Value Property

(HttpInputItem Class  See Also  Send Feedback)

Returns the first value, or null if no value exist.

**Namespace:**  HttpServer

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Value { get; }

Visual Basic (Declaration)

Public ReadOnly Property Value As String

Visual C++

public:
property String^ Value {
    String^ get ();
}

See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpInputItem::Values Property

HttpInputItem Class  See Also  Send Feedback

Returns the list with values. todo: Return a readonly collection

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public mList<string> Values { get; }

Visual Basic (Declaration)

Public ReadOnly Property Values As mList(Of String)

Visual C++

public:
property mList<String^>^ Values {
    mList<String^>^ get ();
}
See Also

HttpInputItem Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HTTP Listener waits for HTTP connections and provide us with HttpListenerContexts using the RequestHandler delegate.

**Namespace:** HttpServer  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public class HttpListener

Visual Basic (Declaration)
Public Class HttpListener

Visual C++
public ref class HttpListener
Inheritance Hierarchy

System...:::Object
  HttpServer...:::HttpListener
See Also

HttpListener Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpListener** type exposes the following members.
Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpListener</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BeginAcceptSocket</strong></td>
<td>Start accepting a socket</td>
</tr>
<tr>
<td><strong>EndAcceptSocket</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Start listen for new connections</td>
</tr>
<tr>
<td><strong>Stop</strong></td>
<td>Stop the listener</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>Write information to the log.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DisconnectHandler</td>
<td>Invoked when a client disconnects</td>
</tr>
<tr>
<td>LogWriter</td>
<td>Gives you a change to receive log entries for all internals of the http lib.</td>
</tr>
<tr>
<td>RequestHandler</td>
<td>This handler will be invoked each time a new connection is accepted.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>ExceptionThrown</td>
<td>Let's receive unhandled exceptions from the threads.</td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  □  Visual Basic
□  Visual C++
□  Include Protected Members
□  Include Inherited Members
C# Webserver
HttpListener Constructor

HttpListener Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpListener(IPAddress, Int32)</td>
<td>Listen for regular http connections</td>
</tr>
<tr>
<td>HttpListener(IPAddress, Int32, X509Certificate)</td>
<td>Launch HttpListener in SSL mode</td>
</tr>
<tr>
<td>HttpListener(IPAddress, Int32, X509Certificate, SslProtocols)</td>
<td>Launch HttpListener in SSL mode</td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpListener Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Listen for regular http connections

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpListener(
    IPAddress address,
    int port
)

Visual Basic (Declaration)

Public Sub New (_
    address As IPAddress, _
    port As Integer _
)

Visual C++

public:
HttpListener(
    IPAddress^ address,
    int port
)

Parameters

address
    Type: System.Net:::IPAddress
    IP Address to accept connections on

port
    Type: System:::Int32
    TCP Port to listen on, default HTTP port is 80.
See Also

HttpListener Class
HttpListener Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
HttpListener Constructor (IPAddress, Int32, X509Certificate)
HttpListener Class  See Also  Send Feedback
Launch HttpListener in SSL mode

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpListener(
    IPAddress address,
    int port,
    X509Certificate certificate
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New (_
    address As IPAddress, _
    port As Integer, _
    certificate As X509Certificate _
)
```

Visual C++

```cpp
public:
HttpListener(
    IPAddress^ address,
    int port,
    X509Certificate^ certificate
)
```

Parameters

address
Type: System.Net:::IPAddress
IP Address to accept connections on

port
Type: System:::Int32
TCP Port to listen on, default HTTPS port is 443

certificate
Certificate to use
See Also

HttpListener Class
HttpListener Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpListener Constructor (IPAddress, Int32, X509Certificate, SslProtocols)

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public HttpListener(
    IPAddress address,
    int port,
    X509Certificate certificate,
    SslProtocols protocol
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    address As IPAddress, _
    port As Integer, _
    certificate As X509Certificate, _
    protocol As SslProtocols _
)
```

**Visual C++**

```cpp
public:
HttpListener(
    IPAddress ^ address,
    int port,
    X509Certificate^ certificate,
    SslProtocols protocol
)
```

**Parameters**

- **address**
  - Type: `System.Net..::.IPAddress`
  - IP Address to accept connections on

- **port**
  - Type: `System..::.Int32`
  - TCP Port to listen on, default HTTPS port is 443
Certificate
  Certificate to use

Protocol
  which https protocol to use, default is Tls.
See Also

HttpListener Class
HttpListener Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpListener` type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BeginAcceptSocket</td>
<td>Start accepting a socket</td>
</tr>
<tr>
<td>EndAcceptSocket</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>Equals</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Start</td>
<td>Start listen for new connections</td>
</tr>
<tr>
<td>Stop</td>
<td>Stop the listener</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>WriteLog</td>
<td>Write information to the log.</td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Start accepting a socket

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected IAsyncResult BeginAcceptSocket(
    AsyncCallback callback,
    Object state
)

Visual Basic (Declaration)

Protected Function BeginAcceptSocket ( _
    callback As AsyncCallback, _
    state As Object _
) As IAsyncResult

Visual C++

protected:
    IAsyncResult^ BeginAcceptSocket(
        AsyncCallback^ callback,
        Object^ state
    )

Parameters

callback
    Type: System::AsyncCallback
    callback being invoked then a new client have been accepted

state
    Type: System::Object
    can be used to pass an object to the callback

Return Value

eresult identifying this async operation.
See Also

HttpListener Class  
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpListener...::EndAcceptSocket Method

HttpListener Class  See Also  Send Feedback

[Missing <summary> documentation for
M:HttpServer.HttpListener.EndAcceptSocket(System.IAsyncResult)]

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected HttpClientContext EndAcceptSocket(IAsyncResult ar)

Visual Basic (Declaration)

Protected Function EndAcceptSocket (_
    ar As IAsyncResult _
) As HttpClientContext

Visual C++

protected:
    HttpClientContext^ EndAcceptSocket(IAsyncResult^ ar)

Parameters

ar

Type: System::::IAsyncResult
    IAsyncResult sent from BeginAcceptSocket(AsyncCallback, Object).

Return Value

HttpClientContext containing connection to http client.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.Security.Authentication...::AuthenticationException</td>
<td>if SSL authentication fails</td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Start listen for new connections

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public void Start(
    int backlog
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Start ( _
    backlog As Integer _
)
```

**Visual C++**

```cpp
public:
void Start(
    int backlog
)
```

### Parameters

backlog

Type: `System::::Int32`

Number of connections that can stand in a queue to be accepted.
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpListener..::.Stop Method

Stop the listener

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Stop()

Visual Basic (Declaration)

Public Sub Stop

Visual C++

public:
void Stop()
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.Net.Sockets.SocketException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpListener...:::WriteLog Method

Write information to the log.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void WriteLog(
    Object source,
    LogPrio prio,
    string message
)

Visual Basic (Declaration)

Public Sub WriteLog ( _
    source As Object, _
    prio As LogPrio, _
    message As String _
)

Visual C++

public:
void WriteLog(
    Object^ source,
    LogPrio^ prio,
    String^ message
)

Parameters

source
    Type: System::Object
    object writing the entry

prio
    Type: HttpServer::LogPrio
    importance of the log entry

message
    Type: System::String
the actual message
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpListener` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DisconnectHandler</td>
<td>Invoked when a client disconnects</td>
</tr>
<tr>
<td>LogWriter</td>
<td>Gives you a change to receive log entries for all internals of the http lib.</td>
</tr>
<tr>
<td>RequestHandler</td>
<td>This handler will be invoked each time a new connection is accepted.</td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpServer Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Invoked when a client disconnects

**Namespace:** [HttpServer](http://HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ClientDisconnectedHandler DisconnectHandler { get; set; }

Visual Basic (Declaration)

Public Property DisconnectHandler As ClientDisconnectedHandler

Visual C++

public:
property ClientDisconnectedHandler^ DisconnectHandler {
ClientDisconnectedHandler^ get ();
void set (ClientDisconnectedHandler^ value);
}


See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Gives you a chance to receive log entries for all internals of the http lib.

**Namespace:**  [HttpServer](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public WriteLogHandler LogWriter { get; set; }

Visual Basic (Declaration)

Public Property LogWriter As WriteLogHandler

Visual C++

public:
property WriteLogHandler^ LogWriter {
    WriteLogHandler^ get ();
    void set (WriteLogHandler^ value);
}
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This handler will be invoked each time a new connection is accepted.

**Namespace:**  [HttpServer](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RequestReceivedHandler RequestHandler { get; set; }

Visual Basic (Declaration)

Public Property RequestHandler As RequestReceivedHandler

Visual C++

public:
property RequestReceivedHandler^ RequestHandler {
    RequestReceivedHandler^ get ();
    void set (RequestReceivedHandler^ value);
}
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpListener` type exposes the following members.
**Events**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExceptionThrown</td>
<td>Let's to receive unhandled exceptions from the threads.</td>
</tr>
</tbody>
</table>
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Let's to receive unhandled exceptions from the threads.

**Namespace:** HttpServer
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public event ExceptionHandler ExceptionThrown
```

**Visual Basic (Declaration)**

```vbnet
Public Event ExceptionThrown As ExceptionHandler
```

**Visual C++**

```cpp
public:
    event ExceptionHandler^ ExceptionThrown {
        void add(ExceptionHandler^ value);
        void remove(ExceptionHandler^ value);
    }
```
Remarks

Exceptions will be thrown during debug mode if this event is not used, exceptions will be printed to console and supressed during release mode.
See Also

HttpListener Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpParam Class

Returns item either from a form or a query string (checks them in that order)

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpParam : HttpInputBase, IEnumerable<HttpInputItem>, IEnumerable

Visual Basic (Declaration)

Public Class HttpParam
    Implements HttpInputBase, IEnumerable(Of HttpInputItem), IEnumerable

Visual C++

public ref class HttpParam : HttpInputBase,
    IEnumerable<HttpInputItem^>, IEnumerable
Inheritance Hierarchy

System..::..Object
    HttpServer..::..HttpGetParam
See Also

HttpParam Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpParam` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpParam</td>
<td>Initialises the class to hold a value either from a post request or a querystring request</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Obsolete.</strong></td>
<td>The add method is not available for HttpParam since HttpParam checks both Request.Form and Request.QueryString</td>
</tr>
<tr>
<td><strong>Add</strong></td>
<td>Checks whether the form or querystring has the specified value</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>(Inherited from Object.)</td>
</tr>
</tbody>
</table>
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🏃 Empty</td>
<td>Representation of a non-initialized HttpParam</td>
</tr>
</tbody>
</table>
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Fetch an item from the form or querystring (in that order).</td>
</tr>
</tbody>
</table>
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initialises the class to hold a value either from a post request or a querystring request.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpParam(
    HttpInputBase form,
    HttpInputBase query
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New ( _
    form As HttpInputBase, _
    query As HttpInputBase _
)
```

Visual C++

```cpp
public:
HttpParam(
    HttpInputBase^ form,
    HttpInputBase^ query
)
```

Parameters

form
Type: `HttpServer::HttpInputBase`


query
Type: `HttpServer::HttpInputBase`

See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpParam** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>s Empty</td>
<td>Representation of a non-initialized HttpParam</td>
</tr>
</tbody>
</table>
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpParam...::Empty Field

HttpParam Class  See Also  Send Feedback

Representation of a non-initialized HttpParam

**Namespace:**  HttpServer

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static readonly HttpParam Empty

Visual Basic (Declaration)

Public Shared ReadOnly Empty As HttpParam

Visual C++

public:
static initonly HttpParam^ Empty
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpParam` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Obsolete.</strong></td>
<td>The add method is not available for HttpParam since HttpParam checks both Request.Form and Request.QueryString</td>
</tr>
<tr>
<td><strong>Add</strong></td>
<td>Checks whether the form or querystring has the specified value</td>
</tr>
<tr>
<td><strong>Contains</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Returns an enumerator that iterates through a collection.</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The add method is not available for HttpParam since HttpParam checks both Request.Form and Request.QueryString.

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

[ObsoleteAttribute("Not implemented for HttpParam")]
public void Add(
    string name,
    string value
)

Visual Basic (Declaration)

<ObsoleteAttribute("Not implemented for HttpParam")> _
Public Sub Add (_
    name As String, _
    value As String _
)

Visual C++

[ObsoleteAttribute(L"Not implemented for HttpParam")]
public:
virtual void Add(  
    String^ name,
    String^ value
) sealed

Parameters

name
    Type: System::String
    name identifying the value

value
    Type: System::String
    value to add

Implements
HttpInputBase...:::Add(String, String)
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::NotImplementedException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains Method

HttpParam Class  See Also  Send Feedback

Checks whether the form or querystring has the specified value

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public bool Contains(
    string name
)
```

Visual Basic (Declaration)

```vbnet
Public Function Contains ( _
    name As String _
) As Boolean
```

Visual C++

```cpp
public:
    virtual bool Contains(
        String^ name
    ) sealed
```

Parameters

name

Type: System::String
Name, case sensitive

Return Value

true if found; otherwise false.

Implements

HttpInputBase::Contains(String)
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Returns an enumerator that iterates through a collection.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```
public IEnumerator GetEnumerator()
```

**Visual Basic (Declaration)**

```
Public Function GetEnumerator As IEnumerator
```

**Visual C++**

```
public:
virtual IEnumerator^ GetEnumerator() sealed
```

**Return Value**

An **IEnumerator** object that can be used to iterate through the collection.

**Implements**

**IEnumerable..::.GetEnumerator()**
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpParam` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Fetch an item from the form or querystring (in that order).</td>
</tr>
</tbody>
</table>
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Fetch an item from the form or querystring (in that order).

**Namespace:** [HttpGetServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpInputItem this[string name] { get; }
```

Visual Basic (Declaration)

```vbnet
Public ReadOnly Default Property Item ( _
    name As String _)
    As HttpInputItem
```

Visual C++

```cpp
public:
virtual property HttpInputItem^ default[String^ name] { 
    HttpInputItem^ get (String^ name) sealed;
}
```

Parameters

name

Type: System::String

Return Value

Item if found; otherwise HttpInputItem.EmptyLanguageNode

Implements

HttpInputBase::Item[[[(String)]]]
See Also

HttpParam Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest Class

Contains serverside http request information.

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public class HttpRequest : ICloneable
```

Visual Basic (Declaration)

```vbnet
Public Class HttpRequest
    Implements ICloneable
```

Visual C++

```cpp
public ref class HttpRequest : ICloneable
```
Inheritance Hierarchy

System..:::Object
HttpGetServer..:::HttpRequest
See Also

HttpRequest Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest Members

[HttpRequest Class]  [Constructors]  [Methods]  [Fields]  [Properties]  [See Also]  [Send Feedback]

The HttpRequest type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpRequest</td>
<td>Initializes a new instance of the HttpRequest class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddHeader</strong></td>
<td>Called during parsing of a HttpRequest.</td>
</tr>
<tr>
<td><strong>Clone</strong></td>
<td>Creates a new object that is a copy of the current instance.</td>
</tr>
<tr>
<td><strong>DecodeBody</strong></td>
<td>Decode body into a form.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
</tbody>
</table>
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🍀 <strong>UriSplitters</strong></td>
<td>Chars used to split an url path into multiple parts.</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcceptTypes</td>
<td>Kind of types accepted by the client.</td>
</tr>
<tr>
<td>Body</td>
<td>Submitted body contents</td>
</tr>
<tr>
<td>BodyIsComplete</td>
<td>Have all body content bytes been received?</td>
</tr>
<tr>
<td>Connection</td>
<td>Kind of connection used for the session.</td>
</tr>
<tr>
<td>ContentLength</td>
<td>Number of bytes in the body</td>
</tr>
<tr>
<td>Cookies</td>
<td>Returns set cookies for the request</td>
</tr>
<tr>
<td>Form</td>
<td>Form parameters.</td>
</tr>
<tr>
<td>Headers</td>
<td>Headers sent by the client. All names are in lower case.</td>
</tr>
<tr>
<td>HttpVersion</td>
<td>Version of http. Probably HttpHelper.HTTP10 or HttpHelper.HTTP11</td>
</tr>
<tr>
<td>IsAjax</td>
<td>Returns true if the request was made by Ajax (Asyncronous Javascript)</td>
</tr>
<tr>
<td>Method</td>
<td>Requested method, always upper case.</td>
</tr>
<tr>
<td>Param</td>
<td>Check's both QueryString and Form after the parameter.</td>
</tr>
<tr>
<td>QueryString</td>
<td>Variables sent in the query string</td>
</tr>
<tr>
<td>Uri</td>
<td>Requested URI (url)</td>
</tr>
<tr>
<td>UriParts</td>
<td>Uri absolute path splitted into parts.</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest Constructor

HttpRequest Class  See Also  Send Feedback

Initializes a new instance of the HttpRequest class

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public HttpRequest()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New
```

**Visual C++**

```cpp
public:
HttpRequest()
```
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpRequest` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UriSplitters</strong></td>
<td>Chars used to split an url path into multiple parts.</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest::UriSplitters Field

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)

Chars used to split an url path into multiple parts.
## Syntax

**C#**

```csharp
public static readonly char[] UriSplitters
```

**Visual Basic (Declaration)**

```vbnet
Public Shared ReadOnly UriSplitters As Char()
```

**Visual C++**

```cpp
public:
static initonly array<wchar_t>^ UriSplitters
```
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpRequest** type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddHeader</td>
<td>Called during parsing of a HttpRequest.</td>
</tr>
<tr>
<td>Clone</td>
<td>Creates a new object that is a copy of the current instance.</td>
</tr>
<tr>
<td>DecodeBody</td>
<td>Decode body into a form.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Called during parsing of a HttpRequest.

**Namespace:**  [HttpServer](http://HttpServer)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public void AddHeader(
    string name,
    string value
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub AddHeader (_
    name As String, _
    value As String _
)
```

**Visual C++**

```cpp
public:
void AddHeader(
    String^ name,
    String^ value
)
```

## Parameters

- **name**
  
  Type: `System::String`
  
  Name of the header, should not be url encoded

- **value**
  
  Type: `System::String`
  
  Value of the header, should not be url encoded
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>[HttpServer.Exceptions...]:<a href="https://example.com">BadRequestException</a></td>
<td>If a header is incorrect.</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest...:::Clone Method
HttpRequest Class  See Also  Send Feedback

Creates a new object that is a copy of the current instance.

**Namespace:**  [HttpServer](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public Object Clone()
```

**Visual Basic (Declaration)**

```vbnet
Public Function Clone As Object
```

**Visual C++**

```cpp
public:
virtual Object^ Clone() sealed
```

**Return Value**

A new object that is a copy of this instance.

**Implements**

`ICloneable...::Clone()`
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest...:::DecodeBody Method

HttpRequest Class  See Also  Send Feedback

Decode body into a form.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void DecodeBody(
    FormDecoderProvider providers
)

Visual Basic (Declaration)

Public Sub DecodeBody ( _
    providers As FormDecoderProvider _
)

Visual C++

public:
void DecodeBody(
    FormDecoderProvider^ providers
)

Parameters

providers
  Type: HttpServer.FormDecoders,...,:; FormDecoderProvider
  A list with form decoders.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.IO:::InvalidDataException</code></td>
<td>If body contents is not valid for the chosen decoder.</td>
</tr>
<tr>
<td><code>System:::InvalidOperationException</code></td>
<td>If body is still being transferred.</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpRequest type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcceptTypes</td>
<td>Kind of types accepted by the client.</td>
</tr>
<tr>
<td>Body</td>
<td>Submitted body contents</td>
</tr>
<tr>
<td>BodyIsComplete</td>
<td>Have all body content bytes been received?</td>
</tr>
<tr>
<td>Connection</td>
<td>Kind of connection used for the session.</td>
</tr>
<tr>
<td>ContentLength</td>
<td>Number of bytes in the body</td>
</tr>
<tr>
<td>Cookies</td>
<td>Returns set cookies for the request</td>
</tr>
<tr>
<td>Form</td>
<td>Form parameters.</td>
</tr>
<tr>
<td>Headers</td>
<td>Headers sent by the client. All names are in lower case.</td>
</tr>
<tr>
<td>HttpVersion</td>
<td>Version of http. Probably HttpHelper.HTTP10 or HttpHelper.HTTP11</td>
</tr>
<tr>
<td>IsAjax</td>
<td>Returns true if the request was made by Ajax (Asyncronous Javascript)</td>
</tr>
<tr>
<td>Method</td>
<td>Requested method, always upper case.</td>
</tr>
<tr>
<td>Param</td>
<td>Check's both QueryString and Form after the parameter.</td>
</tr>
<tr>
<td>QueryString</td>
<td>Variables sent in the query string</td>
</tr>
<tr>
<td>Uri</td>
<td>Requested URI (url)</td>
</tr>
<tr>
<td>UriParts</td>
<td>Uri absolute path splitted into parts.</td>
</tr>
</tbody>
</table>
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest..:::AcceptTypes Property

HttpRequest Class  See Also  Send Feedback

Kind of types accepted by the client.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string[] AcceptTypes { get; }

Visual Basic (Declaration)

Public ReadOnly Property AcceptTypes As String()

Visual C++

public:
property array<String^>^ AcceptTypes {
    array<String^>^ get();
}

See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest.

Submitted body contents

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Stream Body { get; set; }

Visual Basic (Declaration)

Public Property Body As Stream

Visual C++

public:
property Stream^ Body {
    Stream^ get ();
    void set (Stream^ value);
}
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest.::.BodyIsComplete Property

HttpRequest Class  See Also  Send Feedback

Have all body content bytes been received?

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool BodyIsComplete { get; }

Visual Basic (Declaration)

Public ReadOnly Property BodyIsComplete As Boolean

Visual C++

public:
property bool BodyIsComplete {
    bool get ();
}

See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest...:.Connection Property

HttpRequest Class  See Also  Send Feedback

Kind of connection used for the session.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ConnectionType Connection { get; set; }

Visual Basic (Declaration)

Public Property Connection As ConnectionType

Visual C++

public:
property ConnectionType Connection {
    ConnectionType get ();
    void set (ConnectionType value);
}
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest...::ContentLength Property

HttpRequest Class  See Also  Send Feedback

Number of bytes in the body

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int ContentLength { get; set; }

Visual Basic (Declaration)

Public Property ContentLength As Integer

Visual C++

public:
property int ContentLength {
    int get ();
    void set (int value);
}
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest.Cookies Property

Returns set cookies for the request

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RequestCookies Cookies { get; }

Visual Basic (Declaration)

Public ReadOnly Property Cookies As RequestCookies

Visual C++

public:

property RequestCookies^ Cookies {
    RequestCookies^ get ();
}

See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest.Form Property

HttpRequest Class  See Also  Send Feedback

Form parameters.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpForm Form { get; }

Visual Basic (Declaration)

Public ReadOnly Property Form As HttpForm

Visual C++

public:
property HttpForm^ Form {
HttpForm^ get ();
}
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Headers sent by the client. All names are in lower case.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#  
public NameValueCollection Headers { get; }

Visual Basic (Declaration)  
Public ReadOnly Property Headers As NameValueCollection

Visual C++  
public:  
property NameValueCollection^ Headers {  
NameValueCollection^ get ();  
}


See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Version of http. Probably HttpHelper.HTTP10 or HttpHelper.HTTP11

**Namespace:** [HttpGetServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string HttpVersion { get; set; }

Visual Basic (Declaration)

Public Property HttpVersion As String

Visual C++

public:
    property String^ HttpVersion {
        String^ get ();
        void set (String^ value);
    }
See Also

HttpRequest Class
HttpServer Namespace
HttpServer...:..HttpHelper

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest:::IsAjax Property

HttpRequest Class  See Also  Send Feedback

Returns true if the request was made by Ajax (Asyncronous Javascript)

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool IsAjax { get; }

Visual Basic (Declaration)

Public ReadOnly Property IsAjax As Boolean

Visual C++

public:
property bool IsAjax {
    bool get ();
}


See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Requested method, always upper case.

**Namespace:** [HttpServer](#)

**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Method { get; set; }

Visual Basic (Declaration)

Public Property Method As String

Visual C++

public:
property String^ Method {
    String^ get ();
    void set (String^ value);
}

See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest::Param Property

HttpRequest Class  See Also  Send Feedback

Check's both QueryString and Form after the parameter.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public HttpResponseMessage Param { get; }

Visual Basic (Declaration)
Public ReadOnly Property Param As HttpResponseMessage

Visual C++
public:
property HttpResponseMessage^ Param {
    HttpResponseMessage^ get ();
}
See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest.QueryString Property

Variables sent in the query string

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpInput QueryString { get; }

Visual Basic (Declaration)

Public ReadOnly Property QueryString As HttpInput

Visual C++

public:
property HttpInput^ QueryString {
    HttpInput^ get ();
}

See Also

HttpRequest Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequest..::.Uri Property

HttpRequest Class  See Also  Send Feedback

Requested URI (url)

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Uri Uri { get; set; }

Visual Basic (Declaration)

Public Property Uri As Uri

Visual C++

public:
property Uri^ Uri {
    Uri^ get ();
    void set (Uri^ value);
}
See Also

HttpRequest Class
HttpServer Namespace
HttpRequestUriPath()()

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Uri absolute path splitted into parts.

**Namespace:**  [HttpServer](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string[] UriParts { get; }

Visual Basic (Declaration)

Public ReadOnly Property UriParts As String()

Visual C++

public:
property array<String>^ UriParts {
    array<String>^ get();
}
Remarks

If you're using controllers than the first part is controller name, the second part is method name and the third part is Id property.
Examples

// uri is: http://gauffin.com/code/tiny/
Console.WriteLine(request.UriParts[0]);  // result: code
Console.WriteLine(request.UriParts[1]);  // result: tiny
See Also

HttpRequest Class
HttpServer Namespace
HttpRequest...: Uri

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequestParser Class

Parses a HTTP request directly from a stream

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpRequestParser

Visual Basic (Declaration)

Public Class HttpRequestParser

Visual C++

public ref class HttpRequestParser
Inheritance Hierarchy

System..:::Object
HttpServer..:::HttpRequestParser
See Also

HttpRequestParser Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpRequestParser type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpRequestParser</td>
<td>Create a new request parser</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>OnFirstLine</strong></td>
<td>Parse request line</td>
</tr>
<tr>
<td><strong>OnHeader</strong></td>
<td>We've parsed a new header.</td>
</tr>
<tr>
<td><strong>ParseMessage</strong></td>
<td>Parse a message</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>Write information to the log.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td><code>CurrentState</code></td>
<td>Current state in parser.</td>
</tr>
</tbody>
</table>
See Also

HttpRequestParser Class
HttpServer Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new request parser

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpRequestParser(
    HttpRequestParser::::RequestCompletedHandler requestCompleted
    WriteLogHandler logWriter
)

Visual Basic (Declaration)

Public Sub New (_
    requestCompleted As HttpRequestParser::::RequestCompletedHandler
    logWriter As WriteLogHandler _
)

Visual C++

public:
HttpRequestParser(
    HttpRequestParser::::RequestCompletedHandler^ requestCompleted
    WriteLogHandler^ logWriter
)

Parameters

requestCompleted
    Type: HttpRequestParser::::RequestCompletedHandler
delegate called when a complete request have been generated

logWriter
    Type: HttpServer::::WriteLogHandler
delegate receiving log entries.
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpRequestParser` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>OnFirstLine</strong></td>
<td>Parse request line</td>
</tr>
<tr>
<td><strong>OnHeader</strong></td>
<td>We've parsed a new header.</td>
</tr>
<tr>
<td><strong>ParseMessage</strong></td>
<td>Parse a message</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>Write information to the log.</td>
</tr>
</tbody>
</table>
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequestParser::OnFirstLine Method

HttpRequestParser Class  See Also  Send Feedback

Parse request line

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
protected void OnFirstLine(
    string value
)
```

Visual Basic (Declaration)

```vbnet
Protected Sub OnFirstLine (_
    value As String _
)
```

Visual C++

```cpp
protected:
void OnFirstLine(
    String^ value
)
```

Parameters

value

Type: `System::::String`

Remarks

Expects the following format: "Method SP Request-URI SP HTTP-Version CRLF"
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Exceptions...::BadRequestException</code></td>
<td>If line is incorrect</td>
</tr>
</tbody>
</table>
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
We've parsed a new header.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void OnHeader(
    string name,
    string value
)

Visual Basic (Declaration)

Protected Sub OnHeader (_
    name As String, _
    value As String _
)

Visual C++

protected:
void OnHeader(
    String^ name,
    String^ value
)

Parameters

name
Type: System::String
Name in lower case

value
Type: System::String
Value, unmodified.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpException::BadRequestException</code></td>
<td>If content length cannot be parsed.</td>
</tr>
</tbody>
</table>
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequestParser...::ParseMessage Method

HttpRequestParser Class  See Also  Send Feedback

Parse a message

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public int ParseMessage(
    byte[] buffer,
    int offset,
    int size
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function ParseMessage (_
    buffer As Byte(), _
    offset As Integer, _
    size As Integer _
) As Integer
```

**Visual C++**

```cpp
public:
    int ParseMessage(
        array<unsigned char>^ buffer,
        int offset,
        int size
    )
```

**Parameters**

- **buffer**
  - Type: array< System::Byte >[]
  

- **offset**
  - Type: System::Int32
  where in buffer that parsing should start

- **size**
Type: System::Int32
number of bytes to parse

Return Value

Unparsed bytes left in buffer.
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
HttpRequestParser::WriteLog Method
HttpRequestParser Class  See Also  Send Feedback

Write information to the log.

**Namespace:**  HttpHeaders
**Assembly:**  HttpHeaders (in HttpHeaders.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public void WriteLog(
    Object source,
    LogPrio prio,
    string message
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub WriteLog (_
    source As Object, _
    prio As LogPrio, _
    message As String _
)
```

### Visual C++

```cpp
public:
void WriteLog(
    Object^ source,
    LogPrio prio,
    String^ message
)
```

## Parameters

**source**

Type: `System::Object`

object writing the entry

**prio**

Type: `HttpServer::LogPrio`

importance of the log entry

**message**

Type: `System::String`
the actual message
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpRequestParser` type exposes the following members.
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CurrentState</td>
<td>Current state in parser.</td>
</tr>
</tbody>
</table>
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Current state in parser.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpRequestParser::State CurrentState { get; }
```

Visual Basic (Declaration)

```vbnet
Public ReadOnly Property CurrentState As HttpRequestParser::State
```

Visual C++

```cpp
public:
property HttpRequestParser::State CurrentState {
HttpRequestParser::State get ();
}
```
See Also

HttpRequestParser Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpRequestParser::RequestCompletedHandler Delegate

Invoked when a request have been completed.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public delegate void RequestCompletedHandler(
    HttpRequest request
)

Visual Basic (Declaration)

Public Delegate Sub RequestCompletedHandler ( _
    request As HttpRequest _
)

Visual C++

public delegate void RequestCompletedHandler( 
    HttpRequest^ request
)

Parameters

request
    Type: HttpServer::HttpRequest
See Also

HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Current state in the parsing.

**Namespace:** [HttpServer](https://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public enum State

Visual Basic (Declaration)

Public Enumeration State

Visual C++

public enum class State
## Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FirstLine</td>
<td>Should parse the request line</td>
</tr>
<tr>
<td>HeaderName</td>
<td>Searching for a complete header name</td>
</tr>
<tr>
<td>AfterName</td>
<td>Searching for colon after header name (ignoring white spaces)</td>
</tr>
<tr>
<td>Between</td>
<td>Searching for start of header value (ignoring white spaces)</td>
</tr>
<tr>
<td>HeaderValue</td>
<td>Searching for a complete header value (can span over multiple lines, as long as they are prefixed with one/more whitespaces)</td>
</tr>
<tr>
<td>Body</td>
<td>Adding bytes to body</td>
</tr>
</tbody>
</table>
See Also

HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Response that is sent back to the web browser / client. A response can be sent if different ways. The easiest one is to just fill the Body stream with content, everything else will then be taken care of by the framework. The default content-type is text/html, you should change it if you send anything else. The second and slightly more complex way is to send the response as parts. Start with sending the header using the SendHeaders method and then you can send the body using SendBody method, but do not forget to set ContentType and ContentLength before doing so.

**Namespace:** HttpServer  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public class HttpResponseMessage

Visual Basic (Declaration)
Public Class HttpResponseMessage

Visual C++
public ref class HttpResponseMessage
Examples

```java
public void MyHandler(HttpRequest request, HttpResponse response) {
}
```
Inheritance Hierarchy

System...:::Object
HttpServer...:::HttpResponse
See Also

HttpResponse Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpResponse` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpResponse</td>
<td>Initializes a new instance of the <a href="#">HttpResponse</a> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddHeader</td>
<td>Add another header to the document.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>Redirect</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Send</td>
<td>Send headers and body to the browser.</td>
</tr>
<tr>
<td>SendBody</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>SendHeaders</td>
<td>Send headers to the client.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Body</strong></td>
<td>The body stream is used to cache the body contents before sending everything to the client. It's the simplest way to serve documents.</td>
</tr>
<tr>
<td></td>
<td>The chunked encoding modifies the body of a message in order to transfer it as a series of chunks, each with its own size indicator, followed by an OPTIONAL trailer containing entity-header fields. This allows dynamically produced content to be transferred along with the information necessary for the recipient to verify that it has received the full message.</td>
</tr>
<tr>
<td><strong>Connection</strong></td>
<td>Kind of connection</td>
</tr>
<tr>
<td><strong>ContentLength</strong></td>
<td>Size of the body. MUST be specified before sending the header, unless property Chunked is set to true.</td>
</tr>
<tr>
<td><strong>ContentType</strong></td>
<td>Kind of content in the body</td>
</tr>
<tr>
<td><strong>Cookies</strong></td>
<td>Cookies that should be created/changed.</td>
</tr>
<tr>
<td><strong>Encoding</strong></td>
<td>Encoding to use when sending stuff to the client.</td>
</tr>
<tr>
<td><strong>HeadersSent</strong></td>
<td>Headers have been sent to the client-</td>
</tr>
<tr>
<td><strong>KeepAlive</strong></td>
<td>Number of seconds to keep connection alive</td>
</tr>
<tr>
<td><strong>Reason</strong></td>
<td>Information about why a specific status code was used.</td>
</tr>
<tr>
<td><strong>Sent</strong></td>
<td>The whole response have been sent.</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>Status code that is sent to the client.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse Constructor

HTTPResponse Class  See Also  Send Feedback

Initializes a new instance of the HttpRequest class

Namespace:  HttpHeaders
Assembly:  HttpHeaders (in HttpHeaders.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpResponseMessage(
    HttpClientContext context,
    HttpRequest request
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (
    context As HttpClientContext, _
    request As HttpRequest _
)
```

**Visual C++**

```cpp
public:
HttpResponse(
    HttpClientContext^ context,
    HttpRequest^ request
)
```

Parameters

context

Type: **HttpServer..::.HttpClientContext**


request

Type: **HttpServer..::.HttpRequest**

## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System::ArgumentException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpResponseMessage type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddHeader</strong></td>
<td>Add another header to the document.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Redirect</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Send</strong></td>
<td>Send headers and body to the browser.</td>
</tr>
<tr>
<td><strong>SendBody</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>SendHeaders</strong></td>
<td>Send headers to the client.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>

[Object](#): <https://docs.microsoft.com>
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add another header to the document.

**Namespace:** [HttpServer](https://example.com)

**Assembly:** [HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)](https://example.com)
Syntax

C#

public void AddHeader(
    string name,
    string value
)

Visual Basic (Declaration)

Public Sub AddHeader ( _
    name As String, _
    value As String _
)

Visual C++

public:
void AddHeader(
    String^ name,
    String^ value
)

Parameters

name
  Type: System::String
  Name of the header, case sensitive, use lower cases.

value
  Type: System::String
  Header values can span over multiple lines as long as each line starts with a white space. New line chars should be \n
Remarks

Adding any header will override the default ones and those specified by properties.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::InvalidOperationException</code></td>
<td>If headers already been sent.</td>
</tr>
<tr>
<td><code>System:::ArgumentException</code></td>
<td>If value conditions have not been met.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Redirect(String)</td>
<td>redirect to somewhere</td>
</tr>
<tr>
<td>Redirect(Uri)</td>
<td>Redirect client to somewhere else using the 302 status code.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpResponse Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse.Redirect Method (String)

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Redirect(
    string url
)

Visual Basic (Declaration)

Public Sub Redirect ( _
    url As String _
)

Visual C++

public:
void Redirect(
    String^ url
)

Parameters

url
Type: System::String
where the redirect should go
Remarks

No body are allowed when doing redirects.
See Also

HttpResponse Class
Redirect Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Redirect client to somewhere else using the 302 status code.

**Namespace:** [HttpServer](#)  
**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Redirect(
    Uri uri
)

Visual Basic (Declaration)

Public Sub Redirect ( _
    uri As Uri _
)

Visual C++

public:
    void Redirect(
        Uri^ uri
    )

Parameters

uri

Type: System::Uri
Destination of the redirect
Remarks

You can not do anything more with the request when a redirect have been done. This should be your last action.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System::::InvalidOperationException</code></td>
<td>If headers already been sent.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
Redirect Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse..::.Send Method

HttpResponse Class  See Also  Send Feedback

Send headers and body to the browser.

**Namespace:**  [HttpServer](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
# Syntax

**C#**

```csharp
public void Send()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Send
```

**Visual C++**

```cpp
public:
void Send()
```
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System...:::InvalidOperationException</td>
<td>If content have already been sent.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpResponse:::SendBody Method

HttpResponse Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SendBody(array[Byte][][])</td>
<td>Make sure that you have specified ContentLength and sent the headers first.</td>
</tr>
<tr>
<td>SendBody(array[Byte][][], Int32, Int32)</td>
<td>Make sure that you have specified ContentLength and sent the headers first.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpResponse Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse..::.SendBody Method (array<Byte>[]()[])

HttpResponse Class  See Also  Send Feedback

Make sure that you have specified ContentLength and sent the headers first.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void SendBody(
    byte[] buffer
)
```

Visual Basic (Declaration)

```vbnet
Public Sub SendBody ( _
    buffer As Byte() _
)
```

Visual C++

```cpp
public:
void SendBody(
    array<unsigned char>^ buffer
)
```

Parameters

buffer
    Type: array< System::::Byte >[]0[]

Remarks

This method can be used if you want to send body contents without caching them first. This is recommended for larger files to keep the memory usage low.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System...::InvalidOperationException</code></td>
<td>If headers have not been sent.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
SendBody Overload
HttpServer Namespace
HttpResponse:::Send()
HttpResponse:::SendHeaders()
HttpResponse::SendBody Method (array<Byte>[][], Int32, Int32)

HttpResponse Class  See Also  Send Feedback

Make sure that you have specified ContentLength and sent the headers first.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public void SendBody(
    byte[] buffer,
    int offset,
    int count
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub SendBody (_
    buffer As Byte(), _
    offset As Integer, _
    count As Integer _
)
```

**Visual C++**

```cpp
public:
void SendBody(
    array<unsigned char>^ buffer, 
    int offset,
    int count
)
```

**Parameters**

**buffer**

Type: `array<System::::Byte>[]()`


**offset**

Type: `System::::Int32`

offset of first byte to send

**count**
Type: `System::Int32`
number of bytes to send.
Remarks

This method can be used if you want to send body contents without caching them first. This is recommended for larger files to keep the memory usage low.
<table>
<thead>
<tr>
<th><strong>Exception</strong></th>
<th><strong>Condition</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System...::InvalidOperationException</code></td>
<td>If headers have not been sent.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
SendBody Overload
HttpServer Namespace
HttpResponse:::Send()
HttpResponse:::SendHeaders()
Send headers to the client.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public void SendHeaders()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub SendHeaders
```

**Visual C++**

```cpp
public:
    void SendHeaders()
```
# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.IO.InvalidOperationException</td>
<td>If headers already been sent.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpServer Namespace
HttpResponse::AddHeader(String, String)
HttpResponse::Send()
HttpResponse::SendBody(array<Byte>[][])

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpResponse** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Body</strong></td>
<td>The body stream is used to cache the body contents before sending everything to the client. It's the simplest way to serve documents.</td>
</tr>
<tr>
<td><strong>Chunked</strong></td>
<td>The chunked encoding modifies the body of a message in order to transfer it as a series of chunks, each with its own size indicator, followed by an OPTIONAL trailer containing entity-header fields. This allows dynamically produced content to be transferred along with the information necessary for the recipient to verify that it has received the full message.</td>
</tr>
<tr>
<td><strong>Connection</strong></td>
<td>Kind of connection</td>
</tr>
<tr>
<td><strong>ContentLength</strong></td>
<td>Size of the body. MUST be specified before sending the header, unless property Chunked is set to true.</td>
</tr>
<tr>
<td><strong>ContentType</strong></td>
<td>Kind of content in the body</td>
</tr>
<tr>
<td><strong>Cookies</strong></td>
<td>Cookies that should be created/changed.</td>
</tr>
<tr>
<td><strong>Encoding</strong></td>
<td>Encoding to use when sending stuff to the client.</td>
</tr>
<tr>
<td><strong>HeadersSent</strong></td>
<td>Headers have been sent to the client.</td>
</tr>
<tr>
<td><strong>KeepAlive</strong></td>
<td>Number of seconds to keep connection alive</td>
</tr>
<tr>
<td><strong>Reason</strong></td>
<td>Information about why a specific status code was used.</td>
</tr>
<tr>
<td><strong>Sent</strong></td>
<td>The whole response have been sent.</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>Status code that is sent to the client.</td>
</tr>
</tbody>
</table>
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The body stream is used to cache the body contents before sending everything to the client. It's the simplest way to serve documents.

**Namespace:** HttpServer

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Stream Body { get; set; }

Visual Basic (Declaration)

Public Property Body As Stream

Visual C++

public:
property Stream^ Body {
Stream^ get ();
void set (Stream^ value);
}
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The chunked encoding modifies the body of a message in order to transfer it as a series of chunks, each with its own size indicator, followed by an OPTIONAL trailer containing entity-header fields. This allows dynamically produced content to be transferred along with the information necessary for the recipient to verify that it has received the full message.

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Chunked { get; set; }

Visual Basic (Declaration)

Public Property Chunked As Boolean

Visual C++

public:
property bool Chunked {
    bool get ();
    void set (bool value);
}
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Kind of connection

**Namespace:** [HttpServer](http://example.com)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ConnectionType Connection { get; set; }

Visual Basic (Declaration)

Public Property Connection As ConnectionType

Visual C++

public:
    property ConnectionType Connection {
    ConnectionType get ();
    void set (ConnectionType value);
    }
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse...::.ContentLength Property

Size of the body. MUST be specified before sending the header, unless property Chunked is set to true.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public long ContentLength { get; set; }

Visual Basic (Declaration)

Public Property ContentLength As Long

Visual C++

public:
property long long ContentLength {
    long long get ();
    void set (long long value);
}

See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Kind of content in the body

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string ContentType { get; set; }

Visual Basic (Declaration)

Public Property ContentType As String

Visual C++

public:
property String^ ContentType {
    String^ get ();
    void set (String^ value);
}
Remarks

Default is text/html
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Cookies that should be created/changed.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll)  Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public ResponseCookies Cookies { get; }
```

Visual Basic (Declaration)

```vbnet
Public ReadOnly Property Cookies As ResponseCookies
```

Visual C++

```cpp
public:
property ResponseCookies^ Cookies {
    ResponseCookies^ get ();
}
```
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse...:::Encoding Property

**HttpResponse Class**  **See Also**  **Send Feedback**

Encoding to use when sending stuff to the client.

**Namespace:**  [HttpServer](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Encoding Encoding { get; set; }

Visual Basic (Declaration)

Public Property Encoding As Encoding

Visual C++

public:
property Encoding^ Encoding {
    Encoding^ get ();
    void set (Encoding^ value);
}
Remarks

Default is UTF8
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Headers have been sent to the client-

**Namespace:**  [HttpServer](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll)  
**Version:**  1.0.0.0 (1.0.0.0)
Syntax

C#

public bool HeadersSent { get; }

Visual Basic (Declaration)

Public ReadOnly Property HeadersSent As Boolean

Visual C++

public:
property bool HeadersSent {
    bool get ();
}

Remarks

You can not send any additional headers if they have already been sent.
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse.::.KeepAlive Property

**HttpResponse Class**  **See Also**  **Send Feedback**

Number of seconds to keep connection alive

**Namespace:**  [HttpServer](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int KeepAlive { get; set; }

Visual Basic (Declaration)

Public Property KeepAlive As Integer

Visual C++

public:
property int KeepAlive {
    int get ();
    void set (int value);
}
Remarks

Only used if Connection property is set to ConnectionType.KeepAlive
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse..::.Reason Property

HttpResponse Class  See Also  Send Feedback

Information about why a specific status code was used.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

class Reason
{
    public static string Reason { get; set; }
}

Visual Basic (Declaration)

Public Property Reason As String

Visual C++

public:
    property String^ Reason {
        String^ get ()
        void set (String^ value);
    }
}
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse..::.Sent Property

The whole response have been sent.

Namespace:  **HttpServer**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Sent { get; }

Visual Basic (Declaration)

Public ReadOnly Property Sent As Boolean

Visual C++

public:
property bool Sent {
    bool get ();
}
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpResponse..::.Status Property

HttpStatus code that is sent to the client.

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpStatusCode Status { get; set; }

Visual Basic (Declaration)

Public Property Status As HttpStatusCode

Visual C++

public:
property HttpStatusCode Status {
(HttpStatusCode get ();
void set ((HttpStatusCode value);
}
Remarks

Default is HttpStatusCode.Ok
See Also

HttpResponse Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A complete http server, you need to add a module to it to be able to handle incoming requests.

**Namespace:** HttpServer  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public class HttpServer

Visual Basic (Declaration)
Public Class HttpServer

Visual C++
public ref class HttpServer
// this small example will add two web site modules, thus handling // two different sites. In reality you should add Controller modules // two the website modules to be able to handle different requests.
HttpServer server = new HttpServer();
server.Add(new WebSiteModule("www.gauffin.com", "Gauffin Telecom AB"));
server.Add(new WebSiteModule("www.vapadi.se", "Remote PBX"));

// start regular http
server.Start(IPAddress.Any, 80);

// start https
server.Start(IPAddress.Any, 443, myCertificate);
Inheritance Hierarchy

System...:::Object
HttpServer...:::HttpServer
See Also

HttpServer Members
HttpServer Namespace
HttpServer.HttpModules..::HttpModule
HttpServer.HttpModules..::ControllerModule
HttpServer.HttpModules..::FileModule
HttpServer..::HttpListener

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpServer** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpServer</td>
<td>Initializes a new instance of the <strong>HttpServer</strong> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>DecodeBody</strong></td>
<td>Decodes the request body.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.</td>
</tr>
<tr>
<td><strong>ErrorPage</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetRealm</strong></td>
<td>Realms are used by the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">AuthModules</a>.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>HandleRequest</strong></td>
<td>Process an incoming request.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>OnClientDisconnected</strong></td>
<td>Can be overloaded to implement stuff when a client have been connected.</td>
</tr>
<tr>
<td><strong>ProcessAuthentication</strong></td>
<td>Handle authentication</td>
</tr>
<tr>
<td><strong>RequestAuthentication</strong></td>
<td>Will request authentication.</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Stop</strong></td>
<td>shut down the server and listeners</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>write an entry to the log file</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AuthenticationModules</strong></td>
<td>Modules used for authentication. The module that is added first is used as</td>
</tr>
<tr>
<td></td>
<td>the default auth module.</td>
</tr>
<tr>
<td><strong>FormDecoderProviders</strong></td>
<td>Form decoder providers are used to decode request body (which normally</td>
</tr>
<tr>
<td></td>
<td>contains form data).</td>
</tr>
<tr>
<td><strong>ServerName</strong></td>
<td>Server name sent in http responses.</td>
</tr>
<tr>
<td><strong>SessionCookieName</strong></td>
<td>Name of cookie where session id is stored.</td>
</tr>
</tbody>
</table>
### Events

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExceptionThrown</td>
<td>Let's receive unhandled exceptions from the threads.</td>
</tr>
<tr>
<td>LogEntryWritten</td>
<td>Use this event to be able to get log entries into your favorite log library.</td>
</tr>
<tr>
<td>RealmWanted</td>
<td>Realms are used during http authentication. Default realm is same as server name.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpServer Constructor

HttpServer Class  See Also  Send Feedback

Initializes a new instance of the HttpServer class

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpServer();

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
HttpServer();
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpServer** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>DecodeBody</strong></td>
<td>Decodes the request body.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>ErrorPage</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetRealm</strong></td>
<td>Realms are used by the <code>AuthModules</code>.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>HandleRequest</strong></td>
<td>Process an incoming request.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>OnClientDisconnected</strong></td>
<td>Can be overloaded to implement stuff when a client have been connected.</td>
</tr>
<tr>
<td><strong>ProcessAuthentication</strong></td>
<td>Handle authentication</td>
</tr>
<tr>
<td><strong>RequestAuthentication</strong></td>
<td>Will request authentication.</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Stop</strong></td>
<td>shut down the server and listeners</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>WriteLog</strong></td>
<td>write an entry to the log file</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpServer...Add Method

HttpRequest Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Add(HttpModule)</code></td>
<td>Add a <code>HttpModule</code> to the server.</td>
</tr>
<tr>
<td><code>Add(RedirectRule)</code></td>
<td>Adds the specified rule.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add a **HttpModule** to the server.

**Namespace:**  **HttpServer**  
**Assembly:**  **HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)**
Syntax

C#

```csharp
public void Add(HttpModule module)
```

Visual Basic (Declaration)

```vbnet
Public Sub Add(module As HttpModule)
```

Visual C++

```cpp
public:
void Add(HttpModule module)
```

Parameters

- **module**
  - Type: `HttpServer.HttpModules...HttpModule`
  - mode to add
See Also

HttpServer Class
Add Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Adds the specified rule.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Add(
    RedirectRule rule
)

Visual Basic (Declaration)

Public Sub Add ( _
    rule As RedirectRule _
)

Visual C++

public:
    void Add(
        RedirectRule^ rule
    )

Parameters

rule
    Type: HttpServer.Rules::::RedirectRule
    The rule.
See Also

HttpServer Class
Add Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Decodes the request body.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void DecodeBody(
    HttpRequest request
)

Visual Basic (Declaration)

Protected Overridable Sub DecodeBody ( _
    request As HttpRequest _
)

Visual C++

protected:
virtual void DecodeBody(  
    HttpRequest^ request
)

Parameters

request
Type: HttpRequest
The request.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
- Visual C++
- Include Protected Members
- Include Inherited Members

C# Webserver
HttpServer...:.ErrorPage Method

HttpServer Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ErrorPage(HttpResponse,HttpException)</code></td>
<td>Generate a HTTP error page (that will be added to the response body). response status code is also set.</td>
</tr>
<tr>
<td><code>ErrorPage(HttpResponse,HttpStatusCode,String)</code></td>
<td>Generate a HTTP error page (that will be added to the response body). response status code is also set.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate a HTTP error page (that will be added to the response body). response status code is also set.

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void ErrorPage(
    HttpResponse response,
    HttpException err
)

Visual Basic (Declaration)

Protected Overridable Sub ErrorPage ( _
    response As HttpResponse, _
    err As HttpException _
)

Visual C++

protected:
virtual void ErrorPage(
    HttpResponse^ response,
    HttpException^ err
)

Parameters

response
    Type: HttpServer:::HttpResponse
    Response that the page will be generated in.

erro
    Type: HttpServer.Exceptions:::HttpException
    exception.
See Also

HttpServer Class
ErrorPage Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate a HTTP error page (that will be added to the response body). response status code is also set.

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void ErrorPage(
    HttpResponse response,
    HttpStatusCode error,
    string body
)

Visual Basic (Declaration)

Protected Overridable Sub ErrorPage ( _
    response As HttpResponse, _
    error As HttpStatusCode, _
    body As String _
)

Visual C++

protected:
    virtual void ErrorPage(
        HttpResponse^ response,
        HttpStatusCode error,
        String^ body
    )

Parameters

response
    Type: HttpServer::::HttpResponse
    Response that the page will be generated in.

error
    Type: System.Net::::HttpStatusCode
    http error.

body
    Type: System::::String
response body contents.
See Also

HttpServer Class
ErrorPage Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Realms are used by the AuthModules.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual string GetRealm(
    HttpRequest request
)

Visual Basic (Declaration)

Protected Overridable Function GetRealm ( _
    request As HttpRequest _
) As String

Visual C++

protected:
    virtual String^ GetRealm(
        HttpRequest^ request
    )

Parameters

request
    Type: HttpServer:::HttpRequest
    http request

Return Value

domain/realm.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Process an incoming request.

**Namespace:**  [HttpServer](/api/HttpServer)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
protected virtual void HandleRequest(
    HttpClientContext context,
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)
```

**Visual Basic (Declaration)**

```vbnet
Protected Overridable Sub HandleRequest (_
    context As HttpClientContext, _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
)
```

**Visual C++**

```cpp
protected:
    virtual void HandleRequest(
        HttpClientContext^ context,
        HttpRequest^ request,
        HttpResponse^ response,
        HttpSession^ session
    )
```

### Parameters

- **context**
  - Type: `HttpServer::HttpClientContext`
  - Connection to client

- **request**
  - Type: `HttpServer::HttpRequest`
  - Request information
response
Type: `HttpServer::HttpResponse`
response that should be filled

session
Type: `HttpServer::HttpSession`
session information
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Can be overloaded to implement stuff when a client have been connected.

**Namespace:** [HttpServer](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
protected virtual void OnClientDisconnected(
    HttpClientContext client,
    SocketError error
)
```

### Visual Basic (Declaration)

```vbnet
Protected Overridable Sub OnClientDisconnected ( _
    client As HttpClientContext, _
    error As SocketError _
)
```

### Visual C++

```cpp
protected:
    virtual void OnClientDisconnected(
        HttpClientContext^ client,
        SocketError error
    )
```

## Parameters

### client

Type: `HttpServer..::.HttpContext`  
client that disconnected

### error

Type: `System.Net.Sockets..::.SocketError`  
disconnect reason
Remarks

Default implementation does nothing.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Handle authentication

**Namespace:** [HttpServer](#)  
**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual bool ProcessAuthentication(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)

Visual Basic (Declaration)

Protected Overridable Function ProcessAuthentication ( _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
) As Boolean

Visual C++

protected:
    virtual bool ProcessAuthentication(
        HttpRequest^ request,
        HttpResponse^ response,
        HttpSession^ session
    )

Parameters

request
    Type: HttpServer...: HttpRequest

response
    Type: HttpServer...: HttpResponse
session

Type: HttpServer..::HttpSession


Return Value

true if request can be handled; false if not.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Will request authentication.

**Namespace:** [HttpServer](#)  
**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void RequestAuthentication(
    AuthModule mod,
    HttpRequest request,
    HttpResponse response
)

Visual Basic (Declaration)

Protected Overridable Sub RequestAuthentication (
    mod As AuthModule, 
    request As HttpRequest, 
    response As HttpResponse 
)

Visual C++

protected:
virtual void RequestAuthentication(
    AuthModule^ mod,
    HttpRequest^ request,
    HttpResponse^ response
)

Parameters

mod
Type: HttpServer.Authentication..::.AuthModule


request
Type: HttpServer..::.HttpRequest

response
  Type: **HttpResponse**

Remarks

Sends respond to client, nothing else can be done with the response after this.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpServer...: Start Method

HttpServer Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Start(IPAddress, Int32)</code></td>
<td>Start the webserver using regular HTTP.</td>
</tr>
<tr>
<td><code>Start(IPAddress, Int32, X509Certificate)</code></td>
<td>Accept secure connections.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Start the webserver using regular HTTP.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Start(
    IPAddress address,
    int port
)

Visual Basic (Declaration)

Public Sub Start ( _
    address As IPAddress, _
    port As Integer _
)

Visual C++

public:
void Start(
    IPAddress* address,
    int port
)

Parameters

address
  Type: System.Net.:::IPAddress
  IP Address to listen on, use IPAddress.Any to accept connections on all
  ipaddresses/network cards.

port
  Type: System.:::Int32
  Port to listen on. 80 can be a good idea =)
See Also

HttpServer Class
Start Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
HttpServer....Start Method (IPAddress, Int32, X509Certificate)

HttpServer Class  See Also  Send Feedback

Accept secure connections.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public void Start(
    IPAddress address,
    int port,
    X509Certificate certificate
)
```

#### Visual Basic (Declaration)

```vbnet
Public Sub Start (_
    address As IPAddress, _
    port As Integer, _
    certificate As X509Certificate _
)
```

#### Visual C++

```cpp
public:
    void Start(
        IPAddress^ address,
        int port,
        X509Certificate^ certificate
    )
```

### Parameters

#### address

Type: `System.Net.IPAddress`

IP Address to listen on, use IPAddress.Any to accept connections on all ipaddresses/network cards.

#### port

Type: `System::Int32`

Port to listen on. 80 can be a good idea =)

#### certificate
Certificate to use
See Also

HttpServer Class
Start Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
shut down the server and listeners

**Namespace:**  [HttpServer](http://localhost:8080)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Stop()

Visual Basic (Declaration)

Public Sub Stop

Visual C++

public:
void Stop()
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
write an entry to the log file

**Namespace:**  [HttpServer](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void WriteLog(
    LogPrio prio,
    string message
)

Visual Basic (Declaration)

Protected Overridable Sub WriteLog ( _
    _prio As LogPrio, _
    _message As String _
)

Visual C++

protected:
virtual void WriteLog(
    LogPrio prio,
    String^ message
)

Parameters

prio
    Type: HttpServer::LogPrio
    importance of the message

message
    Type: System::String
    log message
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpServer` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AuthenticationModules</strong></td>
<td>Modules used for authentication. The module that is added first is used as the default auth module.</td>
</tr>
<tr>
<td>FormDecoderProviders</td>
<td>Form decoder providers are used to decode request body (which normally contains form data).</td>
</tr>
<tr>
<td>ServerName</td>
<td>Server name sent in http responses.</td>
</tr>
<tr>
<td>SessionCookieName</td>
<td>Name of cookie where session id is stored.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Modules used for authentication. The module that is added first is used as the default auth module.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IList<AuthModule> AuthenticationModules { get; }

Visual Basic (Declaration)

Public Readonly Property AuthenticationModules As IList(Of AuthModule)

Visual C++

public:
property IList<AuthModule>^ AuthenticationModules {
    IList<AuthModule>^ get ();
}
Remarks

Use the corresponding property in the WebSiteModule if you are using multiple websites.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Form decoder providers are used to decode request body (which normally contains form data).

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public FormDecoderProvider FormDecoderProviders { get; }
```

**Visual Basic (Declaration)**

```vbnet
Public ReadOnly Property FormDecoderProviders As FormDecoderProvider
```

**Visual C++**

```cpp
public:
    property FormDecoderProvider^ FormDecoderProviders {
        FormDecoderProvider^ get ();
    }
```
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Server name sent in http responses.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public string ServerName { get; set; }
```

**Visual Basic (Declaration)**

```vbnet
Public Property ServerName As String
```

**Visual C++**

```cpp
public:
    property String^ ServerName {
        String^ get ();
        void set (String^ value);
    }
```
Remarks

Do NOT include version in name, since it makes it easier for hackers.
See Also

HttpRequest Class
HttpRequest Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Name of cookie where session id is stored.

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public string SessionCookieName { get; set; }
```

**Visual Basic (Declaration)**

Public Property SessionCookieName As String

**Visual C++**

```cpp
public:
property String^ SessionCookieName {
    String^ get();
    void set (String^ value);
}
```
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpServer** type exposes the following members.
## Events

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExceptionThrown</td>
<td>Let's to receive unhandled exceptions from the threads.</td>
</tr>
<tr>
<td>LogEntryWritten</td>
<td>Use this event to be able to get log entries into your favorite log library.</td>
</tr>
<tr>
<td>RealmWanted</td>
<td>Realms are used during http authentication. Default realm is same as server name.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Let's receive unhandled exceptions from the threads.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public event ExceptionHandler ExceptionThrown

Visual Basic (Declaration)

Public Event ExceptionThrown As ExceptionHandler

Visual C++

public:
    event ExceptionHandler^ ExceptionThrown {
        void add (ExceptionHandler^ value);
        void remove (ExceptionHandler^ value);
    }
Remarks

Exceptions will be thrown during debug mode if this event is not used, exceptions will be printed to console and suppressed during release mode.
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Use this event to be able to get log entries into your favorite log library.

**Namespace:**  [HttpServer](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public event WriteLogHandler LogEntryWritten

Visual Basic (Declaration)

Public Event LogEntryWritten As WriteLogHandler

Visual C++

public:
    event WriteLogHandler^ LogEntryWritten {
        void add (WriteLogHandler^ value);
        void remove (WriteLogHandler^ value);
    }
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Realms are used during http authentication. Default realm is same as server name.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public event RealmHandler RealmWanted

Visual Basic (Declaration)

Public Event RealmWanted As RealmHandler

Visual C++

public:
    event RealmHandler^ RealmWanted {
        void add (RealmHandler^ value);
        void remove (RealmHandler^ value);
    }
See Also

HttpServer Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

HttpSession Interface

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public interface HttpSession : IDisposable
```

**Visual Basic (Declaration)**

```vbnet
Public Interface HttpSession Implements IDisposable
```

**Visual C++**

```cpp
public interface class HttpSession : IDisposable
```
See Also

 HttpSession Members
 HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpSession type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| Clear | Remove everything from the session  
Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.  
(Inherited from IDisposable.) |
| Dispose | |
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessed</td>
<td>When the session was last accessed. This property is touched by the http server each time the session is requested.</td>
</tr>
<tr>
<td>Count</td>
<td>Number of session variables.</td>
</tr>
<tr>
<td>Id</td>
<td>Session id</td>
</tr>
<tr>
<td>Item</td>
<td>Should</td>
</tr>
</tbody>
</table>
See Also

 HttpSession Interface
 HttpSession Namespace

 This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpSession type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear</td>
<td>Remove everything from the session</td>
</tr>
<tr>
<td>Dispose</td>
<td>Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)</td>
</tr>
</tbody>
</table>
See Also

 HttpSession Interface
 HttpSession Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
Remove everything from the session

**Namespace:**  HttpSession
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

void Clear()

Visual Basic (Declaration)

Sub Clear

Visual C++

void Clear()
See Also

HttpSession Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpSession type exposes the following members.
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessed</td>
<td>When the session was last accessed. This property is touched by the http server each time the session is requested.</td>
</tr>
<tr>
<td>Count</td>
<td>Number of session variables.</td>
</tr>
<tr>
<td>Id</td>
<td>Session id</td>
</tr>
<tr>
<td>Item</td>
<td>Should</td>
</tr>
</tbody>
</table>
See Also

HttpSession Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
When the session was last accessed. This property is touched by the http server each time the session is requested.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

DateTime Accessed { get; set; }

Visual Basic (Declaration)

Property Accessed As DateTime

Visual C++

property DateTime Accessed {
    DateTime get ();
    void set (DateTime value);
}
See Also

HttpSession Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Number of session variables.

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
int Count { get; }
```

Visual Basic (Declaration)

```vbnet
ReadOnly Property Count As Integer
```

Visual C++

```cpp
property int Count {
    int get ();
}
```
See Also

HttpSession Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
C# Webserver

HttpSession:::Id Property

HttpSession Interface  See Also  Send Feedback

Session id

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

string Id { get; }

Visual Basic (Declaration)

ReadOnly Property Id As String

Visual C++

property String^ Id {
    String^ get ();
}

See Also

HttpSession Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Should

Namespace:  [HttpServer](#)  
Assembly:  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
Object this[string name]
{
    get; set;
}
```

Visual Basic (Declaration)

Default Property Item ( _
    name As String _
) As Object

Visual C++

```cpp
property Object^ default[String^ name] {
    Object^ get (String^ name);
    void set (String^ name, Object^ value);
}
```

Parameters

name
    Type: System:::String
    Name of the session variable

Return Value

null if it's not set
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.Runtime.Serialization.SerializationException</code></td>
<td>If the object can't be serialized.</td>
</tr>
</tbody>
</table>
See Also

HttpSession Interface
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Priority for log entries

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public enum LogPrio

Visual Basic (Declaration)

Public Enumeration LogPrio

Visual C++

public enum class LogPrio
## Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trace</td>
<td>Very detailed logs to be able to follow the flow of the program.</td>
</tr>
<tr>
<td>Debug</td>
<td>Logs to help debug errors in the application</td>
</tr>
<tr>
<td>Info</td>
<td>Information to be able to keep track of state changes etc.</td>
</tr>
<tr>
<td>Warning</td>
<td>Something did not go as we expected, but it's no problem.</td>
</tr>
<tr>
<td>Error</td>
<td>Something that should not fail failed, but we can still keep on going.</td>
</tr>
<tr>
<td>Fatal</td>
<td>Something failed, and we cannot handle it properly.</td>
</tr>
</tbody>
</table>
See Also

HttpServer Namespace
HttpServer:::WriteLogHandler

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains all HTTP Methods (according to the HTTP 1.1 specification)

See: http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static class Method

Visual Basic (Declaration)

Public NotInheritable Class Method

Visual C++

public ref class Method abstract sealed
Inheritance Hierarchy

System:::Object
HttpServer:::Method
See Also

Method Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `Method` type exposes the following members.
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Delete</strong></td>
<td>The DELETE method requests that the origin server delete the resource identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Get</strong></td>
<td>The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Header</strong></td>
<td>The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.</td>
</tr>
<tr>
<td><strong>Options</strong></td>
<td>The OPTIONS method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Post</strong></td>
<td>The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.</td>
</tr>
<tr>
<td><strong>Put</strong></td>
<td>The PUT method requests that the enclosed entity be stored under the supplied Request-URI.</td>
</tr>
<tr>
<td><strong>Trace</strong></td>
<td>The TRACE method is used to invoke a remote, application-layer loop-back of the request message.</td>
</tr>
</tbody>
</table>
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Method** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete</td>
<td>The DELETE method requests that the origin server delete the resource identified by the Request-URI.</td>
</tr>
<tr>
<td>Get</td>
<td>The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.</td>
</tr>
<tr>
<td>Header</td>
<td>The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.</td>
</tr>
<tr>
<td>Options</td>
<td>The OPTIONS method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.</td>
</tr>
<tr>
<td>Post</td>
<td>The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.</td>
</tr>
<tr>
<td>Put</td>
<td>The PUT method requests that the enclosed entity be stored under the supplied Request-URI.</td>
</tr>
<tr>
<td>Trace</td>
<td>The TRACE method is used to invoke a remote, application-layer loop-back of the request message.</td>
</tr>
</tbody>
</table>
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The DELETE method requests that the origin server delete the resource identified by the Request-URI.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Delete

Visual Basic (Declaration)

Public Const Delete As String

Visual C++

public:
    literal String ^ Delete
Remarks

This method MAY be overridden by human intervention (or other means) on the origin server. The client cannot be guaranteed that the operation has been carried out, even if the status code returned from the origin server indicates that the action has been completed successfully.

However, the server SHOULD NOT indicate success unless, at the time the response is given, it intends to delete the resource or move it to an inaccessible location.

A successful response SHOULD be 200 (OK) if the response includes an entity describing the status, 202 (Accepted) if the action has not yet been enacted, or 204 (No Content) if the action has been enacted but the response does not include an entity.

If the request passes through a cache and the Request-URI identifies one or more currently cached entities, those entries SHOULD be treated as stale. Responses to this method are not cacheable.
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.

**Namespace:** [HttpServer](http://example.com/HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Get

Visual Basic (Declaration)

Public Const Get As String

Visual C++

public:
    literal String^ Get
**Remarks**

If the Request-URI refers to a data-producing process, it is the produced data which shall be returned as the entity in the response and not the source text of the process, unless that text happens to be the output of the process.

The semantics of the GET method change to a "conditional GET" if the request message includes an If-Modified-Since, If-Unmodified-Since, If-Match, If-None-Match, or If-Range header field. A conditional GET method requests that the entity be transferred only under the circumstances described by the conditional header field(s). The conditional GET method is intended to reduce unnecessary network usage by allowing cached entities to be refreshed without requiring multiple requests or transferring data already held by the client.
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Header

Visual Basic (Declaration)

Public Const Header As String

Visual C++

public:
literal String^ Header
Remarks

The metainformation contained in the HTTP headers in response to a HEAD request SHOULD be identical to the information sent in response to a GET request. This method can be used for obtaining metainformation about the entity implied by the request without transferring the entity-body itself. This method is often used for testing hypertext links for validity, accessibility, and recent modification.
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The OPTIONS method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.

**Namespace:** HttpServer  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Options

Visual Basic (Declaration)

Public Const Options As String

Visual C++

public:
literal String^ Options
Remarks

This method allows the client to determine the options and/or requirements associated with a resource, or the capabilities of a server, without implying a resource action or initiating a resource retrieval.
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Post

Visual Basic (Declaration)

Public Const Post As String

Visual C++

public:
    literal String^ Post
Remarks

POST is designed to allow a uniform method to cover the following functions:

- Annotation of existing resources;
- Posting a message to a bulletin board, newsgroup, mailing list, or similar group of articles;
- Providing a block of data, such as the result of submitting a form, to a data-handling process;
- Extending a database through an append operation.

If a resource has been created on the origin server, the response SHOULD be 201 (Created) and contain an entity which describes the status of the request and refers to the new resource, and a Location header (see section 14.30).

The action performed by the POST method might not result in a resource that can be identified by a URI. In this case, either 200 (OK) or 204 (No Content) is the appropriate response status, depending on whether or not the response includes an entity that describes the result.

Responses to this method are not cacheable, unless the response includes appropriate Cache-Control or Expires header fields. However, the 303 (See Other) response can be used to direct the user agent to retrieve a cacheable resource.
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The PUT method requests that the enclosed entity be stored under the supplied Request-URI.

**Namespace:**  [HttpServer](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Put

Visual Basic (Declaration)

Public Const Put As String

Visual C++

public:
literal String^ Put
Remarks

- If the Request-URI refers to an already existing resource, the enclosed entity SHOULD be considered as a modified version of the one residing on the origin server.
- If the Request-URI does not point to an existing resource, and that URI is capable of being defined as a new resource by the requesting user agent, the origin server can create the resource with that URI.
- If a new resource is created, the origin server MUST inform the user agent via the 201 (Created) response.
- If an existing resource is modified, either the 200 (OK) or 204 (No Content) response codes SHOULD be sent to indicate successful completion of the request.
- If the resource could not be created or modified with the Request-URI, an appropriate error response SHOULD be given that reflects the nature of the problem.

The recipient of the entity MUST NOT ignore any Content-* (e.g. Content-Range) headers that it does not understand or implement and MUST return a 501 (Not Implemented) response in such cases.
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TRACE method is used to invoke a remote, application-layer loop-back of the request message.

**Namespace:** HttpServer  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public const string Trace
```

**Visual Basic (Declaration)**

```vbnet
Public Const Trace As String
```

**Visual C++**

```cpp
public:
    literal String^ Trace
```
See Also

Method Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains all HTTP Methods (according to the HTTP 1.1 specification)

See: http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html

**Namespace:** HttpServer

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

[ObsoleteAttribute("Use Method.XXX instead.")] public static class Methods

Visual Basic (Declaration)

<ObsoleteAttribute("Use Method.XXX instead.")> _
Public NotInheritable Class Methods

Visual C++

[ObsoleteAttribute(L"Use Method.XXX instead.")] public ref class Methods abstract sealed
Inheritance Hierarchy

System:::Object
HttpServer:::Methods
See Also

Methods Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `Methods` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Delete</strong></td>
<td>The DELETE method requests that the origin server delete the resource identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Get</strong></td>
<td>The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Header</strong></td>
<td>The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.</td>
</tr>
<tr>
<td><strong>Options</strong></td>
<td>The OPTIONS method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Post</strong></td>
<td>The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.</td>
</tr>
<tr>
<td><strong>Put</strong></td>
<td>The PUT method requests that the enclosed entity be stored under the supplied Request-URI.</td>
</tr>
<tr>
<td><strong>Trace</strong></td>
<td>The TRACE method is used to invoke a remote, application-layer loop-back of the request message.</td>
</tr>
</tbody>
</table>
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Methods** type exposes the following members.
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Delete</strong></td>
<td>The DELETE method requests that the origin server delete the resource identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Get</strong></td>
<td>The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Header</strong></td>
<td>The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.</td>
</tr>
<tr>
<td><strong>Options</strong></td>
<td>The OPTIONS method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.</td>
</tr>
<tr>
<td><strong>Post</strong></td>
<td>The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.</td>
</tr>
<tr>
<td><strong>Put</strong></td>
<td>The PUT method requests that the enclosed entity be stored under the supplied Request-URI.</td>
</tr>
<tr>
<td><strong>Trace</strong></td>
<td>The TRACE method is used to invoke a remote, application-layer loop-back of the request message.</td>
</tr>
</tbody>
</table>
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The DELETE method requests that the origin server delete the resource identified by the Request-URI.

**Namespace:** [HttpServer](#)  
**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Delete

Visual Basic (Declaration)

Public Const Delete As String

Visual C++

public:
  literal String^ Delete
Remarks

This method MAY be overridden by human intervention (or other means) on the origin server. The client cannot be guaranteed that the operation has been carried out, even if the status code returned from the origin server indicates that the action has been completed successfully.

However, the server SHOULD NOT indicate success unless, at the time the response is given, it intends to delete the resource or move it to an inaccessible location.

A successful response SHOULD be 200 (OK) if the response includes an entity describing the status, 202 (Accepted) if the action has not yet been enacted, or 204 (No Content) if the action has been enacted but the response does not include an entity.

If the request passes through a cache and the Request-URI identifies one or more currently cached entities, those entries SHOULD be treated as stale. Responses to this method are not cacheable.
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI.

**Namespace:** `HttpServer`  
**Assembly:** `HttpServer` (in `HttpServer.dll`) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Get

Visual Basic (Declaration)

Public Const Get As String

Visual C++

public:
literal String^ Get
Remarks

If the Request-URI refers to a data-producing process, it is the produced data which shall be returned as the entity in the response and not the source text of the process, unless that text happens to be the output of the process.

The semantics of the GET method change to a "conditional GET" if the request message includes an If-Modified-Since, If-Unmodified-Since, If-Match, If-None-Match, or If-Range header field. A conditional GET method requests that the entity be transferred only under the circumstances described by the conditional header field(s). The conditional GET method is intended to reduce unnecessary network usage by allowing cached entities to be refreshed without requiring multiple requests or transferring data already held by the client.
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Header

Visual Basic (Declaration)

Public Const Header As String

Visual C++

public:
literal String Header
Remarks

The metainformation contained in the HTTP headers in response to a HEAD request SHOULD be identical to the information sent in response to a GET request. This method can be used for obtaining metainformation about the entity implied by the request without transferring the entity-body itself. This method is often used for testing hypertext links for validity, accessibility, and recent modification.
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The OPTIONS method represents a request for information about the communication options available on the request/response chain identified by the Request-URI.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Options

Visual Basic (Declaration)

Public Const Options As String

Visual C++

public:
literal String^ Options
Remarks

This method allows the client to determine the options and/or requirements associated with a resource, or the capabilities of a server, without implying a resource action or initiating a resource retrieval.
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.

**Namespace:** [HttpServer](http://example.com)

**Assembly:** [HttpServer](http://example.com) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public const string Post

Visual Basic (Declaration)
Public Const Post As String

Visual C++
public:
literal String^ Post
**Remarks**

POST is designed to allow a uniform method to cover the following functions:

- Annotation of existing resources;
- Posting a message to a bulletin board, newsgroup, mailing list, or similar group of articles;
- Providing a block of data, such as the result of submitting a form, to a data-handling process;
- Extending a database through an append operation.

If a resource has been created on the origin server, the response SHOULD be 201 (Created) and contain an entity which describes the status of the request and refers to the new resource, and a Location header (see section 14.30).

The action performed by the POST method might not result in a resource that can be identified by a URI. In this case, either 200 (OK) or 204 (No Content) is the appropriate response status, depending on whether or not the response includes an entity that describes the result.

Responses to this method are not cacheable, unless the response includes appropriate Cache-Control or Expires header fields. However, the 303 (See Other) response can be used to direct the user agent to retrieve a cacheable resource.
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The PUT method requests that the enclosed entity be stored under the supplied Request-URI.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Put

Visual Basic (Declaration)

Public Const Put As String

Visual C++

public:
literal String^ Put
Remarks

- If the Request-URI refers to an already existing resource, the enclosed entity SHOULD be considered as a modified version of the one residing on the origin server.
- If the Request-URI does not point to an existing resource, and that URI is capable of being defined as a new resource by the requesting user agent, the origin server can create the resource with that URI.
- If a new resource is created, the origin server MUST inform the user agent via the 201 (Created) response.
- If an existing resource is modified, either the 200 (OK) or 204 (No Content) response codes SHOULD be sent to indicate successful completion of the request.
- If the resource could not be created or modified with the Request-URI, an appropriate error response SHOULD be given that reflects the nature of the problem.

The recipient of the entity MUST NOT ignore any Content-* (e.g. Content-Range) headers that it does not understand or implement and MUST return a 501 (Not Implemented) response in such cases.
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TRACE method is used to invoke a remote, application-layer loop-back of the request message.

**Namespace:**  HttpServer

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string Trace

Visual Basic (Declaration)

Public Const Trace As String

Visual C++

public:
literal String^ Trace
See Also

Methods Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Delegate used to find a realm/domain.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public delegate string RealmHandler(
    string domain
)

Visual Basic (Declaration)

Public Delegate Function RealmHandler ( _
    domain As String _
) As String

Visual C++

public delegate String^ RealmHandler(
    String^ domain
)

Parameters

domain
  Type: System::String

Return Value
Remarks

Realms are used during http authentication
See Also

HttpServer Namespace
HttpServer.Authentication, ..., AuthModule
HttpServer.Authentication, ..., AuthenticationHandler

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
cookie sent by the client/browser

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class RequestCookie

Visual Basic (Declaration)

Public Class RequestCookie

Visual C++

public ref class RequestCookie
Inheritance Hierarchy

System::<Object
HttpServer::<RequestCookie
HttpServer::<ResponseCookie
See Also

RequestCookie Members
HttpServer Namespace
HttpServer:::ResponseCookie

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RequestCookie` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RequestCookie</code></td>
<td>Constructor.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Gets the cookie HTML representation. (Overrides <a href="#">Object</a>::<a href="#">ToString</a>.)</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>Name</td>
<td>Gets the cookie identifier.</td>
</tr>
<tr>
<td>Value</td>
<td>Cookie value. Set to null to remove cookie.</td>
</tr>
</tbody>
</table>
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Constructor.

**Namespace:**  [HttpServer](http://HttpServer)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RequestCookie(
    string id,
    string content
)

Visual Basic (Declaration)

Public Sub New (_
    id As String, _
    content As String _
)

Visual C++

public:
RequestCookie(
    String^ id,
    String^ content
)

Parameters

id
    Type: System::String
    cookie identifier

content
    Type: System::String
    cookie content
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td>id or content is null</td>
</tr>
<tr>
<td>System:::ArgumentException</td>
<td>id is empty</td>
</tr>
</tbody>
</table>
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The RequestCookie type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Gets the cookie HTML representation. (Overrides <a href="#">Object</a>::ToString().)</td>
</tr>
</tbody>
</table>
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestCookie..::.ToString Method

RequestCookie Class  See Also  Send Feedback

Gets the cookie HTML representation.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

public override string ToString()

**Visual Basic (Declaration)**

Public Overrides Function ToString As String

**Visual C++**

public:
virtual String^ ToString() override

**Return Value**

cookie string
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RequestCookie` type exposes the following members.
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Gets the cookie identifier.</td>
</tr>
<tr>
<td>Value</td>
<td>Cookie value. Set to null to remove cookie.</td>
</tr>
</tbody>
</table>
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestCookie.Name Property

Gets the cookie identifier.

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Name { get; }

Visual Basic (Declaration)

Public ReadOnly Property Name As String

Visual C++

public:
property String^ Name {
    String^ get ();
}
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestCookie..::.Value Property

Cookie value. Set to null to remove cookie.

Namespace: **HttpServer**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Value { get; set; }

Visual Basic (Declaration)

Public Property Value As String

Visual C++

public:
property String^ Value {
    String^ get ();
    void set (String^ value);
}
See Also

RequestCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This class is created as a wrapper, since there are two different cookie types in .Net (Cookie and HttpCookie). The framework might switch class in the future and we don't want to have to replace all instances.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public sealed class RequestCookies : IEnumerable<RequestCookie>, IEnumerable
```

**Visual Basic (Declaration)**

```vbnet
Public NotInheritable Class RequestCookies
    Implements IEnumerable(Of RequestCookie), IEnumerable
```

**Visual C++**

```cpp
public ref class RequestCookies sealed : IEnumerable<RequestCookie^>, IEnumerable
```
Inheritance Hierarchy

System...:::Object
HttpServer...:::RequestCookies
See Also

RequestCookies Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RequestCookies` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="image" /> RequestCookies</td>
<td>Let's copy all the cookies.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear</td>
<td>Remove all cookies.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>GetEnumerator</td>
<td>Gets a collection enumerator on the cookie list.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.                                               (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance.                                                   (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.                                           (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.                               (Inherited from <a href="#">Object</a>)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Count</strong></td>
<td>Gets the count of cookies in the collection.</td>
</tr>
<tr>
<td><strong>Item</strong></td>
<td>Gets the cookie of a given identifier (null if not existing).</td>
</tr>
</tbody>
</table>
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Let's copy all the cookies.

Namespace:  [HttpServer](http://example.com/HttpServer)
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public RequestCookies(
    string cookies
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    cookies As String _
)
```

**Visual C++**

```cpp
public:
RequestCookies(
    String^ cookies
)
```

**Parameters**

cookies

Type: `System::String`

value from cookie header.
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RequestCookies` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Clear</strong></td>
<td>Remove all cookies.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Gets a collection enumerator on the cookie list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
</tbody>
</table>
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Remove all cookies.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public void Clear()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Clear
```

**Visual C++**

```cpp
public:
void Clear()
```
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestCookies...:::GetEnumerator Method

**RequestCookies Class**  See Also  Send Feedback

Gets a collection enumerator on the cookie list.

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IEnumerator GetEnumerator()

Visual Basic (Declaration)

Public Function GetEnumerator As IEnumerable

Visual C++

public:
virtual IEnumerable^ GetEnumerator() sealed

Return Value

collection enumerator

Implements

IEnumerable::GetEnumerator()
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RequestCookies` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Gets the count of cookies in the collection.</td>
</tr>
<tr>
<td>Item</td>
<td>Gets the cookie of a given identifier (null if not existing).</td>
</tr>
</tbody>
</table>
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Gets the count of cookies in the collection.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int Count { get; }

Visual Basic (Declaration)

Public ReadOnly Property Count As Integer

Visual C++

public:
property int Count {
    int get ();
}

See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestCookies...:.Item Property

Namespace:  [HttpServer]
Assembly:  [HttpServer] (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RequestCookie this[int id] { get; }

Visual Basic (Declaration)

Public ReadOnly Default Property Item ( _
    id As String _
) As RequestCookie

Visual C++

public:
    property RequestCookie^ default[String^ id] { 
        RequestCookie^ get (String^ id); 
    }

Parameters

id
    Type: System:::String
See Also

RequestCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Invoked when a client context have received a new HTTP request

**Namespace:**  [HttpServer](http://www.example.com)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public delegate void RequestReceivedHandler(
    HttpClientContext client,
    HttpRequest request
)

Visual Basic (Declaration)

Public Delegate Sub RequestReceivedHandler ( _
    client As HttpClientContext, _
    request As HttpRequest _
)

Visual C++

public delegate void RequestReceivedHandler(
    HttpClientContext^ client,
    HttpRequest^ request
)

Parameters

client
    Type: HttpServer::HttpClientContext
    Client that received the request.

request
    Type: HttpServer::HttpRequest
    Request that was received.
See Also

HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
cookie being sent back to the browser.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ResponseCookie : RequestCookie

Visual Basic (Declaration)

Public Class ResponseCookie
    Inherits RequestCookie

Visual C++

public ref class ResponseCookie : public RequestCookie
Inheritance Hierarchy

System...:::Object
HttpServer...:::RequestCookie
HttpServer...:::ResponseCookie
See Also

ResponseCookie Members
HttpServer Namespace
HttpServer.....ResponseCookie

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ResponseCookie** type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseCookie</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Gets the cookie HTML representation.</td>
</tr>
<tr>
<td></td>
<td>(Overrides <a href="#">RequestCookie</a>::ToString()).</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expires</td>
<td>When the cookie expires. DateTime.MinValue means that the cookie expires when the session does so.</td>
</tr>
<tr>
<td>Name</td>
<td>Gets the cookie identifier. (Inherited from RequestCookie.)</td>
</tr>
<tr>
<td>Path</td>
<td>Cookie is only valid under this path.</td>
</tr>
<tr>
<td>Value</td>
<td>Cookie value. Set to null to remove cookie. (Inherited from RequestCookie.)</td>
</tr>
</tbody>
</table>
See Also

ResponseCookie Class
HttpRequest Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookie Constructor

**ResponseCookie Class**  **See Also**  **Send Feedback**
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ResponseCookie(RequestCookie, DateTime)</code></td>
<td>Create a new cookie</td>
</tr>
<tr>
<td><code>ResponseCookie(String, String, DateTime)</code></td>
<td>Constructor.</td>
</tr>
<tr>
<td><code>ResponseCookie(String, String, DateTime, String, String)</code></td>
<td>Create a new cookie</td>
</tr>
</tbody>
</table>
See Also

ResponseCookie Class
ResponseCookie Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new cookie

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll)  Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public ResponseCookie(
    RequestCookie cookie,
    DateTime expires
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New (
    cookie As RequestCookie, 
    expires As DateTime
)
```

Visual C++

```cpp
public:
ResponseCookie(
    RequestCookie^ cookie,
    DateTime expires
)
```

Parameters

cookie
Type: `HttpServer:::RequestCookie`
Name and value will be used

expires
Type: `System:::DateTime`
when the cookie expires.
See Also

ResponseCookie Class
ResponseCookie Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookie Constructor (String, String, DateTime)

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public ResponseCookie(
    string id,
    string content,
    DateTime expiresAt)
```

Visual Basic (Declaration)

```vbnet
Public Sub New ( _
    id As String, _
    content As String, _
    expiresAt As DateTime _
)
```

Visual C++

```cpp
public: 
ResponseCookie(
    String^ id,
    String^ content,
    DateTime expiresAt
)
```

Parameters

id
Type: **System::String**
cookie identifier

content
Type: **System::String**
cookie content

expiresAt
Type: **System::DateTime**
cookie expiration date. Use DateTime.MinValue for session cookie.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System::ArgumentNullException</td>
<td>id or content is null</td>
</tr>
<tr>
<td>System::ArgumentException</td>
<td>id is empty</td>
</tr>
</tbody>
</table>
See Also

ResponseCookie Class
ResponseCookie Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
ResponseCookie Constructor (String, String, DateTime, String, String)
ResponseCookie Class  See Also  Send Feedback

Create a new cookie

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ResponseCookie(
    string name,
    string value,
    DateTime expires,
    string path,
    string domain
)

Visual Basic (Declaration)

Public Sub New (_
    name As String, _
    value As String, _
    expires As DateTime, _
    path As String, _
    domain As String _
)

Visual C++

public:
ResponseCookie(
    String^ name,
    String^ value,
    DateTime expires,
    String^ path,
    String^ domain
)

Parameters

name
    Type: System::String
    name identifying the cookie

value
Type: **System.String**
cookie value

expires
Type: **System.DateTime**
when the cookie expires. Setting DateTime.MinValue will delete the cookie when the session is closed.

path
Type: **System.String**
Path to where the cookie is valid

domain
Type: **System.String**
Domain that the cookie is valid for.
See Also

ResponseCookie Class
ResponseCookie Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ResponseCookie` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Gets the cookie HTML representation. (Overrides RequestCookie...::ToString().)</td>
</tr>
</tbody>
</table>
See Also

ResponseCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookie...::ToString Method

Gets the cookie HTML representation.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public override string ToString()
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function ToString As String
```

**Visual C++**

```cpp
public:
virtual String^ ToString() override
```

**ReturnValue**

cookie string
See Also

ResponseCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ResponseCookie` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expires</td>
<td>When the cookie expires. DateTime.MinValue means that the cookie expires when the session do so.</td>
</tr>
<tr>
<td>Name</td>
<td>Gets the cookie identifier. (Inherited from RequestCookie.)</td>
</tr>
<tr>
<td>Path</td>
<td>Cookie is only valid under this path.</td>
</tr>
<tr>
<td>Value</td>
<td>Cookie value. Set to null to remove cookie. (Inherited from RequestCookie.)</td>
</tr>
</tbody>
</table>
See Also

ResponseCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookie...::Expires Property

When the cookie expires. DateTime.MinValue means that the cookie expires when the session do so.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public DateTime Expires { get; set; }

Visual Basic (Declaration)

Public Property Expires As DateTime

Visual C++

public:
property DateTime Expires {
    DateTime get ();
    void set (DateTime value);
}
See Also

ResponseCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Cookie is only valid under this path.

Namespace:  HttpServer
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Path { get; set; }

Visual Basic (Declaration)

Public Property Path As String

Visual C++

public:
property String^ Path {
    String^ get ();
    void set (String^ value);
}
See Also

ResponseCookie Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Cookies that should be set.

**Namespace:** [HttpServer](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public sealed class ResponseCookies : IEnumerable<ResponseCookie>, IEnumerable

Visual Basic (Declaration)

Public NotInheritable Class ResponseCookies _
    Implements IEnumerable(Of ResponseCookie), IEnumerable

Visual C++

public ref class ResponseCookies sealed : IEnumerable<ResponseCookie>, IEnumerable

Inheritance Hierarchy

System::Object
   HttpServer::ResponseCookies
See Also

ResponseCookies Members
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

ResponseCookies Members

ResponseCookies Class  Constructors  Methods  Properties  See Also  Send Feedback

The ResponseCookies type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseCookies</td>
<td>Initializes a new instance of the ResponseCookies class</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Add</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Clear</td>
<td>Remove all cookies</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetEnumerator</td>
<td>Gets a collection enumerator on the cookie list.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Gets the count of cookies in the collection.</td>
</tr>
<tr>
<td>Item</td>
<td>Gets the cookie of a given identifier (null if not existing).</td>
</tr>
</tbody>
</table>
See Also

ResponseCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `ResponseCookies` class

**Namespace:**  [HttpServer](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ResponseCookies()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
ResponseCookies()
The `ResponseCookies` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Clear</strong></td>
<td>Remove all cookies. Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetEnumerator</strong></td>
<td>Gets a collection enumerator on the cookie list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
See Also

ResponseCookies Class  
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  
Visual C++  
Include Protected Members  
Include Inherited Members  
C# Webserver  
ResponseCookies...:::Add Method  
ResponseCookies Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add(ResponseCookie)</td>
<td>Adds a cookie in the collection.</td>
</tr>
<tr>
<td>Add(RequestCookie, DateTime)</td>
<td>Copy a request cookie</td>
</tr>
</tbody>
</table>
See Also

ResponseCookies Class
ResponseCookies Members
HttpContext Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add Method (ResponseCookie)

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Add(
    ResponseCookie cookie
)

Visual Basic (Declaration)

Public Sub Add ( _
    cookie As ResponseCookie _
)

Visual C++

public:
void Add(
    ResponseCookie^ cookie
)

Parameters

cookie
Type: HttpServer:::ResponseCookie
cookie to add
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td>cookie is null</td>
</tr>
</tbody>
</table>
See Also

ResponseCookies Class
Add Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Copy a request cookie

**Namespace:** [HttpServer](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void Add(
    RequestCookie cookie,
    DateTime expires
)
```

Visual Basic (Declaration)

```vbnet
Public Sub Add (_
    cookie As RequestCookie, _
    expires As DateTime _
)
```

Visual C++

```cpp
public:
void Add(
    RequestCookie^ cookie,
    DateTime expires
)
```

Parameters

cookie
Type: `HttpServer::RequestCookie`


expires
Type: `System::DateTime`
When the cookie should expire
See Also

ResponseCookies Class
Add Overload
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookies...:::Clear Method

Namespace: HttpServer
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public void Clear()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Clear
```

**Visual C++**

```cpp
public:
void Clear()
```
See Also

ResponseCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Gets a collection enumerator on the cookie list.

**Namespace:** [HttpServer](#)  
**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IEnumarator GetEnumerator()

Visual Basic (Declaration)

Public Function GetEnumerator As IEnumarator

Visual C++

public:
virtual IEnumarator^ GetEnumerator() sealed

ReturnValue

collection enumerator

Implements

IEnumerator,::GetEnumerator()()
See Also

ResponseCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ResponseCookies` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Gets the count of cookies in the collection.</td>
</tr>
<tr>
<td>Item</td>
<td>Gets the cookie of a given identifier (null if not existing).</td>
</tr>
</tbody>
</table>
See Also

ResponseCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookies.Count Property

Gets the count of cookies in the collection.

**Namespace:** [HttpServer](http://www.httpserver.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public int Count { get; }

**Visual Basic (Declaration)**

Public ReadOnly Property Count As Integer

**Visual C++**

public:
property int Count {
    int get ();
}

See Also

ResponseCookies Class
HttpServer Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ResponseCookies..:::Item Property

Gets the cookie of a given identifier (null if not existing).

**Namespace:** [HttpServer](http://example.com/HttpServer)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
```csharp
public ResponseCookie this[string id] { get; set; }
```

**Visual Basic (Declaration)**

Public Default Property Item ( _
    id As String _
) As ResponseCookie

**Visual C++**

public:
property ResponseCookie^ default[String^ id] {
    ResponseCookie^ get (String^ id);
    void set (String^ id, ResponseCookie^ value);
}

**Parameters**

id

Type: System::String
**See Also**

- [ResponseCookies Class](#)
- [HttpServer Namespace](#)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A delegate that helps us keep track of errors in the system.

**Namespace:**  [HttpServer](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public delegate void WriteLogHandler(
    Object source,
    LogPrio priority,
    string message
)

Visual Basic (Declaration)

Public Delegate Sub WriteLogHandler ( _
    source As Object, _
    priority As LogPrio, _
    message As String _
)

Visual C++

public delegate void WriteLogHandler(
    Object^ source,
    LogPrio priority,
    String^ message
)

Parameters

source
    Type: System::Object
    class that writes the log entry.

priority
    Type: HttpServer::LogPrio
    priority for the message

message
    Type: System::String
    log message
See Also

HttpServer Namespace
HttpServer:::ConsoleLogWriter

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.Authentication]
### Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthModule</td>
<td>Authentication modules are used to implement different kind of http authentication. The &quot;basic&quot; authentication scheme is based on the model that the client must authenticate itself with a user-ID and a password for each realm. The realm value should be considered an opaque string which can only be compared for equality with other realms on that server. The server will service the request only if it can validate the user-ID and password for the protection space of the Request-URI. There are no optional authentication parameters.</td>
</tr>
<tr>
<td>BasicAuthentication</td>
<td>Implements HTTP Digest authentication. It's more secure than Basic auth since password is encrypted with a &quot;key&quot; from the server.</td>
</tr>
<tr>
<td>DigestAuthentication</td>
<td></td>
</tr>
</tbody>
</table>
# Delegates

<table>
<thead>
<tr>
<th>Delegate</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthenticationHandler</td>
<td>Delegate used to let authentication modules authenticate the username and password. Let's you decide on a system level if authentication is required. You can also decide if authentication is required in each HttpModule.</td>
</tr>
<tr>
<td>AuthRequiredDelegate</td>
<td></td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Delegate used to let authentication modules authenticate the username and password.

Namespace:  [HttpServer.Authentication](#)
Assembly:  [HttpServer](#) (in [HttpServer.dll](#))  Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public delegate void AuthenticationHandler(
    string realm,
    string userName,
    ref string password,
    out Object login
)

Visual Basic (Declaration)

Public Delegate Sub AuthenticationHandler ( _
    realm As String, _
    userName As String, _
    ByRef password As String, _
    <OutAttribute> ByRef login As Object _
)

Visual C++

public delegate void AuthenticationHandler(
    String^ realm,
    String^ userName,
    String^% password,
    [OutAttribute] Object^% login
)

Parameters

realm
  Type: System::::String
  Realm that the user want to authenticate in

userName
  Type: System::::String
  Username specified by client

password
Type: `System::String` %
Password supplied by the delegete

login
Type: `System::Object` %
object that will be stored in a session variable called AuthenticationTag()() if authentication was successful.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpRequest.Exceptions..ForbiddenException</td>
<td>throw forbidden exception if too many attempts have been made.</td>
</tr>
</tbody>
</table>
See Also

HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Authentication modules are used to implement different kind of http authentication.

**Namespace:**  [HttpServer.Authentication](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract class AuthModule

Visual Basic (Declaration)

Public MustInherit Class AuthModule

Visual C++

public ref class AuthModule abstract
Inheritance Hierarchy

`System..:::Object`
HttpServer.Authentication..:::AuthModule
  `HttpServer.Authentication..:::BasicAuthentication`
  `HttpServer.Authentication..:::DigestAuthentication`
See Also

[AuthModule Members]
[HttpServer.Authentication Namespace]

This project is maintained by Gauffin Telecom AB

[c) 2008 Gauffin Telecom AB
The AuthModule type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthModule</td>
<td>Initializes a new instance of the AuthModule class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th><strong>Name</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Authenticate</td>
<td>An authentication response have been received from the web browser. Check if it's correct</td>
</tr>
<tr>
<td>AuthenticationRequired</td>
<td>Determines if authentication is required.</td>
</tr>
<tr>
<td>CheckAuthentication</td>
<td>Used to invoke the auth delegate that is used to lookup the username/realm.</td>
</tr>
<tr>
<td>CreateResponse</td>
<td>Create a response that can be sent in the WWW-Authenticate header.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name used in http request.</td>
</tr>
</tbody>
</table>
# Events

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnAuthenticate</td>
<td>used to let authentication modules authenticate the username and password</td>
</tr>
<tr>
<td></td>
<td>Let's you decide on a system level if authentication is required. You can also decide if authentication is required in each HttpModule.</td>
</tr>
</tbody>
</table>
See Also

AuthModule Class
HttpServer.Authentication Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AuthModule Constructor

Namespace: HttpServer.Authentication
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected AuthModule()

Visual Basic (Declaration)

Protected Sub New

Visual C++

protected:
AuthModule()
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AuthModule` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authenticate</td>
<td>An authentication response have been received from the web browser. Check if it's correct</td>
</tr>
<tr>
<td>AuthenticationRequired</td>
<td>Determines if authentication is required.</td>
</tr>
<tr>
<td>CheckAuthentication</td>
<td>Used to invoke the auth delegate that is used to lookup the username/realm.</td>
</tr>
<tr>
<td>CreateResponse</td>
<td>Create a response that can be sent in the WWW-Authenticate header.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
An authentication response have been received from the web browser. Check if it's correct

**Namespace:**  [HttpServer.Authentication](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public abstract Object Authenticate(
    string authenticationHeader,
    string realm,
    string httpVerb,
    params Object[] options
)
```

Visual Basic (Declaration)

```vbnet
Public MustOverride Function Authenticate (_
    authenticationHeader As String, _
    realm As String, _
    httpVerb As String, _
    ParamArray options As Object()) _
) As Object
```

Visual C++

```cpp
public:
    virtual Object^ Authenticate(
        String^ authenticationHeader,
        String^ realm,
        String^ httpVerb,
        ... array<Object^>^ options
    ) abstract
```

Parameters

authenticationHeader
    Type: System::String
    Contents from the Authorization header

realm
    Type: System::String
    Realm that should be authenticated
httpVerb
Type: System::String
GET/POST/PUT/DELETE etc.

options
Type: array<System::Object>[][]
options to specific implementations

Return Value

Authentication object that is stored for the request. A user class or something like that.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System::: ArgumentException</td>
<td>if authenticationHeader is invalid</td>
</tr>
<tr>
<td>System::: ArgumentNullException</td>
<td>If any of the parameters is empty or null.</td>
</tr>
</tbody>
</table>
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines if authentication is required.

**Namespace:**  [HttpServer.Authentication](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool AuthenticationRequired(
    HttpRequest request
)

Visual Basic (Declaration)

Public Function AuthenticationRequired ( _
    request As HttpRequest _
) As Boolean

Visual C++

public:
bool AuthenticationRequired(    
    HttpRequest^ request
)

Parameters

request
    Type: HttpRequest
    Http request from browser

Return Value

true if user should be authenticated.
Remarks

throw ForbiddenException from your delegate if no more attempts are allowed.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Exceptions...:ForbiddenException</code></td>
<td>If no more attempts are allowed</td>
</tr>
</tbody>
</table>
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Used to invoke the auth delegate that is used to lookup the username/realm.

**Namespace:**  [HttpServer.Authentication](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected bool CheckAuthentication(
    string realm,
    string userName,
    ref string password,
    out Object login
)

Visual Basic (Declaration)

Protected Function CheckAuthentication ( _
    realm As String, _
    userName As String, _
    ByRef password As String, _
    <OutAttribute> ByRef login As Object _
) As Boolean

Visual C++

protected:
bool CheckAuthentication( 
    String^ realm, 
    String^ userName, 
    String^% password, 
    [OutAttribute] Object^% login
)

Parameters

realm
Type: System::::String
Realm (domain) that user want to authenticate in

userName
Type: System::::String
Username
password
    Type: System.String
    Password used for validation. Some implementations got password in clear text, they are then sent to client.

login
    Type: System.Object
    object that will be stored in the request to help you identify the user if authentication was successful.

Return Value

true if authentication was successful
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a response that can be sent in the WWW-Authenticate header.

**Namespace:** [HttpServer.Authentication](http://example.com/namespaces/HttpServer.Authentication)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract string CreateResponse(
    string realm,
    params Object[] options
)

Visual Basic (Declaration)

Public MustOverride Function CreateResponse ( _
    realm As String, _
    ParamArray options As Object() _
) As String

Visual C++

public:
virtual String^ CreateResponse( 
    String^ realm,
    ... array<Object^>[]^ options
) abstract

Parameters

realm
    Type: System::String
    Realm that the user should authenticate in

options
    Type: array< System::Object >[]
    Array with optional options.

Return Value

A correct auth request.
Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td>If realm is empty or null.</td>
</tr>
</tbody>
</table>
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **AuthModule** type exposes the following members.
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name used in http request.</td>
</tr>
</tbody>
</table>
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
name used in http request.

Namespace:  [HttpGetServer, Authentication]
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

    public abstract string Name { get; }

Visual Basic (Declaration)

    Public MustOverride ReadOnly Property Name As String

Visual C++

    public:
    virtual property String^ Name {
        String^ get () abstract;
    }

See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **AuthModule** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnAuthenticate</strong></td>
<td>used to let authentication modules authenticate the username and password Let's you decide on a system level if authentication is required. You can also decide if authentication is required in each HttpModule.</td>
</tr>
<tr>
<td><strong>OnAuthenticationRequired</strong></td>
<td></td>
</tr>
</tbody>
</table>

See Also

AuthModule Class
HttpServer.Authentication Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
used to let authentication modules authenticate the username and password

**Namespace:**  [HttpServer.Authentication](#)

**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
C#

public event AuthenticationHandler OnAuthenticate

Visual Basic (Declaration)

Public Event OnAuthenticate As AuthenticationHandler

Visual C++

public:
    event AuthenticationHandler^ OnAuthenticate {
        void add (AuthenticationHandler^ value);
        void remove (AuthenticationHandler^ value);
    }
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AuthModule...:::OnAuthenticationRequired Event

AuthModule Class  See Also  Send Feedback

Let's you decide on a system level if authentication is required. You can also decide if authentication is required in each HttpModule.

Namespace:  HttpServer.Authentication
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public event AuthRequiredDelegate OnAuthenticationRequired
```

**Visual Basic (Declaration)**

```vbnet
Public Event OnAuthenticationRequired As AuthRequiredDelegate
```

**Visual C++**

```cpp
public:
    event AuthRequiredDelegate^ OnAuthenticationRequired {
        void add (AuthRequiredDelegate^ value);
        void remove (AuthRequiredDelegate^ value);
    }
```
See Also

AuthModule Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Let's you decide on a system level if authentication is required. You can also decide if authentication is required in each HttpModule.

**Namespace:**  [HttpServer.Authentication](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public delegate bool AuthRequiredDelegate(
    HttpRequest request
)

Visual Basic (Declaration)

Public Delegate Function AuthRequiredDelegate ( _
    request As HttpRequest _
) As Boolean

Visual C++

public delegate bool AuthRequiredDelegate( 
    HttpRequest^ request
)

Parameters

request
    Type: HttpServer::::HttpRequest
    Http request from client

Return Value

true if user should be authenticated.
Remarks

throw ForbiddenException if no more attempts are allowed.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Exceptions...::ForbiddenException</code></td>
<td>If no more attempts are allowed</td>
</tr>
</tbody>
</table>
See Also

HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The "basic" authentication scheme is based on the model that the client must authenticate itself with a user-ID and a password for each realm. The realm value should be considered an opaque string which can only be compared for equality with other realms on that server. The server will service the request only if it can validate the user-ID and password for the protection space of the Request-URI. There are no optional authentication parameters.

Namespace: HttpServer.Authentication
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class BasicAuthentication : AuthModule

Visual Basic (Declaration)

Public Class BasicAuthentication _
    Inherits AuthModule

Visual C++

public ref class BasicAuthentication : public AuthModule
Inheritance Hierarchy

System...:::Object
HttpServer.Authentication...:::AuthModule
  HttpServer.Authentication...:::BasicAuthentication
See Also

BasicAuthentication Members
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The BasicAuthentication type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BasicAuthentication</td>
<td>Initializes a new instance of the BasicAuthentication class</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Authenticate</td>
<td>An authentication response have been received from the web browser. Check if it's correct (Overrides AuthModule:::Authenticate(String, String, String, array&lt;Object&gt;[]))</td>
</tr>
<tr>
<td>AuthenticationRequired</td>
<td>Determines if authentication is required. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td>CheckAuthentication</td>
<td>Used to invoke the auth delegate that is used to lookup the username/realm. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td>CreateResponse</td>
<td>Create a response that can be sent in the WWW-Authenticate header. (Overrides AuthModule:::CreateResponse(String, array&lt;Object&gt;[]))</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name used in http request. (Overrides AuthModule::&lt;Name&gt;.)</td>
</tr>
</tbody>
</table>
### Events

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnAuthenticate</td>
<td>used to let authentication modules authenticate the username and password</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">AuthModule</a>.)</td>
</tr>
<tr>
<td><strong>OnAuthenticationRequired</strong></td>
<td>Let's you decide on a system level if authentication is required. You can also</td>
</tr>
<tr>
<td></td>
<td>decide if authentication is required in each HttpModule.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">AuthModule</a>.)</td>
</tr>
</tbody>
</table>
See Also

BasicAuthentication Class
HttpRequests.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver
BasicAuthentication Constructor

Initializes a new instance of the BasicAuthentication class

Namespace: HttpServer.Authentication
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public BasicAuthentication()

**Visual Basic (Declaration)**

Public Sub New

**Visual C++**

public:
BasicAuthentication()
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `BasicAuthentication` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authenticate</td>
<td>An authentication response have been received from the web browser. Check if it's correct (Overrides AuthModule::&lt;AuthModule/Authenticate(String, String, String, array&lt;Object&gt;[][]) )</td>
</tr>
<tr>
<td>AuthenticationRequired</td>
<td>Determines if authentication is required. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td>CheckAuthentication</td>
<td>Used to invoke the auth delegate that is used to lookup the username/realm. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td>CreateResponse</td>
<td>Create a response that can be sent in the WWW-Authenticate header. (Overrides AuthModule::&lt;AuthModule/CreateResponse(String, array&lt;Object&gt;[][]) )</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
An authentication response have been received from the web browser. Check if it's correct

**Namespace:** [HttpServer.Authentication](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public override Object Authenticate(
    string authenticationHeader,
    string realm,
    string httpVerb,
    params Object[] options
)
```

#### Visual Basic (Declaration)

```vbnet
Public Overrides Function Authenticate ( _
    authenticationHeader As String, _
    realm As String, _
    httpVerb As String, _
    ParamArray options As Object() _
) As Object
```

#### Visual C++

```cpp
public:
    virtual Object^ Authenticate(
        String^ authenticationHeader,
        String^ realm,
        String^ httpVerb,
        ... array<Object^>^ options
    ) override
```

### Parameters

**authenticationHeader**
- Type: `System::::String`
- Contents from the Authorization header

**realm**
- Type: `System::::String`
- Realm that should be authenticated
httpVerb
  Type: `System::String`
  GET/POST/PUT/DELETE etc.

options
  Type: array< `System::Object` >[](){}
  Not used in basic auth

**Return Value**

Authentication object that is stored for the request. A user class or something like that.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentException</td>
<td>if authenticationHeader is invalid</td>
</tr>
<tr>
<td>System:::ArgumentNullException</td>
<td>If any of the parameters is empty or null.</td>
</tr>
</tbody>
</table>
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a response that can be sent in the WWW-Authenticate header.

**Namespace:**  [HttpServer.Authentication](http://HttpServer.Authentication)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public override string CreateResponse(
    string realm,
    params Object[] options
)
```

### Visual Basic (Declaration)

```vbnet
Public Overrides Function CreateResponse ( _
    realm As String, _
    ParamArray options As Object() _
) As String
```

### Visual C++

```cpp
public:
virtual String^ CreateResponse( 
    String^ realm,
    ... array<Object^>^ options
) override
```

## Parameters

**realm**
Type: System::::String
Realm that the user should authenticate in

**options**
Type: array< System::::Object >[]( )
Not used in basic auth

## Return Value

A correct auth request.
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **BasicAuthentication** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name used in http request.</td>
</tr>
<tr>
<td>(Overrides AuthModule...:..Name.)</td>
<td></td>
</tr>
</tbody>
</table>
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
BasicAuthentication.Name Property

Namespace:  **HttpServer.Authentication**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override string Name { get; }
```

Visual Basic (Declaration)

```vbnet
Public Overrides ReadOnly Property Name As String
```

Visual C++

```cpp
public:
virtual property String^ Name {
    String^ get () override;
}
```
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **BasicAuthentication** type exposes the following members.
## Events

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnAuthenticate</strong></td>
<td>used to let authentication modules authenticate the username and password</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">AuthModule</a>).</td>
</tr>
<tr>
<td></td>
<td>Let's you decide on a system level if authentication is required. You can</td>
</tr>
<tr>
<td></td>
<td>also decide if authentication is required in each HttpModule. (Inherited</td>
</tr>
<tr>
<td></td>
<td>from <a href="#">AuthModule</a>).</td>
</tr>
<tr>
<td><strong>OnAuthenticationRequired</strong></td>
<td></td>
</tr>
</tbody>
</table>
See Also

BasicAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DigestAuthentication Class

Implements HTTP Digest authentication. It's more secure than Basic auth since password is encrypted with a "key" from the server.

Namespace: HttpServer.Authentication
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class DigestAuthentication : AuthModule

Visual Basic (Declaration)

Public Class DigestAuthentication _
    Inherits AuthModule

Visual C++

public ref class DigestAuthentication : public AuthModule
Remarks

Keep in mind that the password is encrypted with MD5. Use a combination of SSL and digest auth to be secure.
Inheritance Hierarchy

System::Object
HttpServer.Authentication::AuthModule
HttpServer.Authentication::DigestAuthentication
See Also

DigestAuthentication Members
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The DigestAuthentication type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DigestAuthentication</td>
<td>Initializes a new instance of the DigestAuthentication class</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Authenticate</strong></td>
<td>An authentication response have been received from the web browser. Check if it's correct (Overrides AuthModule:::Authenticate(String, String, String, array&lt;Object&gt;[][]).).</td>
</tr>
<tr>
<td><strong>AuthenticationRequired</strong></td>
<td>Determines if authentication is required. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td><strong>CheckAuthentication</strong></td>
<td>Used to invoke the auth delegate that is used to lookup the username/realm. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td><strong>CreateResponse</strong></td>
<td>Create a response that can be sent in the WWW-Authenticate header. (Overrides AuthModule:::CreateResponse(String, array&lt;Object&gt;[][]).)</td>
</tr>
<tr>
<td><strong>Decode</strong></td>
<td>Decodes authorization header value</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetCurrentNonce</strong></td>
<td>Gets the current nonce.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetMD5HashBinHex2</strong></td>
<td>Gets the Md5 hash bin hex2.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>IsValidNonce</strong></td>
<td>determines if the nonce is valid or has expired.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>DisableNonceCheck</td>
<td>Used by test classes to be able to use hardcoded values</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Image](72x674 to 78x674)</td>
<td>Name used in http request. (Overrides AuthModule:::.Name.)</td>
</tr>
</tbody>
</table>
## Events

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnAuthenticate</strong></td>
<td>used to let authentication modules authenticate the username and password</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">AuthModule</a>.)</td>
</tr>
<tr>
<td></td>
<td>Let's you decide on a system level if authentication is required. You can also</td>
</tr>
<tr>
<td><strong>OnAuthenticationRequired</strong></td>
<td>decide if authentication is required in each HttpModule.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">AuthModule</a>.)</td>
</tr>
</tbody>
</table>
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  C# Webserver

DigestAuthentication Constructor

DigestAuthentication Class  See Also  Send Feedback

Initializes a new instance of the DigestAuthentication class

Namespace:  HttpServer.Authentication
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public DigestAuthentication()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
DigestAuthentication()
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `DigestAuthentication` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🍃 DisableNonceCheck</td>
<td>Used by test classes to be able to use hardcoded values</td>
</tr>
</tbody>
</table>
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
DigestAuthentication::DisableNonceCheck Field
DigestAuthentication Class  See Also  Send Feedback

Used by test classes to be able to use hardcoded values

Namespace:  HttpServer.Authentication
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static bool DisableNonceCheck

Visual Basic (Declaration)

Public Shared DisableNonceCheck As Boolean

Visual C++

public:
static bool DisableNonceCheck
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `DigestAuthentication` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authenticate</td>
<td>An authentication response have been received from the web browser. Check if it's correct. (Overrides AuthModule:::Authenticate(String, String, String, array&lt;Object&gt;[].))</td>
</tr>
<tr>
<td>AuthenticationRequired</td>
<td>Determines if authentication is required. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td>CheckAuthentication</td>
<td>Used to invoke the auth delegate that is used to lookup the username/realm. (Inherited from AuthModule.)</td>
</tr>
<tr>
<td>CreateResponse</td>
<td>Create a response that can be sent in the WWW-Authenticate header. (Overrides AuthModule:::CreateResponse(String, array&lt;Object&gt;[].))</td>
</tr>
<tr>
<td>Decode</td>
<td>Decodes authorization header value</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetCurrentNonce</td>
<td>Gets the current nonce.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetMD5HashBinHex2</td>
<td>Gets the Md5 hash bin hex2.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>IsValidNonce</td>
<td>determines if the nonce is valid or has expired.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>.</td>
</tr>
</tbody>
</table>

(Inherited from **Object**.)
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
An authentication response have been received from the web browser. Check if it's correct

**Namespace:**  [HttpServer.Authentication](#)

**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public override Object Authenticate(
    string authenticationHeader,
    string realm,
    string httpVerb,
    params Object[] options
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function Authenticate ( _
    authenticationHeader As String, _
    realm As String, _
    httpVerb As String, _
    ParamArray options As Object() _
) As Object
```

**Visual C++**

```cpp
public:
virtual Object^ Authenticate(
    String^ authenticationHeader, 
    String^ realm, 
    String^ httpVerb, 
    ... array<Object^>()^ options
) override
```

### Parameters

authenticatorHeader

- **Type:** `System::::String`
- Contents from the Authorization header

realm

- **Type:** `System::::String`
- Realm that should be authenticated
httpVerb
Type: System::String
GET/POST/PUT/DELETE etc.

options
Type: array< System::Object >[]
First option: true if username/password is correct but not cnonce

**Return Value**

Authentication object that is stored for the request. A user class or something like that.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentException</td>
<td>if authenticationHeader is invalid</td>
</tr>
<tr>
<td>System:::ArgumentNullException</td>
<td>If any of the parameters is empty or null.</td>
</tr>
</tbody>
</table>
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a response that can be sent in the WWW-Authenticate header.

**Namespace:** [HttpServer.Authentication](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public override string CreateResponse(
    string realm,
    params Object[] options
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function CreateResponse (_
    realm As String, _
    ParamArray options As Object() _) As String
```

**Visual C++**

```cpp
public:
virtual String^ CreateResponse(
    String^ realm,
    ... array<Object^>^ options
) override
```

**Parameters**

**realm**

Type: `System::::String`

Realm that the user should authenticate in

**options**

Type: `array< System::::Object >[](())[]`

First options specifies if true if username/password is correct but not cnonce.

**Return Value**

A correct auth request.
# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System..::.ArgumentNullException</td>
<td>If realm is empty or null.</td>
</tr>
</tbody>
</table>
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DigestAuthentication::Decode Method

Decodes authorization header value

**Namespace:** [HttpServer.Authentication](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static NameValueCollection Decode(string buffer, Encoding encoding)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function Decode(_
    buffer As String, _
    encoding As Encoding _
) As NameValueCollection
```

Visual C++

```cpp
public:
static NameValueCollection^ Decode(_
    String^ buffer, _
    Encoding^ encoding
)
```

Parameters

buffer
Type: System::::String
header value

encoding
Type: System.Text::::Encoding
Encoding that the buffer is in

Return Value

All headers and their values if successful; otherwise null
Remarks

Can handle lots of whitespaces and new lines without failing.
Examples

NameValueCollection header =
DigestAuthentication.Decode("response="6629fae49393a05397450978507c4ef1
Encoding.ASCII");
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DigestAuthentication..::.GetCurrentNonce Method

DigestAuthentication Class  See Also  Send Feedback

Gets the current nonce.

Namespace:  HttpServer.Authentication
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual string GetCurrentNonce()

Visual Basic (Declaration)

Protected Overridable Function GetCurrentNonce As String

Visual C++

protected:
virtual String^ GetCurrentNonce()

Return Value

See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DigestAuthentication..::.GetMD5HashBinHex2 Method

DigestAuthentication Class  See Also  Send Feedback

Gets the Md5 hash bin hex2.

Namespace:  HttpServer.Authentication
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string GetMD5HashBinHex2(
    string toBeHashed
)

Visual Basic (Declaration)

Public Shared Function GetMD5HashBinHex2 ( _
    toBeHashed As String _
) As String

Visual C++

public:
static String^ GetMD5HashBinHex2(
    String^ toBeHashed
)

Parameters

toBeHashed
    Type: System::String
    To be hashed.

Return Value

[Missing <returns> documentation for
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DigestAuthentication...:::IsValidNonce Method

`DigestAuthentication Class`  `See Also`  `Send Feedback`

determines if the nonce is valid or has expired.

**Namespace:**  `HttpServer.Authentication`

**Assembly:**  `HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)`
Syntax

**C#**

```csharp
protected virtual bool IsValidNonce(
    string nonce
)
```

**Visual Basic (Declaration)**

```vbnet
Protected Overridable Function IsValidNonce ( _
    nonce As String _
) As Boolean
```

**Visual C++**

```cpp
protected:
virtual bool IsValidNonce(
    String^ nonce
)
```

**Parameters**

nonce

Type: `System::String`

nonce value (check wikipedia for info)

**Return Value**

true if the nonce has not expired.
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The DigestAuthentication type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>name used in http request. (Overrides AuthModule:..:Name.)</td>
</tr>
</tbody>
</table>
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
name used in http request.

**Namespace:**  [HttpServer.Authentication](http://HttpServer.Authentication)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override string Name { get; }
```

Visual Basic (Declaration)

```vbnet
Public Overrides ReadOnly Property Name As String
```

Visual C++

```cpp
public:
virtual property String^ Name {
    String^ get () override;
}
```
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `DigestAuthentication` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnAuthenticate</td>
<td>used to let authentication modules</td>
</tr>
<tr>
<td></td>
<td>authenticate the username and password (Inherited from AuthModule.)</td>
</tr>
<tr>
<td></td>
<td>Let's you decide on a system level if authentication is required. You can also</td>
</tr>
<tr>
<td>OnAuthenticationRequired</td>
<td>decide if authentication is required in each HttpModule.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from AuthModule.)</td>
</tr>
</tbody>
</table>
See Also

DigestAuthentication Class
HttpServer.Authentication Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.Controllers]
## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthRequiredAttribute</td>
<td>Marks methods to let framework know that the method is protected</td>
</tr>
<tr>
<td>AuthValidatorAttribute</td>
<td>Method marked with this attribute determines if authentication is required.</td>
</tr>
<tr>
<td>BeforeFilterAttribute</td>
<td>Methods marked with BeforeFilter will be invoked before each request.</td>
</tr>
<tr>
<td>ControllerNameAttribute</td>
<td>This attribute can be used to map a controller to a specific url without using the class name. This attribute tells the controller that the method uses SendHeader and/or SendBody to handle the request.</td>
</tr>
<tr>
<td>RawHandlerAttribute</td>
<td>A controller in the Model-View-Controller pattern. Derive this class and add method with one of the following signatures: &quot;public string MethodName()&quot; or &quot;public void MyMethod()&quot;. The first should return a string with the response, the latter should use SendHeader and SendBody methods to handle the response.</td>
</tr>
<tr>
<td>RequestController</td>
<td>View controllers integrates the templates, by adding Render methods.</td>
</tr>
<tr>
<td>ViewController</td>
<td></td>
</tr>
</tbody>
</table>
## Enumerations

<table>
<thead>
<tr>
<th>Enumeration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FilterPosition</td>
<td>Determines when a before filter is executed.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AuthRequiredAttribute Class

Marks methods to let framework know that the method is protected

**Namespace:** [HttpServer.Controllers](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class AuthRequiredAttribute : Attribute

Visual Basic (Declaration)

Public Class AuthRequiredAttribute 
    Inherits Attribute

Visual C++

public ref class AuthRequiredAttribute : public Attribute
Inheritance Hierarchy

System::Object
System::Attribute
HttpServer.Controllers::AuthRequiredAttribute
See Also

AuthRequiredAttribute Members
HttpServer.Controllers Namespace
HttpServer.Controllers...:: AuthValidatorAttribute
HttpServer.Controllers...:: RequestController

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AuthRequiredAttribute` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthRequiredAttribute</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>IsDefaultAttribute</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
**Properties**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>Level is a value defined by you. It should be used to determine the users that can access the method tagged with the AuthRequired attribute. When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)</td>
</tr>
</tbody>
</table>
See Also

AuthRequiredAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver

AuthRequiredAttribute Constructor

AuthRequiredAttribute Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AuthRequiredAttribute()</code></td>
<td>Initializes a new instance of the <code>AuthRequiredAttribute</code> class.</td>
</tr>
<tr>
<td><code>AuthRequiredAttribute(Int32)</code></td>
<td>Initializes a new instance of the <code>AuthRequiredAttribute</code> class</td>
</tr>
</tbody>
</table>
See Also

AuthRequiredAttribute Class
AuthRequiredAttribute Members
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AuthRequiredAttribute Constructor

Initializes a new instance of the AuthRequiredAttribute class.

Namespace: HttpServer.Controllers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public AuthRequiredAttribute()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
AuthRequiredAttribute()
See Also

AuthRequiredAttribute Class
AuthRequiredAttribute Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AuthRequiredAttribute Constructor (Int32)

Initializes a new instance of the AuthRequiredAttribute class

Namespace: HttpServer.Controllers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public AuthRequiredAttribute(
    int level
)

Visual Basic (Declaration)

Public Sub New (  
    level As Integer  
)

Visual C++

public:
AuthRequiredAttribute(
    int level
)

Parameters

level
Type: System::::Int32
Level is a value defined by you. It should be used to determine the users that can access the method tagged with the AuthRequired attribute.
public enum UserLevel { Guest, User, Admin, SuperAdmin } public class UserController : RequestController { [AuthRequired(UserLevel.User)] public string View() { return "Can also be viewed by users"; } [AuthValidatorAttribute] public bool ValidateUser(int level) { (User)user = Session["user"]; return user != null && user.Status >= level; } }
See Also

AuthRequiredAttribute Class
AuthRequiredAttribute Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AuthRequiredAttribute` type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>IsDefaultAttribute</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

AuthRequiredAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AuthRequiredAttribute` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Level</strong></td>
<td>Level is a value defined by you. It should be used to determine the users that can access the method tagged with the AuthRequired attribute. When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)</td>
</tr>
</tbody>
</table>
See Also

AuthRequiredAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AuthRequiredAttribute::Level Property

Level is a value defined by you. It should be used to determine the users that can access the method tagged with the AuthRequired attribute.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int Level { get; }

Visual Basic (Declaration)

Public ReadOnly Property Level As Integer

Visual C++

public:
property int Level {
    int get ();
}

See Also

[AuthRequiredAttribute Class](#)
[HttpServer.Controllers Namespace](#)

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
Method marked with this attribute determines if authentication is required.

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class AuthValidatorAttribute : Attribute

Visual Basic (Declaration)

Public Class AuthValidatorAttribute _
    Inherits Attribute

Visual C++

public ref class AuthValidatorAttribute : public Attribute
Remarks

The method should take one parameter (int level), return a bool and be protected/private.

You should throw UnauthorizedException if you are using HTTP authentication.
Examples

```csharp
public enum UserLevel { Guest, User, Admin, SuperAdmin }
public class UserController : RequestController {
    [AuthRequired(UserLevel.User)]
    public string View() {
        return "Can also be viewed by users";
    }
    [AuthValidatorAttribute]
    public bool ValidateUser(int level) {
        User user = Session["user"];  
        return user != null && user.Status >= level;
    }
}
```
Inheritance Hierarchy

System..::..Object
System..::..Attribute
HttpServer.Controllers..::..AuthValidatorAttribute
See Also

AuthValidatorAttribute Members
HttpServer.Controllers Namespace
HttpServer.HttpModules..::.ControllerModule
HttpServer..::.HttpServer
HttpServer.Controllers..::.AuthRequiredAttribute
WebSiteModule

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The AuthValidatorAttribute type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthValidatorAttribute</td>
<td>Initializes a new instance of the AuthValidatorAttribute class</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>IsDefaultAttribute</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typeld</td>
<td>When implemented in a derived class, gets a unique identifier (Inherited from Attribute.)</td>
</tr>
</tbody>
</table>
See Also

AuthValidatorAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic Visual C++
C# Webserver

AuthValidatorAttribute Constructor

See Also Send Feedback

Initializes a new instance of the `AuthValidatorAttribute` class

**Namespace:** [HttpServer.Controllers]

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public AuthValidatorAttribute()

**Visual Basic (Declaration)**

Public Sub New

**Visual C++**

public:
AuthValidatorAttribute()
See Also

AuthValidatorAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AuthValidatorAttribute` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>IsDefaultValue</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

AuthValidatorAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **AuthValidatorAttribute** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TypeId</td>
<td>When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)</td>
</tr>
</tbody>
</table>
See Also

AuthValidatorAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Methods marked with BeforeFilter will be invoked before each request.

**Namespace:**  [HttpServer.Controllers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public class BeforeFilterAttribute : Attribute
```

Visual Basic (Declaration)

```vbnet
Public Class BeforeFilterAttribute
    Inherits Attribute
```

Visual C++

```cpp
public ref class BeforeFilterAttribute : public Attribute
```
Remarks

BeforeFilters should take no arguments and return false if controller method should not be invoked.
Inheritance Hierarchy

System..:::Object
System..:::Attribute
HttpServer.Controllers..:::BeforeFilterAttribute
See Also

BeforeFilterAttribute Members
HttpServer.Controllers Namespace
HttpServer.Controllers...::FilterPosition

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `BeforeFilterAttribute` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BeforeFilterAttribute</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Equals</code></td>
<td>Returns a value that indicates whether this instance is equal to a specified object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Attribute</a>.)</td>
</tr>
<tr>
<td><code>Finalize</code></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><code>GetHashCode</code></td>
<td>Returns the hash code for this instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Attribute</a>.)</td>
</tr>
<tr>
<td><code>GetType</code></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><code>IsDefaultAttribute</code></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Attribute</a>.)</td>
</tr>
<tr>
<td><code>Match</code></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Attribute</a>.)</td>
</tr>
<tr>
<td><code>MemberwiseClone</code></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><code>ToString</code></td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Position</td>
<td>Filters position in before filter queue</td>
</tr>
<tr>
<td></td>
<td>When implemented in a derived class, gets a unique identifier</td>
</tr>
<tr>
<td>Typeld</td>
<td>for this <code>Attribute</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Attribute</code>.)</td>
</tr>
</tbody>
</table>
See Also

BeforeFilterAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>BeforeFilterAttribute()</code></td>
<td>Initializes a new instance of the <code>BeforeFilterAttribute</code> class.</td>
</tr>
<tr>
<td><code>BeforeFilterAttribute(FilterPosition)</code></td>
<td>Initializes a new instance of the <code>BeforeFilterAttribute</code> class.</td>
</tr>
</tbody>
</table>
See Also

BeforeFilterAttribute Class
BeforeFilterAttribute Members
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
BeforeFilterAttribute Constructor

BeforeFilterAttribute Class  See Also  Send Feedback

Initializes a new instance of the BeforeFilterAttribute class.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public BeforeFilterAttribute()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New
```

**Visual C++**

```cpp
public:
BeforeFilterAttribute()
```
See Also

BeforeFilterAttribute Class
BeforeFilterAttribute Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
BeforeFilterAttribute Constructor (FilterPosition)

BeforeFilterAttribute Class  See Also  Send Feedback

Initializes a new instance of the BeforeFilterAttribute class.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public BeforeFilterAttribute(
    FilterPosition position
)

Visual Basic (Declaration)
Public Sub New ( _
    position As FilterPosition _
)

Visual C++
public:
BeforeFilterAttribute(
    FilterPosition position
)

Parameters

position
Type: HttpServer.Controllers..::;FilterPosition
Specify if the filter should be invoked among the first filters, in between or among the last.
See Also

BeforeFilterAttribute Class
BeforeFilterAttribute Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **BeforeFilterAttribute** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>IsDefaultAttribute</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from <strong>Attribute</strong>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
See Also

BeforeFilterAttribute Class
HttpServer.Controllers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `BeforeFilterAttribute` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>Filters position in before filter queue</td>
</tr>
<tr>
<td></td>
<td>When implemented in a derived class, gets a unique identifier</td>
</tr>
<tr>
<td>Typeld</td>
<td>for this <code>Attribute</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Attribute</code>.)</td>
</tr>
</tbody>
</table>
See Also

BeforeFilterAttribute Class
HttpServer.Controllers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
BeforeFilterAttribute..:::Position Property

Filters position in before filter queue

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public FilterPosition Position { get; }
```

Visual Basic (Declaration)

```vbnet
Public ReadOnly Property Position As FilterPosition
```

Visual C++

```c++
public:
property FilterPosition Position {
    FilterPosition get ();
}
```
See Also

BeforeFilterAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ControllerNameAttribute Class

This attribute can be used to map a controller to a specific URL without using the class name.

Namespace:  [HttpServer.Controllers]
Assembly:  [HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)]
Syntax

C#

public class ControllerNameAttribute : Attribute

Visual Basic (Declaration)

Public Class ControllerNameAttribute _
    Inherits Attribute

Visual C++

public ref class ControllerNameAttribute : public Attribute
Inheritance Hierarchy

System:::Object
System:::Attribute
HttpServer.Controllers:::ControllerNameAttribute
See Also

ControllerNameAttribute Members
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ControllerNameAttribute` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ControllerNameAttribute</code></td>
<td>Maps a controller to a url without using the controller name.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Equals</code></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><code>Finalize</code></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>GetHashCode</code></td>
<td>Returns the hash code for this instance. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><code>GetType</code></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>IsDefaultAttribute</code></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><code>Match</code></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><code>MemberwiseClone</code></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>ToString</code></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name that the controller should use. When implemented in a derived class, gets a unique identifier.</td>
</tr>
<tr>
<td>Typeld</td>
<td>for this Attribute. (Inherited from Attribute.)</td>
</tr>
</tbody>
</table>
See Also

ControllerNameAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Maps a controller to a url without using the controller name.

**Namespace:**  [HttpServer.Controllers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ControllerNameAttribute(
    string name
)

Visual Basic (Declaration)

Public Sub New (_
    name As String _
)

Visual C++

public:
ControllerNameAttribute(
    String^ name
)

Parameters

name
    Type: System::String
    The name.
Remarks

Per default the class name is used to determine which url to handle. For instance, "class UserController" or "class User" tells the framework that the urls that starts with "/user" should be handled by that controller.

This attribute can be used to circumvent that.
See Also

ControllerNameAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ControllerNameAttribute** type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>IsDefaultAttribute</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from <code>Attribute</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

ControllerNameAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ControllerNameAttribute` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name that the controller should use</td>
</tr>
<tr>
<td></td>
<td>When implemented in a derived class, gets a unique identifier</td>
</tr>
<tr>
<td>Typeld</td>
<td>for this <strong>Attribute</strong>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <strong>Attribute</strong>.)</td>
</tr>
</tbody>
</table>
See Also

ControllerNameAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The name that the controller should use

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Name { get; }

Visual Basic (Declaration)

Public ReadOnly Property Name As String

Visual C++

public:
property String^ Name {
    String^ get ();
}

}
See Also

ControllerNameAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determins when a before filter is executed.

**Namespace:** [HttpServer.Controllers](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public enum FilterPosition

**Visual Basic (Declaration)**

Public Enumeration FilterPosition

**Visual C++**

public enum class FilterPosition
<table>
<thead>
<tr>
<th>Member name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>Filter will be invoked first (unless another filter is added after this one with the First position)</td>
</tr>
<tr>
<td>Between</td>
<td>Invoke after all first filters, and before the last filters.</td>
</tr>
<tr>
<td>Last</td>
<td>Filter will be invoked last (unless another filter is added after this one with the Last position)</td>
</tr>
</tbody>
</table>
See Also

HttpServer.Controllers Namespace
HttpServer.Controllers..::.BeforeFilterAttribute

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This attribute tells the controller that the method uses SendHeader and/or SendBody to handle the request.

**Namespace:**  [HttpServer.Controllers](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public class RawHandlerAttribute : Attribute
```

**Visual Basic (Declaration)**

```vbnet
Public Class RawHandlerAttribute 
    Inherits Attribute
```

**Visual C++**

```cpp
public ref class RawHandlerAttribute : public Attribute
```
Inheritance Hierarchy

System...:::Object
System...:::Attribute
HttpServer.Controllers...:::RawHandlerAttribute
See Also

RawHandlerAttribute Members
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RawHandlerAttribute` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RawHandlerAttribute</td>
<td>Initializes a new instance of the RawHandlerAttribute class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Returns a value that indicates whether this instance is equal to a specified object.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Returns the hash code for this instance.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td><strong>IsDefaultAttribute</strong></td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class.</td>
</tr>
<tr>
<td><strong>Match</strong></td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TypeId</td>
<td>When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)</td>
</tr>
</tbody>
</table>
See Also

RawHandlerAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RawHandlerAttribute Constructor

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RawHandlerAttribute()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
RawHandlerAttribute()
See Also

RawHandlerAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RawHandlerAttribute` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Returns a value that indicates whether this instance is equal to a specified object.</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Returns the hash code for this instance.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td>IsDefaultAttribute</td>
<td>When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class.</td>
</tr>
<tr>
<td>Match</td>
<td>When overridden in a derived class, returns a value that indicates whether this instance equals a specified object.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object.</td>
</tr>
</tbody>
</table>
See Also

RawHandlerAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **RawHandlerAttribute** type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>TypeId</code></td>
<td>When implemented in a derived class, gets a unique identifier for this <code>Attribute</code>. (Inherited from <code>Attribute</code>.)</td>
</tr>
</tbody>
</table>
See Also

RawHandlerAttribute Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A controller in the Model-View-Controller pattern. Derive this class and add method with one of the following signatures: "public string MethodName()" or "public void MyMethod()". The first should return a string with the response, the latter should use SendHeader and SendBody methods to handle the response.

**Namespace:** HttpServer.Controllers  
**Assembly:** HttpServer (in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract class RequestController : HttpHeaders, ICloneable

Visual Basic (Declaration)

Public MustInherit Class RequestController __
    Inherits HttpHeaders __
    Implements ICloneable

Visual C++

public ref class RequestController abstract : public HttpHeaders, ICloneable
Remarks

Last segment of the path is always broken into the properties Id and RequestedType. Although note that the RequestedType can also be empty if no file extension have been specified. A typical use of file extensions in controllers is to specify which type of format to return.
Examples

```csharp
public class MyController : RequestController { public string Hello() { if (RequestedType == "xml") return "<hello>World</hello>"; else return "Hello " + Request.QueryString["user"].Value + ", welcome to my world"; } public void File() { Response.Headers.ContentType = "text/xml"; Response.SendHeader(); } }
```
Inheritance Hierarchy

System..:::Object
HttpServer.HttpModules..:::HttpModule
HttpServer.Controllers..:::RequestController
   HttpServer.Controllers..:::ViewController
See Also

RequestController Members
HttpServer.Controllers Namespace
HttpServer.Controllers...::ControllerNameAttribute
HttpServer.Controllers...::AuthRequiredAttribute
HttpServer.Controllers...::AuthValidatorAttribute

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RequestController` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestController</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CanHandle</strong></td>
<td>Method that determines if an url should be handled or not by the module</td>
</tr>
<tr>
<td><strong>Clone</strong></td>
<td>Make a clone of this controller</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong></td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetMethod</strong></td>
<td>Determines which method to use.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>InvokeMethod</strong></td>
<td>Override this method to be able to process result returned by controller method.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>OnUnhandledException</strong></td>
<td>Override this method if you want to be able to handle unhandled exceptions</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url (Overrides <strong>HttpModule</strong>,:::<strong>Process(HttpRequest, HttpResponse, HttpSession)</strong>.)</td>
</tr>
<tr>
<td><strong>SetupRequest</strong></td>
<td>Will assign all variables that are unique for each session</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthenticationTag</td>
<td>object that was attached during http authentication process.</td>
</tr>
<tr>
<td>ControllerName</td>
<td>Name of this controller (class name without the &quot;Controller&quot; part)</td>
</tr>
<tr>
<td>DefaultMethod</td>
<td>Specifies the method to use if no action have been specified.</td>
</tr>
<tr>
<td>Id</td>
<td>Id is the third part of the uri path.</td>
</tr>
<tr>
<td>MethodName</td>
<td>Method currently being invoked. Always in lower case.</td>
</tr>
<tr>
<td>Request</td>
<td>Request information (like Url, form, querystring etc)</td>
</tr>
<tr>
<td>RequestedExtension</td>
<td>Extension if a filename was specified.</td>
</tr>
<tr>
<td>Response</td>
<td>Response information (that is going to be sent back to the browser/client)</td>
</tr>
<tr>
<td>Session</td>
<td>Session information, is stored between requests as long as the session cookie is valid.</td>
</tr>
</tbody>
</table>
See Also

RequestController Class
HttpServer.Controllers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestController Constructor

RequestController Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RequestController()</code></td>
<td>create a new controller</td>
</tr>
<tr>
<td><code>RequestController(RequestController)</code></td>
<td>create a new request controller</td>
</tr>
</tbody>
</table>
See Also

RequestController Class
RequestController Members
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
create a new controller

**Namespace:**  [HttpServer.Controllers]

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public RequestController()

**Visual Basic (Declaration)**

Public Sub New

**Visual C++**

public:
RequestController()
See Also

RequestController Class
RequestController Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
create a new request controller

**Namespace:** [HttpServer.Controllers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RequestController(  
    RequestController controller  
)

Visual Basic (Declaration)

Public Sub New ( _  
    controller As RequestController _  
)

Visual C++

public:
    RequestController(  
        RequestController^ controller  
    )

Parameters

caller
    Type: HttpServer.Controllers::__RequestController
    prototype to copy information from
See Also

RequestController Class
RequestController Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The RequestController type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanHandle</td>
<td>Method that determines if an url should be handled or not by the module</td>
</tr>
<tr>
<td>Clone</td>
<td>Make a clone of this controller</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td>GetMethod</td>
<td>Determines which method to use.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td>InvokeMethod</td>
<td>Override this method to be able to process result returned by controller method.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td>OnUnhandledException</td>
<td>Override this method if you want to be able to handle unhandled exceptions Method that process the url (Overrides <strong>HttpModule::Process(HttpRequest, HttpResponse, HttpSession)</strong>.)</td>
</tr>
<tr>
<td>Process</td>
<td><strong>HttpModule::Process(HttpRequest, HttpResponse, HttpSession)</strong>.)</td>
</tr>
<tr>
<td>SetupRequest</td>
<td>Will assign all variables that are unique for each session</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
See Also

RequestController Class
HttpServer.Controllers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Method that determines if an url should be handled or not by the module

**Namespace:**  [HttpServer.Controllers](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public virtual bool CanHandle(
    HttpRequest request
)
```

Visual Basic (Declaration)

```vbnet
Public Overridable Function CanHandle ( _
    request As HttpRequest _
) As Boolean
```

Visual C++

```cpp
public:
    virtual bool CanHandle(
        HttpRequest^ request
    )
```

Parameters

- request
  - Type: `HttpServer::::HttpRequest`
  - Url requested by the client.

Return Value

- true if module should handle the url.
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Make a clone of this controller

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  [HttpServer](in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract Object Clone()

Visual Basic (Declaration)

Public MustOverride Function Clone As Object

Visual C++

public:
virtual Object^ Clone() abstract

Return Value

a new controller with the same base information as this one.

Implements

ICloneable::Clone()()
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines which method to use.

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll)  
**Version:**  1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual MethodInfo GetMethod(HttpRequest request)

Visual Basic (Declaration)

Protected Overridable Function GetMethod(_request As HttpRequest_) As MethodInfo

Visual C++

protected:
virtual MethodInfo^ GetMethod(_HttpRequest^ request)

Parameters

request
Type: HttpServer...HttpRequest
Requested resource

Return Value

See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Override this method to be able to process result returned by controller method.

**Namespace:**  [HttpServer.Controllers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void InvokeMethod()

Visual Basic (Declaration)

Protected Overridable Sub InvokeMethod

Visual C++

protected:
virtual void InvokeMethod()
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Override this method if you want to be able to handle unhandled exceptions

**Namespace:** [HttpServer.Controllers](https://example.com/HttpServer.Controllers)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void OnUnhandledException(
    Exception err
)

Visual Basic (Declaration)

Protected Overridable Sub OnUnhandledException ( _
    err As Exception _
)

Visual C++

protected:
virtual void OnUnhandledException(
    Exception^ err
)

Parameters

err
    Type: System::Exception
    thrown exception
Remarks

Don't "eat" exceptions derived from HttpException since they are handled by the framework, unless you are sure of what you are doing.
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Method that process the url

**Namespace:** [HttpServer.Controllers](http://HttpServer.Controllers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool Process(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)

Visual Basic (Declaration)

Public Overrides Function Process ( _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
) As Boolean

Visual C++

public:
virtual bool Process(
    HttpRequest^ request,
    HttpResponse^ response,
    HttpSession^ session
) override

Parameters

request
    Type: HttpServer::HttpRequest
    Uses Uri and QueryString to determine method.

response
    Type: HttpServer::HttpResponse
    Relays response object to invoked method.

session
    Type: HttpServer::HttpSession
Relays session object to invoked method.

Return Value

See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

RequestController...:::SetupRequest Method

RequestController Class  See Also  Send Feedback

Will assign all variables that are unique for each session

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual void SetupRequest(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)

Visual Basic (Declaration)

Protected Overridable Sub SetupRequest (_
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
)

Visual C++

protected:
virtual void SetupRequest(
    HttpRequest^ request,
    HttpResponse^ response,
    HttpSession^ session
)

Parameters

request
Type: HttpRequest

response
Type: HttpResponse
Type: HttpSession

[Missing <param name="session"/> documentation for
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The RequestController type exposes the following members.
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AuthenticationTag</strong></td>
<td>object that was attached during http authentication process.</td>
</tr>
<tr>
<td><strong>ControllerName</strong></td>
<td>Name of this controller (class name without the &quot;Controller&quot; part)</td>
</tr>
<tr>
<td><strong>DefaultMethod</strong></td>
<td>Specifies the method to use if no action have been specified.</td>
</tr>
<tr>
<td><strong>Id</strong></td>
<td>Id is the third part of the uri path.</td>
</tr>
<tr>
<td><strong>MethodName</strong></td>
<td>Method currently being invoked. Always in lower case.</td>
</tr>
<tr>
<td><strong>Request</strong></td>
<td>Request information (like Url, form, querystring etc)</td>
</tr>
<tr>
<td><strong>RequestedExtension</strong></td>
<td>Extension if a filename was specified.</td>
</tr>
<tr>
<td><strong>Response</strong></td>
<td>Response information (that is going to be sent back to the browser/client)</td>
</tr>
<tr>
<td><strong>Session</strong></td>
<td>Session information, is stored between requests as long as the session cookie is valid.</td>
</tr>
</tbody>
</table>
See Also

RequestController Class
HttpServer.Controllers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestController::AuthenticationTag Property

Namespace: HttpServer.Controllers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
```csharp
protected Object AuthenticationTag { get; set; }
```

**Visual Basic (Declaration)**

Protected Property AuthenticationTag As Object

**Visual C++**

protected:
property Object^ AuthenticationTag {
    Object^ get ();
    void set (Object^ value);
}
Remarks

You can also assign this tag yourself if you are using regular http page login.
See Also

RequestController Class
HttpServer.Controllers Namespace
HttpServer.Authentication::AuthModule

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Name of this controller (class name without the "Controller" part)

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string ControllerName { get; }

Visual Basic (Declaration)

Public ReadOnly Property ControllerName As String

Visual C++

public:
property String^ ControllerName {
  String^ get ();
}
}
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Specifies the method to use if no action have been specified.

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string DefaultMethod { get; set; }

Visual Basic (Declaration)

Public Property DefaultMethod As String

Visual C++

public:
property String^ DefaultMethod {
    String^ get ();
    void set (String^ value);
}

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System...:ArgumentException</td>
<td>If specified method do not exist.</td>
</tr>
</tbody>
</table>
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Id is the third part of the uri path.

**Namespace:**  [HttpServer.Controllers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Id { get; }

Visual Basic (Declaration)

Public ReadOnly Property Id As String

Visual C++

public:
property String^ Id {
    String^ get ();
}
Remarks

Is extracted as in: /controllername/methodname/id/

string.Empty if not specified.
Examples
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Method currently being invoked. Always in lower case.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string MethodName { get; }
```

Visual Basic (Declaration)

```vbnet
Public ReadOnly Property MethodName As String
```

Visual C++

```cpp
public:
    property String^ MethodName {
        String^ get ();
    }
```
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RequestController...:::Request Property

Namespace: HttpServer.Controllers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
protected HttpRequest Request { get; }

Visual Basic (Declaration)
Protected Readonly Property Request As HttpRequest

Visual C++
protected:
property HttpRequest^ Request {
    HttpRequest^ get ();
}
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Extension if a filename was specified.

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  [HttpServer.dll](#) (in HttpServer.dll)  
**Version:**  1.0.0.0 (1.0.0.0)
Syntax

C#

public string RequestedExtension { get; }

Visual Basic (Declaration)

Public ReadOnly Property RequestedExtension As String

Visual C++

public:
property String^ RequestedExtension {
	String^ get ();
}


See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Response information (that is going to be sent back to the browser/client)

**Namespace:**  [HttpServer.Controllers]

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
protected HttpResponseMessage Response { get; }
```

**Visual Basic (Declaration)**

```vbnet
Protected ReadOnly Property Response As HttpResponseMessage
```

**Visual C++**

```c++
protected:
property HttpResponseMessage^ Response {
    HttpResponseMessage^ get ();
}
```
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Session information, is stored between requests as long as the session cookie is valid.

Namespace:.HttpServer.Controllers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected HttpSession Session { get; }

Visual Basic (Declaration)

Protected ReadOnly Property Session As HttpSession

Visual C++

protected:
property HttpSession^ Session {
    HttpSession^ get ();
}
See Also

RequestController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
View controllers integrates the templates, by adding Render methods.

**Namespace:** [HttpServer.Controllers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract class ViewController : RequestController

Visual Basic (Declaration)

Public MustInherit Class ViewController _
    Inherits RequestController

Visual C++

public ref class ViewController abstract : public RequestController
Inheritance Hierarchy

System..:::Object
  HttpServer.HttpModules..:::HttpModule
    HttpServer.Controllers..:::RequestController
      HttpServer.Controllers..:::ViewController
See Also

ViewController Members
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ViewController type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViewController</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanHandle</td>
<td>Method that determines if an url should be handled or not by the module</td>
</tr>
<tr>
<td>Clone</td>
<td>Make a clone of this controller</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>GetMethod</td>
<td>Determines which method to use.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td>InvokeMethod</td>
<td>Override this method to be able to process result returned by controller method.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td>OnUnhandledException</td>
<td>Override this method if you want to be able to handle unhandled exceptions</td>
</tr>
<tr>
<td>Process</td>
<td>Method that process the url</td>
</tr>
<tr>
<td>Render</td>
<td>Render template for the currently invoked method.</td>
</tr>
<tr>
<td>RenderAction</td>
<td>Render contents into a template.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RenderErrors</td>
<td>Renders errors from the Errors property into the current method template, or as a javascript alert if the request is ajax.</td>
</tr>
<tr>
<td>RenderJavascript</td>
<td>Switches content-type to &quot;text/javascript&quot; and returns content.</td>
</tr>
<tr>
<td>RenderJsErrors</td>
<td>Creates a javascript &quot;alert&quot; filled with all errors.</td>
</tr>
<tr>
<td>RenderLayout</td>
<td>Renders one of the layouts</td>
</tr>
<tr>
<td>RenderTemplate</td>
<td>Render a template.</td>
</tr>
<tr>
<td>SetupRequest</td>
<td>Invoked each time a new request is about to be invoked.</td>
</tr>
<tr>
<td></td>
<td>(Overrides RequestController::SetupRequest(HttpRequest, HttpResponse, HttpSession).)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arguments</td>
<td>Arguments that are being used in the templates.</td>
</tr>
<tr>
<td>AuthenticationTag</td>
<td>The object that was attached during the HTTP authentication process.</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
<tr>
<td>ControllerName</td>
<td>Name of this controller (class name without the &quot;Controller&quot; part)</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
<tr>
<td>DefaultMethod</td>
<td>Specifies the method to use if no action have been specified.</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
<tr>
<td>Errors</td>
<td>A set of errors that occurred during request processing. Key should be argument name (if argument error, otherwise Empty()()()), value should be the error message.</td>
</tr>
<tr>
<td>Id</td>
<td>Id is the third part of the uri path.</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
<tr>
<td>IncludeLayoutInAjaxRequests</td>
<td>True if we always should render contents inside page layouts when request is ajax.</td>
</tr>
<tr>
<td>Layout</td>
<td>Which page layout to use (without file extension)</td>
</tr>
<tr>
<td>MethodName</td>
<td>Method currently being invoked.</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
<tr>
<td>Request</td>
<td>Request information (like Url, form, querystring etc)</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
<tr>
<td>RequestedExtension</td>
<td>Extension if a filename was specified.</td>
</tr>
<tr>
<td>(Inherited from RequestController.)</td>
<td></td>
</tr>
</tbody>
</table>
Response

Response information (that is going to be sent back to the browser/client) (Inherited from RequestController.)

Session

Session information, is stored between requests as long as the session cookie is valid. (Inherited from RequestController.)

Title

Page title (are added as a parameter to the layout template, use it in <title> html tag.)
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViewController(ViewController)</td>
<td>Create a new ViewController.</td>
</tr>
<tr>
<td>ViewController(TemplateManager)</td>
<td>Create a new ViewController.</td>
</tr>
</tbody>
</table>
See Also

ViewController Class  
ViewController Members  
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new ViewController.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public ViewController(
                        ViewController controller
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New ( _
                  controller As ViewController _
)
```

Visual C++

```cpp
public:
ViewController(
                 ViewController^ controller
)
```

Parameters

controller

Type: `HttpServer.Controllers.::ViewController`

prototype to copy information from
See Also

ViewController Class
ViewController Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new ViewController.

**Namespace:**  HttpServer.Controllers

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ViewController(
    TemplateManager mgr
)

Visual Basic (Declaration)

Public Sub New ( _
    mgr As TemplateManager _
)

Visual C++

public:
ViewController(
    TemplateManager^ mgr
)

Parameters

mgr
Type: HttpHeaders.Rendering...::TemplateManager

See Also

ViewController Class
ViewController Overload
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ViewController type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanHandle</td>
<td>Method that determines if an url should be handled or not by the module</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>Clone</td>
<td>Make a clone of this controller</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetMethod</td>
<td>Determines which method to use.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>InvokeMethod</td>
<td>Override this method to be able to process result returned by controller</td>
</tr>
<tr>
<td></td>
<td>method.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>OnUnhandledException</td>
<td>Override this method if you want to be able to handle unhanded exceptions</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>Process</td>
<td>Method that process the url</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>Render</td>
<td>Render template for the currently invoked method.</td>
</tr>
<tr>
<td>RenderAction</td>
<td>Render contents into a template.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RenderErrors</td>
<td>Renders errors from the <code>Errors</code> property into the current method template, or as a javascript alert if the request is ajax.</td>
</tr>
<tr>
<td>RenderJavascript</td>
<td>Switches content-type to &quot;text/javascript&quot; and returns content.</td>
</tr>
<tr>
<td>RenderJsErrors</td>
<td>Creates a javascript &quot;alert&quot; filled with all errors.</td>
</tr>
<tr>
<td>RenderLayout</td>
<td>Renders one of the layouts</td>
</tr>
<tr>
<td>RenderTemplate</td>
<td>Render a template.</td>
</tr>
<tr>
<td>SetupRequest</td>
<td>Invoked each time a new request is about to be invoked. (Overrides <code>RequestController...:::SetupRequest(HttpRequest, HttpResponse, HttpSession)</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ViewController...:::Render Method

ViewController Class  See Also  Send Feedback

Render template for the currently invoked method.

Namespace:  HttpHeaders.Controllers
Assembly:  HttpHeaders (in HttpHeaders.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected string Render(
    params Object[] args
)

Visual Basic (Declaration)

Protected Function Render ( _
    ParamArray args As Object() _
) As String

Visual C++

protected:
String^ Render(
    ... array<Object^>^ args
)

Parameters

args

Type: array< System::Object >[]( )[]

arguments/parameters used in template

Return Value

template generated content
Remarks

calls RenderActionWithErrors
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Directive

 ViewController...:::RenderAction Method
 ViewController Class  See Also  Send Feedback

Render contents into a template.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
protected virtual string RenderAction(
    string method,
    params Object[] args
)
```

### Visual Basic (Declaration)

```vbnet
Protected Overridable Function RenderAction ( _
    method As String, _
    ParamArray args As Object() _
) As String
```

### Visual C++

```cpp
protected:
virtual String^ RenderAction( 
    String^ method,
    ... array<Object^>^ args
)
```

## Parameters

**method**
Type: `System::String`  
method/template to generate

**args**
Type: array<`System::Object`>[]()  
arguments/parameters used in template

## Return Value

template generated content
Remarks

calls RenderActionWithErrors.
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Renders errors from the Errors property into the current method template, or as a javascript alert if the request is ajax.

Namespace: HttpServer.Controllers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

protected `string` RenderErrors(`
    `string` method,
    params `Object`[] arguments
)

**Visual Basic (Declaration)**

Protected Function RenderErrors (_
    method As `String`, _
    ParamArray arguments As `Object`() _
) As `String`

**Visual C++**

protected:
`String` RenderErrors(`
    `String` method,
    ... array<`Object`[^] arguments
)

**Parameters**

**method**
Type: `System::String`  
name of the currently invoked method.

**arguments**
Type: array< `System::Object`>[]()
arguments used in the method template.

**Return Value**

generated string
Remarks

Creates a javascript alertbox if request is ajax.
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Switches content-type to "text/javascript" and returns content.

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected string RenderJavascript(
    string js
)

Visual Basic (Declaration)

Protected Function RenderJavascript ( _
    js As String _
) As String

Visual C++

protected:
    String^ RenderJavascript(
        String^ js
    )

Parameters

js
    Type: System::.:: String
    javascript to send to the client.

Return Value

js
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ViewController..::.RenderJsErrors Method

ViewController Class  See Also  Send Feedback

Creates a javascript "alert" filled with all errors.

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
protected string RenderJsErrors(
    NameValueCollection errors
)
```

### Visual Basic (Declaration)

```vbnet
Protected Function RenderJsErrors ( _
    errors As NameValueCollection _
) As String
```

### Visual C++

```cpp
protected:
    String^ RenderJsErrors(
        NameValueCollection^ errors
    )
```

### Parameters

- **errors**
  
  Type: `System.Collections.Specialized.NameValueCollection`

### Return Value

- a
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ViewController...:::RenderLayout Method

`ViewController Class`  `See Also`  `Send Feedback`

renders one of the layouts

**Namespace:**  `HttpException.Controllers`

**Assembly:**  `HttpServer (in HttpServer.dll)`  `Version: 1.0.0.0 (1.0.0.0)`
Syntax

C#

protected virtual string RenderLayout(
    string layoutName,
    string contents
)

Visual Basic (Declaration)

Protected Overridable Function RenderLayout ( _
    layoutName As String, _
    contents As String _
) As String

Visual C++

protected:
virtual String^ RenderLayout(
    String^ layoutName,
    String^ contents
)

Parameters

layoutName
    Type: System::String
    layout to render (should be found in the "views\layouts" folder).

contents
    Type: System::String
    contents will be put in the template variable called "text".

Return Value

generated text/html.
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Render a template.

**Namespace:** [HttpServer.Controllers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected string RenderTemplate(
    string controller,
    string method,
    params Object[] args
)

Visual Basic (Declaration)

Protected Function RenderTemplate (_
    controller As String, _
    method As String, _
    ParamArray args As Object() _
) As String

Visual C++

protected:
    String^ RenderTemplate(
        String^ controller,
        String^ method,
        ... array<Object^>^ args
    )

Parameters

controller
    Type: System:::String
    controller name are used as a foldername when looking for the template.

method
    Type: System:::String
    method are used as filename when looking for the template.

args
    Type: array< System:::Object >[]()[]}
arguments that should be passed to the template.

**Return Value**

Remarks

Merges the Arguments property with the "args" parameter and pass those to the template.
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Invoked each time a new request is about to be invoked.

**Namespace:** [HttpServer.Controllers](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
protected override void SetupRequest(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)
```

### Visual Basic (Declaration)

```vbnet
Protected Overrides Sub SetupRequest ( _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
)
```

### Visual C++

```cpp
protected:
    virtual void SetupRequest(
        HttpRequest^ request,
        HttpResponse^ response,
        HttpSession^ session
    ) override
```

## Parameters

### request

Type: `HttpRequest`

[Missing `<param name="request"/>` documentation for

### response

Type: `HttpResponse`

[Missing `<param name="response"/>` documentation for
session

Type: HttpSession

Remarks

Can be used to clear old data.
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ViewController type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arguments</td>
<td>Arguments that are being used in the templates.</td>
</tr>
<tr>
<td>AuthenticationTag</td>
<td>object that was attached during http authentication process.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>ControllerName</td>
<td>Name of this controller (class name without the &quot;Controller&quot; part)</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>DefaultMethod</td>
<td>Specifies the method to use if no action have been specified.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>Errors</td>
<td>A set of errors that occurred during request processing. Key should be</td>
</tr>
<tr>
<td></td>
<td>argument name (if argument error, otherwise Empty())(), value should be</td>
</tr>
<tr>
<td></td>
<td>the error message.</td>
</tr>
<tr>
<td>Id</td>
<td>Id is the third part of the uri path.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>IncludeLayoutInAjaxRequests</td>
<td>True if we always should render contents inside page layouts when request is ajax.</td>
</tr>
<tr>
<td>Layout</td>
<td>Which page layout to use (without file extension)</td>
</tr>
<tr>
<td>MethodName</td>
<td>Method currently being invoked.</td>
</tr>
<tr>
<td></td>
<td>Always in lower case.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>Request</td>
<td>Request information (like Url, form, querystring etc)</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td>RequestedExtension</td>
<td>Extension if a filename was specified.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from RequestController.)</td>
</tr>
<tr>
<td></td>
<td>Response</td>
</tr>
<tr>
<td>---</td>
<td>---------------------</td>
</tr>
<tr>
<td></td>
<td>Session</td>
</tr>
<tr>
<td></td>
<td>Title</td>
</tr>
</tbody>
</table>
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Arguments that are being used in the templates.

**Namespace:** [HttpServer.Controllers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected TemplateArguments Arguments { get; }

Visual Basic (Declaration)

Protected ReadOnly Property Arguments As TemplateArguments

Visual C++

protected:
property TemplateArguments^ Arguments {
    TemplateArguments^ get ();
}

}
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A set of errors that occurred during request processing. Key should be argument name (if argument error, otherwise Empty())(), value should be the error message.

**Namespace:**  [HttpServer.Controllers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected NameValueCollection Errors { get; set; }

Visual Basic (Declaration)

Protected Property Errors As NameValueCollection

Visual C++

protected:
property NameValueCollection^ Errors {
    NameValueCollection^ get ();
    void set (NameValueCollection^ value);
}


Remarks

Errors can be rendered into templates using the WebHelper.RenderError method.
See Also

ViewController Class
HttpServer.Controllers Namespace
HttpServer.Helpers...:::WebHelper

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
True if we always should render contents inside page layouts when request is ajax.

**Namespace:**  [HttpServer.Controllers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public bool IncludeLayoutInAjaxRequests { get; set; }
```

**Visual Basic (Declaration)**

```vbnet
Public Property IncludeLayoutInAjaxRequests As Boolean
```

**Visual C++**

```cpp
public:
property bool IncludeLayoutInAjaxRequests {
    bool get ();
    void set (bool value);
}
```
Remarks

default is false.
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Which page layout to use (without file extension)

Namespace:  HttpServer.Controllers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Layout { get; set; }

Visual Basic (Declaration)

Public Property Layout As String

Visual C++

public:
property String^ Layout {
    String^ get ();
    void set (String^ value);
}
Remarks

Page layouts should be placed in the Views\Layouts folder.

default is "Application"
See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Page title (are added as a parameter to the layout template, use it in <title> html tag.

**Namespace:**  [HttpServer.Controllers](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Title { get; set; }

Visual Basic (Declaration)

Public Property Title As String

Visual C++

public:
property String^ Title {
    String^ get ();
    void set (String^ value);
}

See Also

ViewController Class
HttpServer.Controllers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.Exceptions]
# Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BadRequestException</td>
<td>The request could not be understood by the server due to malformed syntax. The client SHOULD NOT repeat the request without modifications. Text taken from: <a href="http://www.submissionchamber.com/help-guides/error-codes.php">http://www.submissionchamber.com/help-guides/error-codes.php</a> The server understood the request, but is refusing to fulfill it. Authorization will not help and the request SHOULD NOT be repeated. If the request method was not HEAD and the server wishes to make public why the request has not been fulfilled, it SHOULD describe the reason for the refusal in the entity.</td>
</tr>
<tr>
<td>ForbiddenException</td>
<td></td>
</tr>
<tr>
<td>HttpException</td>
<td>All HTTP based exceptions will derive this class. The server encountered an unexpected condition which prevented it from fulfilling the request.</td>
</tr>
<tr>
<td>InternalServerException</td>
<td>The requested resource was not found in the web server. The request requires user authentication. The response MUST include a WWW-Authenticate header field (section 14.47) containing a challenge applicable to the requested resource. The client MAY repeat the request with a suitable Authorization header</td>
</tr>
<tr>
<td>NotFoundException</td>
<td></td>
</tr>
</tbody>
</table>
UnauthorizedException (section 14.8). If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials. If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user SHOULD be presented the entity that was given in the response, since that entity might include relevant diagnostic information. HTTP access authentication is explained in rfc2617: http://www.ietf.org/rfc/rfc2617.txt (description is taken from http://www.submissionchamber.com/help-guides/error-codes.php#sec10.4.2)
The request could not be understood by the server due to malformed syntax. The client SHOULD NOT repeat the request without modifications. Text taken from: http://www.submissionchamber.com/help-guides/error-codes.php

**Namespace:**  [HttpServer.Exceptions](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class BadRequestException : Exception

Visual Basic (Declaration)

Public Class BadRequestException _
    Inherits Exception

Visual C++

public ref class BadRequestException : public Exception
Inheritance Hierarchy

System...:::Object
System...:::Exception
HttpServer.Exceptions...:::HttpException
HttpServer.Exceptions...:::BadRequestException
See Also

BadRequestException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `BadRequestException` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BadRequestException</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/systemserialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception.</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception.</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>status code to use in the response.</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the Exception instance that caused the current exception.</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception.</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error.</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown.</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception.</td>
</tr>
</tbody>
</table>
See Also

BadRequestException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
BadRequestException Constructor

BadRequestException Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BadRequestException(String)</td>
<td>Create a new bad request exception.</td>
</tr>
<tr>
<td>BadRequestException(String, Exception)</td>
<td>Create a new bad request exception.</td>
</tr>
</tbody>
</table>
See Also

BadRequestException Class
BadRequestException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new bad request exception.

**Namespace:** [HttpServer.Exceptions](#)  
**Assembly:** `HttpServer` (in `HttpServer.dll`) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public BadRequestException(
    string errMsg
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    errMsg As String _
)
```

**Visual C++**

```cpp
public:
BadRequestException(
    String^ errMsg
)
```

**Parameters**

`errMsg`

Type: `System::String`

reason to why the request was bad.
See Also

BadRequestException Class
BadRequestException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new bad request exception.

**Namespace:** [HttpServer.Exceptions]

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public BadRequestException(
    string errMsg,
    Exception inner
)

Visual Basic (Declaration)

Public Sub New ( _
    errMsg As String, _
    inner As Exception _
)

Visual C++

public:
BadRequestException(
    String^ errMsg,
    Exception^ inner
)

Parameters

errMsg
   Type: System::String
   reason to why the request was bad.

inner
   Type: System::Exception
   inner exception
See Also

BadRequestException Class
BadRequestException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `BadRequestException` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
</tbody>
</table>
See Also

BadRequestException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **BadRequestException** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>status code to use in the response. (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

BadRequestException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ForbiddenException Class

Members  See Also  Send Feedback

The server understood the request, but is refusing to fulfill it. Authorization will not help and the request SHOULD NOT be repeated. If the request method was not HEAD and the server wishes to make public why the request has not been fulfilled, it SHOULD describe the reason for the refusal in the entity. If the server does not wish to make this information available to the client, the status code 404 (Not Found) can be used instead. Text taken from: http://www.submissionchamber.com/help-guides/error-codes.php

Namespace:  HttpServer.Exceptions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public class ForbiddenException : HttpException
```

**Visual Basic (Declaration)**

```vbnet
Public Class ForbiddenException
    Inherits HttpException
```

**Visual C++**

```cpp
public ref class ForbiddenException : public HttpException
```
Inheritance Hierarchy

System..::.Object
System..::.Exception
HttpServer.Exceptions..::.HttpException
HttpServer.Exceptions..::.ForbiddenException
See Also

ForbiddenException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ForbiddenException` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ForbiddenException</code></td>
<td>Initializes a new instance of the <code>ForbiddenException</code> class.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="#">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="#">Exception</a>)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>When overridden in a derived class, sets the <a href="#">SerializationInfo</a> with information about the exception. (Inherited from <a href="#">Exception</a>)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="#">Exception</a>)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="#">Exception</a>)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>Status code to use in the response. (Inherited from <strong>HttpException</strong>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <strong>Exception</strong> instance that caused the current exception. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <strong>Exception</strong>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from <strong>Exception</strong>.)</td>
</tr>
</tbody>
</table>
See Also

ForbiddenException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ForbiddenException Constructor

Initializes a new instance of the `ForbiddenException` class.

**Namespace:**  [HttpServer.Exceptions](HttpServer.Exceptions)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ForbiddenException(
    string errorMsg
)

Visual Basic (Declaration)

Public Sub New (_
    errorMsg As String _
)

Visual C++

public:
ForbiddenException(
    String^ errorMsg
)

Parameters

errorMsg
    Type: System::String
    error message
See Also

ForbiddenException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ForbiddenException` type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
</tbody>
</table>
See Also

ForbiddenException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ForbiddenException` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception.        (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>Status code to use in the response.                                            (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <code>Exception</code> instance that caused the current exception.         (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception.                         (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception.                           (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

ForbiddenException Class
HttpServer.Exceptions Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
All HTTP based exceptions will derive this class.

**Namespace:** [HttpServer.Exceptions](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpException : Exception

Visual Basic (Declaration)

Public Class HttpException _
    Inherits Exception

Visual C++

public ref class HttpException : public Exception
Inheritance Hierarchy

System..:::Object
System..:::Exception
HttpServer.Exceptions..:::.HttpException
  HttpServer.Exceptions..:::.BadRequestException
  HttpServer.Exceptions..:::.ForbiddenException
  HttpServer.Exceptions..:::.InternalServerException
  HttpServer.Exceptions..:::.NotFoundException
  HttpServer.Exceptions..:::.UnauthorizedException
See Also

HttpException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpException` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpException</code></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.) When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.) Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.) Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
</tbody>
</table>
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>Status code to use in the response.</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <a href="#">Exception</a> instance that caused the current exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
</tbody>
</table>
See Also

HttpException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpException(HttpStatusCode, String)</code></td>
<td>Create a new <code>HttpException</code></td>
</tr>
<tr>
<td><code>HttpException(HttpStatusCode, String, Exception)</code></td>
<td>Create a new <code>HttpException</code></td>
</tr>
</tbody>
</table>
See Also

HttpException Class
HttpException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new HttpException

**Namespace:** [HttpServer.Exceptions](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public HttpException(
    HttpStatusCode code,
    string message
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    code As HttpStatusCode, _
    message As String _
)
```

**Visual C++**

```cpp
public: 
HttpException(
    HttpStatusCode code,
    String^ message
)
```

### Parameters

**code**

Type: `System.Net..::.HttpStatusCode`
http status code (sent in the response)

**message**

Type: `System..::.String`
error description
See Also

HttpException Class
HttpException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new HttpException

**Namespace:**  [HttpServer.Exceptions](#)  
**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#))  
**Version:**  1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public HttpException(
    HttpStatusCode code,
    string message,
    Exception inner
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New ( _
    code As HttpStatusCode, _
    message As String, _
    inner As Exception _
)
```

### Visual C++

```cpp
public:
HttpException(
    HttpStatusCode code,
    String^ message,
    Exception^ inner
)
```

## Parameters

**code**

- Type: `System.Net...::HttpStatusCode`
- http status code (sent in the response)

**message**

- Type: `System...::String`
- error description

**inner**

- Type: `System...::Exception`
inner exception
See Also

HttpException Class
HttpException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpException` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
</tbody>
</table>
See Also

HttpException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpException` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| Data             | Gets a collection of key/value pairs that provide additional user-defined information about the exception.  
                    (Inherited from [Exception](#).) |
| HelpLink         | Gets or sets a link to the help file associated with this exception.  
                    (Inherited from [Exception](#).) |
| HResult          | Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.  
                    (Inherited from [Exception](#).) |
| HttpStatusCode   | status code to use in the response. |
| InnerException   | Gets the [Exception](#) instance that caused the current exception.  
                    (Inherited from [Exception](#).) |
| Message          | Gets a message that describes the current exception.  
                    (Inherited from [Exception](#).) |
| Source           | Gets or sets the name of the application or the object that causes the error.  
                    (Inherited from [Exception](#).) |
| StackTrace       | Gets a string representation of the frames on the call stack at the time the current exception was thrown.  
                    (Inherited from [Exception](#).) |
| TargetSite       | Gets the method that throws the current exception.  
                    (Inherited from [Exception](#).) |
See Also

HttpException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpException...::: HttpStatusCode Property

status code to use in the response.

Namespace:  HttpServer.Exceptions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public HttpStatusCode HttpStatusCode { get; }
```

**Visual Basic (Declaration)**

```vbnet
Public ReadOnly Property HttpStatusCode As HttpStatusCode
```

**Visual C++**

```cpp
public:
    property HttpStatusCode HttpStatusCode { HttpStatusCode get (); }
```
See Also

HttpException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
InternalServerException Class

Members  See Also  Send Feedback

The server encountered an unexpected condition which prevented it from fulfilling the request.

Namespace:  HttpServer.Exceptions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class InternalServerException : Exception

Visual Basic (Declaration)

Public Class InternalServerException _
    Inherits Exception

Visual C++

public ref class InternalServerException : public Exception
Inheritance Hierarchy

System::Object
   System::Exception
      HttpServer.Exceptions::HttpException
         HttpServer.Exceptions::InternalServerError
See Also

InternalServerException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `InternalServerException` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InternalServerError</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="#">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>HelpLink</td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>HResult</td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>HttpStatusCode</td>
<td>status code to use in the response. (Inherited from HttpException.)</td>
</tr>
<tr>
<td>InnerException</td>
<td>Gets the Exception instance that caused the current exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>Message</td>
<td>Gets a message that describes the current exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>Source</td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)</td>
</tr>
<tr>
<td>StackTrace</td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)</td>
</tr>
<tr>
<td>TargetSite</td>
<td>Gets the method that throws the current exception. (Inherited from Exception.)</td>
</tr>
</tbody>
</table>
See Also

InternalServerException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>InternalServerException()</code></td>
<td>Initializes a new instance of the <code>InternalServerException</code> class.</td>
</tr>
<tr>
<td><code>InternalServerException(String)</code></td>
<td>Initializes a new instance of the <code>InternalServerException</code> class.</td>
</tr>
<tr>
<td><code>InternalServerException(String, Exception)</code></td>
<td>Initializes a new instance of the <code>InternalServerException</code> class.</td>
</tr>
</tbody>
</table>
See Also

InternalServerException Class
InternalServerException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
InternalServerException Constructor

Initializes a new instance of the InternalServerException class.

Namespace: HttpServer.Exceptions
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public InternalServerException()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
InternalServerException()
See Also

InternalServerException Class
InternalServerException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
InternalServerException Constructor (String)

Initializes a new instance of the `InternalServerException` class.

Namespace: [HttpServer.Exceptions](#)
Assembly: [HttpServer (in HttpServer.dll)](#) Version: 1.0.0 (1.0.0.0)
Syntax

C#

public InternalServerException(
    string message
)

Visual Basic (Declaration)

Public Sub New (_
    message As String _
)

Visual C++

public:
_InternalServerException(
    String^ message
)

Parameters

message
Type: System::String
error message.
See Also

InternalServerException Class
InternalServerException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++
C# Webserver

InternalServerException Constructor (String, Exception)

Namespace:  HttpServer.Exceptions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public InternalServerException(
    string message,
    Exception inner
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    message As String, _
    inner As Exception _
)
```

**Visual C++**

```cpp
public:
    InternalServerException(
        String^ message, 
        Exception^ inner
    )
```

**Parameters**

**message**

Type: `System::String`
error message.

**inner**

Type: `System::Exception`
inner exception.
See Also

InternalServerException Class
InternalServerException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `InternalServerException` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetBaseException</td>
<td>When overridden in a derived class, returns the <a href="#">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="#">SerializationInfo</a> with information about the exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td>GetObjectData</td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td></td>
</tr>
</tbody>
</table>
See Also

InternalServerException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **InternalServerException** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>HelpLink</td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>HResult</td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>HttpStatusCode</td>
<td>Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td>InnerException</td>
<td>Gets a message that describes the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>Message</td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>Source</td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>StackTrace</td>
<td>Gets the method that throws the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>

---

*Properties*
See Also

InternalServerException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The requested resource was not found in the web server.

**Namespace:**  [HttpServer.Exceptions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class NotFoundException : HttpException

Visual Basic (Declaration)

Public Class NotFoundException _
    Inherits HttpException

Visual C++

public ref class NotFoundException : public HttpException
Inheritance Hierarchy

System..::..Object
  System..::..Exception
    HttpServer.Exceptions..::..HttpException
      HttpNotFoundException
See Also

[NotFoundException Members]
[HttpServer.Exceptions Namespace]

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `NotFoundException` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>NotFoundException</code></td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetBaseException</td>
<td>When overridden in a derived class, returns the <code>Exception</code> that is the root cause of one or more subsequent exceptions. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetObjectData</td>
<td>When overridden in a derived class, sets the <code>SerializationInfo</code> with information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the runtime type of the current instance. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Creates and returns a string representation of the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception.        (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>status code to use in the response. (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <code>Exception</code> instance that caused the current exception.  (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception.                         (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception.                          (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

NotFoundException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NotFoundException(String)</td>
<td>Create a new exception</td>
</tr>
<tr>
<td>NotFoundException(String, Exception)</td>
<td>Create a new exception</td>
</tr>
</tbody>
</table>
See Also

NotFoundException Class
NotFoundException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new exception

**Namespace:**  [HttpServer.Exceptions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public NotFoundException(
    string message
)

Visual Basic (Declaration)

Public Sub New (_
    message As String _
)

Visual C++

public:
NotFoundException(
    String^ message
)

Parameters

message
    Type: System::String
    message describing the error
See Also

NotFoundException Class
NotFoundException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new exception

**Namespace:** [HttpServer.Exceptions](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public NotFoundException(
    string message,
    Exception inner
)

Visual Basic (Declaration)

Public Sub New (_
    message As String, _
    inner As Exception _
)

Visual C++

public:
    NotFoundException(
        String^ message,
        Exception^ inner
    )

Parameters

message
    Type: System::String
    message describing the error

inner
    Type: System::Exception
    inner exception
See Also

NotFoundException Class
NotFoundException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `NotFoundException` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
</tbody>
</table>
See Also

NotFoundException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `NotFoundException` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HRESULT</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>Status code to use in the response. (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

NotFoundException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The request requires user authentication. The response MUST include a WWW-Authenticate header field (section 14.47) containing a challenge applicable to the requested resource. The client MAY repeat the request with a suitable Authorization header field (section 14.8). If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials. If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user SHOULD be presented the entity that was given in the response, since that entity might include relevant diagnostic information. HTTP access authentication is explained in rfc2617: http://www.ietf.org/rfc/rfc2617.txt (description is taken from http://www.submissionchamber.com/help-guides/error-codes.php#sec10.4.2)

Namespace:  HttpServer.Exceptions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public class UnauthorizedException : HttpException
```

**Visual Basic (Declaration)**

```vbnet
Public Class UnauthorizedException
    Inherits HttpException
```

**Visual C++**

```cpp
public ref class UnauthorizedException : public HttpException
```
Inheritance Hierarchy

System...:::Object
System...:::Exception
  HttpServer.Exceptions...:::HttpException
    HttpServer.Exceptions...:::UnauthorizedException
See Also

UnauthorizedException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **UnauthorizedException** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\textbf{UnauthorizedException}$</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetBaseException</td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serializerinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td>GetObjectData</td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td></td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>status code to use in the response. (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

UnauthorizedException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
UnauthorizedException Constructor

UnauthorizedException Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnauthorizedException()()</td>
<td>Create a new unauthorized exception.</td>
</tr>
<tr>
<td>UnauthorizedException(String)</td>
<td>Create a new unauthorized exception.</td>
</tr>
<tr>
<td>UnauthorizedException(String, Exception)</td>
<td>Create a new unauthorized exception.</td>
</tr>
</tbody>
</table>
See Also

UnauthorizedException Class
UnauthorizedException Members
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
UnauthorizedException Constructor

Create a new unauthorized exception.

**Namespace:**  [HttpServer.Exceptions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public UnauthorizedException()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
UnauthorizedException()
See Also

UnauthorizedException Class
UnauthorizedException Overload
HttpServer.Exceptions Namespace
HttpServer.Exceptions,:::UnauthorizedException

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new unauthorized exception.

**Namespace:**  [HttpServer.Exceptions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public UnauthorizedException(
    string message
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (
    message As String
)
```

**Visual C++**

```cpp
public:
UnauthorizedException(
    String^ message
)
```

**Parameters**

message

Type: System::String

reason to why the request was unauthorized.
See Also

UnauthorizedException Class
UnauthorizedException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
UnauthorizedException Constructor (String, Exception)

UnauthorizedException Class

Create a new unauthorized exception.

Namespace:  HttpServer.Exceptions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public UnauthorizedException(
    string message,
    Exception inner
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New (_
    message As String, _
    inner As Exception _
)
```

Visual C++

```cpp
public:
UnauthorizedException(
    String^ message,
    Exception^ inner
)
```

Parameters

message
  Type: System::String
  reason to why the request was unauthorized.

inner
  Type: System::Exception
  inner exception
See Also

UnauthorizedException Class
UnauthorizedException Overload
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **UnauthorizedException** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="/Object">Object</a> is equal to the current <a href="/Object">Object</a>. (Inherited from <a href="/Object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="/Object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="/Object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="/Object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="/Exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="/Exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="/SerializationInfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="/Exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="/Exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Creates a shallow copy of the current <a href="/Object">Object</a>. (Inherited from <a href="/Object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="/Exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td></td>
</tr>
</tbody>
</table>
See Also

UnauthorizedException Class
HttpServer.Exceptions Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `UnauthorizedException` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>HttpStatusCode</strong></td>
<td>Status code to use in the response. (Inherited from <code>HttpException</code>.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

UnauthorizedException Class
HttpServer.Exceptions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.FormDecoders]
## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormDecoderProvider</td>
<td>This provider is used to let us implement any type of form decoding we want without having to rewrite anything else in the server.</td>
</tr>
<tr>
<td>MultipartDecoder</td>
<td>Can handle application/x-www-form-urlencoded</td>
</tr>
<tr>
<td>UrlDecoder</td>
<td>This decoder converts XML documents to form items. Each element becomes a subitem in the form, and each attribute becomes an item.</td>
</tr>
<tr>
<td>XmlDecoder</td>
<td></td>
</tr>
</tbody>
</table>
## Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormDecoder</td>
<td>Interface for form content decoders.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Interface for form content decoders.

**Namespace:** [HttpServer.FormDecoders](http://example.com/HttpServer.FormDecoders)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public interface FormDecoder

Visual Basic (Declaration)

Public Interface FormDecoder

Visual C++

public interface class FormDecoder
See Also

FormDecoder Members
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormDecoder` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanParse</td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td>Decode</td>
<td></td>
</tr>
</tbody>
</table>
See Also

FormDecoder Interface
HttpServer.FormDecoders Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormDecoder` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanParse</td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td>Decode</td>
<td></td>
</tr>
</tbody>
</table>
See Also

FormDecoder Interface
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver

FormDecoder...CanParse Method

FormDecoder Interface  See Also  Send Feedback

Checks if the decoder can handle the mime type

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
bool CanParse(
    string contentType
)
```

**Visual Basic (Declaration)**

```vbnet
Function CanParse ( _
    contentType As String _
) As Boolean
```

**Visual C++**

```cpp
bool CanParse(
    String^ contentType
)
```

## Parameters

**contentType**

Type: `System::String`

Content type (with any additional info like boundry). Content type is always supplied in lower case.

## Return Value

True if the decoder can parse the specified content type
See Also

FormDecoder Interface
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormDecoder...:::Decode Method

[Missing <summary> documentation for

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
HttpForm Decode(
    Stream stream,
    string contentType,
    Encoding encoding
)
```

Visual Basic (Declaration)

Function Decode ( _
    stream As Stream, _
    contentType As String, _
    encoding As Encoding _
) As HttpForm

Visual C++

```c++
HttpForm Decode(
    Stream^ stream,
    String^ contentType,
    Encoding^ encoding
)
```

Parameters

stream
Type: System.IO:::Stream
Stream containing the content

contentType
Type: System:::String
Content type (with any additional info like boundary). Content type is always supplied in lower case

encoding
Type: System.Text:::Encoding
Stream encoding

Return Value

A http form, or null if content could not be parsed.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.IO..::.InvalidDataException</code></td>
<td>If contents in the stream is not valid input data.</td>
</tr>
</tbody>
</table>
See Also

FormDecoder Interface
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This provider is used to let us implement any type of form decoding we want without having to rewrite anything else in the server.

**Namespace:** [HttpServer.FormDecoders](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class FormDecoderProvider

Visual Basic (Declaration)

Public Class FormDecoderProvider

Visual C++

public ref class FormDecoderProvider
Inheritance Hierarchy

System..:::Object
HttpServer.FormDecoders..:::FormDecoderProvider
See Also

FormDecoderProvider Members
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormDecoderProvider` type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormDecoderProvider</td>
<td>Initializes a new instance of the FormDecoderProvider class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Add a decoder.</td>
</tr>
<tr>
<td><strong>Decode</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Number of added decoders.</td>
</tr>
<tr>
<td>Decoders</td>
<td>Use with care.</td>
</tr>
<tr>
<td>DefaultDecoder</td>
<td>Decoder used for unknown content types.</td>
</tr>
</tbody>
</table>
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormDecoderProvider Constructor

Namespace: HttpServer.FormDecoders
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)

Initializes a new instance of the FormDecoderProvider class
Syntax

C#

public FormDecoderProvider()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
FormDecoderProvider()
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormDecoderProvider` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Add a decoder.</td>
</tr>
<tr>
<td><strong>Decode</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add a decoder.

 Namespace:  [HttpServer.FormDecoders]
 Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void Add(
    FormDecoder decoder
)
```

Visual Basic (Declaration)

```vbnet
Public Sub Add ( _
    decoder As FormDecoder _
)
```

Visual C++

```cpp
public:
void Add( 
    FormDecoder^ decoder
)
```

Parameters

decoder

Type: `HttpServer.FormDecoders..::.FormDecoder`

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormDecoderProvider...:::Decode Method

FormDecoderProvider Class  See Also  Send Feedback

[Missing <summary> documentation for

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HttpForm Decode(
    string contentType,
    Stream stream,
    Encoding encoding
)

Visual Basic (Declaration)

Public Function Decode (_
    contentType As String, _
    stream As Stream, _
    encoding As Encoding _
) As HttpForm

Visual C++

public:
HttpForm^ Decode(
    String^ contentType,
    Stream^ stream,
    Encoding^ encoding
)

Parameters

contentType
    Type: System::String
    Should contain boundary and type, as in: multipart/form-data; boundary=---
    ------------------230051238959

stream
    Type: System.IO::Stream
    Stream containing form data.

encoding
Type: `System.Text.Encoding`
Encoding used when decoding the stream

**Return Value**

HttpInput.EmptyLanguageNode if no parser was found. Must always return something (HttpInput.Empty instead of null)
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System..::.ArgumentException</td>
<td>If stream is null or not readable.</td>
</tr>
<tr>
<td>System.IO..::.InvalidDataException</td>
<td>If stream contents cannot be decoded properly.</td>
</tr>
</tbody>
</table>
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The FormDecoderProvider type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Count</strong></td>
<td>Number of added decoders.</td>
</tr>
<tr>
<td><strong>Decoders</strong></td>
<td>Use with care.</td>
</tr>
<tr>
<td><strong>DefaultDecoder</strong></td>
<td>Decoder used for unknown content types.</td>
</tr>
</tbody>
</table>
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Number of added decoders.

**Namespace:**  [HttpServer.FormDecoders](http://www.HttpServer.FormDecoders)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public int Count { get; }
```

**Visual Basic (Declaration)**

```
Public ReadOnly Property Count As Integer
```

**Visual C++**

```csharp
public:
    property int Count {
        int get ();
    }
```
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Use with care.

**Namespace:**  [HttpServer.FormDecoders](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IList<FormDecoder> Decoders { get; }

Visual Basic (Declaration)

Public ReadOnly Property Decoders As IList(Of FormDecoder)

Visual C++

public:
property IList<FormDecoder>^ Decoders {
    IList<FormDecoder>^ get ();
}
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Decoder used for unknown content types.

**Namespace:** [HttpServer.FormDecoders](http://HttpServer.FormDecoders)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
C#

```csharp
public FormDecoder DefaultDecoder { get; set; }
```

Visual Basic (Declaration)

```vbnet
Public Property DefaultDecoder As FormDecoder
```

Visual C++

```cpp
public:
property FormDecoder^ DefaultDecoder {
FormDecoder^ get ();
void set (FormDecoder^ value);
}
```
See Also

FormDecoderProvider Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MultipartDecoder Class

Namespace: HttpServer.FormDecoders
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public class MultipartDecoder : FormDecoder
```

**Visual Basic (Declaration)**

```vbnet
Public Class MultipartDecoder
    Implements FormDecoder
```

**Visual C++**

```c++
public ref class MultipartDecoder : FormDecoder
```
Remarks

http://www.faqs.org/rfcs/rfc1867.html
Inheritance Hierarchy

System..:::.Object
HttpServer.FormDecoders..:::.MultipartDecoder
See Also

MultipartDecoder Members
HttpPostServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `MultipartDecoder` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MultipartDecoder</td>
<td>Initializes a new instance of the MultipartDecoder class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanParse</td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td>Decode</td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.</td>
</tr>
<tr>
<td>Equals</td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
# Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormData</td>
<td>form-data</td>
</tr>
<tr>
<td>MimeType</td>
<td>multipart/form-data</td>
</tr>
</tbody>
</table>
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MultipartDecoder Constructor

Namespace: HttpServer.FormDecoders

Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public MultipartDecoder()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
MultipartDecoder()
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `MultipartDecoder` type exposes the following members.
# Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormData</td>
<td>form-data</td>
</tr>
<tr>
<td>MimeType</td>
<td>multipart/form-data</td>
</tr>
</tbody>
</table>
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MultipartDecoder...:::FormData Field

**MultipartDecoder Class**  See Also  Send Feedback

form-data

**Namespace:**  [HttpServer.FormDecoders](#)

**Assembly:**  HttpServer (in HttpServer.dll)  Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public const string FormData

**Visual Basic (Declaration)**

Public Const FormData As String

**Visual C++**

public:

literal String^ FormData
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MultipartDecoder...:::MimeType Field

Namespace:  [HttpServer.FormDecoders](http://example.com)
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string MimeType

Visual Basic (Declaration)

Public Const MimeType As String

Visual C++

public:
    literal String^ MimeType
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The MultipartDecoder type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CanParse</strong></td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td><strong>Decode</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MultipartDecoder...:::CanParse Method

MultipartDecoder Class  See Also  Send Feedback

Checks if the decoder can handle the mime type

**Namespace:**  [HttpServer.FormDecoders](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool CanParse(
    string contentType
)

Visual Basic (Declaration)

Public Function CanParse ( _
    contentType As String _
) As Boolean

Visual C++

public:
    virtual bool CanParse(
        String^ contentType
    ) sealed

Parameters

contentType
    Type: System::::String
    Content type (with any additional info like boundry). Content type is always supplied in lower case.

Return Value

True if the decoder can parse the specified content type

Implements

FormDecoder::::CanParse(String)
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MultipartDecoder..::.Decode Method

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version:  1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public HttpForm Decode(
    Stream stream,
    string contentType,
    Encoding encoding
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Decode (_
    stream As Stream, _
    contentType As String, _
    encoding As Encoding _
) As HttpForm
```

**Visual C++**

```cpp
public:
    virtual HttpForm^ Decode(
        Stream^ stream, _
        String^ contentType, _
        Encoding^ encoding
    ) sealed
```

### Parameters

**stream**
- Type: `System.IO:::Stream`
- Stream containing the content

**contentType**
- Type: `System:::String`
- Content type (with any additional info like boundry). Content type is always supplied in lower case

**encoding**
Type: `System.Text.Encoding`  
Stream encoding

**Return Value**

A http form, or null if content could not be parsed.

**Implements**

`FormDecoder.Decode(Stream, String, Encoding)`
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.IO:::InvalidDataException</td>
<td>If contents in the stream is not valid input data.</td>
</tr>
<tr>
<td>System:::ArgumentNullException</td>
<td>If any parameter is null</td>
</tr>
</tbody>
</table>
See Also

MultipartDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
UrlDecoder Class

Members  See Also  Send Feedback

Can handle application/x-www-form-urlencoded

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class UrlDecoder : FormDecoder

Visual Basic (Declaration)

Public Class UrlDecoder
    Implements FormDecoder

Visual C++

public ref class UrlDecoder : FormDecoder
Inheritance Hierarchy

System..:::Object
HttpServer.FormDecoders..:::UrlDecoder
See Also

UrlDecoder Members
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The [UrlDecoder](#) type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>UrlDecoder</code></td>
<td>Initializes a new instance of the <code>UrlDecoder</code> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanParse</td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td>Decode</td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Equals</td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

**UrlDecoder Class**
**HttpServer.FormDecoders Namespace**

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
Initializes a new instance of the UrlDecoder class

**Namespace:**  [HttpServer.FormDecoders](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public UrlDecoder()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
UrlDecoder()}
See Also

UrlDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **UrlDecoder** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanParse</td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
</tbody>
</table>
| Decode          | Determines whether the specified `Object` is equal to the current `Object`.
| (Inherited from `Object`.) |
| Equals          | Allows an `Object` to attempt to free resources and perform other cleanup operations before the `Object`
| (Inherited from `Object`.) |
| Finalize        | Allows an `Object` to attempt to free resources and perform other cleanup operations before the `Object`
| (Inherited from `Object`.) |
| GetHashCode     | Serves as a hash function for a particular type.                  |
| (Inherited from `Object`.) |
| GetType         | Gets the `Type` of the current instance.                          |
| (Inherited from `Object`.) |
| MemberwiseClone | Creates a shallow copy of the current `Object`.                   |
| (Inherited from `Object`.) |
| ToString        | Returns a `String` that represents the current `Object`.          |
| (Inherited from `Object`.) |
See Also

UrlDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
CanParse Method

Namespace:  [HttpServer.FormDecoders]
Assembly:  [HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)]

Checks if the decoder can handle the mime type
Syntax

C#

```csharp
public bool CanParse(
    string contentType
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function CanParse (_
    contentType As String _
) As Boolean
```

**Visual C++**

```cpp
public:
    virtual bool CanParse(
        String^ contentType
    ) sealed
```

**Parameters**

contentType

Type: `System::String`

Content type (with any additional info like boundry). Content type is always supplied in lower case.

**Return Value**

True if the decoder can parse the specified content type

**Implements**

`FormDecoder::CanParse(String)`
See Also

UrlDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
UrlDecoder...:::Decode Method

UrlDecoder Class  See Also  Send Feedback

[Missing <summary> documentation for

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public HttpForm Decode(
    Stream stream,
    string contentType,
    Encoding encoding
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Decode ( _
    stream As Stream, _
    contentType As String, _
    encoding As Encoding _
) As HttpForm
```

**Visual C++**

```cpp
public:
    virtual HttpForm Decode(
    Stream^ stream, 
    String^ contentType, 
    Encoding^ encoding 
) sealed
```

**Parameters**

**stream**
Type: `System.IO::Stream`
Stream containing the content

**contentType**
Type: `System::String`
Content type (with any additional info like boundary). Content type is always supplied in lower case

**encoding**
Type: `System.Text.Encoding`  
Stream enconding

**Return Value**

A http form, or null if content could not be parsed.

**Implements**

`FormDecoder.Decode(Stream, String, Encoding)`
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.IO::InvalidDataException</code></td>
<td>If contents in the stream is not valid input data.</td>
</tr>
</tbody>
</table>
See Also

[UrlDecoder Class](#)
[HttpServer.FormDecoders Namespace](#)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This decoder converts XML documents to form items. Each element becomes a subitem in the form, and each attribute becomes an item.

**Namespace:** [HttpServer.FormDecoders](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public class XmlDecoder : FormDecoder
```

**Visual Basic (Declaration)**

```vbnet
Public Class XmlDecoder
    Implements FormDecoder
```

**Visual C++**

```cpp
public ref class XmlDecoder : FormDecoder
```
Remarks

The original xml document is stored in form['__xml__'].Value.
Examples

// xml: somethingdata // result: // form["hello"].Value = "something" //
form["hello"]["id"].Value = 1 // form["hello"]["world"]["id"].Value = 1 //
form["hello"]["world"].Value = "data"
Inheritance Hierarchy

System::Object

HttpServer.FormDecoders::XmlDecoder
See Also

XmlDecoder Members
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `XmlDecoder` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="XmlDecoder" /></td>
<td>Initializes a new instance of the <a href="#">XmlDecoder</a> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CanParse</strong></td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td><strong>Decode</strong></td>
<td>Determines whether the specified <a href="/api/typeof/Object">Object</a> is equal to</td>
</tr>
<tr>
<td></td>
<td>the current <a href="/api/typeof/Object">Object</a>.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Allows an <a href="/api/typeof/Object">Object</a> to attempt to free resources and</td>
</tr>
<tr>
<td></td>
<td>perform other cleanup operations before the <a href="/api/typeof/Object">Object</a> is</td>
</tr>
<tr>
<td></td>
<td>reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="/api/typeof/Object">Object</a> to attempt to free resources and</td>
</tr>
<tr>
<td></td>
<td>perform other cleanup operations before the <a href="/api/typeof/Object">Object</a> is</td>
</tr>
<tr>
<td></td>
<td>reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="/api/typeof/Type">Type</a> of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="/api/typeof/Object">Object</a>.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="/api/typeof/String">String</a> that represents the current <a href="/api/typeof/Object">Object</a>.</td>
</tr>
<tr>
<td><strong>TraverseNode</strong></td>
<td>Recursive function that will go through an xml element and store it's content to the form item.</td>
</tr>
</tbody>
</table>
See Also

*XmlDecoder Class*
*HttpServer.FormDecoders Namespace*

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
XmlDecoder Constructor

Initializes a new instance of the XmlDecoder class

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public XmlDecoder()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New
```

**Visual C++**

```cpp
public:
XmlDecoder()
```
See Also

XmlDecoder Class
HttpPost/FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **XmlDecoder** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanParse</td>
<td>Checks if the decoder can handle the mime type</td>
</tr>
<tr>
<td>Decode</td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Equals</td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>TraverseNode</td>
<td>Recursive function that will go through an xml element and store it's content to the form item.</td>
</tr>
</tbody>
</table>
See Also

XmlDecoder Class
HttpPost.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
XmlDecoder...::CanParse Method

Checks if the decoder can handle the mime type

Namespace:  HttpServer/FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public bool CanParse(
    string contentType
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function CanParse ( _
    contentType As String _
) As Boolean
```

**Visual C++**

```cpp
public:
  virtual bool CanParse(
    String^ contentType
  ) sealed
```

### Parameters

**contentType**

Type: `System::String`

Content type (with any additional info like boundry). Content type is always supplied in lower case.

### Return Value

True if the decoder can parse the specified content type

### Implements

`FormDecoder::CanParse(String)`
See Also

XmlDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for

Namespace:  HttpServer.FormDecoders
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpForm Decode(
    Stream stream,
    string contentType,
    Encoding encoding
)
```

Visual Basic (Declaration)

```vbnet
Public Function Decode ( _
    stream As Stream, _
    contentType As String, _
    encoding As Encoding _
) As HttpForm
```

Visual C++

```cpp
public:
    virtual HttpForm^ Decode(
        Stream^ stream,
        String^ contentType,
        Encoding^ encoding
    ) sealed
```

Parameters

stream
Type: `System.IO::Stream`
Stream containing the content

contentType
Type: `System::String`
Content type (with any additional info like boundary). Content type is always supplied in lower case

encoding
Type: System.Text.Encoding
Stream encoding

Return Value

A http form, or null if content could not be parsed.

Implements

FormDecoder.Decode(Stream, String, Encoding)
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.IO::InvalidDataException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

XmlDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Recursive function that will go through an xml element and store it's content to the form item.

**Namespace:** [HttpServer.FormDecoders](http://HttpServer.FormDecoders)

**Assembly:** [HttpServer](in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void TraverseNode(
      HttpInputBase item,
      XmlNode node
)

Visual Basic (Declaration)

Public Sub TraverseNode ( _
      item As HttpInputBase, _
      node As XmlNode _
)

Visual C++

public:
void TraverseNode(
      HttpInputBase^ item,
      XmlNode^ node
)

Parameters

item
Type: HttpServer::<::HttpInputBase
(parent) Item in form that content should be added to.

node
Type: System.Xml::<::XmlNode
Node that should be parsed.
See Also

XmlDecoder Class
HttpServer.FormDecoders Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.Helpers]
<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormHelper</td>
<td>Helpers making it easier to work with forms.</td>
</tr>
<tr>
<td>FormValidator</td>
<td>Validator is used to validate all input items in a form.</td>
</tr>
<tr>
<td>JavascriptHelperImplementation</td>
<td>Purpose of this class is to create a javascript toolkit independent javascript helper.</td>
</tr>
<tr>
<td>JSHelper</td>
<td>Will contain helper functions for javascript.</td>
</tr>
<tr>
<td>ObjectForm</td>
<td>The object form class takes an object and creates form items for it.</td>
</tr>
<tr>
<td>Validator</td>
<td>General usage validator class.</td>
</tr>
<tr>
<td>WebHelper</td>
<td>Webhelper provides helpers for common tasks in HTML.</td>
</tr>
<tr>
<td>XmlHelper</td>
<td>Helpers to make XML handling easier.</td>
</tr>
</tbody>
</table>
Delegates

<table>
<thead>
<tr>
<th>Delegate</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetIdTitle</td>
<td>Delegate used by <code>Select(String, IEnumerable, GetIdTitle, Object, Boolean)</code> to populate select options.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Helpers making it easier to work with forms.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public static class FormHelper
```

**Visual Basic (Declaration)**

```vbnet
Public NotInheritable Class FormHelper
```

**Visual C++**

```cpp
public ref class FormHelper abstract sealed
```
Inheritance Hierarchy

System..:::Object
HttpServer.Helpers..:::FormHelper
See Also

FormHelper Members
HttpServer.Helpers Namespace
HttpServer.Helpers...::ObjectForm

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **FormHelper** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckBox</td>
<td>Overloaded</td>
</tr>
<tr>
<td>End</td>
<td>form close tag</td>
</tr>
<tr>
<td>Options</td>
<td>Generate a list of HTML options</td>
</tr>
<tr>
<td>RadioButton</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Select</td>
<td>Overloaded</td>
</tr>
<tr>
<td>Start</td>
<td>Overloaded</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>JSImplementation</td>
<td>Used to let the website use different javascript libraries. Default is PrototypeImp</td>
</tr>
</tbody>
</table>
See Also

FormHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **FormHelper** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSImplementation</td>
<td>Used to let the website use different javascript libraries. Default is PrototypeImp</td>
</tr>
</tbody>
</table>
See Also

FormHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper...:::JSImplementation Field

Used to let the website use different javascript libraries. Default is PrototypeImp

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static JavascriptHelperImplementation JSImplementation

Visual Basic (Declaration)

Public Shared JSImplementation As JavascriptHelperImplementation

Visual C++

public:
static JavascriptHelperImplementation^ JSImplementation
See Also

FormHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormHelper` type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckBox</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>End</td>
<td>form close tag</td>
</tr>
<tr>
<td>Options</td>
<td>Generate a list of HTML options</td>
</tr>
<tr>
<td>RadioButton</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Select</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Start</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
See Also

FormHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper...:::CheckBox Method

FormHelper Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CheckBox(String, Object, Boolean, array&lt;String&gt;[])</code></td>
<td>Creates a checkbox.</td>
</tr>
<tr>
<td><code>CheckBox(String, String, Boolean, array&lt;String&gt;[])</code></td>
<td>Creates a checkbox.</td>
</tr>
<tr>
<td><code>CheckBox(String, String, Object, Boolean, array&lt;String&gt;[])</code></td>
<td>Creates a checkbox.</td>
</tr>
</tbody>
</table>
See Also

FormHelper Class
FormHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper...::::CheckBox Method (String, Object, Boolean, array<String>[] [])

FormHelper Class  See Also  Send Feedback

Creates a checkbox.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string CheckBox(
    string name,
    Object value,
    bool isChecked,
    params string[] htmlAttributes
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function CheckBox (_
    name As String, _
    value As Object, _
    isChecked As Boolean, _
    ParamArray htmlAttributes As String() _
) As String
```

Visual C++

```cpp
public:
static String^ CheckBox(
    String^ name,
    Object^ value,
    bool isChecked,
    ... array<String^>^ htmlAttributes
)
```

Parameters

- **name**
  - Type: `System::String`
  - element name

- **value**
  - Type: `System::Object`
  - element value
isChecked
  Type: System::Boolean
  true if checkbox is checked (selected)

htmlAttributes
  Type: array(System::String)[0][0]
  a list with additional attributes (name, value, name, value).

Return Value

a generated radio button
See Also

FormHelper Class
CheckBox Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper..::.CheckBox Method (String, String, Boolean, array<String>[])
# Syntax

## C#

```csharp
public static string CheckBox(
    string name,
    string id,
    bool isChecked,
    params string[] htmlAttributes
)
```

## Visual Basic (Declaration)

```vbnet
Public Shared Function CheckBox( _
    name As String, _
    id As String, _
    isChecked As Boolean, _
    ParamArray htmlAttributes As String() _
) As String
```

## Visual C++

```cpp
public:
static String^ CheckBox(
    String^ name,
    String^ id,
    bool isChecked,
    ... array<String^>^ htmlAttributes
)
```

## Parameters

**name**

Type: `System::String`

Element name

**id**

Type: `System::String`

Element id
isChecked
  Type: `System::Boolean`
  true if checkbox is checked (selected)

htmlAttributes
  Type: array< `System::String` >[]
  a list with additional attributes (name, value, name, value).

**Return Value**

a generated radio button
Remarks

will set value to "1".
See Also

FormHelper Class
CheckBox Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper:::CheckBox Method (String, String, Object, Boolean, array<String>[])::

FormHelper Class    See Also    Send Feedback

Creates a checkbox.

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string CheckBox(
    string name,
    string id,
    Object value,
    bool isChecked,
    params string[] htmlAttributes
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function CheckBox ( _
    name As String, _
    id As String, _
    value As Object, _
    isChecked As Boolean, _
    ParamArray htmlAttributes As String() _
) As String
```

Visual C++

```cpp
public:
static String^ CheckBox( 
    String^ name,
    String^ id,
    Object^ value,
    bool isChecked,
    ... array<String^>^ htmlAttributes
)
```

Parameters

name
Type: System::String
element name

id
Type: System::String
element id
value
Type: System:::Object
element value
isChecked
Type: System:::Boolean
true if checkbox is checked (selected)

htmlAttributes
Type: array< System::String >[]
a list with additional attributes (name, value, name, value).

**Return Value**

a generated radio button
See Also

FormHelper Class
CheckBox Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper Class  See Also  Send Feedback

form close tag

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string End()

Visual Basic (Declaration)

Public Shared Function End As String

Visual C++

public:
static String^ End() 

Return Value

See Also

FormHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper...:..Options Method

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public static string Options(
    IEnumerable collection,
    GetIdTitle getIdTitle,
    Object selectedValue,
    bool firstEmpty
)
```

**Visual Basic (Declaration)**

```vbnet
Public Shared Function Options (_,
    collection As IEnumerable, _,
    getIdTitle As GetIdTitle, _,
    selectedValue As Object, _,
    firstEmpty As Boolean _) As String
)
```

**Visual C++**

```csharp
public:
static String^ Options(
    IEnumerable^ collection,
    GetIdTitle^ getIdTitle,
    Object^ selectedValue,
    bool^ firstEmpty
)
```

**Parameters**

collection
   Type: System.Collections::IEnumerable
collection used to generate options.

getIdTitle
   Type: HttpServer.Helpers::GetIdTitle
delegate used to return id and title from objects.
selectedValue
Type: System::Object
value that should be marked as selected.

firstEmpty
Type: System::Boolean
First row should contain an empty value.

Return Value

See Also

FormHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper....::RadioButton Method

FormHelper Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RadioButton(String, Object, Boolean, array&lt;String&gt;[])</td>
<td>Creates a RadioButton.</td>
</tr>
<tr>
<td>RadioButton(String, String, Object, Boolean, array&lt;String&gt;[][][])</td>
<td>Creates a RadioButton.</td>
</tr>
</tbody>
</table>
See Also

FormHelper Class
FormHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper...:..RadioButton Method (String, Object, Boolean, array<String>[[]])

FormsHelper Class  See Also  Send Feedback

Creates a RadioButton.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public static string RadioButton(
    string name,
    Object value,
    bool isSelected,
    params string[] htmlAttributes
)
```

**Visual Basic (Declaration)**

```vbnet
Public Shared Function RadioButton ( _
    name As String, _
    value As Object, _
    isSelected As Boolean, _
    ParamArray htmlAttributes As String() _
) As String
```

**Visual C++**

```cpp
public:
static String^ RadioButton(
    String^ name,
    Object^ value,
    bool isSelected,
    ... array<String^>^ htmlAttributes
)
```

### Parameters

**name**
- Type: `System:::String`
- element name

**value**
- Type: `System:::Object`
- element value
isSelected
  Type: System::Boolean
  true if radiobutton is selected

htmlAttributes
  Type: array< System::String >[]
  a list with additional attributes (name, value, name, value).

Return Value

  a generated radio button
See Also

FormHelper Class
RadioButton Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper..::.RadioButton Method (String, String, Object, Boolean, array<String>[][])

FormHelper Class  See Also  Send Feedback

Creates a RadioButton.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public static string RadioButton(
    string name,
    string id,
    object value,
    bool isSelected,
    params string[] htmlAttributes
)
```

### Visual Basic (Declaration)

```vbnet
Public Shared Function RadioButton ( _
    name As String, _
    id As String, _
    value As Object, _
    isSelected As Boolean, _
    ParamArray htmlAttributes As String() _
) As String
```

### Visual C++

```c++
public:
static String^ RadioButton(
    String^ name,
    String^ id,
    Object^ value,
    bool isSelected,
    ... array<String^>^ htmlAttributes
)
```

## Parameters

- **name**
  - Type: `System::String`
  - element name

- **id**
Type: System::String
element id
value
Type: System::Object
element value
isSelected
Type: System::Boolean
true if radiobutton is selected
htmlAttributes
Type: array< System::String >[]
a list with additional attributes (name, value, name, value).

**Return Value**

a generated radio button
See Also

FormHelper Class
RadioButton Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select(String, IEnumerable, GetIdTitle, Object, Boolean)</td>
<td>Creates a select list with the values in a collection.</td>
</tr>
<tr>
<td>Select(String, String, IEnumerable, GetIdTitle, Object, Boolean)</td>
<td>Creates a select list with the values in a collection.</td>
</tr>
<tr>
<td>Select(String, String, IEnumerable, GetIdTitle, Object, Boolean, array&lt;String&gt;[])</td>
<td>Creates a select list with the values in a collection.</td>
</tr>
</tbody>
</table>
See Also

FormHelper Class
FormHelper Members
HttpGetServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic

C# Webserver

FormHelper....Select Method (String, IEnumerable, GetIdTitle, Object, Boolean)

FormHelper Class  See Also  Send Feedback

Creates a select list with the values in a collection.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string Select(
    string name,
    IEnumerable collection,
    GetIdTitle getTitle,
    Object selectedValue,
    bool firstEmpty
)

Visual Basic (Declaration)

Public Shared Function Select ( _
    name As String, _
    collection As IEnumerable, _
    getIdTitle As GetIdTitle, _
    selectedValue As Object, _
    firstEmpty As Boolean _
) As String

Visual C++

public:
static String^ Select(
    String^ name,
    IEnumerable^ collection,
    GetIdTitle^ getTitle,
    Object^ selectedValue,
    bool firstEmpty
)

Parameters

name
   Type: System::String
   Name of the SELECT-tag

collection
Type: `System.Collections::IEnumerable`
collection used to generate options.

**getIdTitle**
Type: `HttpServer.Helpers::GetIdTitle`
delegate used to return id and title from objects.

**selectedValue**
Type: `System::Object`
value that should be marked as selected.

**firstEmpty**
Type: `System::Boolean`
First row should contain an empty value.

**Return Value**

string contain a SELECT-tag.
See Also

FormHelper Class
Select Overload
HttpServer.Helpers Namespace
HttpServer.Helpers...:::GetIdTitle

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormHelper.:::Select Method (String, String, IEnumerable, GetIdTitle, Object, Boolean)

FormsHelper Class  Example  See Also  Send Feedback

Creates a select list with the values in a collection.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string Select(
    string name,
    string id,
    IEnumerable collection,
    GetIdTitle getIdTitle,
    Object selectedValue,
    bool firstEmpty
)

Visual Basic (Declaration)

Public Shared Function Select (
    name As String ,
    id As String ,
    collection As IEnumerable ,
    getIdTitle As GetIdTitle ,
    selectedValue As Object ,
    firstEmpty As Boolean )
    As String

Visual C++

public:
static String^ Select(
    String^ name,
    String^ id,
    IEnumerable^ collection,
    GetIdTitle^ getIdTitle,
    Object^ selectedValue,
    bool firstEmpty
)

Parameters

name
    Type: System::String
    Name of the SELECT-tag
id
  Type: System::String
  Id of the SELECT-tag

collection
  Type: System.Collections::IEnumerable
  collection used to generate options.

getIdTitle
  Type: HttpServer.Helpers::GetIdTitle
delegate used to return id and title from objects.

selectedValue
  Type: System::Object
  value that should be marked as selected.

firstEmpty
  Type: System::Boolean
  First row should contain an empty value.

Return Value

string contain a SELECT-tag.
Examples

C#

```csharp
// Class that is going to be used in a SELECT-tag.
public class User
{
    private readonly string _realName;
    private readonly int _id;
    public User(int id, string realName)
    {
        _id = id;
        _realName = realName;
    }
    public string RealName
    {
        get { return _realName; }
    }
    public int Id
    {
        get { return _id; }
    }
}

// Using an inline delegate to generate the select list
public void UserInlineDelegate()
{
    List<User> items = new List<User>();
    items.Add(new User(1, "adam"));
    items.Add(new User(2, "bertial"));
    items.Add(new User(3, "david"));
    string htmlSelect = Select("users", "users", items, delegate(obj
    {
        User use
        id = use
        value =
    }, 2, true);
}

// Using an method as delegate to generate the select list.
public void UseExternalDelegate()
{
    List<User> items = new List<User>();
    items.Add(new User(1, "adam"));
    items.Add(new User(2, "bertial"));
    items.Add(new User(3, "david"));
    string htmlSelect = Select("users", "users", items, delegate(obj
    {
        User use
        id = use
        value =
    }, 2, true);
```
items.Add(new User(2, "bertial"));
items.Add(new User(3, "david"));
string htmlSelect = Select("users", "users", items, UserOptions,}

// delegate returning id and title
public static void UserOptions(object o, out object id, out object t
{
    User user = (User)o;
id = user.Id;
value = user.RealName;
}
See Also

FormHelper Class
Select Overload
HttpServer.Helpers Namespace
HttpServer.Helpers...:::GetIdTitle

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormsHelper...::Select Method (String, String, IEnumerable, GetIdTitle, Object, Boolean, array<String>[], [])

Creates a select list with the values in a collection.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string Select(
    string name,
    string id,
    IEnumerable collection,
    GetIdTitle getIdTitle,
    Object selectedValue,
    bool firstEmpty,
    params string[] htmlAttributes
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function Select ( _
    name As String, _
    id As String, _
    collection As IEnumerable, _
    getIdTitle As GetIdTitle, _
    selectedValue As Object, _
    firstEmpty As Boolean, _
    ParamArray htmlAttributes As String() _
) As String
```

Visual C++

```cpp
public:
static String^ Select(
    String^ name,
    String^ id,
    IEnumerable^ collection,
    GetIdTitle^ getIdTitle,
    Object^ selectedValue,
    bool firstEmpty,
    ... array<String^>^ htmlAttributes
)
```

Parameters

name
Type: `System.String`  
Name of the SELECT-tag

id  
Type: `System.String`  
Id of the SELECT-tag

collection  
Type: `System.Collections.IEnumerable`  
collection used to generate options.

getIdTitle  
Type: `HttpServer.Helpers.GetIdTitle`  
delegate used to return id and title from objects.

selectedIndex  
Type: `System.Object`  
value that should be marked as selected.

firstEmpty  
Type: `System.Boolean`  
First row should contain an empty value.

htmlAttributes  
Type: array< `System.String` >[]()  
name, value collection of extra html attributes.

**Return Value**

string contain a SELECT-tag.
See Also

FormHelper Class  
Select Overload  
HttpContext.Helpers Namespace  
HttpContext.Helpers::GetIdTitle  

This project is maintained by Gauffin Telecom AB  

(c) 2008 Gauffin Telecom AB
FormHelper::Start Method

FormHelper Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Start(String, String, Boolean)</code></td>
<td>Create a <code>&lt;form&gt;</code> tag.</td>
</tr>
<tr>
<td><code>Start(String, String, Boolean, String)</code></td>
<td>Create a <code>&lt;form&gt;</code> tag.</td>
</tr>
</tbody>
</table>
See Also

- FormHelper Class
- FormHelper Members
- HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a <form> tag.

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string Start(
    string name,
    string action,
    bool isAjax
)

Visual Basic (Declaration)

Public Shared Function Start ( _
    name As String, _
    action As String, _
    isAjax As Boolean _
) As String

Visual C++

public:
    static String^ Start(
        String^ name,
        String^ action,
        bool isAjax
    )

Parameters

name
    Type: System::String
    name of form

action
    Type: System::String
    action to invoke on submit

isAjax
    Type: System::Boolean
form should be posted as ajax

**Return Value**

html code
Examples

WebHelper.FormStart("frmLogin", "/user/login", Request.IsAjax);
See Also

FormHelper Class
Start Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a `<form>` tag.

**Namespace:**  [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string Start(
    string name,
    string action,
    bool isAjax,
    string onsubmit
)

Visual Basic (Declaration)

Public Shared Function Start ( _
    name As String, _
    action As String, _
    isAjax As Boolean, _
    onsubmit As String _
) As String

Visual C++

public:
static String^ Start(
    String^ name,
    String^ action,
    bool isAjax,
    String^ onsubmit
)

Parameters

name
Type: System:::String
name of form

action
Type: System:::String
action to invoke on submit
isAjax
  Type: System::Boolean
  form should be posted as ajax

onsubmit
  Type: System::String
  javascript to perform when form is submitted.

Return Value

html code
Examples

WebHelper.FormStart("frmLogin", "/user/login", Request.IsAjax);
See Also

FormHelper Class
Start Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validator is used to validate all input items in a form.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class FormValidator : Validator

Visual Basic (Declaration)

Public Class FormValidator _
    Inherits Validator

Visual C++

public ref class FormValidator : public Validator
Inheritance Hierarchy

System::Object
HttpServer.Helpers::Validator
HttpServer.Helpers::FormValidator
See Also

FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormValidator` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormValidator</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>AlphaNumeric</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Boolean</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Char</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Digits</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Email</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Format</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Hex</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Integer</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Letters</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>LettersExtra</strong></td>
<td>Validate that a string consists of only letters (a-z and A-Z)</td>
</tr>
<tr>
<td><strong>LettersOrDigits</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Required</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>SetDefaultMgr</strong></td>
<td>Set default language. (Inherited from <a href="#">Validator</a>)</td>
</tr>
<tr>
<td><strong>SetForm</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
- **String** Overloaded.
- **ToString** Returns a **String** that represents the current **Object**. (Inherited from **Object**.)
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_errors</td>
<td>A list with all errors found.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>.)</td>
</tr>
<tr>
<td>_modelLang</td>
<td>Language translation for all &quot;name&quot; parameters.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>.)</td>
</tr>
<tr>
<td>_validatorLang</td>
<td>Language for the validator (i.e. &quot;Field {0} is required.&quot;)</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContainsErrors</td>
<td>true of validation generated errors. (Inherited from <a href="#">Validator</a>)</td>
</tr>
<tr>
<td>Errors</td>
<td>Collection of validation errors. (Inherited from <a href="#">Validator</a>)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator Constructor

FormValidator Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FormValidator(HttpInputBase)</td>
<td>Initializes a new instance of the FormValidator class.</td>
</tr>
<tr>
<td>FormValidator(LanguageNode)</td>
<td>Initializes a new instance of the FormValidator class.</td>
</tr>
<tr>
<td>FormValidator(NameValueCollection)</td>
<td>Initializes a new instance of the FormValidator class.</td>
</tr>
<tr>
<td>FormValidator(HttpInputBase, NameValueCollection)</td>
<td>Initializes a new instance of the FormValidator class.</td>
</tr>
<tr>
<td>FormValidator(NameValueCollection, LanguageNode)</td>
<td>Initializes a new instance of the FormValidator class.</td>
</tr>
<tr>
<td>FormValidator(HttpInputBase, NameValueCollection, LanguageNode)</td>
<td>Initializes a new instance of the FormValidator class.</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class  
FormValidator Members  
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator Constructor (HttpInputBase)

Initializes a new instance of the FormValidator class

**Namespace:**  HttpServer.Helpers
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public FormValidator(
    HttpInputBase form
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New ( _
    form As HttpInputBase _
)
```

Visual C++

```cpp
public:
FormValidator(
    HttpInputBase^ form
)
```

Parameters

form

Type: HttpServer...:HttpInputBase
form that validation should be made on.
See Also

FormValidator Class
FormValidator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

FormValidator Constructor ()

Initializes a new instance of the `FormValidator` class.

**Namespace:**  [HttpContext](https://example.com)  
**Assembly:**  [HttpContext](https://example.com) (in HttpContext.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public FormValidator(
    LanguageNode modelLanguage
)

Visual Basic (Declaration)

Public Sub New ( _
    modelLanguage As LanguageNode _
)

Visual C++

public:
FormValidator(
    LanguageNode^ modelLanguage
)

Parameters

modelLanguage
    Type: LanguageNode
    Translation used to translate the "name" parameters in all validation methods.
See Also

FormValidator Class
FormValidator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the **FormValidator** class.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public FormValidator(
    NameValueCollection errors
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New ( _
    errors As NameValueCollection _
)
```

**Visual C++**

```cpp
public:
FormValidator(
    NameValueCollection^ errors
)
```

### Parameters

errors

Type: `System.Collections.Specialized.NameValueCollection`

collection to be filled with errors
See Also

FormValidator Class
FormValidator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator Constructor (HttpInputBase, NameValueCollection)

Initializes a new instance of the FormValidator class

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
syntax

C#

public FormValidator(
    HttpInputBase form,
    NameValueCollection errors
)

Visual Basic (Declaration)

Public Sub New ( _
    form As HttpInputBase, _
    errors As NameValueCollection _
)

Visual C++

public:
FormValidator(
    HttpInputBase^ form,
    NameValueCollection^ errors
)

Parameters

form
Type: HttpServer:::HttpInputBase
form that validation should be made on.

errors
Type: System.Collections.Specialized:::NameValueCollection
collection that all validation errors are added to.
See Also

FormValidator Class
FormValidator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `FormValidator` class.

**Namespace:** [HttpServer.Helpers](HttpServer.Helpers)

**Assembly:** [HttpServer](HttpServer.dll) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public FormValidator(
    NameValueCollection errors,
    LanguageNode modelLanguage
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    errors As NameValueCollection, _
    modelLanguage As LanguageNode _
)
```

**Visual C++**

```cpp
public:
FormValidator(
    NameValueCollection^ errors,
    LanguageNode^ modelLanguage
)
```

**Parameters**

**errors**
   Type: System.Collections.Specialized.NameValueCollection
collection to be filled with errors

**modelLanguage**
   Type: LanguageNode
   Translation used to translate the "name" parameters in all validation methods.
See Also

FormValidator Class
FormValidator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator Constructor (HttpInputBase, NameValueCollection, )

**FormValidator Class**  **See Also**  **Send Feedback**

Initializes a new instance of the **FormValidator** class

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)](#)
## Syntax

### C#

```csharp
public FormValidator(
    HttpInputBase form,
    NameValueCollection errors,
    LanguageNode modelLanguage
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New (_
    form As HttpInputBase, _
    errors As NameValueCollection, _
    modelLanguage As LanguageNode _
)
```

### Visual C++

```cpp
public:
FormValidator(
    HttpInputBase^ form,
    NameValueCollection^ errors,
    LanguageNode^ modelLanguage
)
```

## Parameters

**form**

Type: `HttpServer::::HttpInputBase`

form that validation should be made on.

**errors**

Type: `System.Collections.Specialized::::NameValueCollection`

collection that all validation errors are added to.

**modelLanguage**

Type: `LanguageNode`
Language category used to translate field names.
See Also

FormValidator Class
FormValidator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormValidator` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_errors</td>
<td>A list with all errors found.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>).</td>
</tr>
<tr>
<td>_modelLang</td>
<td>Language translation for all &quot;name&quot; parameters.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>).</td>
</tr>
<tr>
<td>_validatorLang</td>
<td>Language for the validator (i.e. &quot;Field {0} is required.&quot;).</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>).</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FormValidator` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AlphaNumeric</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Boolean</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Char</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Digits</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Email</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Format</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Hex</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Integer</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Letters</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>LettersExtra</strong></td>
<td>Validate that a string consists of only letters (a-z and A-Z)</td>
</tr>
<tr>
<td><strong>LettersOrDigits</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Required</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>SetDefaultMgr</strong></td>
<td>Set default language. (Inherited from <a href="#">Validator</a>.)</td>
</tr>
<tr>
<td><strong>SetForm</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
String

Overloaded.

ToString

Returns a String that represents the current Object. (Inherited from Object.)
See Also

FormValidator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members  C# Webserver  FormValidator...:.AlphaNumeric Method  FormValidator Class  See Also  Send Feedback
### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlphaNumeric(String)</td>
<td>Check whether the specified value is an integer.</td>
</tr>
<tr>
<td>AlphaNumeric(String, Boolean)</td>
<td>Check whether the specified value is an integer.</td>
</tr>
<tr>
<td>AlphaNumeric(String, String, Boolean)</td>
<td>Check whether the specified value is an integer. (Inherited from Validator.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an integer.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public string AlphaNumeric(
    string name
)
```

### Visual Basic (Declaration)

```vbnet
Public Function AlphaNumeric ( _
    name As String _
) As String
```

### Visual C++

```cpp
public:
    String^ AlphaNumeric(
        String^ name
    )
```

## Parameters

- **name**
  - Type: `System::String`
  - Name of the parameter

## Return Value

- value if parameter contains valid characters; string.Empty if not.
See Also

FormValidator Class
AlphaNumeric Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator.:::AlphaNumeric Method (String, Boolean)

Check whether the specified value is an integer.

**Namespace:** [HttpServer.Helpers](#)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string AlphaNumeric(
    string name,
    bool required
)

Visual Basic (Declaration)

Public Function AlphaNumeric (_
    name As String, _
    required As Boolean _
) As String

Visual C++

public: String^ AlphaNumeric(
    String^ name,
    bool required
)

Parameters

name
    Type: System::String
    Name of the parameter

required
    Type: System::Boolean
    Parameter is required (adds an error if it's not specified)

Return Value

value if parameter contains valid characters; string.Empty if not.
See Also

FormValidator Class
AlphaNumeric Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator::Boolean Method

FormValidator Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean(String)</td>
<td>Checks whether a field is true (can also be in native language).</td>
</tr>
<tr>
<td>Boolean(String, Boolean)</td>
<td>Checks whether a field is true (can also be in native language).</td>
</tr>
<tr>
<td>Boolean(String, String, Boolean)</td>
<td>Checks whether a field contains true (can also be in native language).</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Validator</a>.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator...:::Boolean Method (String)

See Also: Send Feedback

Checks whether a field is true (can also be in native language).

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Boolean(
    string name
)

Visual Basic (Declaration)

Public Function Boolean ( _
    name As String _
) As Boolean

Visual C++

public:
    bool Boolean(
        String^ name
    )

Parameters

name

    Type: System::::String
    field name

Return Value

true if value is true; false if value is false or if validation failed.
Remarks

Check validation errors to see if error occurred.
See Also

FormValidator Class
Boolean Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Checks whether a field is true (can also be in native language).

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)](#)
Syntax

C#

public bool Boolean(
    string name,
    bool required
)

Visual Basic (Declaration)

Public Function Boolean ( _
    name As String, _
    required As Boolean _
) As Boolean

Visual C++

public:
    bool Boolean(
        String^ name,
        bool required
    )

Parameters

name
    Type: System::String
    field name

required
    Type: System::Boolean
    field is required (may not be null or empty).

Return Value

true if value is true; false if value is false or if validation failed.
Remarks

Check validation errors to see if error occurred.
See Also

FormValidator Class
Boolean Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator.Char Method

See Also Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Char(String)</td>
<td>Check whether the specified value is an character.</td>
</tr>
<tr>
<td>Char(String, Boolean)</td>
<td>Check whether the specified value is an character.</td>
</tr>
<tr>
<td>Char(String, String)</td>
<td>Check whether the specified value is an character.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Validator.)</td>
</tr>
<tr>
<td>Char(String, String, Boolean)</td>
<td>Check whether the specified value is an character.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Validator.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an character.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public char Char(
    string name
)

Visual Basic (Declaration)

Public Function Char ( _
    name As String _
) As Char

Visual C++

public:
    wchar_t Char(
        String^ name
    )

Parameters

name
    Type: System::String
    Name of the parameter

Return Value

value if parameter is an int; char.MinValue if not.
See Also

FormValidator Class
Char Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an character.

**Namespace:**  [HttpServer.Helpers]

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public char Char(
    string name,
    bool required
)

Visual Basic (Declaration)

Public Function Char ( _
    name As String, _
    required As Boolean _
) As Char

Visual C++

public:
    wchar_t Char(
        String^ name,
        bool required
    )

Parameters

name
    Type: System::String
    Name of the parameter

required
    Type: System::Boolean
    Parameter is required (adds an error if it's not specified)

Return Value

value if parameter is an int; char.MinValue if not.
See Also

FormValidator Class
Char Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator::Digits Method

FormValidator Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digits(String, Boolean)</td>
<td>Check if a value is digits only</td>
</tr>
<tr>
<td>Digits(String, Boolean, String)</td>
<td>Check if a value is digits only</td>
</tr>
<tr>
<td>Digits(String, String, Boolean)</td>
<td>Check if a value is digits only (Inherited from <a href="#">Validator</a>)</td>
</tr>
<tr>
<td>Digits(String, String, Boolean, String)</td>
<td>Check if a value is digits only (Inherited from <a href="#">Validator</a>)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check if a value is digits only

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll)  Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Digits(
    string name,
    bool required
)

Visual Basic (Declaration)

Public Function Digits ( _
    name As String, _
    required As Boolean _
) As String

Visual C++

public:
    String^ Digits(
        String^ name,
        bool required
    )

Parameters

name
    Type: System::String
    Field name.

required
    Type: System::Boolean
    true if field is required (may not be empty)

Return Value

string if validated, otherwise string.Empty
See Also

FormValidator Class
Digits Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check if a value is digits only

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Digits(
    string name,
    bool required,
    string extraAllowedCharacters
)
```

Visual Basic (Declaration)

```vbnet
Public Function Digits (_
    name As String, _
    required As Boolean, _
    extraAllowedCharacters As String _
) As String
```

Visual C++

```cpp
public: 
String^ Digits(
    String^ name,
    bool required,
    String^ extraAllowedCharacters
)
```

Parameters

name
Type: `System::::String`
Field name.

required
Type: `System::::Boolean`
true if field is required (may not be empty)

extraAllowedCharacters
Type: `System::::String`
extra characters that is allowed.

**Return Value**

string if validated, otherwise string.Empty
See Also

FormValidator Class
Digits Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator...::Email Method

FormValidator Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email(String, Boolean)</td>
<td>validates email address using a regexp.</td>
</tr>
</tbody>
</table>
| Email(String, String, Boolean) | validates email address using a regexp.  
(Inherited from [Validator](#)). |
See Also

FormValidator Class
FormValidator Members
HttpContext.Validators Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
validates email address using a regexp.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#)

Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Email(
    string name,
    bool required
)

Visual Basic (Declaration)

Public Function Email ( _
    name As String, _
    required As Boolean _
) As String

Visual C++

public: string Email(
    String^ name,
    bool required
)

Parameters

name
    Type: System::String
    field name

required
    Type: System::Boolean
    field is required (may not be null or empty).

Return Value

value if validation is ok; otherwise string.Empty.
See Also

FormValidator Class
Email Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator::Format Method

FormValidator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format(String, String)</td>
<td>format an error message (Inherited from <a href="#">Validator</a>)</td>
</tr>
<tr>
<td>Format(String, String, String)</td>
<td>format an error message (Inherited from <a href="#">Validator</a>)</td>
</tr>
<tr>
<td>Format(String, String, String, String)</td>
<td>format an error message (Inherited from <a href="#">Validator</a>)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members  C# Webserver  FormValidator...:Hex Method  

FormValidator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex(String, Boolean)</td>
<td>Validates a string to hex</td>
</tr>
<tr>
<td>Hex(String, String, Boolean)</td>
<td>Validates a string to be in trimmed hex letters</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Validator.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServletRequest Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator...:::Hex Method (String, Boolean)

Validates a string to hex

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
class Example {
    public string Hex(
        string name,
        bool required
    )
```

### Visual Basic (Declaration)

```vbnet
Public Function Hex ( _
    name As String,
    required As Boolean _
) As String
```

### Visual C++

```cpp
public:
    String^ Hex(
        String^ name,
        bool required
    )
```

## Parameters

**name**
- Type: `System::String`
- The name of the field to validate

**required**
- Type: `System::Boolean`
- If the field must be set

## Return Value

The value if validated otherwise string.Empty
See Also

FormValidator Class
Hex Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator::Integer Method

FormValidator Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integer(String)</td>
<td>Check whether the specified form item is an integer.</td>
</tr>
<tr>
<td>Integer(String, Boolean)</td>
<td>Check whether the specified form item is an integer.</td>
</tr>
<tr>
<td>Integer(String, String, Boolean)</td>
<td>Check whether the specified value is an integer. (Inherited from Validator.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified form item is an integer.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0)
# Syntax

**C#**

```csharp
public int Integer(
    string name
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Integer ( _
    name As String _
) As Integer
```

**Visual C++**

```cpp
public:
    int Integer(
        String^ name
    )
```

## Parameters

- **name**
  - Type: `System::String`
  - Form parameter to validate

## Return Value

- value if parameter is an int; 0 if not.
See Also

FormValidator Class
Integer Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified form item is an integer.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public int Integer(
    string name,
    bool required
)
```

### Visual Basic (Declaration)

```vbnet
Public Function Integer ( _
    name As String, _
    required As Boolean _
) As Integer
```

### Visual C++

```cpp
public: int Integer(
    String^ name,
    bool required
)
```

## Parameters

- **name**
  - Type: `System::::String`
  - Form parameter to validate

- **required**
  - Type: `System::::Boolean`
  - Parameter is required (adds an error if it's not specified)

## Return Value

- Value if parameter is an int; 0 if not.
See Also

FormValidator Class
Integer Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Letters(String, Boolean)</strong></td>
<td>Validate that a string consists of only letters (including special letters)</td>
</tr>
<tr>
<td><strong>Letters(String, String, Boolean)</strong></td>
<td>Validates a string to consist of letters. Will test for A-Z, a-z, spaces and letters specified in SpecialLetters (Inherited from Validator.)</td>
</tr>
<tr>
<td><strong>Letters(String, String, Boolean)</strong></td>
<td>Validates a string to consist of letters plus specified characters. Basic allowed letters are A-Z and a-z. For extra characters SpecialLetters (Inherited from Validator.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate that a string consists of only letters (including special letters)

**Namespace:**  [HttpServer.Helpers](http://localhost:4200/HttpServer.Helpers)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Letters(
    string name,
    bool required
)
```

Visual Basic (Declaration)

```vbnet
Public Function Letters ( _
    name As String, _
    required As Boolean _
) As String
```

Visual C++

```cpp
public:
    String^ Letters(
        String^ name,
        bool required
    )
```

Parameters

name
Type: System:::String

required
Type: System:::Boolean
If a value must be passed

Return Value

[Missing <returns> documentation for
See Also

FormValidator Class
Letters Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic Visual C++
C# Webserver
FormValidator..:::LettersExtra Method

FormValidator Class  See Also  Send Feedback

Validate that a string consists of only letters (a-z and A-Z)

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string LettersExtra(
    string name,
    string extraCharacters,
    bool required
)

Visual Basic (Declaration)

Public Function LettersExtra ( _
    name As String, _
    extraCharacters As String, _
    required As Boolean _
) As String

Visual C++

public:
String^ LettersExtra(
    String^ name,
    String^ extraCharacters,
    bool required
)

Parameters

name
Type: System::String

[Missing <param name="name"/> documentation for

extraCharacters
Type: System::String
A string of extra character to test against, dont forget language specific characters and spaces if wished for
required

Type: System::Boolean
If a value must be passed

Return Value

See Also

FormValidator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator::LettersOrDigits Method
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LettersOrDigits(String)</td>
<td>Validate that a string only contains letters or digits.</td>
</tr>
<tr>
<td>LettersOrDigits(String, Boolean)</td>
<td>Validate that a string only contains letters or digits.</td>
</tr>
<tr>
<td>LettersOrDigits(String, String, Boolean)</td>
<td>Validate that a string only contains letters or digits. (Inherited from Validator.)</td>
</tr>
</tbody>
</table>

(Inherited from Validator.)
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate that a string only contains letters or digits.

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string LettersOrDigits(
    string name
)

Visual Basic (Declaration)

Public Function LettersOrDigits ( _
    name As String _
) As String

Visual C++

public:
    String^ LettersOrDigits(
        String^ name
    )

Parameters

name
    Type: System::String
    Form parameter name.

Return Value

value if found; otherwise string.Empty
See Also

FormValidator Class
LettersOrDigits Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate that a string only contains letters or digits.

**Namespace:** [HttpServer.Helpers](http://localhost:8080/#httpserver.helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string LettersOrDigits(
    string name,
    bool required
)

Visual Basic (Declaration)

Public Function LettersOrDigits ( _
    name As String, _
    required As Boolean _
) As String

Visual C++

public:
String^ LettersOrDigits(
    String^ name,
    bool required
)

Parameters

name
    Type: System::String
    Name of form parameter to validate.

required
    Type: System::Boolean
    Value is required.

Return Value

value if valid; otherwise string.EmptyLanguageNode.
See Also

FormValidator Class
LettersOrDigits Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator...::Password Method

FormValidator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Password(String)</code></td>
<td>Validate that a string only contains letters or digits or any of the <code>PasswordChars</code>.</td>
</tr>
<tr>
<td><code>Password(String, Boolean)</code></td>
<td>Validate that a string only contains letters or digits or any of the <code>PasswordChars</code>.</td>
</tr>
<tr>
<td><code>Password(String, String, Boolean)</code></td>
<td>Validate that a string only contains letters or digits or any of the <code>PasswordChars</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Validator</code>.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate that a string only contains letters or digits or any of the PasswordChars.

**Namespace:**  [HttpServer.Helpers](http://example.com)

**Assembly:**  [HttpServer](http://example.com) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public string Password(
    string name
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Password ( _
    name As String _
) As String
```

**Visual C++**

```cpp
public:
String^ Password(
    String^ name
)
```

### Parameters

**name**

Type: `System::String`

Form parameter name.

### Return Value

Value if found; otherwise `string.Empty`
See Also

FormValidator Class
Password Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator..::.Password Method (String, Boolean)
See Also Send Feedback

Validate that a string only contains letters or digits or any of the PasswordChars.

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Password(
    string name,
    bool required
)
```

Visual Basic (Declaration)

```vbnet
Public Function Password ( _
    name As String, _
    required As Boolean _
) As String
```

Visual C++

```cpp
public:
String^ Password(
    String^ name,
    bool required
)
```

Parameters

name
Type: `System::String`
Name of form parameter to validate.

required
Type: `System::Boolean`
Value is required.

Return Value

value if valid; otherwise string.Empty.
See Also

FormValidator Class
Password Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FormValidator::Required Method

FormValidator Class  See Also  Send Feedback
### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required(String)</td>
<td>Check's weather a parameter is null or not.</td>
</tr>
<tr>
<td>Required(String, String)</td>
<td>Check's weather a parameter is null or not.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <strong>Validator</strong>.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check's weather a parameter is null or not.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Required(
  string name
)

Visual Basic (Declaration)

Public Function Required ( _
  name As String _
) As Boolean

Visual C++

public:
bool Required(
  String^ name
)

Parameters

name
  Type: System::String
  Parameter in form

Return Value

true if value is not null; otherwise false.
See Also

FormValidator Class
Required Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# | Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
FormValidator::SetForm Method
FormValidator Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SetForm(HttpInputBase)</td>
<td>Switch to a new http input.</td>
</tr>
<tr>
<td>SetForm(HttpInputBase, LanguageNode)</td>
<td>Switch to a new http input.</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Switch to a new http input.

**Namespace:**  [HttpServer.Helpers](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void SetForm(
    HttpInputBase form
)

Visual Basic (Declaration)

Public Sub SetForm ( _
    form As HttpInputBase _
)

Visual C++

public:
void SetForm(
    HttpInputBase^ form
)

Parameters

form
    Type: HttpServer:::HttpInputBase
    form to use validation for now
See Also

FormValidator Class
SetForm Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Switch to a new http input.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer](#)  (in HttpServer.dll)  Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void SetForm(
    HttpInputBase form,
    LanguageNode modelLanguage
)

Visual Basic (Declaration)

Public Sub SetForm ( _
    form As HttpInputBase, _
    modelLanguage As LanguageNode _
)

Visual C++

public:
void SetForm(
    HttpInputBase^ form,
    LanguageNode^ modelLanguage
)

Parameters

form
    Type: HttpServer::HttpInputBase
    form to use validation for now

modelLanguage
    Type: LanguageNode
    language for the validation
See Also

FormValidator Class
SetForm Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>String(String)</code></td>
<td>Validate a string parameter in the form</td>
</tr>
<tr>
<td><code>String(String, Boolean)</code></td>
<td>Validate a string value</td>
</tr>
<tr>
<td><code>String(String, String, Boolean)</code></td>
<td>Validate a string parameter in the form (Inherited from <a href="#">Validator</a>)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
FormValidator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate a string parameter in the form

**Namespace:** [HttpClient.Helpers](https://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string String(
    string name
)

Visual Basic (Declaration)

Public Function String (_
    name As String _
) As String

Visual C++

public:
String^ String(
    String^ name
)

Parameters

name

    Type: System::String
    Form parameter name.

Return Value

vaue if found; otherwise string.Empty
See Also

FormValidator Class
String Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate a string value

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
# Syntax

## C#

```csharp
public string String(
    string name,
    bool required
)
```

## Visual Basic (Declaration)

```vbnet
Public Function String ( _
    name As String, _
    required As Boolean _
) As String
```

## Visual C++

```cpp
public: String^ String(
    String^ name,
    bool required
)
```

## Parameters

**name**
- Type: `System::String`
  Name of form parameter to validate.

**required**
- Type: `System::Boolean`
  Value is required.

## Return Value

value if valid; otherwise `string.Empty`. 
See Also

FormValidator Class
String Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **FormValidator** type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContainsErrors</td>
<td>true of validation generated errors. (Inherited from <a href="#">Validator</a>.)</td>
</tr>
<tr>
<td>Errors</td>
<td>Collection of validation errors. (Inherited from <a href="#">Validator</a>.)</td>
</tr>
</tbody>
</table>
See Also

FormValidator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Delegate used by `Select(String, IEnumerable, GetIdTitle, Object, Boolean)` to populate select options.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public delegate void GetIdTitle(
    object obj,
    out object id,
    out string title
)
```

### Visual Basic (Declaration)

```vbnet
Public Delegate Sub GetIdTitle (_
    obj As Object, _
    <OutAttribute> ByRef id As Object, _
    <OutAttribute> ByRef title As String _
)
```

### Visual C++

```cpp
public delegate void GetIdTitle(
    object^ obj,
    [OutAttribute] object^% id,
    [OutAttribute] string^% title
)
```

### Parameters

- **obj**
  - Type: `System::Object`
  - current object (for instance a User).

- **id**
  - Type: `System::Object %`
  - Text that should be displayed in the value part of a `<option>`-tag.

- **title**
  - Type: `System::String %`
  - Text shown in the select list.
Examples

// Class that is going to be used in a SELECT-tag. public class User { private readonly string _realName; private readonly int _id; public User(int id, string realName) { _id = id; _realName = realName; } public string RealName { get { return _realName; } } public int Id { get { return _id; } } } // Using an inline delegate to generate the select list public void UserInlineDelegate() { List<User> items = new List<User>(); items.Add(new User(1, "adam"); items.Add(new User(2, "bertial"); items.Add(new User(3, "david"); string htmlSelect = Select("users", "users", items, delegate(object o, out object id, out object value) { User user = (User)o; id = user.Id; value = user.RealName; }, 2, true); } // Using an external method as delegate to generate the select list. public void UseExternalDelegate() { List<User> items = new List<User>(); items.Add(new User(1, "adam"); items.Add(new User(2, "bertial"); items.Add(new User(3, "david"); string htmlSelect = Select("users", "users", items, UserOptions, 1, true); } // delegate returning id and title public static void UserOptions(object o, out object id, out object title) { User user = (User)o; id = user.Id; value = user.RealName; } ///
See Also

HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Purpose of this class is to create a javascript toolkit independent javascript helper.

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract class JavascriptHelperImplementation

Visual Basic (Declaration)

Public MustInherit Class JavascriptHelperImplementation

Visual C++

public ref class JavascriptHelperImplementation abstract
Inheritance Hierarchy

System:::Object
HttpServer.Helpers:::JavascriptHelperImplementation
HttpServer.Helpers.Implementations:::PrototypeImp
See Also

JavascriptHelperImplementation Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The JavascriptHelperImplementation type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JavascriptHelperImplementation</td>
<td>Initializes a new instance of the JavascriptHelperImplementation class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AjaxFormOnSubmit</td>
<td>javascript action that should be added to the &quot;onsubmit&quot; event in the form tag.</td>
</tr>
<tr>
<td>AjaxRequest</td>
<td>Requests a url through ajax</td>
</tr>
<tr>
<td>AjaxUpdater</td>
<td>Ajax requests that updates an element with the fetched content</td>
</tr>
<tr>
<td>CloseDialog</td>
<td>Close a javascript dialog window/div.</td>
</tr>
<tr>
<td>CreateDialog</td>
<td>Creates a new modal dialog window</td>
</tr>
<tr>
<td>DialogLink</td>
<td>A link that pop ups a Dialog (overlay div)</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GenerateOptions</td>
<td>Generates a list with JS options.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  C# Webserver
JavascriptHelperImplementation Constructor

JavascriptHelperImplementation Class  See Also  Send Feedback

Initializes a new instance of the JavascriptHelperImplementation class

Namespace:  HttpServer.Helpers  
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected JavascriptHelperImplementation()

Visual Basic (Declaration)

Protected Sub New

Visual C++

protected:
JavascriptHelperImplementation()
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `JavascriptHelperImplementation` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AjaxFormOnSubmit</td>
<td>javascript action that should be added to the &quot;onsubmit&quot; event in the form tag.</td>
</tr>
<tr>
<td>AjaxRequest</td>
<td>Requests a url through ajax</td>
</tr>
<tr>
<td>AjaxUpdater</td>
<td>Ajax requests that updates an element with the fetched content</td>
</tr>
<tr>
<td>CloseDialog</td>
<td>Close a javascript dialog window/div.</td>
</tr>
<tr>
<td>CreateDialog</td>
<td>Creates a new modal dialog window</td>
</tr>
<tr>
<td>DialogLink</td>
<td>A link that pop ups a Dialog (overlay div)</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GenerateOptions</td>
<td>Generates a list with JS options.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
javascript action that should be added to the "onsubmit" event in the form tag.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  HttpServer (in HttpServer.dll)  
**Version:**  1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract string AjaxFormOnSubmit()

Visual Basic (Declaration)

Public MustOverride Function AjaxFormOnSubmit As String

Visual C++

public:
virtual String^ AjaxFormOnSubmit() abstract

Return Value

See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Requests a url through ajax

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public abstract string AjaxRequest(
    string url,
    params string[] options
)
```

**Visual Basic (Declaration)**

```vbnet
Public MustOverride Function AjaxRequest ( _
    url As String, _
    ParamArray options As String() _
) As String
```

**Visual C++**

```cpp
public:
virtual String^ AjaxRequest(
    String^ url,
    ... array<String^>^ options
) abstract
```

### Parameters

**url**

Type: `System::String`  
url to fetch

**options**

Type: `array< System::String >[][]`  
optional options in format "key, value, key, value", used in JS request object.

### Return Value

a link tag
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Ajax requests that updates an element with the fetched content

**Namespace:** [HttpServer.Helpers](http://example.com/HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public abstract string AjaxUpdater(
    string url,
    string targetId,
    params string[] options
)
```

**Visual Basic (Declaration)**

```vbnet
Public MustOverride Function AjaxUpdater ( _
    url As String, _
    targetId As String, _
    ParamArray options As String() _
) As String
```

**Visual C++**

```c++
public:
    virtual String^ AjaxUpdater(
        String^ url,
        String^ targetId,
        ... array<String^>^ options
    ) abstract
```

**Parameters**

**url**
- Type: `System::String`
  - Url to fetch content from

**targetId**
- Type: `System::String`
  - Element to update

**options**
- Type: `array< System::String >[][]`
optional options in format "key, value, key, value", used in JS updater object.

**Return Value**

A link tag.
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
CloseDialog Method

JavascriptHelperImplementation Class  See Also  Send Feedback

Close a javascript dialog window/div.

Namespace:  [HttpServer.Helpers](#)
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```
public abstract string CloseDialog()
```

**Visual Basic (Declaration)**

```
Public MustOverride Function CloseDialog As String
```

**Visual C++**

```
public:
virtual String & CloseDialog() abstract
```

### Return Value

JavaScript for closing a dialog.
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  
Visual C++  
C# Webserver  
JavascriptHelperImplementation..:::CreateDialog Method  
JavascriptHelperImplementation Class  See Also  Send Feedback

Creates a new modal dialog window

**Namespace:**  HttpServer.Helpers  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public abstract string CreateDialog(
    string url,
    string title,
    params string[] options
)
```

### Visual Basic (Declaration)

```vbnet
Public MustOverride Function CreateDialog (_
    url As String, _
    title As String, _
    ParamArray options As String() _
) As String
```

### Visual C++

```cpp
public:
    virtual String^ CreateDialog(
        String^ url,
        String^ title,
        ... array<String^>^ options
    ) abstract
```

## Parameters

### url

- **Type:** `System::String`
  - url to open in window.

### title

- **Type:** `System::String`
  - window title (may not be supported by all js implementations)

### options

- **Type:** `array< System::String >[]`
Return Value
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A link that pop ups a Dialog (overlay div)

Namespace:  **HttpServer.Helpers**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract string DialogLink(
    string url,
    string title,
    params string[] htmlAttributes
)

Visual Basic (Declaration)

Public MustOverride Function DialogLink ( _
    url As String, _
    title As String, _
    ParamArray htmlAttributes As String() _
) As String

Visual C++

public:
virtual String^ DialogLink(
    String^ url,
    String^ title,
    ... array<String^>[] htmlAttributes
)
abstract

Parameters

url
   Type: System:::String
   url to contents of dialog

title
   Type: System:::String
   link title

htmlAttributes
   Type: array< System:::String >[](null)
name/value of html attributes

Return Value

A "a"-tag that pops a dialog when clicked
Examples

WebHelper.DialogLink("/user/show/1", "show user", "onmouseover", "alert('booh!');");
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generates a list with JS options.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
protected virtual void GenerateOptions(
    StringBuilder sb,
    string[] options,
    bool startWithComma
)
```

### Visual Basic (Declaration)

```vbnet
Protected Overridable Sub GenerateOptions (_
    sb As StringBuilder, _
    options As String(), _
    startWithComma As Boolean _
)
```

### Visual C++

```cpp
protected:
virtual void GenerateOptions(
    StringBuilder^ sb,
    array<String^>^ options,
    bool startWithComma
)
```

## Parameters

### sb

Type: `System.Text.StringBuilder`  
StringBuilder that the options should be added to.

### options

Type: `array<System::String>[]`  


### startWithComma
Type: `System::Boolean` true if we should start with a comma.
See Also

JavascriptHelperImplementation Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Will contain helper functions for javascript.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)](#)
Syntax

C#

public static class JSHelper

Visual Basic (Declaration)

Public NotInheritable Class JSHelper

Visual C++

public ref class JSHelper abstract sealed
Inheritance Hierarchy

System...:::Object
HttpServer.Helpers...:::JSHelper
See Also

JSHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **JSHelper** type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AjaxRequest</td>
<td>Requests a url through ajax</td>
</tr>
<tr>
<td>AjaxUpdater</td>
<td>Ajax requests that updates an element with the fetched content</td>
</tr>
<tr>
<td>CloseDialog</td>
<td>Close a javascript dialog window/div.</td>
</tr>
<tr>
<td>CreateDialog</td>
<td>Opens contents in a dialog window.</td>
</tr>
</tbody>
</table>
See Also

JSHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The JSHelper type exposes the following members.
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AjaxRequest</strong></td>
<td>Requests a url through ajax</td>
</tr>
<tr>
<td><strong>AjaxUpdater</strong></td>
<td>Ajax requests that updates an element with the fetched content</td>
</tr>
<tr>
<td><strong>CloseDialog</strong></td>
<td>Close a javascript dialog window/div.</td>
</tr>
<tr>
<td><strong>CreateDialog</strong></td>
<td>Opens contents in a dialog window.</td>
</tr>
</tbody>
</table>
See Also

JSHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Requests a url through ajax

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string AjaxRequest(
    string url,
    params string[] options
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function AjaxRequest (_
    url As String, _
    ParamArray options As String() _
) As String
```

Visual C++

```cpp
public:
static String^ AjaxRequest(
    String^ url,
    ... array<String^>^ options
)
```

Parameters

url

Type: `System::::String`  
url to fetch

options

Type: array< `System::::String` >[()][]  
optional options in format "key, value, key, value", used in JS request object.

Return Value

a link tag
Remarks

onclick attribute is used by this method.
See Also

JSHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Ajax requests that updates an element with the fetched content

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string AjaxUpdater(
    string url,
    string targetId,
    params string[] options
)

Visual Basic (Declaration)

Public Shared Function AjaxUpdater ( _
    url As String, _
    targetId As String, _
    ParamArray options As String() _
) As String

Visual C++

public:
static String^ AjaxUpdater(
    String^ url,
    String^ targetId,
    ... array<String^>^ options
)

Parameters

url
Type: System::String
Url to fetch content from

targetId
Type: System::String
element to update

options
Type: array< System::String >[]()}
options in format "key, value, key, value"

Return Value

A link tag.
See Also

JSHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Close a javascript dialog window/div.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string CloseDialog()

Visual Basic (Declaration)

Public Shared Function CloseDialog As String

Visual C++

public:
static String^ CloseDialog()

Return Value

javascript for closing a dialog.
See Also

JSHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Opens contents in a dialog window.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public static string CreateDialog(
    string url,
    string title,
    params string[] options
)
```

### Visual Basic (Declaration)

```vbnet
Public Shared Function CreateDialog (_
    url As String, _
    title As String, _
    ParamArray options As String() _
) As String
```

### Visual C++

```cpp
public:
static String^ CreateDialog(
    String^ url,
    String^ title,
    ... array<String^>^ options
)
```

## Parameters

### url

Type: **System::String**

url to contents of dialog

### title

Type: **System::String**

link title

### options

Type: array< **System::String** >[][]
name, value, name, value, all parameter names should end with colon.

Return Value

See Also

JSHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The object form class takes an object and creates form items for it.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ObjectForm

Visual Basic (Declaration)

Public Class ObjectForm

Visual C++

public ref class ObjectForm
Inheritance Hierarchy

System::Object

HttpServer.Helpers::ObjectForm
See Also

ObjectForm Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ObjectForm Members

The **ObjectForm** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectForm</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begin</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Cb</td>
<td>Generate a checkbox</td>
</tr>
<tr>
<td>End</td>
<td>html end form tag</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>Hidden</td>
<td>Hiddens the specified property name. (Inherited from Object.)</td>
</tr>
<tr>
<td>Label</td>
<td>Labels the specified property name.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Pb</td>
<td>password box</td>
</tr>
<tr>
<td>Select</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Submit</td>
<td>Write a submit tag.</td>
</tr>
<tr>
<td>Tb</td>
<td>Generates a text box.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
ObjectForm Constructor

ObjectForm Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectForm(String, Object)</td>
<td>Initializes a new instance of the ObjectForm class.</td>
</tr>
<tr>
<td>ObjectForm(String, String, Object)</td>
<td>Initializes a new instance of the ObjectForm class.</td>
</tr>
<tr>
<td>ObjectForm(String, String, Object, String)</td>
<td>Initializes a new instance of the ObjectForm class.</td>
</tr>
</tbody>
</table>
See Also

ObjectForm Class
ObjectForm Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ObjectForm Constructor (String, Object)

Initializes a new instance of the ObjectForm class.

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public ObjectForm(
    string action,
    Object obj
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New (_
    action As String,
    obj As Object_
)
```

### Visual C++

```cpp
public:
ObjectForm(
    String^ action,
    Object^ obj
)
```

## Parameters

**action**
- Type: `System::String`
- form action.

**obj**
- Type: `System::Object`
- object to get values from.
See Also

ObjectForm Class
ObjectForm Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

ObjectForm Constructor (String, String, Object)

ObjectForm Class  See Also  Send Feedback

Initializes a new instance of the ObjectForm class.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ObjectForm(
    string action,
    string name,
    Object obj
)

Visual Basic (Declaration)

Public Sub New ( _
    action As String, _
    name As String, _
    obj As Object _
)

Visual C++

public:
ObjectForm(
    String^ action,
    String^ name,
    Object^ obj
)

Parameters

action
    Type: System::String
    action to do when form is posted.

name
    Type: System::String
    form name *and* id.

obj
    Type: System::Object
object to get values from
See Also

ObjectForm Class
ObjectForm Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ObjectForm Constructor (String, String, Object, String)

ObjectForm Class  See Also  Send Feedback

Initializes a new instance of the ObjectForm class.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ObjectForm(
    string action,
    string name,
    Object obj,
    string method
)

Visual Basic (Declaration)

Public Sub New ( _
    action As String, _
    name As String, _
    obj As Object, _
    method As String _
)

Visual C++

public:
    ObjectForm(
        String^ action,
        String^ name,
        Object^ obj,
        String^ method
    )

Parameters

action
    Type: System::String
    action to do when form is posted.

name
    Type: System::String
    form name *and* id.
obj
Type: System::Object

[Missing <param name="obj"/> documentation for

method
Type: System::String

[Missing <param name="method"/> documentation for
See Also

ObjectForm Class
ObjectForm Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ObjectForm** type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begin</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Cb</td>
<td>Generate a checkbox</td>
</tr>
<tr>
<td>End</td>
<td>html end form tag</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.                             (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance.                                 (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Hidden</td>
<td>Hiddens the specified property name.</td>
</tr>
<tr>
<td>Label</td>
<td>Labels the specified property name.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.                          (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Pb</td>
<td>password box</td>
</tr>
<tr>
<td>Select</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Submit</td>
<td>Write a submit tag.</td>
</tr>
<tr>
<td>Tb</td>
<td>Generates a text box.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.             (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

ObjectForm Class
HttpServer.Helpers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Begin()()</code></td>
<td>write out the FORM-tag.</td>
</tr>
<tr>
<td><code>Begin(Boolean)</code></td>
<td>Writeout the form tag</td>
</tr>
</tbody>
</table>
See Also

ObjectForm Class
ObjectForm Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
write out the FORM-tag.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Begin()

Visual Basic (Declaration)

Public Function Begin As String

Visual C++

public:
string^ Begin()

Return Value

generated html code
See Also

ObjectForm Class
Begin Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Writeout the form tag

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Begin(
    bool isAjax
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Begin ( _
    isAjax As Boolean _
) As String
```

**Visual C++**

```cpp
public:
String^ Begin(
    bool isAjax
)
```

**Parameters**

isAjax

Type: System:::Boolean

form should be posted through ajax.

**Return Value**

generated html code
See Also

ObjectForm Class
Begin Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ObjectForm..::.Cb Method

Generate a checkbox

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public string Cb(
    string propertyName,
    string value,
    params Object[] options
)
```

### Visual Basic (Declaration)

```vbnet
Public Function Cb ( _
    propertyName As String, _
    value As String, _
    ParamArray options As Object() _
) As String
```

### Visual C++

```cpp
public: 
    String^ Cb(
        String^ propertyName,
        String^ value,
        ... array<Object^>^ options
    )
```

## Parameters

- **propertyName**
  - Type: `System::String`
  - property in object

- **value**
  - Type: `System::String`
  - checkbox value

- **options**
  - Type: array< `System::Object` >[](0)[]}
additional html attributes.

Return Value

generated html code
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ObjectForm Class  See Also  Send Feedback

html end form tag

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string End()

Visual Basic (Declaration)

Public Function End As String

Visual C++

public:
String^ End()

Return Value

html
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Hiddens the specified property name.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Hidden(
    string propertyName,
    params Object[] options
)
```

Visual Basic (Declaration)

```vbnet
Public Function Hidden (_
    ByVal propertyName As String, _
    ParamArray options As Object() _
) As String
```

Visual C++

```cpp
public:
String^ Hidden(
    String^ propertyName,
    ... array<Object^>^ options
)
```

Parameters

propertyName
Type: `System::String`
Name of the property.

options
Type: array< `System::Object` >[]
The options.

Return Value

generated html code
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Labels the specified property name.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Label(
    string propertyName,
    string label
)

Visual Basic (Declaration)

Public Function Label ( _
    propertyName As String, _
    label As String _
) As String

Visual C++

public:
    String^ Label(
        String^ propertyName,
        String^ label
    )

Parameters

propertyName
  Type: System::String
  property in object.

label
  Type: System::String
  caption

Return Value

generated html code
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
ObjectForm...::Pb Method

ObjectForm Class  See Also  Send Feedback

password box

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Pb(string propertyName,
    params Object[] options)
```

Visual Basic (Declaration)

```vbnet
Public Function Pb(_
    propertyName As String,
    ParamArray options As Object() _
) As String
```

Visual C++

```cpp
public: 
String^ Pb(_
    String^ propertyName, 
    ... array<Object^>^ options)
```

Parameters

**propertyName**
Type: `System::::String`


**options**
Type: `array< System::::Object >[]()`


Return Value
generated html code
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
ObjectForm...::Select Method

ObjectForm Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select(String, String, String, array&lt;Object&gt;[][])</td>
<td>Write a html select tag</td>
</tr>
<tr>
<td>Select(String, IEnumerable, String, String, array&lt;Object&gt;[][])</td>
<td>Selects the specified property name.</td>
</tr>
</tbody>
</table>
See Also

ObjectForm Class
ObjectForm Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ObjectForm..::.Select Method (String, String, String, array<Object>[])[]

ObjectForm Class  See Also  Send Feedback

Write a html select tag

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Select(
    string propertyName,
    string idColumn,
    string titleColumn,
    params Object[] options
)

Visual Basic (Declaration)

Public Function Select ( _
    propertyName As String, _
    idColumn As String, _
    titleColumn As String, _
    ParamArray options As Object() _
) As String

Visual C++

public:
String^ Select(
    String^ propertyName,
    String^ idColumn,
    String^ titleColumn,
    ... array<Object^>^ options
)

Parameters

propertyName
    Type: System:::String
    object property.

idColumn
    Type: System:::String
    id column
titleColumn
   Type: System::String
   The title column.

options
   Type: array< System::Object >[]()
   The options.

Return Value

See Also

ObjectForm Class
Select Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Select Method (String, IEnumerable, String, String, array<Object>[][])

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Select(
    string propertyName,
    IEnumerable items,
    string idColumn,
    string titleColumn,
    params Object[] options
)

Visual Basic (Declaration)

Public Function Select ( _
    propertyName As String, _
    items As IEnumerable, _
    idColumn As String, _
    titleColumn As String, _
    ParamArray options As Object() _
) As String

Visual C++

public:
    String^ Select(
        String^ propertyName,
        IEnumerable^ items,
        String^ idColumn,
        String^ titleColumn,
        ... array<Object^>^ options
    )

Parameters

propertyName
    Type: System::String
    Name of the property.

items
Type: `System.Collections..::.IEnumerable`  
The items.

`idColumn`  
Type: `System..::.String`  
The id column.

`titleColumn`  
Type: `System..::.String`  
The title column.

`options`  
Type: `array< System..::.Object >[]()`  
The options.

**Return Value**

[Missing `<returns>` documentation for  
See Also

ObjectForm Class
Select Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Write a submit tag.

**Namespace:** [HttpServer.Helpers](http://localhost:8080/)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Submit(
    string value
)
```

Visual Basic (Declaration)

```vbnet
Public Function Submit ( 
    value As String 
) As String
```

Visual C++

```cpp
public:
    String^ Submit(
        String^ value
    )
```

Parameters

value
  Type: System::String
  button caption

Return Value

html submit tag
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generates a text box.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Tb(
    string propertyName,
    params Object[] options
)

Visual Basic (Declaration)

Public Function Tb ( _
    propertyName As String, _
    ParamArray options As Object() _
) As String

Visual C++

public:
    String^ Tb(
        String^ propertyName,
        ... array<Object^>^ options
    )

Parameters

propertyName
Type: System::String


options
Type: array< System::Object >[]([])


Return Value

generated html code
See Also

ObjectForm Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
General usage validator class.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class Validator

Visual Basic (Declaration)

Public Class Validator

Visual C++

public ref class Validator
Remarks

Check all FieldXXXXX constants to see what you need to translate.
Inheritance Hierarchy

System:::Object
HttpServer.Helpers:::Validator
   HttpServer.Helpers:::FormValidator
See Also

Validator Members
HttpGet.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Validator** type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validator</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDefaultPhrases</td>
<td>add default language phrases (only english)</td>
</tr>
<tr>
<td>AlphaNumeric</td>
<td>Check whether the specified value is an integer.</td>
</tr>
<tr>
<td>Boolean</td>
<td>Checks whether a field contains true (can also be in native language).</td>
</tr>
<tr>
<td>Char</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Contains</td>
<td>Checks whether a string contains a specific character.</td>
</tr>
<tr>
<td>Digits</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Email</td>
<td>Validates email address using a regexp.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>Format</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>Hex</td>
<td>Validates a string to be in trimmed hex letters</td>
</tr>
<tr>
<td>Integer</td>
<td>Check whether the specified value is an integer.</td>
</tr>
<tr>
<td>Letters</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>LettersOrDigits</td>
<td>Validate that a string only contains letters or digits.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Password</td>
<td>Validate that a string only contains letters or digits or any of the PasswordChars.</td>
</tr>
<tr>
<td>Required</td>
<td>Check's weather a parameter is null or not.</td>
</tr>
<tr>
<td><strong>SetDefaultMgr</strong></td>
<td>Set default language.</td>
</tr>
<tr>
<td><strong>String</strong></td>
<td>Validate a string parameter in the form</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
# Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_errors</td>
<td>A list with all errors found.</td>
</tr>
<tr>
<td>_modelLang</td>
<td>Language translation for all &quot;name&quot; parameters.</td>
</tr>
<tr>
<td>_validatorLang</td>
<td>Language for the validator (i.e. &quot;Field {0} is required.&quot;)</td>
</tr>
<tr>
<td>EmailRegEx</td>
<td>Regex used to validate emails.</td>
</tr>
<tr>
<td>FieldAlphaNumeric</td>
<td>'{0}' may only contain writable characters.</td>
</tr>
<tr>
<td>FieldBoolean</td>
<td>'{0}' must be {1} or {2}.</td>
</tr>
<tr>
<td>FieldDigits</td>
<td>'{0}' may only contain digits</td>
</tr>
<tr>
<td>FieldDigitsExtra</td>
<td>'{0}' may only contain digits and the following characters: '{1}'</td>
</tr>
<tr>
<td>FieldEmail</td>
<td>'{0}' is not a valid email address.</td>
</tr>
<tr>
<td>FieldHex</td>
<td>Name of the string to display when testing hex fail</td>
</tr>
<tr>
<td>FieldLetterOrDigit</td>
<td>'{0}' can only contain letter or digits.</td>
</tr>
<tr>
<td>FieldLetters</td>
<td>Name of string to display when testing letters fail</td>
</tr>
<tr>
<td>FieldLettersSpecial</td>
<td>Name of string to display when testing letters fail</td>
</tr>
<tr>
<td>FieldNumber</td>
<td>'{0}' is not a number.</td>
</tr>
<tr>
<td>FieldPassword</td>
<td>'{0}' can only contain letter or digits or one of the following chars: {1}</td>
</tr>
<tr>
<td>FieldRequired</td>
<td>'{0}' is required.</td>
</tr>
<tr>
<td>FieldValueFalse</td>
<td>Used to check if a field is false.</td>
</tr>
<tr>
<td>FieldValueTrue</td>
<td>Text: True</td>
</tr>
<tr>
<td>Language</td>
<td>Language manager used if no one else have been specified.</td>
</tr>
<tr>
<td>PasswordChars</td>
<td>Extra characters that are allowed in passwords.</td>
</tr>
<tr>
<td>SpecialLetters</td>
<td>A collection of special letters.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>ContainsErrors</td>
<td>true of validation generated errors.</td>
</tr>
<tr>
<td>Errors</td>
<td>Collection of validation errors.</td>
</tr>
</tbody>
</table>
See Also

Validator Class
HttpServer.Helpers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver

Validator Constructor

Validator Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validator()()</td>
<td>Create a new validator</td>
</tr>
<tr>
<td>Validator(LanguageNode)</td>
<td>Create a new validator.</td>
</tr>
<tr>
<td>Validator(NameValueCollection)</td>
<td>Create a new validator.</td>
</tr>
<tr>
<td>Validator(NameValueCollection, LanguageNode)</td>
<td>Create a new validator.</td>
</tr>
</tbody>
</table>
See Also

Validator Class
Validator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new validator

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Validator()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
Validator()
See Also

Validator Class
Validator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new validator.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Validator(
    LanguageNode modelLanguage
)

Visual Basic (Declaration)

Public Sub New (_
    modelLanguage As LanguageNode _
)

Visual C++

public:
Validator(
    LanguageNode^ modelLanguage
)

Parameters

modelLanguage
    Type: LanguageNode
        language translation for all "name" parameters
See Also

Validator Class
Validator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new validator.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public Validator(
    NameValueCollection errors
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
errors As NameValueCollection _
)
```

**Visual C++**

```cpp
public:
Validator(
    NameValueCollection^ errors
)
```

### Parameters

**errors**

Type: `System.Collections.Specialized.NameValueCollection`

collection to be filled with errors
See Also

Validator Class
Validator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new validator.

**Namespace:** [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Validator(
    NameValueCollection errors,
    LanguageNode modelLanguage
)

Visual Basic (Declaration)

Public Sub New (_
    errors As NameValueCollection, _
    modelLanguage As LanguageNode _
)

Visual C++

public:
    Validator(
        NameValueCollection^ errors,
        LanguageNode^ modelLanguage
    )

Parameters

errors
    Type: System.Collections.Specialized.NameValueCollection
    collection to be filled with errors

modelLanguage
    Type: LanguageNode
    language translation for all "name" parameters
See Also

Validator Class
Validator Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Validator** type exposes the following members.
# Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_errors</td>
<td>A list with all errors found.</td>
</tr>
<tr>
<td>_modelLang</td>
<td>Language translation for all &quot;name&quot; parameters.</td>
</tr>
<tr>
<td>_validatorLang</td>
<td>Language for the validator (i.e. &quot;Field {0} is required.&quot;)</td>
</tr>
<tr>
<td>EmailRegEx</td>
<td>Regex used to validate emails.</td>
</tr>
<tr>
<td>FieldAlphaNumeric</td>
<td>'{0}' may only contain writable characters.</td>
</tr>
<tr>
<td>FieldBoolean</td>
<td>'{0}' must be {1} or {2}.</td>
</tr>
<tr>
<td>FieldDigits</td>
<td>'{0}' may only contain digits.</td>
</tr>
<tr>
<td>FieldDigitsExtra</td>
<td>'{0}' may only contain digits and the following characters: '{1}'</td>
</tr>
<tr>
<td>FieldEmail</td>
<td>'{0}' is not a valid email address.</td>
</tr>
<tr>
<td>FieldHex</td>
<td>Name of the string to display when testing hex fail</td>
</tr>
<tr>
<td>FieldLetterOrDigit</td>
<td>'{0}' can only contain letter or digits.</td>
</tr>
<tr>
<td>FieldLetters</td>
<td>Name of string to display when testing letters fail</td>
</tr>
<tr>
<td>FieldLettersSpecial</td>
<td>Name of string to display when testing letters fail</td>
</tr>
<tr>
<td>FieldNumber</td>
<td>'{0}' is not a number.</td>
</tr>
<tr>
<td>FieldPassword</td>
<td>'{0}' can only contain letter or digits or one of the following chars: {1}</td>
</tr>
<tr>
<td>FieldRequired</td>
<td>'{0}' is required.</td>
</tr>
<tr>
<td>FieldValueFalse</td>
<td>Used to check if a field is false.</td>
</tr>
<tr>
<td>FieldValueTrue</td>
<td>Text: True</td>
</tr>
<tr>
<td>Language</td>
<td>Language manager used if no one else have been specified.</td>
</tr>
<tr>
<td>PasswordChars</td>
<td>Extra characters that are allowed in passwords.</td>
</tr>
<tr>
<td>SpecialLetters</td>
<td>A collection of special letters.</td>
</tr>
</tbody>
</table>
See Also

Validator Class
HttpServer.Helpers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A list with all errors found.

**Namespace:** [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected readonly NameValueCollection _errors

Visual Basic (Declaration)

Protected ReadOnly _errors As NameValueCollection

Visual C++

protected:
initonly NameValueCollection^ _errors
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Language translation for all "name" parameters.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected LanguageNode _modelLang

Visual Basic (Declaration)

Protected _modelLang As LanguageNode

Visual C++

protected:
LanguageNode^ _modelLang
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Language for the validator (i.e. "Field {0} is required.")

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
protected LanguageNode _validatorLang
```

**Visual Basic (Declaration)**

```vbnet
Protected _validatorLang As LanguageNode
```

**Visual C++**

```cpp
protected:
LanguageNode^ _validatorLang
```
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Regex used to validate emails.

**Namespace:** [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static readonly string EmailRegEx

Visual Basic (Declaration)

Public Shared ReadOnly EmailRegEx As String

Visual C++

public:
static initonly String^ EmailRegEx
See Also

- Validator Class
- HttpServer.Helpers Namespace
- Validator:::Email(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' may only contain writable characters.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer](#)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldAlphaNumeric

Visual Basic (Declaration)

Public Const FieldAlphaNumeric As String

Visual C++

class FieldAlphaNumeric
{
public:
    String^ FieldAlphaNumeric;
}
Remarks

ascii 32-126, 128-167, 224-238.
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' must be {1} or {2}.

**Namespace:** [HttpServer.Helpers](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldBoolean

Visual Basic (Declaration)

Public Const FieldBoolean As String

Visual C++

public:
literal String^ FieldBoolean
See Also

Validator Class
HttpServer.Helpers Namespace
Validator.<::<Boolean(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FieldDigits Field

'{0}' may only contain digits

**Namespace:** [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

public const string FieldDigits

**Visual Basic (Declaration)**

Public Const FieldDigits As String

**Visual C++**

public:
literal String^ FieldDigits
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' may only contain digits and the following characters: '{1}'

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldDigitsExtra

Visual Basic (Declaration)

Public Const FieldDigitsExtra As String

Visual C++

public:
literal String^ FieldDigitsExtra
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' is not a valid email address.

**Namespace:** [HttpServer.Helpers](http://example.com)

**Assembly:** [HttpServer](http://example.com) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public const string FieldEmail
```

Visual Basic (Declaration)

```vbnet
Public Const FieldEmail As String
```

Visual C++

```cpp
public:
    literal String^ FieldEmail
```
See Also

Validator Class
HttpServer.Helpers Namespace
Validator:::Email(String, String, Boolean)
Validator:::EmailRegEx

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Name of the string to display when testing hex fail

**Namespace:**  [HttpServer.Helpers](http://HttpServer.Helpers)  
**Assembly:**  [HttpServer](http://HttpServer) (in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldHex

Visual Basic (Declaration)

Public Const FieldHex As String

Visual C++

public:
literal String^ FieldHex
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' can only contain letter or digits.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#))  Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldLetterOrDigit

Visual Basic (Declaration)

Public Const FieldLetterOrDigit As String

Visual C++

public:
literal String^ FieldLetterOrDigit
See Also

Validator Class
HttpServer.Helpers Namespace
Validator:::LettersOrDigits(String, String, Boolean)
Validator:::String(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FieldLetters Field

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)

Name of string to display when testing letters fail
Syntax

C#

public const string FieldLetters

Visual Basic (Declaration)

Public Const FieldLetters As String

Visual C++

public:
literal String^ FieldLetters
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Name of string to display when testing letters fail

**Namespace:**  [HttpServer.Helpers](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public const string FieldLettersSpecial

**Visual Basic (Declaration)**

Public Const FieldLettersSpecial As String

**Visual C++**

public:
literal String^ FieldLettersSpecial
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' is not a number.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldNumber

Visual Basic (Declaration)

Public Const FieldNumber As String

Visual C++

public:
  literal String^ FieldNumber
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' can only contain letter or digits or one of the following chars: {1}

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldPassword

Visual Basic (Declaration)

Public Const FieldPassword As String

Visual C++

public:
    literal String^ FieldPassword
See Also

Validator Class
HttpServer.Helpers Namespace
Validator...:::PasswordChars
Validator...:::Password(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
'{0}' is required.

**Namespace:**  [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:**  [HttpServer (in HttpServer.dll)](http://HttpServer (in HttpServer.dll)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldRequired

Visual Basic (Declaration)

Public Const FieldRequired As String

Visual C++

public:
literal String^ FieldRequired
See Also

Validator Class
HttpServer.Helpers Namespace
Validator:::Required(String, String)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Used to check if a field is false.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldValueFalse

Visual Basic (Declaration)

Public Const FieldValueFalse As String

Visual C++

public:
literal String^ FieldValueFalse
See Also

Validator Class
HttpServer.Helpers Namespace
Validator..:::Boolean(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Text: True

Namespace: [HttpServer.Helpers](#)
Assembly: [HttpServer](#) in [HttpServer.dll] Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string FieldValueTrue

Visual Basic (Declaration)

Public Const FieldValueTrue As String

Visual C++

public:
literal String^ FieldValueTrue
See Also

Validator Class
HttpServer.Helpers Namespace
Validator..:::Boolean(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Language manager used if no one else have been specified.

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer.dll] Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static LanguageNode Language

Visual Basic (Declaration)

Public Shared Language As LanguageNode

Visual C++

public:
static LanguageNode^ Language
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Extra characters that are allowed in passwords.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string PasswordChars

Visual Basic (Declaration)

Public Shared PasswordChars As String

Visual C++

public:
static String^ PasswordChars
See Also

Validator Class
HttpServer.Helpers Namespace
Validator...:::Password(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A collection of special letters

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public const string SpecialLetters

Visual Basic (Declaration)

Public Const SpecialLetters As String

Visual C++

public:
    literal String^ SpecialLetters
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Validator** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDefaultPhrases</td>
<td>add default language phrases (only english)</td>
</tr>
<tr>
<td>AlphaNumeric</td>
<td>Check whether the specified value is an integer.</td>
</tr>
<tr>
<td>Boolean</td>
<td>Checks whether a field contains true (can also be in native language).</td>
</tr>
<tr>
<td>Char</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Contains</td>
<td>Checks whether a string contains a specific character.</td>
</tr>
<tr>
<td>Digits</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Email</td>
<td>Validates email address using a regexp.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td>(Inherited from Object.)</td>
<td></td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>(Inherited from Object.)</td>
<td></td>
</tr>
<tr>
<td>Format</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>(Inherited from Object.)</td>
<td></td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td>(Inherited from Object.)</td>
<td></td>
</tr>
<tr>
<td>Hex</td>
<td>Validates a string to be in trimmed hex letters</td>
</tr>
<tr>
<td>Integer</td>
<td>Check whether the specified value is an integer.</td>
</tr>
<tr>
<td>Letters</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>LettersOrDigits</td>
<td>Validate that a string only contains letters or digits.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td>(Inherited from Object.)</td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td>Validate that a string only contains letters or digits or any of the PasswordChars.</td>
</tr>
<tr>
<td>Required</td>
<td>Check's weather a parameter is null or not.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td><strong>SetDefaultMgr</strong></td>
<td>Set default language.</td>
</tr>
<tr>
<td><strong>String</strong></td>
<td>Validate a string parameter in the form.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <em>String</em> that represents the current <em>Object</em>. (Inherited from <em>Object</em>.)</td>
</tr>
</tbody>
</table>
See Also

Validator Class
HttpServer.Helpers Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validator...: AddDefaultPhrases Method

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public static void AddDefaultPhrases()

Visual Basic (Declaration)
Public Shared Sub AddDefaultPhrases

Visual C++
public:
static void AddDefaultPhrases()
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an integer.

**Namespace:** [HttpServer.Helpers](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

public string AlphaNumeric(
    string name,
    string value,
    bool required
)

**Visual Basic (Declaration)**

Public Function AlphaNumeric (_
    name As String, _
    value As String, _
    required As Boolean _
) As String

**Visual C++**

public:
    String^ AlphaNumeric( _
        String^ name,
        String^ value,
        bool required
    )

### Parameters

**name**
- Type: System::::String
- Name of the parameter

**value**
- Type: System::::String
- Parameter value to validate

**required**
- Type: System::::Boolean
Parameter is required (adds an error if it's not specified)

**Return Value**

value if parameter contains valid characters; string.Empty if not.
Remarks

ascii 32-126, 128-167, 224-238.
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Checks whether a field contains true (can also be in native language).

**Namespace:** [HttpServer.Helpers](#)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Boolean(
    string name,
    string value,
    bool required
)

Visual Basic (Declaration)

Public Function Boolean (  
    name As String,  
    value As String,  
    required As Boolean  
) As Boolean

Visual C++

public:
bool Boolean(
    String^ name,
    String^ value,
    bool required
)

Parameters

name
    Type: System::String
    field name

value
    Type: System::String
    value to validate

required
    Type: System::Boolean
field is required (may not be null or empty).

Return Value

true if value is true; false if value is false or if validation failed.
Remarks

Check validation errors to see if error occurred.
See Also

Validator Class
HttpServer.Helpers Namespace
Validator...:::FieldValueTrue

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
Validator Char Method

Validator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Char(String, String)</td>
<td>Check whether the specified value is an character.</td>
</tr>
<tr>
<td>Char(String, String, Boolean)</td>
<td>Check whether the specified value is an character.</td>
</tr>
</tbody>
</table>
See Also

Validator Class
Validator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an character.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public char Char(
    string name,
    string value
)
```

Visual Basic (Declaration)

```vbnet
Public Function Char ( _
    name As String, _
    value As String _
) As Char
```

Visual C++

```cpp
public:
    wchar_t Char(
        String^ name,
        String^ value
    )
```

Parameters

name
Type: System::String
Name of the parameter

value
Type: System::String
Parameter value to validate

Return Value

value if parameter is an int; char.MinValue if not.
See Also

Validator Class
Char Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an character.

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public char Char(
    string name,
    string value,
    bool required
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Char (_
    name As String, _
    value As String, _
    required As Boolean _
) As Char
```

**Visual C++**

```cpp
public:
    wchar_t Char(
        String^ name,
        String^ value,
        bool required
    )
```

### Parameters

**name**

Type: `System::String`

Name of the parameter

**value**

Type: `System::String`

Parameter value to validate

**required**

Type: `System::Boolean`
Parameter is required (adds an error if it's not specified)

**Return Value**

value if parameter is an int; char.MinValue if not.
See Also

[Validator Class]
[Char Overload]
[HttpServer.Helpers Namespace]

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Checks whether a string contains a specific character.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public static bool Contains(string s, char ch)
```

**Visual Basic (Declaration)**

```vbnet
Public Shared Function Contains (_
    s As String, _
    ch As Char _
) As Boolean
```

**Visual C++**

```cpp
public:
static bool Contains(
    String^ s, 
    wchar_t ch
)
```

### Parameters

- **s**
  
  Type: **System::String**
  
  source

- **ch**
  
  Type: **System::Char**
  
  character to find.

### Return Value

true if found; otherwise false.
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digits(String, String, Boolean)</td>
<td>Check if a value is digits only</td>
</tr>
<tr>
<td>Digits(String, String, Boolean, String)</td>
<td>Check if a value is digits only</td>
</tr>
</tbody>
</table>
See Also

Validator Class
Validator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check if a value is digits only

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Digits(
    string name,
    string value,
    bool required
)

Visual Basic (Declaration)

Public Function Digits ( _
    name As String, _
    value As String, _
    required As Boolean _
) As String

Visual C++

public:
    String^ Digits(
        String^ name,
        String^ value,
        bool required
    )

Parameters

name
    Type: System:::String
    Field name.

value
    Type: System:::String
    value to validate

required
    Type: System:::Boolean
true if field is required (may not be empty)

**Return Value**

string if validated, otherwise string.Empty
See Also

Validator Class
Digits Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check if a value is digits only

**Namespace:**  [HttpServer.Helpers](http://HttpServer.Helpers)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Digits(
    string name,
    string value,
    bool required,
    string extraAllowedCharacters
)

Visual Basic (Declaration)

Public Function Digits ( _
    name As String, _
    value As String, _
    required As Boolean, _
    extraAllowedCharacters As String _
) As String

Visual C++

public:
    String^ Digits(
        String^ name,
        String^ value,
        bool required,
        String^ extraAllowedCharacters
    )

Parameters

name
Type: System::String
Field name.

value
Type: System::String
value to validate
required
    Type: `System::Boolean`
    true if field is required (may not be empty)

extraAllowedCharacters
    Type: `System::String`
    extra characters that is allowed.

**Return Value**

string if validated, otherwise string.Empty
See Also

Validator Class
Digits Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
validates email address using a regexp.

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string Email(
    string name,
    string value,
    bool required
)
```

Visual Basic (Declaration)

```vbnet
Public Function Email (_
    name As String, _
    value As String, _
    required As Boolean _
) As String
```

Visual C++

```cpp
public: String^ Email(
    String^ name,
    String^ value,
    bool required
)
```

Parameters

name
Type: System::String
field name

value
Type: System::String
value to validate

required
Type: System::Boolean
field is required (may not be null or empty).

**Return Value**

value if validation is ok; otherwise string.Empty.
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members

C# Webserver

Validator...::Format Method

Validator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format(String, String)</td>
<td>format an error message</td>
</tr>
<tr>
<td>Format(String, String, String)</td>
<td>format an error message</td>
</tr>
<tr>
<td>Format(String, String, String, String)</td>
<td>format an error message</td>
</tr>
</tbody>
</table>
See Also

Validator Class
Validator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
format an error message

Namespace:   HttpServer.Helpers
Assembly:   HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual string Format(
    string format,
    string fieldName
)

Visual Basic (Declaration)

Protected Overridable Function Format ( _
    format As String, _
    fieldName As String _
) As String

Visual C++

protected:
virtual String ^ Format(
    String ^ format,
    String ^ fieldName
)

Parameters

format
    Type: System::String
    string containing the formatting

fieldName
    Type: System::String
    name of the field that the error occurred for.

Return Value

formatted string
See Also

- Validator Class
- Format Overload
- HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver

Validator...: Format Method (String, String, String)

Validator Class  See Also  Send Feedback

format an error message

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual string Format(
    string format,
    string fieldName,
    string extra
)

Visual Basic (Declaration)

Protected Overridable Function Format ( _
    format As String, _
    fieldName As String, _
    extra As String _
) As String

Visual C++

protected:
virtual String^ Format(
    String^ format,
    String^ fieldName,
    String^ extra
)

Parameters

format
    Type: System:::String
    string containing the formatting

fieldName
    Type: System:::String
    name of the field that the error occurred for.

extra
    Type: System:::String
extra value in the formatting

Return Value

formatted string
See Also

Validator Class
Format Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++
C# Webserver

Validator...:: Format Method (String, String, String, String)

Validator Class  See Also  Send Feedback

format an error message

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected virtual string Format(
    string format,
    string fieldName,
    string extra,
    string extra2
)

Visual Basic (Declaration)

Protected Overridable Function Format ( _
    format As String, _
    fieldName As String, _
    extra As String, _
    extra2 As String _
) As String

Visual C++

protected:
    virtual String^ Format(
        String^ format,
        String^ fieldName,
        String^ extra,
        String^ extra2
    )

Parameters

format
    Type: System::String
    string containing the formatting

fieldName
    Type: System::String
    name of the field that the error occurred for.
extra
  Type: System.String
  extra value in the formatting

extra2
  Type: System.String
  third parameter in the formatting

Return Value

formatted string
See Also

Validator Class
Format Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

Validator...:::Hex Method

Validator Class  See Also  Send Feedback

Validates a string to be in trimmed hex letters

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Hex(
    string name,
    string value,
    bool required
)

Visual Basic (Declaration)

Public Function Hex (_
    name As String, _
    value As String, _
    required As Boolean _
) As String

Visual C++

public:
    String^ Hex(
        String^ name,
        String^ value,
        bool required
    )

Parameters

name
    Type: System:::String
    The name of the parameter

value
    Type: System:::String
    The value to test

required
    Type: System:::Boolean
If the parameter is required

**Return Value**

The value or string.Empty if validation failed
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check whether the specified value is an integer.

**Namespace:** [HttpServer.Helpers](http://HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public int Integer(
    string name,
    string value,
    bool required
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Integer (_
    name As String, _
    value As String, _
    required As Boolean _
) As Integer
```

**Visual C++**

```cpp
public:
    int Integer(
        String^ name,
        String^ value,
        bool required
    )
```

### Parameters

**name**

Type: `System::String`

Name of the parameter

**value**

Type: `System::String`

Parameter value to validate

**required**

Type: `System::Boolean`
Paramater is required (adds an error if it's not specified)

**Return Value**

value if parameter is an int; 0 if not.
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members  C# Webserver  
Validator...Letters Method

Validator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Letters(String, String, Boolean)</td>
<td>Validates a string to consist of letters. Will test for A-Z, a-z, spaces and letters specified in SpecialLetters</td>
</tr>
<tr>
<td>Letters(String, String, String, Boolean)</td>
<td>Validates a string to consist of letters plus specified characters. Basic allowed letters are A-Z and a-z. For extra characters SpecialLetters</td>
</tr>
</tbody>
</table>
See Also

Validator Class
Validator Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver

Validator...:::Letters Method (String, String, Boolean)

**Validator Class**  **See Also**  **Send Feedback**

Validates a string to consist of letters. Will test for A-Z, a-z, spaces and letters specified in **SpecialLetters**

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public string Letters(
    string name,
    string value,
    bool required
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Letters (_
    name As String, _
    value As String, _
    required As Boolean _
) As String
```

**Visual C++**

```cpp
public:
    String^ Letters(
        String^ name,
        String^ value,
        bool required
    )
```

**Parameters**

**name**

Type: `System::::String`  
field name

**value**

Type: `System::::String`  
value to validate

**required**

Type: `System::::Boolean`
field is required

Return Value

[Missing <returns> documentation for
See Also

Validator Class
Letters Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validator...::Letters Method (String, String, String, Boolean)

Validates a string to consist of letters plus specified characters. Basic allowed letters are A-Z and a-z. For extra characters SpecialLetters

**Namespace:** [HttpServer.Helpers](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public string Letters(
    string name,
    string value,
    string extraAllowedCharacters,
    bool required
)
```

#### Visual Basic (Declaration)

```vbnet
Public Function Letters ( _
    name As String, _
    value As String, _
    extraAllowedCharacters As String, _
    required As Boolean _
) As String
```

#### Visual C++

```cpp
public: String^ Letters( 
    String^ name,
    String^ value,
    String^ extraAllowedCharacters,
    bool required
)
```

### Parameters

- **name**
  - **Type:** `System::String`
  - The name of the parameter

- **value**
  - **Type:** `System::String`
  - The value to validate
extraAllowedCharacters
  Type: System::String
  A string consisting of extra allowed characters

required
  Type: System::Boolean
  If the value is required

Return Value

The validated string or string.Empty if validation failed
See Also

Validator Class
Letters Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate that a string only contains letters or digits.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public string LettersOrDigits(
    string name,
    string value,
    bool required
)
```

Visual Basic (Declaration)

```vbnet
Public Function LettersOrDigits ( _
    name As String, _
    value As String, _
    required As Boolean _
) As String
```

Visual C++

```cpp
public: 
String^ LettersOrDigits(
    String^ name,
    String^ value,
    bool required
)
```

Parameters

name
Type: System::String
Form parameter name.

value
Type: System::String
Tvalue to validate

required
Type: System::Boolean
may not be null or empty if true.

**Return Value**

value if valid; otherwise string.EmptyLanguageNode
See Also

Validator Class
HttpServer.Helpers Namespace
Validator::String(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate that a string only contains letters or digits or any of the PasswordChars.

**Namespace:** HttpServer.Helpers  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public string Password(
    string name,
    string value,
    bool required
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Password ( _
    name As String, _
    value As String, _
    required As Boolean _
) As String
```

**Visual C++**

```cpp
public: 
String^ Password(
    String^ name,
    String^ value,
    bool required
)
```

### Parameters

**name**
- Type: `System::String`
- Form parameter name.

**value**
- Type: `System::String`
- value to validate

**required**
- Type: `System::Boolean`
field may not be empty or null if true

Return Value

value if valid; otherwise string.Empty
See Also

Validator Class
HttpServer.Helpers Namespace
Validator::: String(String, String, Boolean)
Validator::: LettersOrDigits(String, String, Boolean)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check's weather a parameter is null or not.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Required(
  string name,
  string value
)

Visual Basic (Declaration)

Public Function Required ( _
  name As String, _
  value As String _
) As Boolean

Visual C++

public:
bool Required(
  String^ name,
  String^ value
)

Parameters

name
  Type: System..::.String
  Parameter in form

value
  Type: System..::.String
  value that cannot be null or empty.

Return Value

true if value is not null/empty; otherwise false.
See Also

**Validator Class**  
**HttpServer.Helpers Namespace**

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Set default language.

**Namespace:**  [HttpServer.Helpers](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void SetDefaultMgr()

Visual Basic (Declaration)

Protected Sub SetDefaultMgr

Visual C++

protected:
void SetDefaultMgr()
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Validate a string parameter in the form

**Namespace:**  [HttpServer.Helpers]
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public string String(
    string name,
    string value,
    bool required
)
```

### Visual Basic (Declaration)

```vbnet
Public Function String ( _
    name As String, _
    value As String, _
    required As Boolean _
) As String
```

### Visual C++

```cpp
public: String^ String(
    String^ name,
    String^ value,
    bool required
)
```

## Parameters

**name**

Type: `System::::String`

Form parameter name.

**value**

Type: `System::::String`

Value to validate as string.

**required**

Type: `System::::Boolean`
field may not be empty or null if true.

Return Value

t value if valid; otherwise string.Empty
See Also

- [Validator Class](#)
- [HttpServer.Helpers Namespace](#)
- [Validator..::.LettersOrDigits(String, String, Boolean)](#)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Validator** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContainsErrors</td>
<td>true of validation generated errors.</td>
</tr>
<tr>
<td>Errors</td>
<td>Collection of validation errors.</td>
</tr>
</tbody>
</table>
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
true of validation generated errors.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool ContainsErrors { get; }

Visual Basic (Declaration)

Public ReadOnly Property ContainsErrors As Boolean

Visual C++

public:
property bool ContainsErrors {
    bool get ();
}
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Collection of validation errors.

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

class PublicTipo
{
    public NameValueCollection Errors { get; }
}

Visual Basic (Declaration)

Public ReadOnly Property Errors As NameValueCollection

Visual C++

public:
property NameValueCollection^ Errors {
    NameValueCollection^ get ();
}
See Also

Validator Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Webhelper provides helpers for common tasks in HTML.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static class WebHelper

Visual Basic (Declaration)

Public NotInheritable Class WebHelper

Visual C++

public ref class WebHelper abstract sealed
Inheritance Hierarchy

System...:::Object
HttpServer.Helpers...:::WebHelper
See Also

WebHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `WebHelper` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AjaxRequest</strong></td>
<td>Creates a link that invokes through ajax.</td>
</tr>
<tr>
<td><strong>AjaxUpdater</strong></td>
<td>Builds a link that updates an element with the fetched ajax content.</td>
</tr>
<tr>
<td><strong>BuildLink</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>CloseDialog</strong></td>
<td>Close a javascript dialog window/div.</td>
</tr>
<tr>
<td><strong>CreateDialog</strong></td>
<td>Create/Open a dialog box using ajax</td>
</tr>
<tr>
<td><strong>DialogLink</strong></td>
<td>A link that pop ups a Dialog (overlay div)</td>
</tr>
<tr>
<td><strong>Errors</strong></td>
<td>Render errors into a UL with class &quot;errors&quot;</td>
</tr>
<tr>
<td><strong>FormStart</strong></td>
<td><em>Obsolete.</em> Create a <code>&lt;form&gt;</code> tag.</td>
</tr>
<tr>
<td><strong>Link</strong></td>
<td>Create a link tag.</td>
</tr>
<tr>
<td><strong>List</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Select</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSImplementation</td>
<td>Used to let the website use different javascript libraries. Default is PrototypeImp</td>
</tr>
</tbody>
</table>
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The WebHelper type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSImplementation</td>
<td>Used to let the website use different javascript libraries. Default is PrototypeImp</td>
</tr>
</tbody>
</table>
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
WebHelper...: JSSImplementation Field

WebHelper Class  See Also  Send Feedback

Used to let the website use different javascript libraries. Default is PrototypeImp

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static JavascriptHelperImplementation JSImplementation

Visual Basic (Declaration)

Public Shared JSImplementation As JavascriptHelperImplementation

Visual C++

public:
static JavascriptHelperImplementation^ JSImplementation
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **WebHelper** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AjaxRequest</strong></td>
<td>Creates a link that invokes through ajax.</td>
</tr>
<tr>
<td><strong>AjaxUpdater</strong></td>
<td>Builds a link that updates an element with the fetched ajax content.</td>
</tr>
<tr>
<td><strong>BuildLink</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>CloseDialog</strong></td>
<td>Close a javascript dialog window/div.</td>
</tr>
<tr>
<td><strong>CreateDialog</strong></td>
<td>Create/Open a dialog box using ajax</td>
</tr>
<tr>
<td><strong>DialogLink</strong></td>
<td>A link that pop ups a Dialog (overlay div)</td>
</tr>
<tr>
<td><strong>Errors</strong></td>
<td>Render errors into a UL with class &quot;errors&quot;</td>
</tr>
<tr>
<td><strong>Obsolete.</strong></td>
<td></td>
</tr>
<tr>
<td><strong>FormStart</strong></td>
<td>Create a &lt;form&gt; tag.</td>
</tr>
<tr>
<td><strong>Link</strong></td>
<td>Create a link tag.</td>
</tr>
<tr>
<td><strong>List</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Select</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Creates a link that invokes through ajax.

**Namespace:** [HttpServer.Helpers](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string AjaxRequest(
    string url,
    string title,
    params string[] options
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function AjaxRequest ( _
    url As String, _
    title As String, _
    ParamArray options As String() _
) As String
```

Visual C++

```cpp
public:
static String^ AjaxRequest(
    String^ url,
    String^ title,
    ... array<String^>^ options
)
```

Parameters

url
Type: System::::String
url to fetch

title
Type: System::::String
link title

options
Type: array< System::::String >[]()}
optional options in format "key, value, key, value". Javascript options starts with ':'.

**Return Value**

da link tag
Examples

WebHelper.AjaxRequest("/users/add/", "Add user", ":method", "post", "onclick", "validate('this');");
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Builds a link that updates an element with the fetched ajax content.

**Namespace:** [HttpServer.Helpers](http://example.com/HttpServer.Helpers)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string AjaxUpdater(
    string url,
    string title,
    string targetId,
    params string[] options
)

Visual Basic (Declaration)

Public Shared Function AjaxUpdater (_
    url As String, _
    title As String, _
    targetId As String, _
    ParamArray options As String(_) _)  _
) As String

Visual C++

public:
static String^ AjaxUpdater(
    String^ url,
    String^ title,
    String^ targetId,
    ... array<String^>^ options
)

Parameters

url
    Type: System:::String
    Url to fetch content from

title
    Type: System:::String
    link title
targetId
   Type: System::<String
   html element to update with the results of the ajax request.

options
   Type: array< System::<String >[]>()
   optional options in format "key, value, key, value"

**Return Value**

A link tag.
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
WebHelper...:..BuildLink Method
WebHelper Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BuildLink(String, String, array&lt;String&gt;[])</td>
<td>Build a link</td>
</tr>
<tr>
<td>BuildLink(String, String, String, array&lt;String&gt;[])</td>
<td>Builds a link</td>
</tr>
</tbody>
</table>
See Also

WebHelper Class
WebHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Build a link

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string BuildLink(
    string url,
    string title,
    params string[] htmlAttributes
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function BuildLink (_
    url As String, _
    title As String, _
    ParamArray htmlAttributes As String() _
) As String
```

Visual C++

```cpp
public:
static String^ BuildLink(
    String^ url,
    String^ title,
    ... array<String^>^ htmlAttributes
)
```

Parameters

url
Type: `System::String`  
url to go to.

title
Type: `System::String`  
title of link (displayed text)

htmlAttributes
Type: array`< System::String >>[]`
extra html attributes.

**Return Value**

a complete link
See Also

WebHelper Class
BuildLink Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Builds a link

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public static string BuildLink(
    string url,
    string title,
    string extra,
    params string[] htmlAttributes
)
```

### Visual Basic (Declaration)

```vbnet
Public Shared Function BuildLink ( _
    url As String, _
    title As String, _
    extra As String, _
    ParamArray htmlAttributes As String() _
) As String
```

### Visual C++

```csharp
public:
static String^ BuildLink(
    String^ url,
    String^ title,
    String^ extra,
    ... array<String^>^ htmlAttributes
)
```

## Parameters

- **url**
  - Type: `System::String`
  - url to go to.

- **title**
  - Type: `System::String`
  - title of link (displayed text)
extra
   Type: `System::String`
   extra options/text included in attribute part of the link tag.

htmlAttributes
   Type: array< `System::String` >[{}]
   extra html attributes.

**Return Value**

a complete link
See Also

WebHelper Class
BuildLink Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
WebHelper...:::CloseDialog Method

WebHelper Class  See Also  Send Feedback

Close a javascript dialog window/div.

**Namespace:**  HttpServer.Helpers
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string CloseDialog()

Visual Basic (Declaration)

Public Shared Function CloseDialog As String

Visual C++

public:
static String^ CloseDialog()

Return Value

javascript for closing a dialog.
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create/Open a dialog box using ajax

**Namespace:**  [HttpServer.Helpers](http://example.com)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public static string CreateDialog(
    string url,
    string title,
    params string[] parameters
)
```

**Visual Basic (Declaration)**

```vbnet
Public Shared Function CreateDialog (_
    url As String, _
    title As String, _
    ParamArray parameters As String() _
) As String
```

**Visual C++**

```cpp
public:
static String^ CreateDialog( 
    String^ url, 
    String^ title, 
    ... array<String^>^ parameters 
)
```

### Parameters

**url**

Type: `System::String`


**title**

Type: `System::String`

parameters

    Type: array< System::::String >[]

[Missing <param name="parameters"/> documentation for

Return Value

[Missing <returns> documentation for
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A link that pop ups a Dialog (overlay div)

**Namespace:**  [HttpServer.Helpers](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string DialogLink(
    string url,
    string title,
    params string[] htmlAttributes
)

Visual Basic (Declaration)

Public Shared Function DialogLink ( _
    url As String, _
    title As String, _
    ParamArray htmlAttributes As String() _
) As String

Visual C++

public:
static String^ DialogLink(
    String^ url,
    String^ title,
    ... array<String^>^ htmlAttributes
)

Parameters

url
  Type: System:::String
  url to contents of dialog

title
  Type: System:::String
  link title

htmlAttributes
  Type: array< System:::String >[][]
name/value of html attributes.

**Return Value**

A "a"-tag that popups a dialog when clicked
Examples

WebHelper.DialogLink("/user/show/1", "show user", "onmouseover",
"alert('booh!');");
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Render errors into a UL with class "errors"

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string Errors(
    NameValueCollection errors
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function Errors ( _
    errors As NameValueCollection _
) As String
```

Visual C++

```cpp
public:
static String^ Errors(
    NameValueCollection^ errors
)
```

Parameters

errors

Type: System.Collections.Specialized.NameValueCollection

[Missing <param name="errors"/> documentation for

Return Value

[Missing <returns> documentation for
]
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a `<form>` tag.

**Namespace:**  [HttpServer.Helpers](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

[ObsoleteAttribute("Moved to FormHelper")]
public static string FormStart(
    string name,
    string action,
    bool isAjax
)

Visual Basic (Declaration)

<ObsoleteAttribute("Moved to FormHelper")> _
Public Shared Function FormStart ( _
    name As String, _
    action As String, _
    isAjax As Boolean _
) As String

Visual C++

[ObsoleteAttribute(L"Moved to FormHelper")]
public:
static String^ FormStart(
    String^ name,
    String^ action,
    bool isAjax
)

Parameters

name
   Type: System::String
   name of form

action
   Type: System::String
   action to invoke on submit
isAjax
   Type: System::Boolean
   form should be posted as ajax

Return Value

html code
Examples

WebHelper.FormStart("frmLogin", "/user/login", Request.IsAjax);
See Also

WebHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a link tag.

**Namespace:** [HttpServer.Helpers]

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string Link(
    string url,
    string title,
    params string[] htmlAttributes
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function Link ( _
    url As String, _
    title As String, _
    ParamArray htmlAttributes As String() _
) As String
```

Visual C++

```cpp
public:
static String^ Link(
    String^ url,
    String^ title,
    ... array<String^>^ htmlAttributes
)
```

Parameters

url
Type: `System::::String`
url to go to

title
Type: `System::::String`
link title (text that is displayed)

htmlAttributes
Type: array< `System::::String` >[][])
html attributes, name, value, name, value

Return Value

html code
Examples

WebHelper.Link("/user/show/1", "Show user", "id", "showUser", "onclick", "return confirm('Are you shure?');");
See Also

[WebHelper Class]
[HttpServer.Helpers Namespace]

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
WebHelper...:List Method

WebHelper Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>List(IEnumerable&lt;(Of Object&gt;)&gt;, String)</td>
<td>Render errors into a UL with class &quot;errors&quot;</td>
</tr>
<tr>
<td>List(NameValueCollection, String)</td>
<td>Render errors into a UL with class &quot;errors&quot;</td>
</tr>
</tbody>
</table>
See Also

WebHelper Class
WebHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Render errors into a UL with class "errors"

**Namespace:**  [HttpServer.Helpers](#)  
**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public static string List(
    IEnumerable<object> theList, 
    string className
)
```

Visual Basic (Declaration)

```vbnet
Public Shared Function List (_
    theList As IEnumerable(Of Object), _
    className As String _
) As String
```

Visual C++

```csharp
public:
static String^ List(
    IEnumerable<Object^>^ theList, 
    String^ className
)
```

Parameters

theList
    Type: System.Collections.Generic.IEnumerable<object>
    items to list

className
    Type: System.String
    class used by UL-tag.

Return Value

an unordered html list.
See Also

WebHelper Class
List Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Render errors into a UL with class "errors"

**Namespace:**  [HttpServer.Helpers](#)
**Assembly:**  [HttpServer.dll](#) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string List(
    NameValueCollection theList,
    string className
)

Visual Basic (Declaration)

Public Shared Function List ( _
    theList As NameValueCollection, _
    className As String _
) As String

Visual C++

public:
static String^ List(
    NameValueCollection^ theList,
    String^ className
)

Parameters

theList

Type: System.Collections.Specialized.NameValueCollection
items to list

className

Type: System.String
class used by UL-tag.

Return Value

an unordered html list.
See Also

WebHelper Class
List Overload
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members  C# Webserver  WebHelper...Select Method  
WebHelper Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Select(String, ICollection, GetIdTitle, Object, Boolean)</strong></td>
<td>Obsolete. Creates a select list with the values in a collection.</td>
</tr>
<tr>
<td><strong>Select(String, String, ICollection, GetIdTitle, Object, Boolean)</strong></td>
<td>Obsolete. Creates a select list with the values in a collection.</td>
</tr>
</tbody>
</table>
See Also

WebHelper Class
WebHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
WebHelper...:::Select Method (String, ICollection, GetIdTitle, Object, Boolean)

WebHelper Class  See Also  Send Feedback

Creates a select list with the values in a collection.

Namespace:  HttpServer.Helpers
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
[ObsoleteAttribute("Moved to FormHelper")]
public static string Select(
    string name,
    ICollection collection,
    GetIdTitle getIdTitle,
    Object selectedValue,
    bool firstEmpty
)
```

Visual Basic (Declaration)

```vbnet
<ObsoleteAttribute("Moved to FormHelper")>
Public Shared Function Select ( _
    name As String, _
    collection As ICollection, _
    getIdTitle As GetIdTitle, _
    selectedValue As Object, _
    firstEmpty As Boolean _
) As String
```

Visual C++

```cpp
[ObsoleteAttribute(L"Moved to FormHelper")] public:
static String^ Select(
    String^ name,
    ICollection^ collection,
    GetIdTitle^ getIdTitle,
    Object^ selectedValue,
    bool^ firstEmpty
)
```

Parameters

name
Type: `System::String`
Name of the SELECT-tag
collection
Type: System.Collections::ICollection
collection used to generate options.

getIdTitle
Type: HttpServer.Helpers::GetIdTitle
delegate used to return id and title from objects.

selectedValue
Type: System::Object
value that should be marked as selected.

firstEmpty
Type: System::Boolean
First row should contain an empty value.

**Return Value**

string contain a SELECT-tag.
See Also

WebHelper Class
Select Overload
HttpServer.Helpers Namespace
WebHelper...::Select(String, String, ICollection, GetByIdTitle, Object, Boolean)
HttpServer.Helpers...::GetIdTitle

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
WebHelper...Select Method (String, String, ICollection, GetIdTitle, Object, Boolean)

Creates a select list with the values in a collection.

Namespace: HttpServer.Helpers
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
[ObsoleteAttribute("Moved to FormHelper")] public static string Select(
    string name,
    string id,
    ICollection collection,
    GetIdTitle getIdTitle,
    Object selectedValue,
    bool firstEmpty
)
```

### Visual Basic (Declaration)

```vbnet
<ObsoleteAttribute("Moved to FormHelper")] _
Public Shared Function Select ( _
    name As String, _
    id As String, _
    collection As ICollection, _
    getIdTitle As GetIdTitle, _
    selectedValue As Object, _
    firstEmpty As Boolean _
) As String
```

### Visual C++

```csharp
[ObsoleteAttribute(L"Moved to FormHelper")] public:
static String^ Select(
    String^ name,
    String^ id,
    ICollection^ collection,
    GetIdTitle^ getIdTitle,
    Object^ selectedValue,
    bool firstEmpty
)
```

## Parameters

- **name**
Type: `System:::String`  
Name of the SELECT-tag

id  
Type: `System:::String`  
Id of the SELECT-tag

collection  
Type: `System.Collections:::IICollection`  
collection used to generate options.

getIdTitle  
Type: `HttpServer.Helpers:::GetIdTitle`  
delegate used to return id and title from objects.

selectedValue  
Type: `System:::Object`  
value that should be marked as selected.

firstEmpty  
Type: `System:::Boolean`  
First row should contain an empty value.

**Return Value**

string contain a SELECT-tag.
Examples

C#

// Class that is going to be used in a SELECT-tag.
public class User
{
    private readonly string _realName;
    private readonly int _id;

    public User(int id, string realName)
    {
        _id = id;
        _realName = realName;
    }

    public string RealName
    {
        get { return _realName; }  
    }

    public int Id
    {
        get { return _id; }  
    }
}

// Using an inline delegate to generate the select list
public void UserInlineDelegate()
{
    List<User> items = new List<User>();
    items.Add(new User(1, "adam"));
    items.Add(new User(2, "bertial"));
    items.Add(new User(3, "david"));

    string htmlSelect = Select("users", "users", items, delegate(obj
    {
        User use
        id = use
        value =
    }, 2, true);
}

// Using an method as delegate to generate the select list.
public void UseExternalDelegate()
{
    List<User> items = new List<User>();
    items.Add(new User(1, "adam"));
items.Add(new User(2, "bertial"));
items.Add(new User(3, "david"));
string htmlSelect = Select("users", "users", items, UserOptions,

// delegate returning id and title
public static void UserOptions(object o, out object id, out object title)
{
    User user = (User)o;
    id = user.Id;
    value = user.RealName;
}
See Also

**WebHelper Class**
**Select Overload**
**HttpServer.Helpers Namespace**
**WebHelper..::.Select(String, ICollection, GetIdTitle, Object, Boolean)**
**HttpServer.Helpers..::.GetIdTitle**

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
Helpers to make XML handling easier

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static class XmlHelper

Visual Basic (Declaration)

Public NotInheritable Class XmlHelper

Visual C++

public ref class XmlHelper abstract sealed
Inheritance Hierarchy

System::Object
HttpServer.Helpers::XmlHelper
See Also

XmlHelper Members
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `XmlHelper` type exposes the following members.
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serialize</td>
<td>Serializes object to XML.</td>
</tr>
</tbody>
</table>
See Also

XmlHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **XmlHelper** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serialize</td>
<td>Serializes object to XML.</td>
</tr>
</tbody>
</table>
See Also

XmlHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Serializes object to XML.

**Namespace:** [HttpServer.Helpers](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string Serialize(
   Object value
)

Visual Basic (Declaration)

Public Shared Function Serialize ( _
   value As Object _
) As String

Visual C++

public:
static String^ Serialize( 
   Object^ value
)

Parameters

value
   Type: System::Object
   object to serialize.

Return Value

xml
Remarks

Removes namespaces and adds indentation
See Also

XmlHelper Class
HttpServer.Helpers Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
### Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="#" alt="PrototypeImp" /></td>
<td>PrototypeJS implementation of the javascript functions.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
PrototypeJS implementation of the javascript functions.

**Namespace:**  [HttpServer.Helpers.Implementations](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class PrototypeImp : JavascriptHelperImplementation

Visual Basic (Declaration)

Public Class PrototypeImp _
    Inherits JavascriptHelperImplementation

Visual C++

public ref class PrototypeImp : public JavascriptHelperImplementation
Inheritance Hierarchy

System..:::Object
HttpServer.Helpers..:::JavascriptHelperImplementation
HttpServer.Helpers.Implementations..:::PrototypeImp
See Also

PrototypeImp Members
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **PrototypeImp** type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PrototypeImp</td>
<td>Initializes a new instance of the PrototypeImp class</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>AjaxFormOnSubmit</strong></td>
<td>javascript action that should be added to the &quot;onsubmit&quot; event in the form tag. (Overrides JavascriptHelperImplementation::AjaxFormOnSubmit)</td>
</tr>
<tr>
<td><strong>AjaxRequest</strong></td>
<td>Requests a url through ajax (Overrides JavascriptHelperImplementation::AjaxRequest(String, array&lt;String&gt;[][]) )</td>
</tr>
<tr>
<td><strong>AjaxUpdater</strong></td>
<td>Ajax requests that updates an element with the fetched content (Overrides JavascriptHelperImplementation::AjaxUpdater(String, String, array&lt;String&gt;[][]) )</td>
</tr>
<tr>
<td><strong>CloseDialog</strong></td>
<td>Close a javascript dialog window/div. (Overrides JavascriptHelperImplementation::CloseDialog())</td>
</tr>
<tr>
<td><strong>CreateDialog</strong></td>
<td>create a modal dialog (usually using DIVs) (Overrides JavascriptHelperImplementation::CreateDialog(String, String, array&lt;String&gt;[][]) )</td>
</tr>
<tr>
<td><strong>DialogLink</strong></td>
<td>A link that pop ups a Dialog (overlay div) (Overrides JavascriptHelperImplementation::DialogLink(String, String, array&lt;String&gt;[][]) )</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GenerateOptions</strong></td>
<td>Generates a list with JS options. (Inherited from JavascriptHelperImplementation.)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
C# Visual C++
C# Webserver
PrototypeImp Constructor

PrototypeImp Class  See Also  Send Feedback

Initializes a new instance of the PrototypeImp class

**Namespace:**  [HttpServer.Helpers.Implementations](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public PrototypeImp()
```

Visual Basic (Declaration)

```vbnet
Public Sub New
```

Visual C++

```cpp
public:
PrototypeImp()
```
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **PrototypeImp** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AjaxFormOnSubmit</strong></td>
<td>javascript action that should be added to the &quot;onsubmit&quot; event in the form tag. (Overrides JavascriptHelperImplementation...::AjaxFormOnSubmit())</td>
</tr>
<tr>
<td><strong>AjaxRequest</strong></td>
<td>Requests a url through ajax (Overrides JavascriptHelperImplementation...::AjaxRequest(String, array&lt;String&gt;[][])[])</td>
</tr>
<tr>
<td><strong>AjaxUpdater</strong></td>
<td>Ajax requests that updates an element with the fetched content (Overrides JavascriptHelperImplementation...::AjaxUpdater(String, String, array&lt;String&gt;[][])[])</td>
</tr>
<tr>
<td><strong>CloseDialog</strong></td>
<td>Close a javascript dialog window/div. (Overrides JavascriptHelperImplementation...::CloseDialog() [])</td>
</tr>
<tr>
<td><strong>CreateDialog</strong></td>
<td>create a modal dialog (usually using DIVs) (Overrides JavascriptHelperImplementation...::CreateDialog(String, String, array&lt;String&gt;[][])[])</td>
</tr>
<tr>
<td><strong>DialogLink</strong></td>
<td>A link that pop ups a Dialog (overlay div) (Overrides JavascriptHelperImplementation...::DialogLink(String, S array&lt;String&gt;[][])[])</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GenerateOptions</strong></td>
<td>Generates a list with JS options. (Inherited from JavascriptHelperImplementation.)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.</td>
</tr>
</tbody>
</table>
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
javascript action that should be added to the "onsubmit" event in the form tag.

**Namespace:**  [HttpServer.Helpers.Implementations](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override string AjaxFormOnSubmit()

Visual Basic (Declaration)

Public Overrides Function AjaxFormOnSubmit As String

Visual C++

public:
virtual String^ AjaxFormOnSubmit() override

Return Value

See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Requests a url through ajax

Namespace:  [HttpServer.Helpers.Implementations](#)
Assembly:  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override string AjaxRequest(
    string url,
    params string[] options
)
```

Visual Basic (Declaration)

```vbnet
Public Overrides Function AjaxRequest ( _
    url As String, _
    ParamArray options As String() _
) As String
```

Visual C++

```cpp
public:
virtual String^ AjaxRequest(
    String^ url,
    ... array<String^>[] options
) override
```

Parameters

url

Type: `System::String`
url to fetch

options

Type: array< `System::String` >[0][]
optional options in format "key, value, key, value", used in JS request object.

Return Value

a link tag
Remarks

onclick attribute is used by this method.
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Ajax requests that updates an element with the fetched content

**Namespace:**  [HttpServer.Helpers.Implementations](http://example.com/path)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override string AjaxUpdater(string url, string targetId, params string[] options)
```

Visual Basic (Declaration)

```vbnet
Public Overrides Function AjaxUpdater ( _
    url As String, _
    targetId As String, _
    ParamArray options As String() _
) As String
```

Visual C++

```cpp
public:
virtual String^ AjaxUpdater( 
    String^ url, 
    String^ targetId, 
    ... array<String^>^ options
) override
```

Parameters

url
Type: System::String
Url to fetch content from

targetId
Type: System::String
element to update

options
Type: array< System::String >[]()}
options in format "key, value, key, value"

Return Value

A link tag.
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Close a javascript dialog window/div.

**Namespace:**  [HttpServer.Helpers.Implementations](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public override string CloseDialog()
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function CloseDialog As String
```

**Visual C++**

```cpp
public:
virtual String^ CloseDialog() override
```

**Return Value**

javascript for closing a dialog.
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
create a modal dialog (usually using DIVs)

**Namespace:**  [HttpServer.Helpers.Implementations](http://example.com)
**Assembly:**  [HttpServer (in HttpServer.dll)](http://example.com) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public override string CreateDialog(
    string url,
    string title,
    params string[] options
)
```

### Visual Basic (Declaration)

```vbnet
Public Overrides Function CreateDialog (_
    url As String, _
    title As String, _
    ParamArray options As String() _
) As String
```

### Visual C++

```cpp
public:
virtual String^ CreateDialog(
    String^ url,
    String^ title,
    ... array<String^>^ options
) override
```

## Parameters

### url

- **Type:** `System::String`
- url to fetch

### title

- **Type:** `System::String`
- dialog title

### options

- **Type:** array< `System::String` >[]()
javascript/html attributes. javascript options ends with colon ':

Return Value

[Missing <returns> documentation for
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A link that pop ups a Dialog (overlay div)

Namespace:  HttpServer.Helpers.Implementations
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public override string DialogLink(
    string url,
    string title,
    params string[] htmlAttributes
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function DialogLink ( _
    url As String, _
    title As String, _
    ParamArray htmlAttributes As String() _
) As String
```

**Visual C++**

```cpp
public:
virtual String^ DialogLink(
    String^ url,
    String^ title,
    ... array<String^>^ htmlAttributes
) override
```

**Parameters**

- **url**
  - Type: **System::String**
  - url to contents of dialog

- **title**
  - Type: **System::String**
  - link title

- **htmlAttributes**
  - Type: array< **System::String** >[]()
name, value, name, value

Return Value

A "a"-tag that popups a dialog when clicked
Remarks

Requires Control.Modal found here: http://livepipe.net/projects/control_modal/

And the following javascript (load it in application.js):

```
C#

Event.observe(window, 'load',
    function() {
        document.getElementsByClassName('modal').each(function(link){
            ...
        });
    }
);
```
Examples

WebHelper.DialogLink("/user/show/1", "show user", "onmouseover", "alert('booh!');");
See Also

PrototypeImp Class
HttpServer.Helpers.Implementations Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ControllerModule</strong></td>
<td>A controller module is a part of the ModelViewController design pattern. It gives you a way to create user friendly urls.</td>
</tr>
<tr>
<td><strong>FileModule</strong></td>
<td>The purpose of this module is to serve files.</td>
</tr>
<tr>
<td><strong>HttpModule</strong></td>
<td>A HttpModule can be used to serve urls. The module itself decides if it should serve a url or not. In this way, you can get a very flexible http app since you can let multiple modules serve almost similar urls.</td>
</tr>
<tr>
<td><strong>HttpModuleExceptionEventArgs</strong></td>
<td>Used to inform http server that</td>
</tr>
<tr>
<td><strong>ReverseProxyModule</strong></td>
<td>A reverse proxy are used to act as a bridge between local (protected/hidden) websites and public clients. A typical usage is to allow web servers on non standard ports to still be available to the public clients, or allow web servers on private ips to be available.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A controller module is a part of the ModelViewController design pattern. It gives you a way to create user friendly urls.

**Namespace:**  [HttpServer.HttpModules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ControllerModule : IHttpModule

Visual Basic (Declaration)

Public Class ControllerModule _
    Inherits IHttpModule

Visual C++

public ref class ControllerModule : public IHttpModule
Remarks

The controller module uses the flyweight pattern which means that the memory usage will continue to increase until the module have enough objects in memory to serve all concurrent requests. The objects are reused and will not be freed.
Examples

ControllerModule module = new ControllerModule(); module.Add(new UserController()); module.Add(new SearchController()); myWebsite.Add(module);
Inheritance Hierarchy

- `System..:::Object`
- `HttpServer.HttpModules..:::HttpModule`
- `HttpServer.HttpModules..:::ControllerModule`
See Also

ControllerModule Members
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ControllerModule` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ControllerModule</code></td>
<td>Initializes a new instance of the <code>ControllerModule</code> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>The controller module uses the prototype design pattern to be able to create new controller objects for requests if the stack is empty.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url (Overrides <code>HttpModule...Process(HttpRequest, HttpResponse, HttpSession)</code>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Get a prototype</td>
</tr>
</tbody>
</table>
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ControllerModule Constructor

Initializes a new instance of the ControllerModule class

**Namespace:**  [HttpServer.HttpModules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public ControllerModule()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New
```

**Visual C++**

```cpp
public:
ControllerModule()
```
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ControllerModule` type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>The controller module uses the prototype design pattern to be able to create new controller objects for requests if the stack is empty.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url (Overrides HttpModule...::Process(HttpRequest, HttpResponse, HttpSession).)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The controller module uses the prototype design pattern to be able to create new controller objects for requests if the stack is empty.

**Namespace:**  [HttpServer.HttpModules](http://example.com/HttpServer.HttpModules)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public void Add(
    RequestController prototype
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Add (_
    prototype As RequestController _
)
```

**Visual C++**

```cpp
public:
void Add(
    RequestController^ prototype
)
```

**Parameters**

prototype
Type: `HttpServer.Controllers...::RequestController`
A prototype which will be cloned for each request
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System..::.ArgumentNullException</td>
<td></td>
</tr>
<tr>
<td>System..::.InvalidProgramException</td>
<td>If a controller with that name have been added already.</td>
</tr>
</tbody>
</table>
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ControllerModule::Process Method

Namespace:  HttpServer.HttpModules
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool Process(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)

Visual Basic (Declaration)

Public Overrides Function Process ( _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
) As Boolean

Visual C++

public:
virtual bool Process(
    HttpRequest^ request,
    HttpResponse^ response,
    HttpSession^ session
) override

Parameters

request
    Type: HttpServer::HttpRequest
    Information sent by the browser about the request

response
    Type: HttpServer::HttpResponse
    Information that is being sent back to the client.

session
    Type: HttpServer::HttpSession
Session used to

Return Value

[Missing <returns> documentation for
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ControllerModule` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Get a prototype</td>
</tr>
</tbody>
</table>
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get a prototype

**Namespace:**  [HttpServer.HttpModules](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RequestController this[string controllerName]
{
    get;
}

Visual Basic (Declaration)

Public ReadOnly Default Property Item ( _
    controllerName As String _
) As RequestController

Visual C++

public:
    property RequestController^ default[String^ controllerName] {
        RequestController^ get (String^ controllerName);
    }

Parameters

controllerName
    Type: System::String
    in lowercase, without "Controller"

Return Value

The controller if found; otherwise null.
Example

RequestController userController = controllerModule["user"]; //fetches the class UserController
See Also

ControllerModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The purpose of this module is to serve files.

**Namespace:** [HttpServer.HttpModules](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public class FileModule : HttpModule

Visual Basic (Declaration)
Public Class FileModule
    Inherits HttpModule

Visual C++
public ref class FileModule : public HttpModule
Inheritance Hierarchy

System..:::Object
HttpServer.HttpModules..:::HttpModule
HttpServer.HttpModules..:::FileModule
See Also

FileModule Members
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **FileModule** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FileModule</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddDefaultMimeTypes</strong></td>
<td>Mimetypes that this class can handle per default</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if the request should be handled by this module. Invoked by the HttpServer</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetFileExtension</strong></td>
<td>return a file extension from an absolute uri path (or plain filename)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url (Overrides <strong>HttpContext::Process(HttpRequest, HttpResponse, HttpSession)</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ForbiddenChars</td>
<td>characters that may not exist in a path.</td>
</tr>
<tr>
<td>MimeTypes</td>
<td>List with all mime-type that are allowed.</td>
</tr>
</tbody>
</table>
See Also

FileModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FileModule Constructor

FileModule Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>FileModule(String, String)</code></td>
<td>Initializes a new instance of the <code>FileModule</code> class</td>
</tr>
<tr>
<td><code>FileModule(String, String, Boolean)</code></td>
<td>Initializes a new instance of the <code>FileModule</code> class</td>
</tr>
</tbody>
</table>
See Also

FileModule Class
FileModule Members
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
FileModule Constructor (String, String)

Namespace:  HttpServer.HttpModules
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public FileModule(
    string baseUri,
    string basePath
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New (_
    baseUri As String, _
    basePath As String _
)
```

Visual C++

```cpp
public:
FileModule(
    String^ baseUri, 
    String^ basePath
)
```

Parameters

baseUri
Type: System::String

[Missing <param name="baseUri"/> documentation for
]

basePath
Type: System::String

[Missing <param name="basePath"/> documentation for
]
See Also

FileModule Class
FileModule Overload
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `FileModule` class

**Namespace:**  [HttpServer.HttpModules](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public FileModule(
    string baseUri,
    string basePath,
    bool useLastModifiedHeader
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New (_
    baseUri As String, _
    basePath As String, _
    useLastModifiedHeader As Boolean _
)
```

Visual C++

```cpp
public:
FileModule(
    String^ baseUri,
    String^ basePath,
    bool useLastModifiedHeader
)
```

Parameters

baseUri
Type: `System::String`
Uri to serve, for instance "/files/"

basePath
Type: `System::String`
Path on hard drive where we should start looking for files

useLastModifiedHeader
Type: `System::Boolean`
If true a Last-Modifed header will be sent upon requests urging webbrowser to cache files
See Also

FileModule Class
FileModule Overload
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `FileModule` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AddDefaultMimeTypes</code></td>
<td>Mimetypes that this class can handle per default</td>
</tr>
<tr>
<td><code>CanHandle</code></td>
<td>Determines if the request should be handled by this module. Invoked by the HttpServer</td>
</tr>
<tr>
<td><code>Equals</code></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td><code>Finalize</code></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><code>GetFileExtension</code></td>
<td>return a file extension from an absolute uri path (or plain filename)</td>
</tr>
<tr>
<td><code>GetHashCode</code></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><code>GetType</code></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td><code>MemberwiseClone</code></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td><code>Process</code></td>
<td>Method that process the url (Overrides <code>HttpModule::&lt;cmp::Process(HttpRequest, HttpResponse, HttpSession&gt;)</code>.)</td>
</tr>
<tr>
<td><code>ToString</code></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
</tbody>
</table>
See Also

FileModule Class
HttpServer.HttpModules Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Mimetypes that this class can handle per default

**Namespace:**  [HttpServer.HttpModules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void AddDefaultMimeTypes()

Visual Basic (Declaration)

Public Sub AddDefaultMimeTypes

Visual C++

public:
void AddDefaultMimeTypes()
See Also

FileSync Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines if the request should be handled by this module. Invoked by the HttpServer

**Namespace:** [HttpServer.HttpModules](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public bool CanHandle(Uri uri)
```

**Visual Basic (Declaration)**

```vbnet
Public Function CanHandle ( _
    uri As Uri _
) As Boolean
```

**Visual C++**

```cpp
public:
bool CanHandle(
    Uri^ uri
)
```

### Parameters

**uri**

Type: `System::Uri`


### Return Value

true if this module should handle it.
See Also

FileModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
namespace: HttpServer.HttpModules
assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public static string GetFileExtension(
    string uri
)
```

### Visual Basic (Declaration)

```vbnet
Public Shared Function GetFileExtension ( _
    uri As String _
) As String
```

### Visual C++

```cpp
public:
static String^ GetFileExtension(
    String^ uri
)
```

## Parameters

**uri**

Type: **System::String**


## Return Value

See Also

FileModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Method that process the url

**Namespace:**  [HttpServer.HttpModules](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override bool Process(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function Process (_
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
) As Boolean
```

**Visual C++**

```cpp
public:
    virtual bool Process(
        HttpRequest^ request,
        HttpResponse^ response,
        HttpSession^ session
    ) override
```

**Parameters**

*request*

Type: `HttpRequest`

Information sent by the browser about the request

*response*

Type: `HttpResponse`

Information that is being sent back to the client.

*session*

Type: ` HttpSession`
Session used to

**Return Value**

[Missing <returns> documentation for
See Also

FileModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **FileModule** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ForbiddenChars</td>
<td>characters that may not exist in a path.</td>
</tr>
<tr>
<td>MimeTypes</td>
<td>List with all mime-type that are allowed.</td>
</tr>
</tbody>
</table>
See Also

FileModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
characters that may not exist in a path.

**Namespace:** [HttpServer.HttpModules](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string[] ForbiddenChars { get; set; }

Visual Basic (Declaration)

Public Property ForbiddenChars As String()

Visual C++

public:
property array<String^>^ ForbiddenChars {
 array<String^>^ get ();
 void set (array<String^>^ value);
}
Examples

fileMod.ForbiddenChars = new string[] { "\", '..', ':' };
See Also

**FileModule Class**
**HttpServer.HttpModules Namespace**

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
List with all mime-type that are allowed.

**Namespace:** [HttpServer.HttpModules](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public IDictionary<string, string> MimeTypes { get; }

Visual Basic (Declaration)

Public ReadOnly Property MimeTypes As IDictionary(Of String, String)

Visual C++

public:
property IDictionary<String^, String^>^ MimeTypes {
            IDictionary<String^, String^>^ get ();
        }
Remarks

All other mime types will result in a Forbidden http status code.
See Also

FileModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A HttpModule can be used to serve urls. The module itself decides if it should serve a url or not. In this way, you can get a very flexible http app since you can let multiple modules serve almost similar urls.

Namespace: `HttpServer.HttpModules`
Assembly: `HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)`
## Syntax

### C#

```csharp
public abstract class HttpModule
```

### Visual Basic (Declaration)

```vbnet
Public MustInherit Class HttpModule
```

### Visual C++

```cpp
public ref class HttpModule abstract
```
Remarks

Throw UnauthorizedException if you are using a AuthenticationModule and want to prompt for username/password.
Inheritance Hierarchy

System:::Object
HttpServer.HttpModules:::HttpModule
  HttpServer.Controllers:::RequestController
  HttpServer.HttpModules:::ControllerModule
  HttpServer.HttpModules:::FileModule
  HttpServer.HttpModules:::ReverseProxyModule
See Also

HttpModule Members
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpModule` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpModule</td>
<td>Initializes a new instance of the <a href="#">HttpModule</a> class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

HttpModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initialize a new instance of the `HttpModule` class

**Namespace:** `HttpServer.HttpModules`

**Assembly:** `HttpServer` (in `HttpServer.dll`) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
protected HttpModule()
```

**Visual Basic (Declaration)**

```vbnet
Protected Sub New
```

**Visual C++**

```cpp
protected:
HttpModule()
```
See Also

HttpModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpModule` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

HttpModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Method that process the url

**Namespace:**  [HttpServer.HttpModules](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public abstract bool Process(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)
```

**Visual Basic (Declaration)**

```vbnet
Public MustOverride Function Process ( _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
) As Boolean
```

**Visual C++**

```cpp
public:
  virtual bool Process(
      HttpRequest^ request,
      HttpResponse^ response,
      HttpSession^ session
  ) abstract
```

### Parameters

**request**
- Type: `HttpRequest`
  - Information sent by the browser about the request

**response**
- Type: `HttpResponse`
  - Information that is being sent back to the client.

**session**
- Type: `HttpSession`
Session used to

Return Value

ture if this module handled the request.
See Also

HttpModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Used to inform http server that

**Namespace:** HttpServer.HttpModules
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HttpModuleExceptionEventArgs : EventArgs

Visual Basic (Declaration)

Public Class HttpModuleExceptionEventArgs
    Inherits EventArgs

Visual C++

public ref class HttpModuleExceptionEventArgs : public EventArgs
Inheritance Hierarchy

System..:::Object
System..:::EventArgs
HttpServer.HttpModules..:::HttpModuleExceptionEventArgs
See Also

HttpModuleExceptionEventArgs Members
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpModuleExceptionEventArgs` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpModuleExceptionEventArgs</td>
<td>Event arguments used when an exception is thrown by a module</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exception</td>
<td>Exception thrown in a module</td>
</tr>
</tbody>
</table>
See Also

HttpModuleExceptionEventArgs Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpModuleExceptionEventArgs Constructor

Event arguments used when an exception is thrown by a module

**Namespace:** [HttpServer.HttpModules](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpModuleExceptionEventArgs(
    Exception e
)
```

Visual Basic (Declaration)

```vbnet
Public Sub New ( _
    e As Exception _
)
```

Visual C++

```cpp
public:
HttpModuleExceptionEventArgs(
    Exception^ e
)
```

Parameters

e
  Type: System::Exception
  the exception
See Also

HttpModuleExceptionEventArgs Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HttpModuleExceptionEventArgs** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

HttpModuleExceptionEventArgs Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HttpModuleExceptionEventArgs` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exception</td>
<td>Exception thrown in a module</td>
</tr>
</tbody>
</table>
See Also

HttpModuleExceptionEventArgs Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpModuleExceptionEventArgs Class  See Also  Send Feedback

Exception thrown in a module

**Namespace:**  [HttpServer.HttpModules](http://example.com/HttpServer.HttpModules)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Exception Exception { get; }

Visual Basic (Declaration)

Public ReadOnly Property Exception As Exception

Visual C++

public:
property Exception^ Exception {
    Exception^ get ();
}

See Also

HttpModuleExceptionEventArgs Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A reverse proxy are used to act as a bridge between local (protected/hidden) websites and public clients. A typical usage is to allow web servers on non standard ports to still be available to the public clients, or allow web servers on private ips to be available.

**Namespace:** [HttpServer.HttpModules](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ReverseProxyModule : IHttpModule

Visual Basic (Declaration)

Public Class ReverseProxyModule _
    Inherits IHttpModule

Visual C++

public ref class ReverseProxyModule : public IHttpModule
Inheritance Hierarchy

System:::Object
   HttpServer.HttpModules:::HttpModule
      HttpServer.HttpModules:::ReverseProxyModule
See Also

ReverseProxyModule Members
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ReverseProxyModule type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReverseProxyModule</td>
<td>Initializes a new instance of the ReverseProxyModule class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url (Overrides <a href="https://docs.microsoft.com/en-us/dotnet/api/system.web.httpmodule.process">HttpModule.:::Process(HttpRequest, HttpResponse, HttpSession)</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

ReverseProxyModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ReverseProxyModule Constructor

Initializes a new instance of the ReverseProxyModule class

**Namespace:**  [HttpServer.HttpModules]

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ReverseProxyModule(
    string source,
    string destination
)

Visual Basic (Declaration)

Public Sub New (_
    source As String, _
    destination As String _
)

Visual C++

public:
ReverseProxyModule(
    String^ source,
    String^ destination
)

Parameters

source
Type: System::String
Base url requested from browser

destination
Type: System::String
Base url on private web server
Examples

_server.Add(new
See Also

ReverseProxyModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ReverseProxyModule type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Method that process the url</td>
</tr>
<tr>
<td></td>
<td>(Overides <code>HttpModule..::.Process(HttpRequest, HttpResponse, HttpSession).</code>)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

ReverseProxyModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
ReverseProxyModule...:::Process Method
ReverseProxyModule Class  See Also  Send Feedback

Method that process the url

Namespace:  HttpServer.HttpModules
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool Process(
    HttpRequest request,
    HttpResponse response,
    HttpSession session
)

Visual Basic (Declaration)

Public Overrides Function Process ( _
    request As HttpRequest, _
    response As HttpResponse, _
    session As HttpSession _
) As Boolean

Visual C++

public:
virtual bool Process(
    HttpRequest^ request,
    HttpResponse^ response,
    HttpSession^ session
) override

Parameters

request
    Type: HttpServer::HttpRequest
    Information sent by the browser about the request

response
    Type: HttpServer::HttpResponse
    Information that is being sent back to the client.

session
    Type: HttpServer::HttpSession
Session used to

**Return Value**

See Also

ReverseProxyModule Class
HttpServer.HttpModules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.Rendering]
## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArgumentContainer</td>
<td>A container class grouping mapping name, argument type and the argument object</td>
</tr>
<tr>
<td>CodeGeneratorException</td>
<td>Contains information on where in the template the error occurred, and what the error was.</td>
</tr>
<tr>
<td>CompileException</td>
<td>Exception thrown if compilation fails</td>
</tr>
<tr>
<td>TemplateArguments</td>
<td>Arguments being used in a template.</td>
</tr>
<tr>
<td>TemplateCompiler</td>
<td>The compiler is responsible of creating a render object which can be cached and used over and over again.</td>
</tr>
<tr>
<td>TemplateManager</td>
<td>Purpose if this class is to take template objects and keep them in memory. It will also take a filename and the code generator to use if when the template have been changed on disk.</td>
</tr>
</tbody>
</table>
## Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateGenerator</td>
<td>A code generator is used to convert template code to something that we can work with, as html or c# code.</td>
</tr>
<tr>
<td>TinyTemplate</td>
<td>Interface for dynamically generated templates.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A container class grouping mapping name, argument type and the argument object

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class ArgumentContainer

Visual Basic (Declaration)

Public Class ArgumentContainer

Visual C++

public ref class ArgumentContainer
Inheritance Hierarchy

System:::Object
HttpServer.Rendering:::ArgumentContainer
See Also

ArgumentContainer Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ArgumentContainer** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArgumentContainer</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>SetObject</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Attribute for retrieving the name. The name cannot be set however because an argument is defined by its name changing the name would be changing the arguments meaning, thus an argument needing a name change should rather be recreated. Returns or changes the argument object. If the object is to be changed to null the type must be passed aswell, in that case <strong>SetObject(Object, Type)</strong>.</td>
</tr>
<tr>
<td>Object</td>
<td>Returns the type of the argument object. The property cannot be set since the type depends on and must correspond to the type of the object.</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ArgumentContainer(String, Object)</code></td>
<td>Constructor to initiate an ArgumentContainer, will set the ArgumentType to the value of the argumentObject's type</td>
</tr>
<tr>
<td><code>ArgumentContainer(String, Object, Type)</code></td>
<td>Alternative constructor allowing argumentObject parameter to be null</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
ArgumentContainer Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++
C#  Webserver

ArgumentContainer Constructor (String, Object)

ArgumentContainer Class  See Also  Send Feedback

Constructor to initiate an ArgumentContainer, will set the ArgumentType to the value of the argumentObject's type

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public ArgumentContainer(
    string argumentName,
    Object argumentObject
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New (_
    argumentName As String,
    argumentObject As Object
)
```

### Visual C++

```cpp
public:
ArgumentContainer(
    String^ argumentName,
    Object^ argumentObject
)
```

## Parameters

**argumentName**
- Type: `System::String`
- Cannot be null

**argumentObject**
- Type: `System::Object`
- Cannot be null
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System..::.ArgumentNullException</td>
<td>If argumentName or argumentObject is null</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
ArgumentContainer Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ArgumentContainer Constructor (String, Object, Type)

Alternative constructor allowing argumentObject parameter to be null

**Namespace:** [HttpServer.Rendering](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public ArgumentContainer(
    string argumentName,
    Object argumentObject,
    Type argumentType
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New (
    argumentName As String,
    argumentObject As Object,
    argumentType As Type
)
```

### Visual C++

```cpp
public:
ArgumentContainer(
    String* argumentName,
    Object* argumentObject,
    Type* argumentType
)
```

## Parameters

**argumentName**
- Type: `System::String`
- Cannot be null

**argumentObject**
- Type: `System::Object`


**argumentType**
Type: System:::Type
Cannot be null
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.NullReferenceException</code></td>
<td>If argumentName or argumentType is null</td>
</tr>
<tr>
<td><code>System..::.ArgumentException</code></td>
<td>If argumentObject != null and argumentType != typeof(argumentObject)</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
ArgumentContainer Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ArgumentContainer type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>SetObject</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver

ArgumentContainer::SetObject Method
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SetObject</strong>(Obj)</td>
<td>Sets the object and type of the argument, equal to ArgumentContainer.Object = ...</td>
</tr>
<tr>
<td><strong>SetObject</strong>(Obj, Type)</td>
<td>Sets the argument object and type. Type of the object and argumentType must correspond.</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
ArgumentContainer Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ArgumentContainer...:::SetObject Method (Object)

Sets the object and type of the argument, equal to ArgumentContainer.Object = ...

**Namespace:** [HttpServer.Rendering](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void SetObject(
    Object argumentObject
)

Visual Basic (Declaration)

Public Sub SetObject ( _
    argumentObject As Object _
)

Visual C++

public:
void SetObject(
    Object^ argumentObject
)

Parameters

argumentObject
    Type: System::Object
    Cannot be null
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td>If argumentObject is null</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class  
SetObject Overload  
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Sets the argument object and type. Type of the object and argumentType must correspond.

**Namespace:** [HttpServer.Rendering](http://example.com)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void SetObject(
    Object argumentObject,
    Type argumentType
)

Visual Basic (Declaration)

Public Sub SetObject (_
    argumentObject As Object, _
    argumentType As Type _
)

Visual C++

public:
void SetObject(
    Object^ argumentObject,
    Type^ argumentType
)

Parameters

argumentObject
    Type: System::Object

    [Missing <param name="argumentObject"/> documentation for

argumentType
    Type: System::Type
    Cannot be null
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::ArgumentNullException</code></td>
<td>If <code>argumentType</code> is null</td>
</tr>
<tr>
<td><code>System:::ArgumentException</code></td>
<td>If <code>typeof(argumentObject)</code> differs from <code>argumentType</code> and <code>object != null</code></td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
SetObject Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ArgumentContainer` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Attribute for retrieving the name. The name cannot be set however because an argument is defined by its name changing the name would be changing the arguments meaning, thus an argument needing a name change should rather be recreated</td>
</tr>
<tr>
<td>Object</td>
<td>Returns or changes the argument object. If the object is to be changed to null the type must be passed aswell, in that case <strong>SetObject(Object, Type)</strong></td>
</tr>
<tr>
<td>Type</td>
<td>Returns the type of the argument object. The property cannot be set since the type depends on and must correspond to the type of the object</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
HttpServer.Rendering Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ArgumentContainer..::.Name Property

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)

Attribute for retrieving the name. The name cannot be set however because an argument is defined by its name changing the name would be changing the arguments meaning, thus an argument needing a name change should rather be recreated.
Syntax

**C#**

```csharp
public string Name { get; }
```

**Visual Basic (Declaration)**

```vbnet
Public ReadOnly Property Name As String
```

**Visual C++**

```c++
public:
property String^ Name {
    String^ get ();
}
```
See Also

ArgumentContainer Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ArgumentContainer:::Object Property

Returns or changes the argument object. If the object is to be changed to null the type must be passed as well, in that case **SetObject(Object, Type)**

**Namespace:** HttpServer.Rendering

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public Object Object { get; set; }
```

Visual Basic (Declaration)

```vbnet
Public Property Object As Object
```

Visual C++

```cpp
public:
property Object^ Object {
    Object^ get ();
    void set (Object^ value);
}
```
# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::ArgumentNullException</td>
<td>If set value is null</td>
</tr>
</tbody>
</table>
See Also

ArgumentContainer Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ArgumentContainer...:::Type Property

ArgumentContainer Class  See Also  Send Feedback

Returns the type of the argument object. The property cannot be set since the type depends on and must correspond to the type of the object

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Type Type { get; }

Visual Basic (Declaration)

Public ReadOnly Property Type As Type

Visual C++

public:
property Type^ Type {
    Type^ get ();
}
}
See Also

ArgumentContainer Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains information on where in the template the error occurred, and what the error was.

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class CodeGeneratorException : Exception

Visual Basic (Declaration)

Public Class CodeGeneratorException _
    Inherits Exception

Visual C++

public ref class CodeGeneratorException : public Exception
Inheritance Hierarchy

System..:::Object
System..:::Exception
HttpServer.Rendering..:::CodeGeneratorException
See Also

CodeGenException Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `CodeGenException` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CodeGeneratorException</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="methods" alt="" /> Finalize</td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>).</td>
</tr>
<tr>
<td><img src="methods" alt="" /> GetBaseException</td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>).</td>
</tr>
<tr>
<td><img src="methods" alt="" /> GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>). When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception.</td>
</tr>
<tr>
<td><img src="methods" alt="" /> GetObjectData</td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>). Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>). Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>).</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>HelpLink</td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>HResult</td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>InnerException</td>
<td>Gets the Exception instance that caused the current exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>Line</td>
<td>Returns the actual line where the error originated</td>
</tr>
<tr>
<td>LineNumber</td>
<td>Line number in template</td>
</tr>
<tr>
<td>Message</td>
<td>Gets a message that describes the current exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td>Source</td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)</td>
</tr>
<tr>
<td>StackTrace</td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)</td>
</tr>
<tr>
<td>TargetSite</td>
<td>Gets the method that throws the current exception. (Inherited from Exception.)</td>
</tr>
</tbody>
</table>
See Also

CodeGenException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members  C# Webserver  CodeGeneratorException Constructor  CodeGeneratorException Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CodeGenException(Int32, String)</code></td>
<td>Initializes a new instance of the <code>CodeGenException</code> class.</td>
</tr>
<tr>
<td><code>CodeGenException(Int32, String, String)</code></td>
<td>Initializes a new instance of the <code>CodeGenException</code> class.</td>
</tr>
</tbody>
</table>
See Also

CodeGenException Class
CodeGenException Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `CodeGeneratorException` class.

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public CodeGeneratorException(
    int lineNumber,
    string error
)

Visual Basic (Declaration)

Public Sub New ( _
    lineNumber As Integer, _
    error As String _
)

Visual C++

public:
CodeGeneratorException(  
    int lineNumber,
    String^ error
)

Parameters

lineNumber
    Type: System::Int32
    Line that the error appeared on.

error
    Type: System::String
    error description.
See Also

CodeGenException Class
CodeGenException Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
CodeGeneratorException Constructor (Int32, String, String)

CodeGeneratorException Class  See Also  Send Feedback

Initializes a new instance of the `CodeGeneratorException` class.

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public CodeGeneratorException(
    int lineNumber,
    string line,
    string error
)

Visual Basic (Declaration)

Public Sub New (_
    lineNumber As Integer, _
    line As String, _
    error As String _
)

Visual C++

public:
    CodeGeneratorException(
        int lineNumber,
        String^ line,
        String^ error
    )

Parameters

lineNumber
Type: System:::Int32
Line that the error appeared on.

line
Type: System:::String
line contents.

error
Type: System:::String
error description.
See Also

CodeGenException Class
CodeGenException Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `CodeGeneratorException` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td></td>
</tr>
</tbody>
</table>
See Also

CodeGenException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `CodeGenException` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>HelpLink</td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>HResult</td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>InnerException</td>
<td>Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>Line</td>
<td>Returns the actual line where the error originated.</td>
</tr>
<tr>
<td>LineNumber</td>
<td>Line number in template</td>
</tr>
<tr>
<td>Message</td>
<td>Gets a message that describes the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>Source</td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>StackTrace</td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from <code>Exception</code>.)</td>
</tr>
<tr>
<td>TargetSite</td>
<td>Gets the method that throws the current exception. (Inherited from <code>Exception</code>.)</td>
</tr>
</tbody>
</table>
See Also

CodeGenException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
CodeGenException.Line Property

CodeGenException Class  See Also  Send Feedback

Returns the actual line where the error originated

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Line { get; }

Visual Basic (Declaration)

Public ReadOnly Property Line As String

Visual C++

public:
property String^ Line {
    String^ get ();
}


See Also

CodeGenException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
CodeGenException..::..LineNumber Property
CodeGenException Class  See Also  Send Feedback

Line number in template

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public int LineNumber { get; }
```

Visual Basic (Declaration)

```vbnet
Public ReadOnly Property LineNumber As Integer
```

Visual C++

```c++
public:
    property int LineNumber {
        int get ();
    }
```
See Also

CodeGenException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Exception thrown if compilation fails

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class CompileException : Exception

Visual Basic (Declaration)

Public Class CompileException _
    Inherits Exception

Visual C++

public ref class CompileException : public Exception
Inheritance Hierarchy

System..::.Object
System..::.Exception
HttpServer.Rendering..::.CompileException
See Also

CompileException Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **CompileException** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CompileException</td>
<td>Create a compile exception</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>When overridden in a derived class, returns the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. When overridden in a derived class, sets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.serialization.serializationinfo">SerializationInfo</a> with information about the exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.exception">Exception</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td></td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the Exception instance that caused the current exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)</td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception. (Inherited from Exception.)</td>
</tr>
</tbody>
</table>
See Also

CompileException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a compile exception

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public CompileException(
    string errorMessage
)

Visual Basic (Declaration)

Public Sub New ( _
    errorMessage As String _
)

Visual C++

public:
CompileException(
    String^ errorMessage
)

Parameters

errorMessage
    Type: System::String
    error message
See Also

CompileException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `CompileException` type exposes the following members.
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetBaseException</strong></td>
<td>Returns the <a href="#">Exception</a> that is the root cause of one or more subsequent exceptions. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetObjectData</strong></td>
<td>Sets the <a href="#">SerializationInfo</a> with information about the exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the runtime type of the current instance. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Creates and returns a string representation of the current exception. (Inherited from <a href="#">Exception</a>.)</td>
</tr>
</tbody>
</table>
See Also

CompileException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `CompileException` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Inherited From</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>Gets a collection of key/value pairs that provide additional user-defined information about the exception.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>HelpLink</strong></td>
<td>Gets or sets a link to the help file associated with this exception.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>HResult</strong></td>
<td>Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>InnerException</strong></td>
<td>Gets the <strong>Exception</strong> instance that caused the current exception.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>Gets a message that describes the current exception.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>Source</strong></td>
<td>Gets or sets the name of the application or the object that causes the error.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>StackTrace</strong></td>
<td>Gets a string representation of the frames on the call stack at the time the current exception was thrown.</td>
<td><strong>Exception</strong></td>
</tr>
<tr>
<td><strong>TargetSite</strong></td>
<td>Gets the method that throws the current exception.</td>
<td><strong>Exception</strong></td>
</tr>
</tbody>
</table>
See Also

CompileException Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Arguments being used in a template.

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class TemplateArguments

Visual Basic (Declaration)

Public Class TemplateArguments

Visual C++

public ref class TemplateArguments
Inheritance Hierarchy

- System..::.Object
  - HttpServer.Rendering..::.TemplateArguments
See Also

TemplateArguments Members
HttpServer.Rendering Namespace
HttpServer.Rendering::::TemplateManager
HttpServer.Rendering::::TinyTemplate

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `TemplateArguments` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateArguments</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Clear</td>
<td>Clears all added arguments. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Clears added arguments (Overrides <code>Object::FinalizeQQQ</code>.)</td>
</tr>
<tr>
<td>GetArguments</td>
<td>Retrieves the arguments (Does so now by copying the values to a new array, could be optimized?)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Update</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Function to make it possible to index out known arguments</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
HttpContext.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateArguments()()()</td>
<td>Initializes the class without any set arguments.</td>
</tr>
<tr>
<td>TemplateArguments(TemplateArguments)</td>
<td>Initializes the class with all the arguments of the parameter class.</td>
</tr>
<tr>
<td>TemplateArguments(array&lt;Object&gt;[])[]</td>
<td>Initializes the class with the objects in the parameterlist. Note that each object that is null must be followed by a type.</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
TemplateArguments Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes the class without any set arguments.

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public TemplateArguments()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
TemplateArguments()
See Also

TemplateArguments Class
TemplateArguments Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateArguments Constructor (TemplateArguments)

TemplateArguments Class  See Also  Send Feedback

Initializes the class with all the arguments of the parameter class.

**Namespace:**  [HttpServer.Rendering](http://example.com/HttpServer.Rendering)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public TemplateArguments(
    TemplateArguments arguments
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub New (_
    arguments As TemplateArguments _
)
```

### Visual C++

```cpp
public:
TemplateArguments(
    TemplateArguments^ arguments
)
```

## Parameters

- **arguments**
  - **Type:** `HttpServer.Rendering...TemplateArguments`
  - Cannot be null
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System::ArgumentNullException</code></td>
<td>If arguments is null</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
TemplateArguments Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateArguments Constructor (array<Object>[][])

Initializes the class with the objects in the parameterlist. Note that each object that is null must be followed by a type.

Namespace:  **HttpServer.Rendering**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public TemplateArguments(
    params Object[] arguments
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New ( _
    ParamArray arguments As Object() _
)
```

**Visual C++**

```cpp
public:
TemplateArguments(
    ... array<Object[]>^ arguments
)
```

### Parameters

arguments

Type: array< System::::Object >[]( )[]

Should contain ordered pairs/truples of string, object and type where type is optional if the object isn´t null
Examples

new TemplateArguments("Dir", ",", "Users", null, typeof(User));
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::ArgumentException</code></td>
<td>If optional type differs from type of object (if object != null) or argument name is duplicate</td>
</tr>
<tr>
<td><code>System:::ArgumentException</code></td>
<td>If the order of objects is incorrect</td>
</tr>
<tr>
<td><code>System:::ArgumentNullException</code></td>
<td>If any argument name or type is null</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
TemplateArguments Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TemplateArguments type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Clear</td>
<td>Clears all added arguments</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Clears added arguments</td>
</tr>
<tr>
<td>GetArguments</td>
<td>Retrieves the arguments (Does so now by copying the values to a new array, could be optimized?)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>Update</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
See Also

[TemplateArguments Class]
[HttpServer.Rendering Namespace]

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
TemplateArguments...:::Add Method
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add(String, Object)</td>
<td>Adds an argument. (Will throw ArgumentException on duplicates since adding an argument twice points to incorrect code, for ways around this Update(String, Object))</td>
</tr>
<tr>
<td>Add(String, Object, Type)</td>
<td>Adds an argument. Allows for argumentObject to be null (Will throw ArgumentException Add(String, Object))</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
TemplateArguments Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateArguments.Add Method (String, Object)

Adds an argument. (Will throw ArgumentException on duplicates since adding an argument twice points to incorrect code, for ways around this Update(String, Object))

Namespace: HttpServer.Rendering
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Add(
    string argumentName,
    Object argumentObject
)

Visual Basic (Declaration)

Public Sub Add (_
    argumentName As String, _
    argumentObject As Object _
)

Visual C++

public:
void Add(
    String^ argumentName,
    Object^ argumentObject
)

Parameters

argumentName
    Type: System::String
    Cannot be null

argumentObject
    Type: System::Object
    Cannot be null
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System..::..NullReferenceException</td>
<td>If argumentName or argumentObject is null</td>
</tr>
<tr>
<td>System..::..ArgumentException</td>
<td>If an argument named argumentName already exists</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
Add Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateArguments....Add Method (String, Object, Type)

TemplateArguments Class  See Also  Send Feedback

Adds an argument. Allows for argumentObject to be null (Will throw ArgumentException
Add(String, Object)

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Add(
    string argumentName,
    object argumentObject,
    Type argumentType
)

Visual Basic (Declaration)

Public Sub Add ( _
    argumentName As String, _
    argumentObject As Object, _
    argumentType As Type _
)

Visual C++

public:
void Add(
    String^ argumentName,
    Object^ argumentObject,
    Type^ argumentType
)

Parameters

argumentName
Type: System::String
Cannot be null

argumentObject
Type: System::Object


argumentType
Type: System:::Type
Cannot be null
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::NullReferenceException</td>
<td>If argumentName or argumentType is null</td>
</tr>
<tr>
<td>System:::ArgumentException</td>
<td>If an argument named argumentName already exists or argumentObject != null and typeof(argumentObject) differs from argumentType</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
Add Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Clears all added arguments

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Clear()

Visual Basic (Declaration)

Public Sub Clear

Visual C++

public:
void Clear()
See Also

TemplateArguments Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  C# Webserver

TemplateArguments...:.:Finalize Method

_TemplateArguments Class  See Also  Send Feedback

Cleans added arguments

Namespace:  **HttpServer.Rendering**  
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
protected override void Finalize()

Visual Basic (Declaration)
Protected Overrides Sub Finalize

Visual C++
protected:
virtual void Finalize() override
See Also

TemplateArguments Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Retrieve the arguments (Does so now by copying the values to a new array, could be optimized?)

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ArgumentContainer[] GetArguments()

Visual Basic (Declaration)

Public Function GetArguments As ArgumentContainer()

Visual C++

public:
array<ArgumentContainer>^ ^ GetArguments()

Return Value

An array containing arguments with name, object and type
See Also

TemplateArguments Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  Include Protected Members  Include Inherited Members

C# Webserver  TemplateArguments...:..Update Method

TemplateArguments Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update(TemplateArguments)</td>
<td>A function that merges two argument holders updating and adding values</td>
</tr>
<tr>
<td>Update(String, Object)</td>
<td>Updates an already added argument</td>
</tr>
<tr>
<td>Update(String, Object, Type)</td>
<td>Updates an already added argument, allows for argumentObject to be null</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
TemplateArguments Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A function that merges two argument holders updating and adding values

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public void Update(
    TemplateArguments arguments
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Update (_
    arguments As TemplateArguments _
)
```

**Visual C++**

```cpp
public:
void Update(
    TemplateArguments^ arguments
)
```

### Parameters

**arguments**

Type: [HttpServer.Rendering..:::TemplateArguments](#)

# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentNullException</code></td>
<td>If arguments is null</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
Update Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Updates an already added argument

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Update(
    string argumentName,
    Object argumentObject
)

Visual Basic (Declaration)

Public Sub Update (_
    argumentName As String, _
    argumentObject As Object _
)

Visual C++

public:
void Update(
    String^ argumentName,
    Object^ argumentObject
)

Parameters

argumentName
    Type: System::String
    Cannot be null

argumentObject
    Type: System::Object
    Cannot be null
**Exceptions**

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentException</code></td>
<td>If no argument named argumentName exists</td>
</tr>
<tr>
<td><code>System..::.ArgumentNullException</code></td>
<td>If argumentName or argumentObject is null</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
Update Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateArguments...::.Update Method (String, Object, Type)

Updates an already added argument, allows for argumentObject to be null

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public void Update(
    string argumentName,
    Object argumentObject,
    Type argumentType
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Update (_
    argumentName As String, _
    argumentObject As Object, _
    argumentType As Type _
)
```

**Visual C++**

```cpp
public:
void Update(
    String^ argumentName,
    Object^ argumentObject,
    Type^ argumentType
)
```

### Parameters

**argumentName**
- Type: `System::::String`
- Cannot be null

**argumentObject**
- Type: `System::::Object`

**argumentType**

Type: System:::Type
Cannot be null
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentException</code></td>
<td>If an argument named <code>argumentName</code> doesn’t exist or <code>argumentObject</code> != null and <code>typeof(argumentObject)</code> differs from <code>argumentType</code></td>
</tr>
<tr>
<td><code>System..::.ArgumentNullException</code></td>
<td>If <code>argumentName</code> or <code>argumentType</code> is null</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
Update Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TemplateArguments type exposes the following members.
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Function to make it possible to index out known arguments</td>
</tr>
</tbody>
</table>
See Also

TemplateArguments Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateArguments..::.Item Property

Namespace: **HttpServer.Rendering**  
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ArgumentContainer this[string name] { get; }

Visual Basic (Declaration)

Public ReadOnly Default Property Item ( _
    name As String _
) As ArgumentContainer

Visual C++

public:
    property ArgumentContainer^ default[String^ name] { 
    ArgumentContainer^ get (String^ name);
    }

Parameters

name
    Type: System:::String
    The name of an added argument

Return Value

Null if no ArgumentContainer by name was found
See Also

TemplateArguments Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The compiler is responsible of creating a render object which can be cached and used over and over again.

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class TemplateCompiler

Visual Basic (Declaration)

Public Class TemplateCompiler

Visual C++

public ref class TemplateCompiler
Inheritance Hierarchy

System..:::Object
HttpServer.Rendering..:::TemplateCompiler
See Also

TemplateCompiler Members
HttpServer.Rendering Namespace
HttpServer.Rendering:::TemplateManager
HttpServer.Rendering:::TemplateGenerator

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TemplateCompiler** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateCompiler</td>
<td>Create a new template compiler</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddType</td>
<td>This type should be included, so it may be called from the scripts (namespace and assembly).</td>
</tr>
<tr>
<td>Compile</td>
<td>Compiles the specified args.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetTypeName</td>
<td>Used to get correct names for generics.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚙ TemplateBase</td>
<td>Base C# code for a template object.</td>
</tr>
</tbody>
</table>
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new template compiler

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll)  
**Version:**  1.0.0.0 (1.0.0.0)
Syntax

C#

public TemplateCompiler()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
TemplateCompiler()
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `TemplateCompiler` type exposes the following members.
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateBase</td>
<td>Base C# code for a template object.</td>
</tr>
</tbody>
</table>
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Base C# code for a template object.

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public static string TemplateBase

Visual Basic (Declaration)

Public Shared TemplateBase As String

Visual C++

public:
static String^ TemplateBase
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TemplateCompiler type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddType</strong></td>
<td>This type should be included, so it may be called from the scripts (namespace and assembly).</td>
</tr>
<tr>
<td><strong>Compile</strong></td>
<td>Compiles the specified args.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>GetTypeName</strong></td>
<td>Used to get correct names for generics.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
</tbody>
</table>
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This type should be included, so it may be called from the scripts (namespace and assembly).

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public void AddType(
    Type type
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub AddType (_
    type As Type _
)
```

**Visual C++**

```cpp
public:
void AddType(
    Type^ type
)
```

**Parameters**

type

Type: `System::::Type`

[Missing `<param name="type"/>` documentation for "M:HttpServer.Rendering.TemplateCompiler.AddType(System.Type)"]
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Compiles the specified args.

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```
public TinyTemplate Compile(
    TemplateArguments args,
    string template,
    string templateId
)
```

**Visual Basic (Declaration)**

```
Public Function Compile (_
    args As TemplateArguments, _
    template As String, _
    templateId As String _
) As TinyTemplate
```

**Visual C++**

```
public:
    TinyTemplate^ Compile(
        TemplateArguments^ args,
        String^ template,
        String^ templateId
    )
```

### Parameters

**args**

Type: `HttpServer.Rendering:::TemplateArguments`
Arguments, should contain "name, value, name, value" etc.

**template**

Type: `System:::String`
C# code that will be included in the generated template class

**templateId**

Type: `System:::String`
Id of the template class

**Return Value**

Tiny template if successful; otherwise null.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpExceptionRendering.CompileException</code></td>
<td>If compilation fails</td>
</tr>
<tr>
<td><code>System.ArgumentException</code></td>
<td>If args are incorrect</td>
</tr>
</tbody>
</table>
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  C# Webserver
TemplateCompiler...:::GetTypeName Method
TemplateCompiler Class  See Also  Send Feedback

Used to get correct names for generics.

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public string GetTypeName(
    Type type
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function GetTypeName ( _
    type As Type _
) As String
```

**Visual C++**

```cpp
public:
    String^ GetTypeName(
        Type^ type
    )
```

### Parameters

type

Type: `System::::Type`

[Missing `<param name="type"/>` documentation for
"M:HttpServer.Rendering.TemplateCompiler.GetTypeName(System.Type)""]

### Return Value

[Missing `<returns>` documentation for
M:HttpServer.Rendering.TemplateCompiler.GetTypeName(System.Type)]
See Also

TemplateCompiler Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A code generator is used to convert template code to something that we can work with, as html or c# code.

**Namespace:**  [HttpServer.Rendering](http://example.com/HttpServer.Rendering)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public interface TemplateGenerator

Visual Basic (Declaration)

Public Interface TemplateGenerator

Visual C++

public interface class TemplateGenerator
See Also

TemplateGenerator Members
HttpServer.Rendering Namespace
HttpServer.Rendering:::TemplateManager

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `TemplateGenerator` type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GenerateCode</td>
<td>Generate C# code from the template.</td>
</tr>
<tr>
<td>Parse</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
See Also

TemplateGenerator Interface
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TemplateGenerator** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GenerateCode</td>
<td>Generate C# code from the template.</td>
</tr>
<tr>
<td>Parse</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
See Also

TemplateGenerator Interface
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate C# code from the template.

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
void GenerateCode(
    TextWriter writer
)
```

**Visual Basic (Declaration)**

```vbnet
Sub GenerateCode (_
    writer As TextWriter _
)
```

**Visual C++**

```c++
void GenerateCode(
    TextWriter^ writer
)
```

**Parameters**

`writer`

Type: `System.IO::TextWriter`

A textwriter that the generated code will be written to.
Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::InvalidOperationException</td>
<td>If the template have not been parsed first.</td>
</tr>
<tr>
<td>HttpServer.Rendering:::CodeGenException</td>
<td>If template is incorrect</td>
</tr>
</tbody>
</table>
See Also

TemplateGenerator Interface
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateGenerator::.:Parse Method

See Also

Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parse(TextReader)</td>
<td>Parse a file and convert into our own template object code.</td>
</tr>
<tr>
<td>Parse(String)</td>
<td>Parse a file and convert into our own template object code.</td>
</tr>
</tbody>
</table>
See Also

TemplateGenerator Interface
TemplateGenerator Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a file and convert into our own template object code.

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

void Parse(
    TextReader reader
)

Visual Basic (Declaration)

Sub Parse ( _
    reader As TextReader _
)

Visual C++

void Parse(
    TextReader^ reader
)

Parameters

reader

Type: System.IO:::TextReader
A textreader containing our template
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HttpException.Rendering::CodeGenException</strong></td>
<td>If something is incorrect in the template.</td>
</tr>
</tbody>
</table>
See Also

TemplateGenerator Interface
Parse Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a file and convert into our own template object code.

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
void Parse(
            string fullPath
)
```

**Visual Basic (Declaration)**

```vbnet
Sub Parse (_
            fullPath As String _
)
```

**Visual C++**

```cpp
void Parse(
            String^ fullPath
)
```

**Parameters**

`fullPath`

Type: `System::String`

Path and filename to a template
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpServer.Rendering...:::CodeGeneratorException</td>
<td>If something is incorrect in the template.</td>
</tr>
<tr>
<td>System.IO...:::FileNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System.IO...:::DirectoryNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System...:::UnauthorizedAccessException</td>
<td></td>
</tr>
<tr>
<td>System.IO...:::PathTooLongException</td>
<td></td>
</tr>
<tr>
<td>System...:::NotSupportedException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

TemplateGenerator Interface
Parse Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Purpose if this class is to take template objects and keep them in memory. It will also take a filename and the code generator to use if when the template have been changed on disk.

Namespace:  [HttpServerRendering](HttpServerRendering)
Assembly:  [HttpServer](HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class TemplateManager

Visual Basic (Declaration)

Public Class TemplateManager

Visual C++

public ref class TemplateManager
Inheritance Hierarchy

System..:::Object
HttpServer.Rendering..:::TemplateManager
See Also

TemplateManager Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TemplateManager** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TemplateManager</td>
<td>Initializes a new instance of the TemplateManager class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Add a template generator</td>
</tr>
<tr>
<td><strong>AddType</strong></td>
<td>This type should be included, so it may be called from the scripts (namespace and assembly).</td>
</tr>
<tr>
<td><strong>Compile</strong></td>
<td>Compiles the specified code.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GenerateCode</strong></td>
<td>Will generate code from the template. Next step is to compile the code.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Render</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>RenderPartial</strong></td>
<td>Render a partial</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PathPrefix</td>
<td>A prefix for all files rendered.</td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateManager Constructor

Initializes a new instance of the TemplateManager class

Namespace: HttpServer.Rendering
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public TemplateManager()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
TemplateManager()
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TemplateManager** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add</strong></td>
<td>Add a template generator</td>
</tr>
<tr>
<td><strong>AddType</strong></td>
<td>This type should be included, so it may be called from the scripts (namespace and assembly).</td>
</tr>
<tr>
<td><strong>Compile</strong></td>
<td>Compiles the specified code.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GenerateCode</strong></td>
<td>Will generate code from the template. Next step is to compile the code.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td><strong>Render</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>RenderPartial</strong></td>
<td>Render a partial</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add a template generator

**Namespace:**  [HttpServer.Rendering](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void Add(
    string fileExtension,
    TemplateGenerator generator
)
```

Visual Basic (Declaration)

```vbnet
Public Sub Add ( _
    fileExtension As String, _
    generator As TemplateGenerator _
)
```

Visual C++

```cpp
public:
void Add(
    String^ fileExtension,
    TemplateGenerator^ generator
)
```

Parameters

fileExtension
- Type: `System::String`
- File extension without the dot.

generator
- Type: `HttpServer.Rendering::TemplateGenerator`
- Generator to handle the extension
Examples

cache.Add("haml", new HamlGenerator());
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::InvalidOperationException</code></td>
<td>If the generator already exists.</td>
</tr>
<tr>
<td><code>System:::ArgumentException</code></td>
<td>If file extension is incorrect</td>
</tr>
<tr>
<td><code>System:::ArgumentNullException</code></td>
<td>If generator is not specified.</td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This type should be included, so it may be called from the scripts (namespace and assembly).

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void AddType(
    Type type
)
```

Visual Basic (Declaration)

```vbnet
Public Sub AddType (_
    type As Type _
)
```

Visual C++

```cpp
public:
void AddType(
    Type^ type
)
```

Parameters

type
Type: System::::Type

[Missing <param name="type"/> documentation for "M:HttpServer.Rendering.TemplateManager.AddType(System.Type)" ]
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Compiles the specified code.

**Namespace:** [HttpServer.Rendering](http://example.com/HttpServer.Rendering)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
protected TinyTemplate Compile(
    string code,
    TemplateArguments arguments,
    string templateId
)
```

### Visual Basic (Declaration)

```vbnet
Protected Function Compile (_
    code As String,
    arguments As TemplateArguments,
    templateId As String_)
) As TinyTemplate
```

### Visual C++

```cpp
protected:
    TinyTemplate^ Compile( 
        String^ code,  
        TemplateArguments^ arguments,  
        String^ templateId 
    )
```

## Parameters

**code**

Type: `System:::String`  
c# code generated from a template.

**arguments**

Type: `HttpRequestRendering:::TemplateArguments`  
Arguments as in name, value, name, value, name, value

**templateId**

Type: `System:::String`
Required by TemplateCompiler.Compile to identify the compiled template

**Return Value**

Template
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpExceptionRendering...::CompileException</code></td>
<td>If compilation fails</td>
</tr>
</tbody>
</table>
See Also

TemplateName Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Will generate code from the template. Next step is to compile the code.

**Namespace:**  [HttpServer.Rendering](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected string GenerateCode(string fullPath)

Visual Basic (Declaration)

Protected Function GenerateCode (_
    fullPath As String _
) As String

Visual C++

protected:
String^ GenerateCode(
    String^ fullPath
)

Parameters

fullPath
    Type: System::String
    Path and filename to template.

Return Value

[Missing <returns> documentation for
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SystemargesException</td>
<td>If no template generator exists for the specified extension.</td>
</tr>
<tr>
<td>System.IO.FileNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System.InvalidOperationException</td>
<td></td>
</tr>
<tr>
<td>HttpServer.Rendering.CodeGeneratorException</td>
<td>If parsing/compiling fails</td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TemplateManager::Render Method

TemplateManager Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Render(String, TemplateArguments)</code></td>
<td>Generate HTML from a template.</td>
</tr>
<tr>
<td><code>Render(String, TemplateArguments, String)</code></td>
<td>Generate HTML from a template.</td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
TemplateManager Members
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate HTML from a template.

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public string Render(
    string filename,
    TemplateArguments args
)
```

**Visual Basic (Declaration)**

```vbnet
Public Function Render ( _
    filename As String, _
    args As TemplateArguments _
) As String
```

**Visual C++**

```cpp
public:
    String^ Render(
        String^ filename,
        TemplateArguments^ args
    )
```

### Parameters

**filename**
- Type: `System::String`
  - Path and filename

**args**
- Type: `HttpServer.Rendering::TemplateArguments`
  - Variables used in the template. Should be specified as "name, value, name, value" where name is variable name and value is variable contents.

### Return Value

string html = cache.Generate("views\users\view.haml", new
    TemplateArguments("user", dbUser, "isAdmin", dbUser.IsAdmin));
# Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.IO...:FileNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System...:InvalidOperationException</td>
<td></td>
</tr>
<tr>
<td>HttpServer.Rendering...:CompileException</td>
<td></td>
</tr>
<tr>
<td>System...:ArgumentException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
Render Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate HTML from a template.

Namespace:  HttpServer.Rendering
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected string Render(
    string filename,
    TemplateArguments args,
    string partialIdentifier
)

Visual Basic (Declaration)

Protected Function Render ( _
    filename As String, _
    args As TemplateArguments, _
    partialIdentifier As String _
) As String

Visual C++

protected:
String^ Render(
    String^ filename,
    TemplateArguments^ args,
    String^ partialIdentifier
)

Parameters

filename
    Type: System::::String
    Path and filename

args
    Type: HttpServer.Rendering::::TemplateArguments
    Variables used in the template. Should be specified as "name, value, name, value" where name is variable name and value is variable contents.

partialIdentifier
Type: System.String
An identifier to use for identifying a partial render, this parameter can be null

Return Value

Examples

string html = cache.Generate("views\users\view.haml", new TemplateArguments("user", dbUser, "isAdmin", dbUser.IsAdmin), null);
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.IO:::FileNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System:::InvalidOperationException</td>
<td></td>
</tr>
<tr>
<td>HttpServer.Rendering:::CompileException</td>
<td></td>
</tr>
<tr>
<td>System:::ArgumentException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
Render Overload
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Render a partial

**Namespace:**  [HttpServer.Rendering](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string RenderPartial(
    string filename,
    TemplateArguments args,
    string partialIdentifier
)

Visual Basic (Declaration)

Public Function RenderPartial (_
    filename As String, _
    args As TemplateArguments, _
    partialIdentifier As String _
) As String

Visual C++

public: 
String^ RenderPartial( 
    String^ filename, 
    TemplateArguments^ args, 
    String^ partialIdentifier 
)

Parameters

filename
    Type: System::String
    Path and filename

args
    Type: HttpServer.Rendering:::TemplateArguments
    Variables used in the template. Should be specified as "name, value, name, value" where name is variable name and value is variable contents.

partialIdentifier
Type: System::String
The name of the partial to render, if not found/not generated it will be compiled anew

Return Value

## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.IO::&lt;FileNotFoundException&gt;</td>
<td></td>
</tr>
<tr>
<td>System::&lt;InvalidOperationException&gt;</td>
<td></td>
</tr>
<tr>
<td>HttpServer.Rendering::&lt;CompileException&gt;</td>
<td></td>
</tr>
<tr>
<td>System::&lt;ArgumentException&gt;</td>
<td></td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `TemplateManager` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PathPrefix</td>
<td>A prefix for all files rendered.</td>
</tr>
</tbody>
</table>
See Also

TemplateManager Class
HttpServletRequest.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A prefix for all files rendered.

Namespace: **HttpServer.Rendering**
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string PathPrefix { get; set; }

Visual Basic (Declaration)

Public Property PathPrefix As String

Visual C++

public:
property String^ PathPrefix {
    String^ get ();
    void set (String^ value);
}
Remarks

PathPrefix may not be null, only string.EmptyLanguageNode.
Examples

// will look after template in Environment.CurrentDirectory + "views\" <ControllerName>\templateName.*" mgr.PathPrefix = "views\";
See Also

TemplateManager Class
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Interface for dynamically generated templates.

**Namespace:** [HttpServer.Rendering](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public interface TinyTemplate

Visual Basic (Declaration)

Public Interface TinyTemplate

Visual C++

public interface class TinyTemplate
See Also

TinyTemplate Members
HttpServer.Rendering Namespace
HttpServer.Rendering:::TemplateManager

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TinyTemplate** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invoke</td>
<td>Run the template to generate HTML code.</td>
</tr>
</tbody>
</table>
See Also

TinyTemplate Interface
HttpServer.Rendering Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TinyTemplate** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invoke</td>
<td>Run the template to generate HTML code.</td>
</tr>
</tbody>
</table>
See Also

TinyTemplate Interface
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Run the template to generate HTML code.

**Namespace:** [HttpServer.Rendering](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
string Invoke(
    TemplateArguments args,
    TemplateManager hiddenTemplateManager
)
```

Visual Basic (Declaration)

Function Invoke (_
    args As TemplateArguments, _
    hiddenTemplateManager As TemplateManager _
) As String

Visual C++

```cpp
String^ Invoke(
    TemplateArguments^ args,
    TemplateManager^ hiddenTemplateManager
)
```

Parameters

args
    Type: `HttpServer.Rendering::::TemplateArguments`
    arguments passed to the template

hiddenTemplateManager
    Type: `HttpServer.Rendering::::TemplateManager`
    template manager (a manager is used to generate templates)

Return Value

HTML code.
See Also

TinyTemplate Interface
HttpServer.Rendering Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
C# Webserver
HttpServer.Rendering.Haml Namespace
Send Feedback

## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌱 LineInfo</td>
<td>Contains line text and state information about a line in a HAML template.</td>
</tr>
<tr>
<td>🌱 NodeList</td>
<td>A list of prototype nodes.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HamlGenerator Class

Generates C#/HTML from haml code. Haml doc:
http://haml.hamptoncatlin.com/docs/rdoc/classes/Haml.html

Namespace: HttpServer.Rendering.Haml
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class HamlGenerator : TemplateGenerator

Visual Basic (Declaration)

Public Class HamlGenerator
    Implements TemplateGenerator

Visual C++

public ref class HamlGenerator : TemplateGenerator
Inheritance Hierarchy

System..:::Object
HttpServer.Rendering.Haml..:::HamlGenerator
See Also

HamlGenerator Members
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `HamlGenerator` type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HamlGenerator</td>
<td>create a new haml generator.</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckIntenation</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>CheckMultiLine</td>
<td>check if current line is a multiline</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GenerateCode</td>
<td>Generate C# code from the template.</td>
</tr>
<tr>
<td>GenerateHtml</td>
<td>Generate html code from the template. Code is encased in &lt;% and &lt;%=</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetWord</td>
<td>Get the first word (letters and digits only) from the specified offset.</td>
</tr>
<tr>
<td>HandlePlacement</td>
<td>Check indentation / node placement</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>Parse</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>ParseNode</td>
<td>Parse a node //todo: improve doc</td>
</tr>
<tr>
<td>PreParse</td>
<td>PreParse goes through the text add handles intedation and all multiline</td>
</tr>
<tr>
<td></td>
<td>cases.</td>
</tr>
<tr>
<td>PrintDocument</td>
<td>print the entire document</td>
</tr>
<tr>
<td>PrintNode</td>
<td>Print line information to the console</td>
</tr>
<tr>
<td>ReadLine</td>
<td>Read next line from file</td>
</tr>
<tr>
<td>Spaces</td>
<td>Generates a string with spaces.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object.</td>
</tr>
</tbody>
</table>
(Inherited from **Object**.)
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RootNode</td>
<td>Property to retrieve the root node for the latest parsed document</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
create a new haml generator.

**Namespace:**  [HttpServer.Rendering.Haml](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public HamlGenerator()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
HamlGenerator()
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HamlGenerator** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CheckIntendation</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>CheckMultiLine</strong></td>
<td>check if current line is a multiline</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GenerateCode</strong></td>
<td>Generate C# code from the template.</td>
</tr>
<tr>
<td><strong>GenerateHtml</strong></td>
<td>Generate html code from the template. Code is encased in <code>&lt;%</code> and <code>&lt;%=</code></td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetWord</strong></td>
<td>Get the first word (letters and digits only) from the specified offset.</td>
</tr>
<tr>
<td><strong>HandlePlacement</strong></td>
<td>Check indentation / node placement</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ParseNode</strong></td>
<td>Parse a node //todo: improve doc</td>
</tr>
<tr>
<td><strong>PreParse</strong></td>
<td>PreParse goes through the text add handles intendment and all multiline cases.</td>
</tr>
<tr>
<td><strong>PrintDocument</strong></td>
<td>print the entire document</td>
</tr>
<tr>
<td><strong>PrintNode</strong></td>
<td>Print line information to the console</td>
</tr>
<tr>
<td><strong>ReadLine</strong></td>
<td>Read next line from file</td>
</tr>
<tr>
<td><strong>Spaces</strong></td>
<td>Generates a string with spaces.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>.</td>
</tr>
</tbody>
</table>
(Inherited from Object.)
See Also

Ham1Generator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#
Visual Basic
Visual C++
Include Protected Members
Include Inherited Members

C# Webserver
HamlGenerator..:::.CheckIntendation Method

HamlGenerator Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckIntendation(LineInfo)</td>
<td>Check indentation</td>
</tr>
<tr>
<td>CheckIntendation(LineInfo, Int32, Int32)</td>
<td>Check and validate indentation</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
HamlGenerator Members
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check indentation

**Namespace:** [HttpServer.Rendering.Haml](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void CheckIntendation(
    LineInfo line
)

Visual Basic (Declaration)

Protected Sub CheckIntendation ( _
    line As LineInfo _
)

Visual C++

protected:
void CheckIntendation(
    LineInfo^ line
)

Parameters

line
    Type: HttpServer.Rendering.Haml:::LineInfo
    fills line with intend info
See Also

HamlGenerator Class
CheckIntendation Overload
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic  Visual C++  C# Webserver

HamlGenerator..:::CheckIntendation Method (LineInfo, Int32%, Int32%)

HamlGenerator Class  See Also  Send Feedback

Check and validate indendation

Namespace:  HttpServer.Rendering.Haml
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void CheckIntendation(
    LineInfo line,
    out int ws,
    out int intendation
)

Visual Basic (Declaration)

Protected Sub CheckIntendation ( _
    line As LineInfo, _
    <OutAttribute> ByRef ws As Integer, _
    <OutAttribute> ByRef intendation As Integer _
)

Visual C++

protected:
void CheckIntendation(
    LineInfo^ line,
    [OutAttribute] int% ws,
    [OutAttribute] int% intendation
)

Parameters

line
    Type: HttpServer.Rendering.Haml:::LineInfo
    line to check

ws
    Type: System:::Int32 %
    number of whitespaces

intendation
    Type: System:::Int32 %
number of indendations (2 ws = 1 intend, 1 tab = 1 intend)
See Also

HamlGenerator Class
CheckIntendation Overload
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
check if current line is a multiline

**Namespace:** [HttpServer.Rendering.Haml](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void CheckMultiLine(
    LineInfo prevLine,
    LineInfo line
)

Visual Basic (Declaration)

Protected Sub CheckMultiLine (_
    prevLine As LineInfo, _
    line As LineInfo _
)

Visual C++

protected:
void CheckMultiLine(
    LineInfo^ prevLine,
    LineInfo^ line
)

Parameters

prevLine
    Type: HttpServer.Rendering.Haml..:::LineInfo
    previous line

line
    Type: HttpServer.Rendering.Haml..:::LineInfo
    current line
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate C# code from the template.

**Namespace:**  [HttpServer.Rendering.Haml](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public void GenerateCode(
    TextWriter writer
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub GenerateCode ( _
    writer As TextWriter _
)
```

**Visual C++**

```cpp
public:
virtual void GenerateCode(
    TextWriter^ writer
) sealed
```

### Parameters

**writer**

Type: `System.IO::TextWriter`  
A textwriter that the generated code will be written to.

### Implements

`TemplateGenerator::GenerateCode(TextWriter)`
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::InvalidOperationException</code></td>
<td>If the template have not been parsed first.</td>
</tr>
<tr>
<td><code>HttpServer.Rendering:::CodeGenException</code></td>
<td>If template is incorrect</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate html code from the template. Code is encapsed in <% and <%=.

**Namespace:**  [HttpServer.Rendering.Haml](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void GenerateHtml(
    TextWriter writer
)

Visual Basic (Declaration)

Public Sub GenerateHtml ( _
    writer As TextWriter _
)

Visual C++

public:
void GenerateHtml(
    TextWriter^ writer
)

Parameters

writer
    Type: System.IO.TextWriter
    A textwriter that the generated code will be written to.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System:::InvalidOperationException</code></td>
<td>If the template have not been parsed first.</td>
</tr>
<tr>
<td><code>HttpServer.Rendering:::CodeGeneratorException</code></td>
<td>If template is incorrect</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get the first word (letters and digits only) from the specified offset.

**Namespace:** [HttpServer.Rendering.Haml](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public static string GetWord(
    string data,
    int offset
)
```

**Visual Basic (Declaration)**

```vbnet
Public Shared Function GetWord ( _
    data As String, _
    offset As Integer _
) As String
```

**Visual C++**

```csharp
public:
    static String^ GetWord(
        String^ data,
        int offset
    )
```

**Parameters**

data

Type: `System::String`


offset

Type: `System::Int32`


**Return Value**
[Missing <returns> documentation for
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Check indentation / node placement

**Namespace:** [HttpServer.Rendering.Haml](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void HandlePlacement()

Visual Basic (Declaration)

Protected Sub HandlePlacement

Visual C++

protected:
void HandlePlacement()
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Parse(TextReader)</code></td>
<td>Parse a file and convert into our own template object code.</td>
</tr>
<tr>
<td><code>Parse(String)</code></td>
<td>Parse a file and convert into our own template object code.</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
HamlGenerator Members
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a file and convert into our own template object code.

**Namespace:**  [HttpServer.Rendering.Haml](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Parse(
    TextReader reader
)

Visual Basic (Declaration)

Public Sub Parse ( _
    reader As TextReader _
)

Visual C++

public:
virtual void Parse(
    TextReader^ reader
) sealed

Parameters

reader

Type: System.IO.TextReader
A textreader containing our template

Implements

TemplateGenerator..::.Parse(TextReader)
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering.:::CodeGenException</code></td>
<td>If something is incorrect in the template.</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
Parse Overload
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a file and convert into our own template object code.

Namespace:  HttpServer.Rendering.Haml
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Parse(
    string fullPath
)

Visual Basic (Declaration)

Public Sub Parse ( _
    fullPath As String _
)

Visual C++

public:
virtual void Parse(
    String^ fullPath
) sealed

Parameters

fullPath
Type: System::__::String
Path and filename to a template

Implements

TemplateGenerator:::Parse(String)
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering:::CodeGenException</code></td>
<td>If something is incorrect in the template.</td>
</tr>
<tr>
<td><code>System.IO:::FileNotFoundException</code></td>
<td></td>
</tr>
<tr>
<td><code>System.IO:::DirectoryNotFoundException</code></td>
<td></td>
</tr>
<tr>
<td><code>System:::UnauthorizedAccessException</code></td>
<td></td>
</tr>
<tr>
<td><code>System.IO:::PathTooLongException</code></td>
<td></td>
</tr>
<tr>
<td><code>System:::NotSupportedException</code></td>
<td></td>
</tr>
<tr>
<td><code>System:::ArgumentException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
Parse Overload
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a node //todo: improve doc

**Namespace:** [HttpServer.Rendering.Haml](HttpServer.Rendering.Haml)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
protected void ParseNode(
    LineInfo theLine,
    NodeList prototypes,
    Node parent,
   TextNode textNode
)
```

**Visual Basic (Declaration)**

```vbnet
Protected Sub ParseNode (_
    theLine As LineInfo, _
    prototypes As NodeList, _
    parent As Node, _
    textNode AsTextNode _
)
```

**Visual C++**

```cpp
protected:
void ParseNode(
    LineInfo^ theLine,
    NodeList^ prototypes,
    Node^ parent,
    TextNode^ textNode
)
```

### Parameters

- **theLine**
  - Type: `HttpServer.Rendering.Haml::LineInfo`


- **prototypes**
  - Type: `HttpServer.Rendering.Haml::NodeList`
parent
Type: HttpServer.Rendering.Haml.Nodes::Node

textNode
Type: HttpServer.Rendering.Haml.Nodes::TextNode
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
PreParse goes through the text and handles indentation and all multiline cases.

**Namespace:** [HttpServer.Rendering.Haml](https://example.com/namespace)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void PreParse(
    TextReader reader
)

Visual Basic (Declaration)

Protected Sub PreParse ( _
    reader As TextReader _
)

Visual C++

protected:
void PreParse(
    TextReader^ reader
)

Parameters

reader
    Type: System.IO.TextReader
    Reader containing the text
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
print the entire document

**Namespace:**  [HttpServer.Rendering.Haml](HttpServer.Rendering.Haml)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void PrintDocument()

Visual Basic (Declaration)

Public Sub PrintDocument

Visual C++

public:
void PrintDocument()
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Print line information to the console

**Namespace:**  [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void PrintNode(
    LineInfo line
)
```

Visual Basic (Declaration)

```vbnet
Public Sub PrintNode (_
    line As LineInfo _
)
```

Visual C++

```cpp
public:
    void PrintNode(
        LineInfo^ line
    )
```

Parameters

line

Type: `HttpServer.Rendering.Haml::LineInfo`

See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HamlGenerator...:::ReadLine Method

HamlGenerator Class  See Also  Send Feedback

Read next line from file

Namespace:  HttpServer.Rendering.Haml
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected bool ReadLine()

Visual Basic (Declaration)

Protected Function ReadLine As Boolean

Visual C++

protected:
bool ReadLine()

Return Value

true if line could be read; false if EOF.
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HamlGenerator..:::Spaces Method

Generates a string with spaces.

**Namespace:** [HttpServer.Rendering.Haml]

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Spaces(int count)

Visual Basic (Declaration)

Public Function Spaces ( count As Integer ) As String

Visual C++

public: String^ Spaces(int count)

Parameters

count

Type: System::Int32
number of spaces.

Return Value

string of spaces.
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **HamlGenerator** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RootNode</td>
<td>Property to retrieve the root node for the latest parsed document</td>
</tr>
</tbody>
</table>
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Property to retrieve the root node for the latest parsed document

**Namespace:** [HttpServer.Rendering.Haml](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Node RootNode { get; }

Visual Basic (Declaration)

Public ReadOnly Property RootNode As Node

Visual C++

public:
property Node^ RootNode {
            Node^ get ();
}
See Also

HamlGenerator Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains line text and state information about a line in a HAML template.

**Namespace:**  [HttpServer.Rendering.Haml](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class LineInfo

Visual Basic (Declaration)

Public Class LineInfo

Visual C++

public ref class LineInfo
Inheritance Hierarchy

System..:::Object
HttpServer.Rendering.Haml..:::LineInfo
See Also

LineInfo Members
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `LineInfo` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LineInfo</td>
<td>Initializes a new instance of the LineInfo class.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Append</td>
<td>Append another line</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Set</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>SetParsedData</td>
<td>Parsed line contents (without whitespaces in the beginning)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>TrimRight</td>
<td></td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AppendNextLine</td>
<td>If the next line should be appended to us (multi line)</td>
</tr>
<tr>
<td>Children</td>
<td>Do NOT add yourself using the Add methods of the linkedList. Parent property will add node.</td>
</tr>
<tr>
<td>Data</td>
<td>All line data generated on one line</td>
</tr>
<tr>
<td>HasChildren</td>
<td>Has one or more children (intended more that this one)</td>
</tr>
<tr>
<td>Intendation</td>
<td>Number of intends (two spaces = 1, one tab = 1)</td>
</tr>
<tr>
<td>LineNumber</td>
<td>Line number</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node (one level up in intendation)</td>
</tr>
<tr>
<td>SelfClosed</td>
<td>True if node is selfclosed (i.e. <code>&lt;br /&gt;</code>)</td>
</tr>
<tr>
<td>UnfinishedRule</td>
<td>Rule have not got all needed information yet, keep appending lines to this LineInfo until rule says that it's done.</td>
</tr>
<tr>
<td>UnparsedData</td>
<td>Untouched line text</td>
</tr>
<tr>
<td>WhiteSpaces</td>
<td>Number of whitespaces before actual entry beings.</td>
</tr>
</tbody>
</table>
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `LineInfo` class.

**Namespace:**  [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)  
**Assembly:**  [HttpServer](http://HttpServer) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public LineInfo(
    int lineNumber,
    string unparsedData
)

Visual Basic (Declaration)

Public Sub New ( _
    ByVal lineNumber As Integer, _
    ByVal unparsedData As String _
)

Visual C++

public:
LineInfo(
    int lineNumber,
    String^ unparsedData
)

Parameters

lineNumber
  Type: `System::::Int32`
  The line number.

unparsedData
  Type: `System::::String`
  The unparsed data (line contents).
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `LineInfo` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Append</td>
<td>Append another line</td>
</tr>
<tr>
<td></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>Equals</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Set</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>SetParsedData</td>
<td>Parsed line contents (without whitespaces in the beginning)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>TrimRight</td>
<td></td>
</tr>
</tbody>
</table>
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Append another line

**Namespace:**  [HttpServer.Rendering.Haml](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Append(
    [LineInfo] line
)

Visual Basic (Declaration)

Public Sub Append (_
    line As [LineInfo] _
)

Visual C++

public:
    void Append(
        [LineInfo^] line
    )

Parameters

line
  Type: [HttpGetServer.Rendering.Haml::::LineInfo]

See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
LineInfo:::..Set Method

LineInfo Class  See Also  Send Feedback
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set(Int32, Int32)</td>
<td>Set intendment info to previously added line text.</td>
</tr>
<tr>
<td>Set(String, Int32, Int32)</td>
<td>Assign line text</td>
</tr>
</tbody>
</table>
See Also

- LineInfo Class
- LineInfo Members
- HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Set intedation info to previously added line text.

**Namespace:**  [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Set(
    int whiteSpaces,
    int intendation
)

Visual Basic (Declaration)

Public Sub Set ( _
    whiteSpaces As Integer, _
    intendation As Integer _
)

Visual C++

public:
void Set(
    int whiteSpaces,
    int intendation
)

Parameters

whiteSpaces
Type: System::::Int32


intendation
Type: System::::Int32

See Also

LineInfo Class
Set Overload
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
☐ Visual C++
C# Webserver
LineInfo...::..Set Method (String, Int32, Int32)

Namespace:  HttpServer.Rendering.Haml
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public void Set(
    string line,
    int whiteSpaces,
    int intendation
)
```

Visual Basic (Declaration)

```vbnet
Public Sub Set ( _
    line As String, _
    whiteSpaces As Integer, _
    intendation As Integer _
)
```

Visual C++

```cpp
public:
void Set(  
    String^ line,
    int whiteSpaces,
    int intendation
)
```

Parameters

- **line**
  Type: `System::String`


- **whiteSpaces**
  Type: `System::Int32`

intendation

Type: System::Int32

See Also

LineInfo Class
Set Overload
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parsed line contents (without whitespaces in the beginning)

**Namespace:** [HttpServer.Rendering.Haml](http://example.com)
**Assembly:** [HttpServer](http://example.com) (in `HttpServer.dll`) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void SetParsedData(
    string data
)

Visual Basic (Declaration)

Protected Sub SetParsedData ( _
    data As String _
)

Visual C++

protected:
void SetParsedData(
    String^ data
)

Parameters

data
    Type: System::String
text contents
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
LineInfo.TrimRight Method

Namespace: HttpServer.Rendering.Haml
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void TrimRight(
    int count
)

Visual Basic (Declaration)

Public Sub TrimRight ( _
    count As Integer _
)

Visual C++

public:
void TrimRight(
    int count
)

Parameters

count

Type: System::Int32

## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System...::InvalidOperationException</td>
<td>If extra lines have been added.</td>
</tr>
</tbody>
</table>
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `LineInfo` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AppendNextLine</td>
<td>If the next line should be appended to us (multi line)</td>
</tr>
<tr>
<td>Children</td>
<td>Do NOT add yourself using the Add methods of the linkedList. Parent property will add node.</td>
</tr>
<tr>
<td>Data</td>
<td>All line data generated on one line</td>
</tr>
<tr>
<td>HasChildren</td>
<td>Has one or more children (intended more that this one)</td>
</tr>
<tr>
<td>Intendation</td>
<td>Number of intends (two spaces = 1, one tab = 1)</td>
</tr>
<tr>
<td>LineNumber</td>
<td>Line number</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node (one level up in intendment)</td>
</tr>
<tr>
<td>SelfClosed</td>
<td>True if node is selfclosed (i.e. &lt;br /&gt;)</td>
</tr>
<tr>
<td>UnfinishedRule</td>
<td>Rule have not got all needed information yet, keep appending lines to this LineInfo until rule says that it's done.</td>
</tr>
<tr>
<td>UnparsedData</td>
<td>Untouched line text</td>
</tr>
<tr>
<td>WhiteSpaces</td>
<td>Number of whitespaces before actual entry beings.</td>
</tr>
</tbody>
</table>
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
If the next line should be appended to us (multi line)

**Namespace:** [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public bool AppendNextLine { get; set; }
```

Visual Basic (Declaration)

```vbnet
Public Property AppendNextLine As Boolean
```

Visual C++

```cpp
public:
property bool AppendNextLine {
    bool get ();
    void set (bool value);
}
```
See Also

[LineInfo Class](#)
[HttpServer.Rendering.Haml Namespace](#)

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
Do NOT add yourself using the Add methods of the linkedList. Parent property will add node.

**Namespace:** [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
C#

public LinkedList<LineInfo> Children { get; }

Visual Basic (Declaration)

Public ReadOnly Property Children As LinkedList(Of LineInfo)

Visual C++

public:
property LinkedList<LineInfo>^ Children {
    LinkedList<LineInfo>^ get ();
}
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
All line data generated on one line

**Namespace:**  [HttpServer.Rendering.Haml](HttpServer.Rendering.Haml)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Data { get; }

Visual Basic (Declaration)

Public ReadOnly Property Data As String

Visual C++

public:
property String^ Data {
   String^ get ();
}

See Also

- LineInfo Class
- HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Has one or more children (intended more that this one)

**Namespace:** [HttpServer.Rendering.Haml](https://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool HasChildren { get; }  

Visual Basic (Declaration)

Public ReadOnly Property HasChildren As Boolean  

Visual C++

public:
    property bool HasChildren { 
        bool get ();
    }
See Also

LineColor Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
LineInfo.........Intendation Property

Namespace: HttpServer.Rendering.Haml
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int Intendation { get; }

Visual Basic (Declaration)

Public ReadOnly Property Intention As Integer

Visual C++

public:
property int Intendation {
    int get ();
}


See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
LineInfo::LineNumber Property

Line number

**Namespace:** [HttpServer.Rendering.Haml](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int LineNumber { get; }

Visual Basic (Declaration)

Public ReadOnly Property LineNumber As Integer

Visual C++

public:
property int LineNumber {
    int get ();
}

See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
LineInfo..::.Parent Property

Namespace:  HttpServer.Rendering.Haml
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public LineInfo Parent { get; set; }

Visual Basic (Declaration)

Public Property Parent As LineInfo

Visual C++

public:
    property LineInfo^ Parent {
        LineInfo^ get ();
        void set (LineInfo^ value);
    }
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
True if node is selfclosed (i.e. `<br/>`)

**Namespace:** [HttpServer.Rendering.Haml](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool SelfClosed { get; set; }

Visual Basic (Declaration)

Public Property SelfClosed As Boolean

Visual C++

public:
property bool SelfClosed {
   bool get ();
   void set (bool value);
}
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Rule have not got all needed information yet, keep appending lines to this LineInfo until rule says that it's done.

**Namespace:** [HttpServer.Rendering.Haml](#)  
**Assembly:** [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public Rule UnfinishedRule { get; set; }
```

Visual Basic (Declaration)

```vbnet
Public Property UnfinishedRule As Rule
```

Visual C++

```cpp
public:
    property Rule^ UnfinishedRule {
        Rule^ get ();
        void set (Rule^ value);
    }
```
See Also

-LineInfo Class
-HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Namespace: HttpServer.Rendering.Haml
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string UnparsedData { get; }

Visual Basic (Declaration)

Public ReadOnly Property UnparsedData As String

Visual C++

public:
property String^ UnparsedData {
    String^ get ();
}

{
See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Number of whitespaces before actual entry beings.

**Namespace:**  [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int WhiteSpaces { get; }

Visual Basic (Declaration)

Public ReadOnly Property WhiteSpaces As Integer

Visual C++

public:
property int WhiteSpaces {
    int get ();
}

See Also

LineInfo Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A list of prototype nodes.

Namespace: [HttpServer.Rendering.Haml](#)
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class NodeList

Visual Basic (Declaration)

Public Class NodeList

Visual C++

public ref class NodeList
Inheritance Hierarchy

System::Object
HttpServer.Rendering.Haml::NodeList
See Also

NodeList Members
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **NodeList** type exposes the following members.
# Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NodeList</td>
<td>Initializes a new instance of the NodeList class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Add a prototype</td>
</tr>
<tr>
<td>CreateNode</td>
<td>Creates the node.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetPrototype</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
See Also

NodeList Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `NodeList` class

**Namespace:**  [HttpServer.Rendering.Haml](http://HttpServer.Rendering.Haml)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public NodeList()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New
```

**Visual C++**

```cpp
public:
NodeList()
```
See Also

NodeList Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **NodeList** type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Add a prototype</td>
</tr>
<tr>
<td>CreateNode</td>
<td>Creates the node.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetPrototype</td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

NodeList Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add a prototype

**Namespace:** [HttpServer.Rendering.Haml](http://example.com/namespace)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public void Add(
    Node node
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Add (_, node As Node _)
```

**Visual C++**

```cpp
public:
void Add(
    Node^ node
)
```

### Parameters

`node`

Type: `HttpServer.Rendering.Haml.Nodes::Node`

prototype node
See Also

NodeList Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
NodeList::CreateNode Method

Creates the node.

**Namespace:** [HttpServer.Rendering.Haml](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Node CreateNode(
    string word,
    Node parent
)

Visual Basic (Declaration)

Public Function CreateNode ( _
    word As String, _
    parent As Node _
) As Node

Visual C++

public:
    Node^ CreateNode(
        String^ word,
        Node^ parent
    )

Parameters

word
    Type: System::String
    node identifier.

parent
    Type: HttpServer.Rendering.Haml.Nodes::Node
    parent node.

Return Value

created node if identifier was found; otherwise null.
See Also

- NodeList Class
- HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
NodeList..::.GetPrototype Method

Namespace: HttpServer.Rendering.Haml
Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Node GetPrototype(string word, bool firstNode)

Visual Basic (Declaration)

Public Function GetPrototype(_
    word As String, _
    firstNode As Boolean _
) As Node

Visual C++

public:
Node^ GetPrototype(_
    String^ word, _
    bool firstNode
)

Parameters

word

Type: System::String


firstNode

Type: System::Boolean

first node on line

Return Value

[Missing <returns> documentation for]
See Also

NodeList Class
HttpServer.Rendering.Haml Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttributeNode</td>
<td>Contains HTML attributes.</td>
</tr>
<tr>
<td>AttributeNode:::Attribute</td>
<td>A attribute</td>
</tr>
<tr>
<td>ChildNode</td>
<td>Child nodes may not be the first node on a line</td>
</tr>
<tr>
<td>DisplayCodeNode</td>
<td>The follow node allows users to display code in a haml document</td>
</tr>
<tr>
<td>Node</td>
<td>A node represents a controller code and contents. For example a html tag, an attribute list or something else.</td>
</tr>
<tr>
<td>PartialNode</td>
<td>A partial is a HAML template that is inserted into other HAML templates.</td>
</tr>
<tr>
<td>TagNode</td>
<td>Represents a HTML tag.</td>
</tr>
<tr>
<td>TextNode</td>
<td>A text only node.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AttributeNode Class

Members  See Also  Send Feedback

Contains HTML attributes.

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public class AttributeNode : ChildNode
```

**Visual Basic (Declaration)**

```vbnet
Public Class AttributeNode _
    Inherits ChildNode
```

**Visual C++**

```cpp
public ref class AttributeNode : public ChildNode
```
Inheritance Hierarchy

System..::.Object
See Also

(AttributeNode Members)
(HttpServer.Rendering.Haml.Nodes Namespace)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **AttributeNode** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AttributeName</code></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddMe</strong></td>
<td>Creates a DIV node and add's the specified node to it.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">ChildNode</a>.)</td>
</tr>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>determines if this node can handle the line (by checking the first word);</td>
</tr>
<tr>
<td></td>
<td>(Overrides [Node::CanHandle(String, Boolean)].)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetAttribute</strong></td>
<td>Get an attribute</td>
</tr>
<tr>
<td><strong>GetEndPos</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Object</a>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node.</td>
</tr>
<tr>
<td></td>
<td>(Overrides [Node::Parse(NodeList, Node, LineInfo, Int32%)].)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Convert node to HTML (with ASP-tags) (Overrrides Node..:::ToHtml()()().)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AllChildrenCount</td>
<td>Count our children and our childrens children and so on... (Inherited from Node.)</td>
</tr>
<tr>
<td>Attributes</td>
<td>html attributes</td>
</tr>
<tr>
<td>Children</td>
<td>All child nodes for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td>IsTextNode</td>
<td>This is a plain text node (Inherited from ChildNode.)</td>
</tr>
<tr>
<td>LastModifier</td>
<td>Last modifier for this node (Inherited from Node.)</td>
</tr>
<tr>
<td>LineInfo</td>
<td>Should only be set for each parent. (Inherited from Node.)</td>
</tr>
<tr>
<td>ModifierCount</td>
<td>Number of modifiers for this line. (Inherited from Node.)</td>
</tr>
<tr>
<td>Modifiers</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node. (Inherited from Node.)</td>
</tr>
</tbody>
</table>
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AttributeNode Constructor

AttributeNode Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AttributeNode(Node)</code></td>
<td>create an attribute node</td>
</tr>
<tr>
<td><code>AttributeNode(Node, List&lt;Of &lt;(AttributeNode,::,Attribute)&gt;&gt;)</code></td>
<td>Create a new node</td>
</tr>
</tbody>
</table>
See Also

AttributeNode Class
AttributeNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
create an attribute node

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public AttributeNode(
    Node parent
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    parent As Node _
)
```

**Visual C++**

```cpp
public:
AttributeNode(
    Node^ parent
)
```

**Parameters**

parent

Type: `HttpServer.Rendering.Haml.Nodes::Node`
See Also

AttributeNode Class
AttributeNode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new node

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public AttributeNode(  
    Node parent,  
    List<AttributeNode::::Attribute> col  
)

Visual Basic (Declaration)

Public Sub New ( _  
    parent As Node, _  
    col As List(Of AttributeNode::::Attribute) _  
)

Visual C++

public:  
AttributeNode(  
    Node^ parent,  
    List<AttributeNode::::Attribute^>^ col  
)

Parameters

parent
    Type: HttpServer.Rendering.Haml.Nodes::::Node
    parent node

col
    Type: System.Collections.Generic::::List<(Of  
        (AttributeNode::::Attribute)>)
    collection of attributes
See Also

AttributeNode Class
AttributeNode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AttributeNode` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddMe</strong></td>
<td>Creates a DIV node and add's the specified node to it. (Inherited from <a href="#">ChildNode</a>).</td>
</tr>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word); (Overrides <a href="#">Node::CanHandle(String, Boolean)</a>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetAttribute</strong></td>
<td>Get an attribute</td>
</tr>
<tr>
<td><strong>GetEndPos</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Overrides <a href="#">Node::Parse(NodeList, Node, LineInfo, Int32%)</a>.)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
| **ToHtml** | Convert node to HTML (with ASP-tags)  
(Overrides `Node::ToHtml()`.) |
| **ToString** | Returns a `String` that represents the current `Object`.  
(Inherited from `Object`.) |
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AttributeNode...:::CanHandle Method

Determines if this node can handle the line (by checking the first word);

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public override bool CanHandle(
    string word,
    bool firstNode
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function CanHandle (_
    word As String, _
    firstNode As Boolean _
) As Boolean
```

**Visual C++**

```c++
public:
virtual bool CanHandle(
    String^ word,
    bool firstNode
) override
```

### Parameters

- **word**
  - Type: `System::String`
  - Controller char (word)

- **firstNode**
  - Type: `System::Boolean`
  - first node on line

### Return Value

true if text belongs to this node type
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get an attribute

Namespace:  [HttpServer.Rendering.Haml.Nodes]
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public AttributeNode::Attribute GetAttribute(
    string name
)
```

### Visual Basic (Declaration)

```vbnet
Public Function GetAttribute ( _
    name As String _
) As AttributeNode::Attribute
```

### Visual C++

```c++
public:
AttributeNode::Attribute GetAttribute(
    String^ name
)
```

## Parameters

name

Type: `System::String`
name of the attribute (case sensitive)

## Return Value

attribute if found; otherwise null.
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver

AttributeNode....::GetEndPos Method

AttributeNode Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GetEndPos(Int32, String)</code></td>
<td>Get endposition for this modifier. (Inherited from ChildNode.)</td>
</tr>
<tr>
<td><code>GetEndPos(Int32, String, Char)</code></td>
<td>Get endposition for this modifier. (Inherited from ChildNode.)</td>
</tr>
</tbody>
</table>
See Also

AttributeNode Class
AttributeNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse node contents add return a fresh node.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public override Node Parse(
    NodeList prototypes,
    Node parent,
    LineInfo line,
    ref int offset
)
```

### Visual Basic (Declaration)

```vbnet
Public Overrides Function Parse ( _
    prototypes As NodeList, _
    parent As Node, _
    line As LineInfo, _
    ByRef offset As Integer _
) As Node
```

### Visual C++

```cpp
public:
    virtual Node^ Parse(
        NodeList^ prototypes,
        Node^ parent,
        LineInfo^ line,
        int% offset
    ) override
```

## Parameters

### prototypes
- **Type:** `HttpServer.Rendering.Haml..:::NodeList`
- List containing all node types

### parent
- **Type:** `HttpServer.Rendering.Haml.Nodes..:::Node`
- Node that this is a subnode to. Can be null
line
  Type: [HttpServer.Rendering.Haml:::.LineInfo]
  Line to parse

offset
  Type: [System:::.Int32]
  Where to start the parsing. Should be set to where the next node should start parsing.

Return Value

A node corresponding to the bla bla; null if parsing failed.
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>[HttpServer.Rendering.:::CodeGeneratorException]</td>
<td></td>
</tr>
</tbody>
</table>
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# □ □ Visual Basic
□ Visual C++
□ Include Protected Members
□ Include Inherited Members
C# Webserver

AttributeNode:::ToCode Method

AttributeNode Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ToCode(Boolean%)</code></td>
<td>Converts the node to C# code (Inherited from <code>Node</code>).</td>
</tr>
<tr>
<td><code>ToCode(Boolean%, Boolean)</code></td>
<td>Convert the node to C# code (Inherited from <code>Node</code>).</td>
</tr>
<tr>
<td><code>ToCode(Boolean%, Boolean, Boolean)</code></td>
<td>Convert the node to C# code (Overrides Node...::ToCode(Boolean%, Boolean, Boolean).)</td>
</tr>
</tbody>
</table>
See Also

AttributeNode Class
AttributeNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert the node to C# code


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
C#

protected override string ToCode(
    ref bool inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
)

Visual Basic (Declaration)

Protected Overrides Function ToCode ( _
    ByRef inString As Boolean, _
    smallEnough As Boolean, _
    smallEnoughIsDefaultValue As Boolean _
) As String

Visual C++

protected:
virtual String^ ToCode(
    bool% inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
) override

Parameters

inString
    Type: System::Boolean %
    True if we are inside the internal stringbuilder

smallEnough
    Type: System::Boolean
    true if all subnodes fit on one line

smallEnoughIsDefaultValue
    Type: System::Boolean
smallEnough is a default value, recalc it

Return Value

c# code
See Also

AttributeNode Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert node to HTML (with ASP-tags)

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override string ToHtml()

Visual Basic (Declaration)

Public Overrides Function ToHtml As String

Visual C++

public:
    virtual String^ ToHtml() override

Return Value

HTML string
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **AttributeNode** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllChildrenCount</td>
<td>Count our children and our childrens children and so on</td>
</tr>
<tr>
<td>Attributes</td>
<td>html attributes</td>
</tr>
<tr>
<td>Children</td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td>IsTextNode</td>
<td>This is a plain text node</td>
</tr>
<tr>
<td>LastModifier</td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td>LineInfo</td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td>ModifierCount</td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td>Modifiers</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node.</td>
</tr>
</tbody>
</table>
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
html attributes

Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public List<AttributeNode..:::Attribute> Attributes { get; }

Visual Basic (Declaration)

Public ReadOnly Property Attributes As List(Of AttributeNode..:::At

Visual C++

public:
property List<AttributeNode..:::Attribute>^ Attributes {
    List<AttributeNode..:::Attribute>^ get ();
}
See Also

AttributeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A attribute

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class Attribute

Visual Basic (Declaration)

Public Class Attribute

Visual C++

public ref class Attribute
Inheritance Hierarchy

System..::.Object

See Also

AttributeNode::Attribute Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains HTML attributes.

The `AttributeName::Attribute` type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttributeNode:::Attribute</td>
<td>Initializes a new instance of the AttributeNode:::Attribute class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the attribute</td>
</tr>
<tr>
<td>Simple</td>
<td>Value is a simple word or quoted text</td>
</tr>
<tr>
<td>Value</td>
<td>Value, can be a statement, variable or quoted text.</td>
</tr>
</tbody>
</table>
See Also

AttributeNode,:::Attribute Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializer a new instance of the `AttributeNode` class

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Attribute()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
Attribute()
See Also

AttributeNode:::Attribute Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AttributeNode...::Attribute` type exposes the following members.
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>Name of the attribute</td>
</tr>
<tr>
<td><strong>Simple</strong></td>
<td>value is a simple word or quoted text</td>
</tr>
<tr>
<td><strong>Value</strong></td>
<td>Value, can be a statement, variable or quoted text.</td>
</tr>
</tbody>
</table>
See Also

AttributeNode::<Attribute Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Name of the attribute

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Name

Visual Basic (Declaration)

Public Name As String

Visual C++

public:
String^ Name
See Also

AttributeNode, Attribute Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
value is a simple word or quoted text

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Simple

Visual Basic (Declaration)

Public Simple As Boolean

Visual C++

public:
bool Simple
See Also

AttributeNode, Attribute Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Value, can be a statement, variable or quoted text.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public string Value
```

**Visual Basic (Declaration)**

```vbnet
Public Value As String
```

**Visual C++**

```cpp
public: String^ Value
```
See Also

AttributeNode::< Attribute Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members

C# Webserver

AttributeNode...:::Attribute Methods

AttributeNode...:::Attribute

See Also
Send Feedback

The AttributeNode...:::Attribute type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also


This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Child nodes may not be the first node on a line

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://localhost:8080)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract class ChildNode : Node

Visual Basic (Declaration)

Public MustInherit Class ChildNode _
    Inherits Node

Visual C++

public ref class ChildNode abstract : public Node
Inheritance Hierarchy

System::Object

HttpServer.Rendering.Haml.Nodes::Node
HttpServer.Rendering.Haml.Nodes::ChildNode

HttpServer.Rendering.Haml.Nodes::AttributeNode
See Also

ChildNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ChildNode` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildNode</td>
<td>Child nodes may not be the first node on a line</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddMe</strong></td>
<td>Creates a DIV node and add's the specified node to it.</td>
</tr>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word);</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection.CLUSION.</td>
</tr>
<tr>
<td><strong>GetEndPos</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>.</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node.</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Generate HTML code (with ASP tags)</td>
</tr>
</tbody>
</table>

Returns a **String** that represents the current **Object**.
ToString (Inherited from Object.)
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllChildrenCount</td>
<td>Count our children and our childrens children and so on...</td>
</tr>
<tr>
<td>Children</td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td>IsTextNode</td>
<td>This is a plain text node</td>
</tr>
<tr>
<td>LastModifier</td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td>LineInfo</td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td>ModifierCount</td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td>Modifiers</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node.</td>
</tr>
</tbody>
</table>

(Inherited from Node.)
See Also

ChildNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public ChildNode(
    Node parent
)

Visual Basic (Declaration)

Public Sub New (_
    parent As Node _
)

Visual C++

public:
ChildNode(
    Node^ parent
)

Parameters

parent
    Type: HttpServer.Rendering.Haml.Nodes..::Node
    parent node
See Also

ChildNode Class
HttpServer.Rendering.HamlNodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `ChildNode` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddMe</strong></td>
<td>Creates a DIV node and add's the specified node to it. Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>AddModifier</strong></td>
<td></td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word); (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetEndPos</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intedination level for this node. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Generate HTML code (with ASP tags) (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>.</td>
</tr>
</tbody>
</table>
ToString (Inherited from Object.)
See Also

ChildNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
ChildNode..::..AddMe Method

**FirstChild Class**  **See Also**  **Send Feedback**

Creates a DIV node and add's the specified node to it.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public Node AddMe(
    NodeList prototypes,
    Node parent,
    LineInfo line,
    Node me
)
```

#### Visual Basic (Declaration)

```vbnet
Public Function AddMe ( _
    prototypes As NodeList, _
    parent As Node, _
    line As LineInfo, _
    me As Node _
) As Node
```

#### Visual C++

```cpp
public: 
    Node^ AddMe( 
        NodeList^ prototypes, 
        Node^ parent, 
        LineInfo^ line, 
        Node^ me 
    )
```

### Parameters

prototypes
- **Type:** `HttpServer.Rendering.Haml::::NodeList`
- Contains all prototypes for each control char. used to instanciate new nodes.

parent
- **Type:** `HttpServer.Rendering.Haml.Nodes::::Node`
- parent node
line
Type: HttpServer.Rendering.Haml:::LineInfo
current line information

me
Type: HttpServer.Rendering.Haml.Nodes:::Node
node to add to the DIV node

Return Value

current node
See Also

ChildNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
ChildNode...::GetEndPos Method

ChildNode Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetEndPos(Int32, String)</td>
<td>Get endposition for this modifier.</td>
</tr>
<tr>
<td>GetEndPos(Int32, String, Char)</td>
<td>Get endposition for this modifier.</td>
</tr>
</tbody>
</table>
See Also

**ChildNode Class**
**ChildNode Members**
**HttpServer.Rendering.Haml.Nodes Namespace**

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get endposition for this modifier.

Namespace:  [HttpServer.Rendering.Haml.Nodes](http://example.com)
Assembly:  [HttpServer](http://example.com) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
protected int GetEndPos(
    int offset,
    string line
)
```

**Visual Basic (Declaration)**

```vbnet
Protected Function GetEndPos ( _
    offset As Integer, _
    line As String _
) As Integer
```

**Visual C++**

```cpp
protected:
    int GetEndPos(
        int offset,
        String^ line
    )
```

**Parameters**

- **offset**
  - Type: `System::Int32`
  - where to start searching

- **line**
  - Type: `System::String`
  - contents of the current line

**Return Value**

- index on the current line
See Also

ChildNode Class
GetEndPos Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get endposition for this modifier.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected int GetEndPos(
    int offset,
    string line,
    char terminator
)

Visual Basic (Declaration)

Protected Function GetEndPos ( _
    offset As Integer, _
    line As String, _
    terminator As Char _
) As Integer

Visual C++

protected:
int GetEndPos(
    int offset,
    String^ line,
    wchar_t terminator
)

Parameters

offset
    Type: System::Int32
    where to start searching

line
    Type: System::String
    contents of the current line

terminator
    Type: System::Char
char that tells us that this is the end position

Return Value

index on the current line
See Also

ChildNode Class
GetEndPos Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/1" alt="" /> <img src="https://example.com/2" alt="" /> <strong>ToCode(Boolean%)</strong></td>
<td>Converts the node to C# code (Inherited from Node.)</td>
</tr>
<tr>
<td><img src="https://example.com/3" alt="" /> <img src="https://example.com/4" alt="" /> <strong>ToCode(Boolean%, Boolean)</strong></td>
<td>Convert the node to C# code (Inherited from Node.)</td>
</tr>
<tr>
<td><img src="https://example.com/5" alt="" /> <img src="https://example.com/6" alt="" /> <strong>ToCode(Boolean%, Boolean, Boolean)</strong></td>
<td>Generate C# code of the node (Inherited from Node.)</td>
</tr>
</tbody>
</table>
See Also

ChildNode Class
ChildNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **ChildNode** type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our childrens children and so</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>This is a plain text node</td>
</tr>
<tr>
<td></td>
<td>(Overrides Node::IsTextNode.)</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers</td>
</tr>
<tr>
<td></td>
<td>are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from Node.)</td>
</tr>
</tbody>
</table>
See Also

ChildNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This is a plain text node

**Namespace:** [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override bool IsTextNode { get; }
```

Visual Basic (Declaration)

```vbnet
Public Overrides ReadOnly Property IsTextNode As Boolean
```

Visual C++

```c++
public:
virtual property bool IsTextNode {
    bool get () override;
}
```
See Also

ChildNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The follow node allows users to display code in a haml document


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class DisplayCodeNode : ChildNode

Visual Basic (Declaration)

Public Class DisplayCodeNode _
Inherits ChildNode

Visual C++

public ref class DisplayCodeNode : public ChildNode
Examples

#main Welcome =user.FirstName
Inheritance Hierarchy

System::Object
HttpServer.Rendering.Haml.Nodes::Node
HttpServer.Rendering.Haml.Nodes::ChildNode
See Also

DisplayCodeNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `DisplayCodeNode` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DisplayCodeNode</td>
<td>Contains C# code that will be rendered into the view.</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddMe</strong></td>
<td>Creates a DIV node and add's the specified node to it. (Inherited from <code>ChildNode</code>.)</td>
</tr>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <code>Node</code>.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>determines if this node can handle the line (by checking the first word); (Overrides <code>Node::CanHandle(String, Boolean)</code>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetEndPos</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from <code>Node</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Overrides <code>Node::Parse(NodeList, Node, LineInfo, Int32%)</code>.)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Generate HTML for this node (with asp tags for</td>
</tr>
</tbody>
</table>
### ToHtml

(Overrides Node.::ToHtml().)

### ToString

Returns a String that represents the current Object.
(Inherited from Object.)
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our childrens children and so on...</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>Determines if this is a textnode (containing plain text) (Overrids <strong>ChildNode::IsTextNode</strong>.)</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node. (Inherited from <strong>Node</strong>.)</td>
</tr>
</tbody>
</table>
See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains C# code that will be rendered into the view.

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public DisplayCodeNode(
    Node parent
)

Visual Basic (Declaration)

Public Sub New ( _
    parent As Node _
)

Visual C++

public:
DisplayCodeNode(
    Node^ parent
)

Parameters

parent
    Type: [HttpServer.Rendering.Haml.Nodes.::Node](#) Parent node
See Also

[DisplayCodeNode Class](#)
[HttpServer.Rendering.Haml.Nodes Namespace](#)

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The DisplayCodeNode type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddMe</strong></td>
<td>Creates a DIV node and add's the specified node to it. (Inherited from ChildNode.)</td>
</tr>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>determines if this node can handle the line (by checking the first word); (Overrides Node::CanHandle(String, Boolean).)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetEndPos</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Overrides Node::Parse(NodeList, Node, LineInfo, Int32%).)</td>
</tr>
</tbody>
</table>
| **ToCode**   | Overloaded. Generate HTML for this node (with asp tags for
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ToHtml</code></td>
<td>Returns HTML code (Overrides <code>Node...:ToHtml()()</code>)</td>
</tr>
<tr>
<td><code>ToString</code></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines if this node can handle the line (by checking the first word);

Namespace:  [HttpServer.Rendering.Haml.Nodes](#)
Assembly:  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public override bool CanHandle(
    string word,
    bool firstNode
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function CanHandle ( _
    word As String, _
    firstNode As Boolean _
) As Boolean
```

**Visual C++**

```cpp
public:
virtual bool CanHandle(
    String^ word,
    bool firstNode
) override
```

### Parameters

- **word**
  - Type: `System::String`  
  - Controller char (word)

- **firstNode**
  - Type: `System::Boolean`  
  - first node on line

### Return Value

true if text belongs to this node type
See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
DisplayCodeNode.....GetEndPos Method
DisplayCodeNode Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetEndPos(Int32, String)</td>
<td>Get endposition for this modifier. (Inherited from ChildNode.)</td>
</tr>
<tr>
<td>GetEndPos(Int32, String, Char)</td>
<td>Get endposition for this modifier. (Inherited from ChildNode.)</td>
</tr>
</tbody>
</table>
See Also

DisplayCodeNode Class
DisplayCodeNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DisplayCodeNode..::.Parse Method

DisplayCodeNode Class  See Also  Send Feedback

Parse node contents add return a fresh node.

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override Node Parse(
    NodeList prototypes,
    Node parent,
    LineInfo line,
    ref int offset
)

Visual Basic (Declaration)

Public Overrides Function Parse (_
    prototypes As NodeList, _
    parent As Node, _
    line As LineInfo, _
    ByRef offset As Integer _
) As Node

Visual C++

public:
    virtual Node^ Parse(
    NodeList^ prototypes,
    Node^ parent,
    LineInfo^ line,
    int% offset
) override

Parameters

prototypes
    Type: HttpServer.Rendering.Haml:::.NodeList
    List containing all node types

parent
    Type: HttpServer.Rendering.Haml.Nodes:::.Node
    Node that this is a subnode to. Can be null
line
  Type: `HttpServer.Rendering.Haml.Like.LineInfo`
  Line to parse

offset
  Type: `System.Int32`
  Where to start the parsing. Should be set to where the next node should start parsing.

**Return Value**

A node corresponding to the bla bla; null if parsing failed.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering...::CodeGenException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DisplayCodeNode..::.ToCode Method

DisplayCodeNode Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ToCode(Boolean%)</td>
<td>Converts the node to c# code (Inherited from Node.)</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean)</td>
<td>Convert the node to c# code (Inherited from Node.)</td>
</tr>
<tr>
<td></td>
<td>= is placed at the end of a tag definition, after class, id, and attribute declarations. It’s just a shortcut for inserting Ruby code into an element. It works the same as = without a tag: it inserts the result of the Ruby code into the template. However, if the result is short enough, it is displayed entirely on one line. (Overrides Node...; ToCode(Boolean%, Boolean, Boolean).)</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean, Boolean)</td>
<td></td>
</tr>
</tbody>
</table>
See Also

DisplayCodeNode Class
DisplayCodeNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DisplayCodeNode...:.ToCode Method (Boolean%, Boolean, Boolean)

DisplayCodeNode Class  See Also  Send Feedback

= is placed at the end of a tag definition, after class, id, and attribute
declarations. It’s just a shortcut for inserting Ruby code into an element. It works
the same as = without a tag: it inserts the result of the Ruby code into the
template. However, if the result is short enough, it is displayed entirely on one
line.

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected override string ToCode(
    ref bool inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
)

Visual Basic (Declaration)

Protected Overrides Function ToCode ( _
    ByRef inString As Boolean, _
    smallEnough As Boolean, _
    smallEnoughIsDefaultValue As Boolean _
) As String

Visual C++

protected:
    virtual String^ ToCode(
        bool% inString,
        bool smallEnough,
        bool smallEnoughIsDefaultValue
    ) override

Parameters

inString
    Type: System:::Boolean %
    True if we are inside the internal stringbuilder

smallEnough
    Type: System:::Boolean
    true if all subnodes fit on one line

smallEnoughIsDefaultValue
    Type: System:::Boolean
smallEnough is a default value, recalc it

Return Value

c# code
See Also

DisplayCodeNode Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DisplayCodeNode...:::ToHtml Method

**DisplayCodeNode Class**  **See Also**  **Send Feedback**

Generate HTML for this node (with asp tags for code)

**Namespace:**  **HttpServer.Rendering.Haml.Nodes**  
**Assembly:**  **HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)**
### Syntax

**C#**

```csharp
public override string ToHtml()
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function ToHtml As String
```

**Visual C++**

```cpp
public:
virtual String^ ToHtml() override
```

### Return Value

[Missing <returns> documentation for
See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The DisplayCodeNode type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our childrens children and so on... (Inherited from <a href="#">Node</a>)</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node. (Inherited from <a href="#">Node</a>)</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>Determines if this is a textnode (containing plain text) (Overrides <a href="#">ChildNode::IsTextNode</a>)</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node. (Inherited from <a href="#">Node</a>)</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent. (Inherited from <a href="#">Node</a>)</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line. (Inherited from <a href="#">Node</a>)</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <a href="#">Node</a>)</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node. (Inherited from <a href="#">Node</a>)</td>
</tr>
</tbody>
</table>
See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
DisplayCodeNode..:::.IsTextNode Property

Determines if this is a textnode (containg plain text)

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool IsTextNode { get; }

Visual Basic (Declaration)

Public Overrides ReadOnly Property IsTextNode As Boolean

Visual C++

public:
virtual property bool IsTextNode {
    bool get () override;
}

See Also

DisplayCodeNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A node represents a controller code and contents. For example a html tag, an attribute list or something else.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract class Node

Visual Basic (Declaration)

Public MustInherit Class Node

Visual C++

public ref class Node abstract
Inheritance Hierarchy

System:::.Object
See Also

Node Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The Node type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node</td>
<td>Add a new node</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AddModifier</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td>CanHandle</td>
<td>Determines if this node can handle the line (by checking the first word);</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>GetIntendation</td>
<td>Get intendation level for this node.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Parse</td>
<td>Parse node contents add return a fresh node.</td>
</tr>
<tr>
<td>ToCode</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>ToHtml</td>
<td>Generate HTML code (with ASP tags)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllChildrenCount</td>
<td>Count our children and our childrens children and so on...</td>
</tr>
<tr>
<td>Children</td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td>IsTextNode</td>
<td>Returns true if the node is applicable after another node as text instead of being applied as a modifier</td>
</tr>
<tr>
<td>LastModifier</td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td>LineInfo</td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td>ModifierCount</td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td>Modifiers</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node.</td>
</tr>
</tbody>
</table>
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Add a new node

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Node(
    Node parent
)

Visual Basic (Declaration)

Public Sub New (_
    parent As Node _
)

Visual C++

public:
Node(
    Node^ parent
)

Parameters

parent
    Type: HttpServer.Rendering.Haml.Nodes::Node
    parent node.
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Node** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word);</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node.</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Generate HTML code (with ASP tags)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
</tbody>
</table>
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public void AddModifier(
    Node node
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub AddModifier (_
    node As Node _
)
```

### Visual C++

```cpp
public:
void AddModifier(
    Node^ node
)
```

## Parameters

`node`

Type: `HttpServer.Rendering.Haml.Nodes::Node`

Modifier node
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Node::CanHandle Method
determines if this node can handle the line (by checking the first word);

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract bool CanHandle(
    string word,
    bool firstNode
)

Visual Basic (Declaration)

Public MustOverride Function CanHandle ( _
    word As String, _
    firstNode As Boolean _
) As Boolean

Visual C++

public:
    virtual bool CanHandle(
        String^ word,
        bool firstNode
    ) abstract

Parameters

word
    Type: System::String
    Controller char (word)

firstNode
    Type: System::Boolean
    First node on line, used since some nodes cannot exist on their own on a line.

Return Value

true if text belongs to this node type
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Get intedation level for this node.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected int GetIntendation()

Visual Basic (Declaration)

Protected Function GetIntendation As Integer

Visual C++

protected: int GetIntendation()

Return Value

See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse node contents add return a fresh node.


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract Node Parse(
  NodeList prototypes,
  Node parent,
  LineInfo line,
  ref int offset
)

Visual Basic (Declaration)

Public MustOverride Function Parse ( _
  prototypes As NodeList, _
  parent As Node, _
  line As LineInfo, _
  ByRef offset As Integer _
) As Node

Visual C++

public:
    virtual Node^ Parse( 
        NodeList^ prototypes,
        Node^ parent,
        LineInfo^ line,
        int% offset
    ) abstract

Parameters

prototypes
  Type: HttpContext.Topic.Haml..:::ListNode
  List containing all node types

parent
  Type: HttpContext.Topic.Haml.Nodes..:::Node
  Node that this is a subnode to. Can be null
line
  Type: HttpServer.Rendering.Haml.:::LineInfo
  Line to parse

offset
  Type: System.:::Int32 %
  Where to start the parsing. Should be set to where the next node should start parsing.

Return Value

A node corresponding to the bla bla; null if parsing failed.
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpException.Rendering...::CodeGenException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Node...: ToCode Method

See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ToCode(Boolean%)</code></td>
<td>Converts the node to c# code</td>
</tr>
<tr>
<td><code>ToCode(Boolean%, Boolean)</code></td>
<td>Convert the node to c# code</td>
</tr>
<tr>
<td><code>ToCode(Boolean%, Boolean, Boolean)</code></td>
<td>Generate C# code of the node</td>
</tr>
</tbody>
</table>
See Also

Node Class
Node Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Converts the node to C# code

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public virtual string ToCode(
    ref bool inString
)

Visual Basic (Declaration)

Public Overridable Function ToCode ( _
    ByRef inString As Boolean _
) As String

Visual C++

public:
virtual String^ ToCode(
    bool% inString
)

Parameters

inString
    Type: System::::Boolean %
    true if we are inside the internal stringbuilder

Return Value

c# code
See Also

Node Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert the node to C# code


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public virtual string ToCode(
    ref bool inString,
    bool smallEnough
)

Visual Basic (Declaration)

Public Overridable Function ToCode ( _
    ByRef inString As Boolean, _
    smallEnough As Boolean _
) As String

Visual C++

public:
    virtual String^ ToCode(
        bool% inString,
        bool smallEnough
    )

Parameters

inString
    Type: System::Boolean %
    True if we are inside the internal stringbuilder

smallEnough
    Type: System::Boolean
    true if all subnodes fit on one line

Return Value

c# code
See Also

Node Class
ToCode Overload

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate C# code of the node

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
protected abstract string ToCode(
    ref bool inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
)

Visual Basic (Declaration)
Protected MustOverride Function ToCode ( _
    ByRef inString As Boolean, _
    smallEnough As Boolean, _
    smallEnoughIsDefaultValue As Boolean _
) As String

Visual C++
protected:
virtual String^ ToCode(
    bool% inString,
    bool% smallEnough,
    bool% smallEnoughIsDefaultValue
) abstract

Parameters

inString
    Type: System::Boolean %
    true if we are inside the internal stringbuilder

smallEnough
    Type: System::Boolean
    code is small enough to fit on one row.

smallEnoughIsDefaultValue
    Type: System::Boolean
smallEnough is a default value, recalc it

Return Value

c# code
See Also

Node Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate HTML code (with ASP tags)

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public abstract string ToHtml()
```

**Visual Basic (Declaration)**

```vbnet
Public MustOverride Function ToHtml As String
```

**Visual C++**

```cpp
public:
virtual String^ ToHtml() abstract
```

**Return Value**

See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `Node` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our children's children and so on...</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>Returns true if the node is applicable after another node as text instead of being applied as a modifier</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node.</td>
</tr>
</tbody>
</table>
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Node...::AllChildrenCount Property

Count our children and our childrens children and so on...

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com/namespace)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public int AllChildrenCount { get; }

Visual Basic (Declaration)

Public ReadOnly Property AllChildrenCount As Integer

Visual C++

public:
property int AllChildrenCount {
    int get ();
}


See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
All child nodes for this node.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public LinkedList<Node> Children { get; }

Visual Basic (Declaration)

Public ReadOnly Property Children As LinkedList(Of Node)

Visual C++

public:
property LinkedList<Node>^ Children {
    LinkedList<Node>^ get ();
}
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Node...: IsTextNode Property

Node Class  See Also  Send Feedback

Returns true if the node is applicable after another node as text instead of being applied as a modifier

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract bool IsTextNode { get; }

Visual Basic (Declaration)

Public MustOverride ReadOnly Property IsTextNode As Boolean

Visual C++

public:
virtual property bool IsTextNode {
    bool get () abstract;
}
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Last modifier for this node

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Node LastModifier { get; }

Visual Basic (Declaration)

Public Readonly Property LastModifier As Node

Visual C++

public:
property Node^ LastModifier {
    Node^ get ();
}

See Also

Node Class
Node:::Modifiers

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Should only be set for each parent.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public LineInfo LineInfo { get; set; }
```

### Visual Basic (Declaration)

```vbnet
Public Property LineInfo As LineInfo
```

### Visual C++

```c++
public:
property LineInfo^ LineInfo {
    LineInfo^ get ();
    void set (LineInfo^ value);
}
```
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Node...::ModifierCount Property

**Node Class**  See Also  Send Feedback

Number of modifiers for this line.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public int ModifierCount { get; }
```

**Visual Basic (Declaration)**

```vbnet
Public ReadOnly Property ModifierCount As Integer
```

**Visual C++**

```cpp
public:
property int ModifierCount {
    int get ();
}
```
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.

Assembly: HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected LinkedList<Node> Modifiers { get; }

Visual Basic (Declaration)

Protected ReadOnly Property Modifiers As LinkedList(Of Node)

Visual C++

protected:

property LinkedList<Node>^ Modifiers {
                   LinkedList<Node>^ get();
}
See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parent node.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public Node Parent { get; }

Visual Basic (Declaration)

Public ReadOnly Property Parent As Node

Visual C++

public:
property Node^ Parent {
    Node^ get ();
}

See Also

Node Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A partial is a HAML template that is inserted into other HAML templates.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class PartialNode : Node

Visual Basic (Declaration)

Public Class PartialNode
    Inherits Node

Visual C++

public ref class PartialNode : public Node
Inheritance Hierarchy

System..:::Object


See Also

PartialNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The PartialNode type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartialNode</td>
<td>create a new partial node.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddModifier</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td>CanHandle</td>
<td>Determines if this node can handle the line (by checking the first word); (Overrides Node:::CanHandle(String, Boolean).)</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetCode</td>
<td>Helper function to retrieve the code for the partial request</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetIntendation</td>
<td>Get indention level for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Parse</td>
<td>Parse node contents add return a fresh node. (Overrides Node:::Parse(NodeList, Node, LineInfo, Int32%).)</td>
</tr>
<tr>
<td>ThrowError</td>
<td>Throw an exception with predefined information</td>
</tr>
<tr>
<td>ToCode</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>ToHtml</td>
<td>Convert node to HTML (with ASP-tags) (Overrides Node:::ToHtml().)</td>
</tr>
</tbody>
</table>
ToString

Returns a String that represents the current Object.
(Inherited from Object.)
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllChildrenCount</td>
<td>Count our children and our childrens children and so...</td>
</tr>
<tr>
<td>Children</td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td>IsTextNode</td>
<td>This is a plain text node (Overrides Node::IsTextNode.)</td>
</tr>
<tr>
<td>LastModifier</td>
<td>Last modifier for this node (Inherited from Node.)</td>
</tr>
<tr>
<td>LineInfo</td>
<td>Should only be set for each parent. (Inherited from Node.)</td>
</tr>
<tr>
<td>ModifierCount</td>
<td>Number of modifiers for this line. (Inherited from Node.)</td>
</tr>
<tr>
<td>Modifiers</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node.</td>
</tr>
<tr>
<td>Target</td>
<td>Returns the page/controller target for the node</td>
</tr>
</tbody>
</table>

*(Inherited from Node.)*
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
create a new partial node.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public PartialNode(
    Node parent
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New ( _
    parent As Node _
)
```

**Visual C++**

```cpp
public:
    PartialNode(
        Node^ parent
    )
```

**Parameters**

parent

Type: `HttpServer.Rendering.Haml.Nodes..:::Node`

parent node
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **PartialNode** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddModifier</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td>CanHandle</td>
<td>Determines if this node can handle the line (by checking the first word); (Overrides Node::.CanHandle(String, Boolean).)</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetCode</td>
<td>Helper function to retrieve the code for the partial request.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td>GetIntendation</td>
<td>Get intendation level for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td>Parse</td>
<td>Parse node contents add return a fresh node. (Overrides Node:::Parse(NodeList, Node, LineInfo, Int32%).)</td>
</tr>
<tr>
<td>ThrowError</td>
<td>Throw an exception with predefined information</td>
</tr>
<tr>
<td>ToCode</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>ToHtml</td>
<td>Convert node to HTML (with ASP-tags) (Overrides Node:::ToHtml().)</td>
</tr>
</tbody>
</table>
ToString

Returns a String that represents the current Object.
(Inherited from Object.)
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines if this node can handle the line (by checking the first word);

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public override bool CanHandle(
    string word,
    bool firstNode
)
```

### Visual Basic (Declaration)

```vbnet
Public Overrides Function CanHandle ( _
    word As String, _
    firstNode As Boolean _
) As Boolean
```

### Visual C++

```cpp
public:
    virtual bool CanHandle(
        String^ word,
        bool firstNode
    ) override
```

## Parameters

**word**
- Type: `System::::String`
- Controller char (word)

**firstNode**
- Type: `System::::Boolean`
- First node on line.

## Return Value

True if text belongs to this node type
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
PartialNode..:::.GetCode Method

**PartialNode Class**  **See Also**  **Send Feedback**

Helper function to retrieve the code for the partial request

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected string GetCode()

Visual Basic (Declaration)

Protected Function GetCode As String

Visual C++

protected:
String^ GetCode()

Return Value

A string representing the code for invocating a render of the partial
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpServer.Rendering.........</td>
<td>CompileException</td>
</tr>
</tbody>
</table>
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse node contents add return a fresh node.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public override Node Parse(
    NodeList prototypes,
    Node parent,
    LineInfo line,
    ref int offset
)
```

### Visual Basic (Declaration)

```vbnet
Public Overrides Function Parse ( _
    prototypes As NodeList, _
    parent As Node, _
    line As LineInfo, _
    ByRef offset As Integer _
) As Node
```

### Visual C++

```cpp
public:
    virtual Node^ Parse(
        NodeList^ prototypes,
        Node^ parent,
        LineInfo^ line,
        int% offset
    ) override
```

## Parameters

### prototypes

Type: `HttpServer.Rendering.Haml..::.NodeList`  
A list with node types

### parent

Type: `HttpServer.Rendering.Haml.Nodes..::.Node`  
Node that this is a subnode to. Can be null
line
Type: `HttpServer.Rendering.Haml:::LineInfo`
Line to parse

offset
Type: `System:::Int32` %
Where to start the parsing. Will be set to where the next node should start parsing

**Return Value**

A node corresponding to the bla bla; null if parsing failed.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering...::CodeGenarorException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
PartialNode..:::.ThrowException Method

PartialNode Class  See Also  Send Feedback

Throw an exception with predefined information

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected void ThrowError(
    string reason
)

Visual Basic (Declaration)

Protected Sub ThrowError ( _
    reason As String _
)

Visual C++

protected:
void ThrowError(
    String^ reason
)

Parameters

reason
Type: System::String
why the exception was thrown
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
PartialNode...:::ToCode Method

PartialNode Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ToCode(Boolean%)</td>
<td>Converts the node to C# code (Inherited from <code>Node</code>.)</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean)</td>
<td>Convert the node to C# code (Inherited from <code>Node</code>.)</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean, Boolean)</td>
<td>Convert the node to C# code (Overrides <code>Node:::ToCode(Boolean%, Boolean, Boolean)</code>.)</td>
</tr>
</tbody>
</table>
See Also

PartialNode Class
PartialNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert the node to C# code

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected override string ToCode(
    ref bool inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
)

Visual Basic (Declaration)

Protected Overrides Function ToCode ( _
    ByVal inString As Boolean, _
    smallEnough As Boolean, _
    smallEnoughIsDefaultValue As Boolean _
) As String

Visual C++

protected:
    virtual String^ ToCode(
    bool% inString,
    bool% smallEnough,
    bool% smallEnoughIsDefaultValue
    ) override

Parameters

inString
    Type: System::::Boolean %
    True if we are inside the internal StringBuilder

smallEnough
    Type: System::::Boolean
    true if all subnodes fit on one line

smallEnoughIsDefaultValue
    Type: System::::Boolean
smallEnough is a default value, recalc it

**Return Value**

c# code
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpServer.Rendering...::CompileException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

PartialNode Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert node to HTML (with ASP-tags)

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override string ToHtml()

Visual Basic (Declaration)

Public Overrides Function ToHtml As String

Visual C++

public:
virtual String^ ToHtml() override

Return Value

HTML string
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `PartialNode` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our childrens children and so on...</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>This is a plain text node</td>
</tr>
<tr>
<td></td>
<td>(Overrides <a href="#">Node.::IsTextNode</a>.)</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Target</strong></td>
<td>Returns the page/controller target for the node.</td>
</tr>
</tbody>
</table>
See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This is a plain text node

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool IsTextNode { get; }

Visual Basic (Declaration)

Public Overrides ReadOnly Property IsTextNode As Boolean

Visual C++

public:
virtual property bool IsTextNode {
    bool get () override;
}

See Also

PartialNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
PartialNode::$::Target Property

PartialNode Class  See Also  Send Feedback

Returns the page/controller target for the node

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public string Target { get; }
```

**Visual Basic (Declaration)**

```vbnet
Public ReadOnly Property Target As String
```

**Visual C++**

```cpp
public:
property String^ Target {
    String^ get ();
}
```
See Also

PartialNode Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TagNode Class

Members  See Also  Send Feedback

Represents a HTML tag.

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

public class TagNode : Node

**Visual Basic (Declaration)**

Public Class TagNode  
    Inherits Node

**Visual C++**

public ref class TagNode : public Node
Inheritance Hierarchy

System:::Object
HttpServer.Rendering.Haml.Nodes:::Node
HttpServer.Rendering.Haml.Nodes:::TagNode
See Also

TagNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TagNode type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TagNode</td>
<td>Create a new HTML tag node.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word); (Overrides Node:::CanHandle(String, Boolean).)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Overrides Node:::Parse(NodeList, Node, LineInfo, Int32%).)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Convert node to HTML (with ASP-tags) (Overrides Node:::ToHtml(0)).</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our childrens children and so</td>
</tr>
<tr>
<td></td>
<td>on...</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>This is a plain text node</td>
</tr>
<tr>
<td></td>
<td>(Overrides <a href="#">Node::IsTextNode</a>.)</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers</td>
</tr>
<tr>
<td></td>
<td>are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>tag name</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
</tbody>
</table>
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Create a new HTML tag node.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```
public TagNode(
    Node parent
)
```

**Visual Basic (Declaration)**

```
Public Sub New (
    parent As Node
)
```

**Visual C++**

```
public:
TagNode(
    Node^ parent
)
```

**Parameters**

parent

Type: `HttpServer.Rendering.Haml.Nodes::Node`

parent node
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TagNode type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>determines if this node can handle the line (by checking the first word); (Overides <strong>Node</strong>:::CanHandle(String, Boolean).)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from <strong>Node</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Overides <strong>Node</strong>:::Parse(NodeList, Node, LineInfo, Int32%)).</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Convert node to HTML (with ASP-tags) (Overides <strong>Node</strong>:::ToHtml()).</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TagNode...::CanHandle Method

TagNode Class  See Also  Send Feedback

determines if this node can handle the line (by checking the first word);

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override bool CanHandle(
    string word,
    bool firstNode
)
```

Visual Basic (Declaration)

```vbnet
Public Overrides Function CanHandle ( _
    word As String, _
    firstNode As Boolean _
) As Boolean
```

Visual C++

```cpp
public:
virtual bool CanHandle(
    String^ word,
    bool firstNode
) override
```

Parameters

word
Type: `System::String`
Controller char (word)

firstNode
Type: `System::Boolean`
first node on line.

Return Value

true if text belongs to this node type
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TagNode... Parse Method

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)

Parse node contents add return a fresh node.
Syntax

C#

public override Node Parse(
    NodeList prototypes,
    Node parent,
    LineInfo line,
    ref int offset
)

Visual Basic (Declaration)

Public Overrides Function Parse ( _
    prototypes As NodeList, _
    parent As Node, _
    line As LineInfo, _
    ByRef offset As Integer _
) As Node

Visual C++

public:
    virtual Node^ Parse(
        NodeList^ prototypes,
        Node^ parent,
        LineInfo^ line,
        int% offset
    ) override

Parameters

prototypes
    Type: HttpServer.Rendering.Haml::$_::NodeList
    A list with node types

parent
    Type: HttpServer.Rendering.Haml.Nodes::$_::Node
    Node that this is a subnode to. Can be null
line
Type: `HttpServer.Rendering.Haml..:::LineInfo`
Line to parse

offset
Type: `System..:::Int32` %
Where to start the parsing. Will be set to where the next node should start parsing

**Return Value**

A node corresponding to the bla bla; null if parsing failed.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering...CodeGenException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TagNode...ToCode Method

TagNode Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ToCode(Boolean%)</td>
<td>Converts the node to C# code (Inherited from Node.)</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean)</td>
<td>Convert the node to C# code (Inherited from Node.)</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean, Boolean)</td>
<td>Convert the node to C# code (Overrides Node...ToCode(Boolean%, Boolean, Boolean).)</td>
</tr>
</tbody>
</table>
See Also

TagNode Class
TagNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert the node to C# code

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected override string ToCode(
    ref bool inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
)

Visual Basic (Declaration)

Protected Overrides Function ToCode ( _
    ByRef inString As Boolean, _
    smallEnough As Boolean, _
    smallEnoughIsDefaultValue As Boolean _
) As String

Visual C++

protected:
virtual String^ ToCode(
    bool% inString,
    bool% smallEnough,
    bool% smallEnoughIsDefaultValue
) override

Parameters

inString
    Type: System:::Boolean %
    True if we are inside the internal stringbuilder

smallEnough
    Type: System:::Boolean
    true if all subnodes fit on one line

smallEnoughIsDefaultValue
    Type: System:::Boolean
smallEnough is a default value, recalc it

**Return Value**

c# code
See Also

TagNode Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert node to HTML (with ASP-tags)

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public override string ToHtml()

Visual Basic (Declaration)
Public Overrides Function ToHtml As String

Visual C++
public:
virtual String^ ToHtml() override

ReturnValue

HTML string
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TagNode** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AllChildrenCount</strong></td>
<td>Count our children and our childrens children and so on...</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Children</strong></td>
<td>All child nodes for this node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>IsTextNode</strong></td>
<td>This is a plain text node</td>
</tr>
<tr>
<td></td>
<td>(Overrides <a href="#">Node::IsTextNode</a>.)</td>
</tr>
<tr>
<td><strong>LastModifier</strong></td>
<td>Last modifier for this node</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>LineInfo</strong></td>
<td>Should only be set for each parent.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>ModifierCount</strong></td>
<td>Number of modifiers for this line.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Modifiers</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers</td>
</tr>
<tr>
<td></td>
<td>are attribute lists, class names and IDs.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>tag name</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Parent node.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <a href="#">Node</a>.)</td>
</tr>
</tbody>
</table>
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
This is a plain text node


**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public override bool IsTextNode { get; }
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Readonly Property IsTextNode As Boolean
```

**Visual C++**

```c++
public:
virtual property bool IsTextNode {
    bool get () override;
}
```
See Also

TagNode Class
HttpPostRendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TagNode::Name Property

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public string Name { get; set; }
```

#### Visual Basic (Declaration)

```vbnet
Public Property Name As String
```

#### Visual C++

```cpp
public:
property String^ Name {
    String^ get ();
    void set (String^ value);
}
```
See Also

TagNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
A text only node.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class TexNode : Node

Visual Basic (Declaration)

Public Class TextNode _
    Inherits Node

Visual C++

public ref class TexNode : public Node
Inheritance Hierarchy

System...:::Object
See Also

TextNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TextNode** type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>TextNode</code></td>
<td>Initializes a new instance of the <code>TextNode</code> class</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word); ( Overrides Node:::CanHandle(String, Boolean). )</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from Node.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object. (Inherited from Object.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. ( Overrides Node:::Parse(NodeList, Node, LineInfo, Int32%).)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Generate HTML for this node. (Overrides Node:::ToHtml().()</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object. (Inherited from Object.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllChildrenCount</td>
<td>Count our children and our childrens children and so on... (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>Children</td>
<td>All child nodes for this node. (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>IsTextNode</td>
<td>Is this a text node (containing plain text)? (Overrides <a href="#">Node::IsTextNode</a>.)</td>
</tr>
<tr>
<td>LastModifier</td>
<td>Last modifier for this node (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>LineInfo</td>
<td>Should only be set for each parent. (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>ModifierCount</td>
<td>Number of modifiers for this line. (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>Modifiers</td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent node. (Inherited from <a href="#">Node</a>.)</td>
</tr>
<tr>
<td>Text</td>
<td>The text.</td>
</tr>
</tbody>
</table>
See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initialize a new instance of the `TextNode` class.

**Namespace:** [HttpServer.Rendering.Haml.Nodes](http://example.com)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public TextNode(
    Node parent,
    string text
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    parent As Node, _
    text As String _
)
```

**Visual C++**

```cpp
public:
TextNode(
    Node^ parent,
    String^ text
)
```

### Parameters

**parent**

Type: `HttpServer.Rendering.Haml.Nodes:::Node`

parent node

**text**

Type: `System:::String`

plain text
See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **TextNode** type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddModifier</strong></td>
<td>Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs. (Inherited from <code>Node</code>.)</td>
</tr>
<tr>
<td><strong>CanHandle</strong></td>
<td>Determines if this node can handle the line (by checking the first word); (Overrides <code>Node::CanHandle(String, Boolean)</code>.)</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetIntendation</strong></td>
<td>Get intendation level for this node. (Inherited from <code>Node</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Parse node contents add return a fresh node. (Overrides <code>Node::Parse(NodeList, Node, LineInfo, Int32%)</code>.)</td>
</tr>
<tr>
<td><strong>ToCode</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToHtml</strong></td>
<td>Generate HTML for this node. (Overrees <code>Node::ToHtml()()()</code>)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>. (Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
TextNode::CanHandle Method
determines if this node can handle the line (by checking the first word);

**Namespace:** [HttpServer.Rendering.Haml.Nodes](#)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool CanHandle(  
    string word,  
    bool firstNode
)

Visual Basic (Declaration)

Public Overrides Function CanHandle ( _  
    word As String, _  
    firstNode As Boolean _
) As Boolean

Visual C++

public:  
    virtual bool CanHandle(  
        String^ word,  
        bool firstNode
    ) override

Parameters

word
    Type: System::String
    Controller char (word)

firstNode
    Type: System::Boolean
    true if this is the first node on the line.

Return Value

true if text belongs to this node type
See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver

TextNode...:::Parse Method

See Also  Send Feedback

Parse node contents add return a fresh node.

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public override Node Parse(
    NodeList prototypes,
    Node parent,
    LineInfo line,
    ref int offset)
```

Visual Basic (Declaration)

```vbnet
Public Overrides Function Parse(
    _
    prototypes As NodeList, _
    parent As Node, _
    line As LineInfo, _
    ByRef offset As Integer _
) As Node
```

Visual C++

```cpp
public:
virtual Node^ Parse(
    NodeList^ prototypes,
    Node^ parent,
    LineInfo^ line,
    int% offset
) override
```

Parameters

prototypes
Type: `HttpServer.Rendering.Haml..::.NodeList`
List containing all node types

parent
Type: `HttpServer.Rendering.Haml.Nodes..::.Node`
Node that this is a subnode to. Can be null
line
  Type: HttpServer.Rendering.Haml.::.LineInfo
  Line to parse

offset
  Type: System.::.Int32 %
  Where to start the parsing. Should be set to where the next node should start parsing.

**Return Value**

A node corresponding to the bla bla; null if parsing failed.
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering...::CodeGenException</code></td>
<td></td>
</tr>
</tbody>
</table>
See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
TextNode...:ToCode Method

TextNode Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ToCode(Boolean%)</td>
<td>Converts the node to c# code (Inherited from <a href="#">Node</a>).</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean)</td>
<td>Convert the node to c# code (Inherited from <a href="#">Node</a>).</td>
</tr>
<tr>
<td>ToCode(Boolean%, Boolean, Boolean)</td>
<td>Convert the node to c# code (Overrides <a href="#">Node::ToCode(Boolean%, Boolean, Boolean)</a>).</td>
</tr>
</tbody>
</table>
See Also

TextNode Class
TextNode Members
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Convert the node to C# code

**Namespace:** [HttpClient.Rendering.Haml.Nodes](#)

**Assembly:** HttpClient (in HttpClient.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

protected override string ToCode(
    ref bool inString,
    bool smallEnough,
    bool smallEnoughIsDefaultValue
)

Visual Basic (Declaration)

Protected Overrides Function ToCode ( _
    ByRef inString As Boolean, _
    smallEnough As Boolean, _
    smallEnoughIsDefaultValue As Boolean _
) As String

Visual C++

protected:
    virtual String^ ToCode( _
        bool% inString,
        bool% smallEnough,
        bool% smallEnoughIsDefaultValue
    ) override

Parameters

inString
    Type: System:::Boolean %
    True if we are inside the internal stringbuilder

smallEnough
    Type: System:::Boolean
    true if all subnodes fit on one line

smallEnoughIsDefaultValue
    Type: System:::Boolean
todo: add description

Return Value

c# code
See Also

TextNode Class
ToCode Overload
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate HTML for this node.

Assembly:  **HttpServer** (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override string ToHtml()

Visual Basic (Declaration)

Public Overrides Function ToHtml As String

Visual C++

public:
virtual String^ ToHtml() override

Return Value

See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `TextNode` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| **AllChildrenCount** | Count our children and our childrens children and so on...  
(Inherited from [Node](#).)                                                                                              |
| **Children**       | All child nodes for this node.  
(Inherited from [Node](#).)                                                                                                      |
| **IsTextNode**     | Is this a text node (containing plain text)?  
(Overrides [Node::IsTextNode](#).)                                                                                               |
| **LastModifier**   | Last modifier for this node  
(Inherited from [Node](#).)                                                                                                        |
| **LineInfo**       | Should only be set for each parent.  
(Inherited from [Node](#).)                                                                                                        |
| **ModifierCount**  | Number of modifiers for this line.  
(Inherited from [Node](#).)                                                                                                        |
| **Modifiers**      | Modifiers are used to modify the first node on the line. Typical modifiers are attribute lists, class names and IDs.  
(Inherited from [Node](#).)                                                                                                     |
| **Parent**         | Parent node.  
(Inherited from [Node](#).)                                                                                                    |
| **Text**           | The text.                                                                                                                                   |
See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Is this a text node (containing plain text)?


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool IsTextNode { get; }

Visual Basic (Declaration)

Public Overrides ReadOnly Property IsTextNode As Boolean

Visual C++

public:
virtual property bool IsTextNode {
    bool get () override;
}

See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The text.

**Namespace:**  [HttpServer.Rendering.Haml.Nodes](http://www.example.com)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Text { get; set; }

Visual Basic (Declaration)

Public Property Text As String

Visual C++

public:
property String^ Text {
    String^ get ();
    void set (String^ value);
}

See Also

TextNode Class
HttpServer.Rendering.Haml.Nodes Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
# Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttributesRule</td>
<td>Contains an (html) attribute list.</td>
</tr>
<tr>
<td>NewLineRule</td>
<td>Rule that says that something :)</td>
</tr>
<tr>
<td>Rule</td>
<td>Rules are used during preparse</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Contains an (html) attribute list.

**Namespace:**  [HttpServer.Rendering.Haml.Rules](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public class AttributesRule : Rule
```

Visual Basic (Declaration)

```vbnet
Public Class AttributesRule
    Inherits Rule
```

Visual C++

```cpp
public ref class AttributesRule : public Rule
```
Inheritance Hierarchy

System..:::Object
See Also

Attributes
Rule Members

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AttributesRule` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttributesRule</td>
<td>Initializes a new instance of the AttributesRule class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>IsMultiLine</td>
<td>Determines if this node spans over multiple lines. (Overides <a href="https://docs.microsoft.com/en-us/dotnet/api/system.routing.lineinfonode.ismultiline">Rule.::.IsMultiLine(LineInfo, Boolean)</a>.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

AttributesRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the AttributesRule class

**Namespace:**  [HttpServer.Rendering.Haml.Rules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public AttributesRule()
```

Visual Basic (Declaration)

```vbnet
Public Sub New
```

Visual C++

```cpp
public:
AttributesRule()
```
See Also

AttributesRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `AttributesRule` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.</td>
</tr>
<tr>
<td><strong>IsMultiLine</strong></td>
<td>Determines if this node spans over multiple lines.</td>
</tr>
<tr>
<td></td>
<td>(Overrides <code>Rule::&lt;::IsMultiLine(LineInfo, Boolean)</code>.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.</td>
</tr>
</tbody>
</table>
See Also

AttributesRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
AttributesRule..::.IsMultiLine Method

Determines if this node spans over multiple lines.

**Namespace:**  [HttpServer.Rendering.Haml.Rules]

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool IsMultiLine(
    LineInfo line,
    bool isContinued
)

Visual Basic (Declaration)

Public Overrides Function IsMultiLine ( _
    line As LineInfo, _
    isContinued As Boolean _
) As Boolean

Visual C++

public:
virtual bool IsMultiLine(
    LineInfo^ line,
    bool isContinued
) override

Parameters

line
Type: HttpServer.Rendering.Haml..::.LineInfo
contains line information (and text)

isContinued
Type: System..::.Boolean
true if the previous line was continued.

Return Value

true if this line continues onto the next.
See Also

AttributesRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Rule that says that something :)  

**Namespace:**  [HttpServer.Rendering.Haml.Rules](#)  
**Assembly:**  [HttpServer](#) (in [HttpServer.dll](#)) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class NewLineRule : Rule

Visual Basic (Declaration)

Public Class NewLineRule
    Inherits Rule

Visual C++

public ref class NewLineRule : public Rule
Inheritance Hierarchy

System..:::Object
See Also

NewLineRule Members

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `NewLineRule` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NewLineRule</td>
<td>Initializes a new instance of the NewLineRule class</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows a <a href="https://learn.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://learn.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://learn.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://learn.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="https://learn.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://learn.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td>IsMultiLine</td>
<td>Determines if this node spans over multiple lines. (Overrides <a href="https://learn.microsoft.com/en-us/dotnet/api/system.text.rule">Rule..::..IsMultiLine(LineInfo, Boolean)</a>.)</td>
</tr>
</tbody>
</table>
See Also

NewLineRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public NewLineRule()
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New
```

**Visual C++**

```cpp
public:
NewLineRule()
```
See Also

NewLineRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `NewLineRule` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <strong>Object</strong> is equal to the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <strong>Object</strong> to attempt to free resources and perform other cleanup operations before the <strong>Object</strong> is reclaimed by garbage collection. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <strong>Type</strong> of the current instance. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>IsMultiLine</strong></td>
<td>Determines if this node spans over multiple lines. (Overrides Rule::&lt;&gt;::IsMultiLine(LineInfo, Boolean).)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <strong>String</strong> that represents the current <strong>Object</strong>. (Inherited from <strong>Object</strong>.)</td>
</tr>
</tbody>
</table>
See Also

NewLineRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines if this node spans over multiple lines.

**Namespace:**  [HttpServer.Rendering.Haml.Rules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public override bool IsMultiLine(
    LineInfo line,
    bool isContinued
)

Visual Basic (Declaration)

Public Overrides Function IsMultiLine ( _
    line As LineInfo, _
    isContinued As Boolean _
) As Boolean

Visual C++

public: bool IsMultiLine(
    LineInfo^ line,
    bool isContinued
) override

Parameters

line
    Type: HttpServer.Rendering.Haml..::.LineInfo
        contains line information (and text)

isContinued
    Type: System..::.Boolean
        true if the previous line was continued.

Return Value

true if this line continues onto the next.
See Also

NewLineRule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Rules are used during preparse


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public abstract class Rule

Visual Basic (Declaration)
Public MustInherit Class Rule

Visual C++
public ref class Rule abstract
Inheritance Hierarchy

System::<Object
HttpServer.Rendering.Haml.Rules::<Rule
  HttpServer.Rendering.Haml.Rules::<AttributesRule
See Also

Rule Members

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `Rule` type exposes the following members.
Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule</td>
<td>Initializes a new instance of the Rule class</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>IsMultiLine</strong></td>
<td>Determines if this node spans over multiple lines.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

Rule Class
HttpRequest.Rendering.Haml.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Rule Constructor

Initializes a new instance of the Rule class

Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

protected Rule()

**Visual Basic (Declaration)**

Protected Sub New

**Visual C++**

protected:

Rule()
See Also

Rule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Rule** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>IsMultiLine</strong></td>
<td>Determines if this node spans over multiple lines.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
</tbody>
</table>
See Also

Rule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Determines if this node spans over multiple lines.


**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public abstract bool IsMultiLine(
    LineInfo line,
    bool isContinued
)

Visual Basic (Declaration)

Public MustOverride Function IsMultiLine ( _
    line As LineInfo, _
    isContinued As Boolean _
) As Boolean

Visual C++

public:
    virtual bool IsMultiLine(
    LineInfo^ line,
    bool isContinued
    ) abstract

Parameters

line
    Type: HttpHeaders.Rendering.Haml..::LineInfo
    contains line information (and text)

isContinued
    Type: System..::Boolean
    true if rule have previously inited a multiline.

Return Value

true if this line continues onto the next.
See Also

Rule Class

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#   Visual Basic
Visual C++
C# Webserver
HttpServer.Rendering.Tiny Namespace
Send Feedback

# Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🍃 TinyGenerator</td>
<td>Generates C# rendering object using ASP similar tags in the HTML code.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generates C# rendering object using ASP similar tags in the HTML code.

**Namespace:**  [HttpServer.Rendering.Tiny](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class TinyGenerator : TemplateGenerator

Visual Basic (Declaration)

Public Class TinyGenerator
    Implements TemplateGenerator

Visual C++

public ref class TinyGenerator : TemplateGenerator
Inheritance Hierarchy

```
System..:::Object
HttpServer.Rendering.Tiny..:::TinyGenerator
```
See Also

TinyGenerator Members
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TinyGenerator type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TinyGenerator</td>
<td>Initializes a new instance of the TinyGenerator class</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GenerateCode</strong></td>
<td>Generate C# code from the template.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

TinyGenerator Class
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the TinyGenerator class

**Namespace:** HttpServer.Rendering.Tiny
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public TinyGenerator()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
TinyGenerator()
See Also

TinyGenerator Class
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The TinyGenerator type exposes the following members.
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GenerateCode</strong></td>
<td>Generate C# code from the template.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
</tbody>
</table>
See Also

TinyGenerator Class
HttpServer.Rendering.Tiny Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Generate C# code from the template.

**Namespace:** [HttpServer.Rendering.Tiny](http://localhost:8080/)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

### C#

```csharp
public void GenerateCode(
    TextWriter writer
)
```

### Visual Basic (Declaration)

```vbnet
Public Sub GenerateCode ( _
    writer As TextWriter _
)
```

### Visual C++

```cpp
public:
virtual void GenerateCode(
    TextWriter^ writer
) sealed
```

## Parameters

**writer**

- **Type:** `System.IO.TextWriter`
- A textwriter that the generated code will be written to.

## Implements

- `TemplateGenerator.GenerateCode(TextWriter)`
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System:::InvalidOperationException</td>
<td>If the template have not been parsed first.</td>
</tr>
<tr>
<td>HttpServer.Rendering:::CodeGeneratorException</td>
<td>If template is incorrect</td>
</tr>
</tbody>
</table>
See Also

TinyGenerator Class
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
# Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parse(TextReader)</td>
<td>Parse a file and convert into our own template object code.</td>
</tr>
<tr>
<td>Parse(String)</td>
<td>Parse a file and convert into our own template object code.</td>
</tr>
</tbody>
</table>
See Also

TinyGenerator Class
TinyGenerator Members
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a file and convert into our own template object code.

Namespace:  [HttpServer.Rendering.Tiny](#)
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public void Parse(
    TextReader reader
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Parse (_
    reader As TextReader _
)
```

**Visual C++**

```cpp
public:
    virtual void Parse(
        TextReader^ reader
    ) sealed
```

**Parameters**

reader

Type: System.IO.TextReader

A textreader containing our template

**Implements**

`TemplateGenerator.Parse(TextReader)`
## Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>HttpServer.Rendering...::CodeGenException</code></td>
<td>If something is incorrect in the template.</td>
</tr>
</tbody>
</table>
See Also

TinyGenerator Class
Parse Overload
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Parse a file and convert into our own template object code.

**Namespace:**  [HttpServer.Rendering.Tiny](#)

**Assembly:**  [HttpServer](#) (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Parse(
    string fullPath
)

Visual Basic (Declaration)

Public Sub Parse ( _
    fullPath As String _
)

Visual C++

public:
virtual void Parse(
    String^ fullPath
) sealed

Parameters

fullPath
Type: System::String
Path and filename to a template

Implements

TemplateGenerator::Parse(String)
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpServer.Rendering..::.CodeGenErrorException</td>
<td>If something is incorrect in the template.</td>
</tr>
<tr>
<td>System.IO..::.FileNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System.IO..::.DirectoryNotFoundException</td>
<td></td>
</tr>
<tr>
<td>System..::.UnauthorizedAccessException</td>
<td></td>
</tr>
<tr>
<td>System.IO..::.PathTooLongException</td>
<td></td>
</tr>
<tr>
<td>System..::.NotSupportedException</td>
<td></td>
</tr>
</tbody>
</table>
See Also

TinyGenerator Class
Parse Overload
HttpServer.Rendering.Tiny Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RedirectRule</td>
<td>redirects from one url to another.</td>
</tr>
<tr>
<td>RegexRedirectRule</td>
<td>Class to make dynamic binding of redirects. Instead of having to specify a</td>
</tr>
<tr>
<td></td>
<td>number of similar redirect rules a regular expression can be used to identify</td>
</tr>
<tr>
<td></td>
<td>redirect urls and their targets.</td>
</tr>
</tbody>
</table>
## Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule</td>
<td>Rules are used to perform operations before a request is being handled. Rules can be used to create routing etc.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

*(c) 2008 Gauffin Telecom AB*
redirects from one url to another.

**Namespace:**  [HttpServer.Rules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public class RedirectRule : Rule
```

**Visual Basic (Declaration)**

```vbnet
Public Class RedirectRule
    Implements Rule
```

**Visual C++**

```cpp
public ref class RedirectRule : Rule
```
Inheritance Hierarchy

System..:::Object
HttpServer.Rules..:::RedirectRule
  HttpServer.Rules..:::RegexRedirectRule
See Also

RedirectRule Members
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **RedirectRule** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RedirectRule</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Equals</code></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>Finalize</code></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>GetHashCode</code></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>GetType</code></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>MemberwiseClone</code></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><code>Process</code></td>
<td>Process the incoming request.</td>
</tr>
<tr>
<td><code>ToString</code></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromUrl</td>
<td>string to match request url with.</td>
</tr>
<tr>
<td>ShouldRedirect</td>
<td>true if we should redirect.</td>
</tr>
<tr>
<td>ToUrl</td>
<td>Where to redirect.</td>
</tr>
</tbody>
</table>
See Also

RedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RedirectRule Constructor

RedirectRule Class  See Also  Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RedirectRule(String, String)</td>
<td>Initializes a new instance of the RedirectRule class.</td>
</tr>
<tr>
<td>RedirectRule(String, String, Boolean)</td>
<td>Initializes a new instance of the RedirectRule class.</td>
</tr>
</tbody>
</table>
See Also

RedirectRule Class
RedirectRule Members
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RedirectRule Constructor (String, String)

Initializes a new instance of the RedirectRule class.

**Namespace:**  HttpServer.Rules
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RedirectRule(
    string fromUrl,
    string toUrl
)

Visual Basic (Declaration)

Public Sub New (_
    fromUrl As String, _
    toUrl As String _
)

Visual C++

public:
RedirectRule(
    String^ fromUrl,
    String^ toUrl
)

Parameters

fromUrl
Type: System::String
Absolute path (no servername)

toUrl
Type: System::String
Absolute path (no servername)
Examples

server.Add(new RedirectRule("/", "/user/index"));
See Also

RedirectRule Class
RedirectRule Overload
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RedirectRule Constructor (String, String, Boolean)

RedirectRule Class  Example  See Also  Send Feedback

Initializes a new instance of the RedirectRule class.

**Namespace:**  HttpServer.Rules

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RedirectRule(
    string fromUrl,
    string toUrl,
    bool shouldRedirect
)

Visual Basic (Declaration)

Public Sub New ( _
    fromUrl As String, _
    toUrl As String, _
    shouldRedirect As Boolean _
)

Visual C++

public:
RedirectRule(
    String^ fromUrl,
    String^ toUrl,
    bool shouldRedirect
)

Parameters

fromUrl
Type: System::::String
Absolute path (no servername)

toUrl
Type: System::::String
Absolute path (no servername)

shouldRedirect
Type: System::::Boolean
true if request should be redirected, false if the request uri should be replaced.
Examples

server.Add(new RedirectRule("/", "/user/index"));
See Also

RedirectRule Class
RedirectRule Overload
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The RedirectRule type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="#">Type</a> of the current instance.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Process the incoming request.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</td>
</tr>
<tr>
<td>(Inherited from <a href="#">Object</a>.)</td>
<td></td>
</tr>
</tbody>
</table>
See Also

RedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Process the incoming request.

**Namespace:**  [HttpServer.Rules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public virtual bool Process(
    HttpRequest request,
    HttpResponse response
)

Visual Basic (Declaration)

Public Overridable Function Process ( _
    request As HttpRequest, _
    response As HttpResponse _
) As Boolean

Visual C++

public:
virtual bool Process(
    HttpRequest^ request,
    HttpResponse^ response
)

Parameters

request
    Type: HttpServer::HttpRequest
    incoming http request

response
    Type: HttpServer::HttpResponse
    outgoing http response

Return Value

true if response should be sent to the browser directly (no other rules or modules will be processed).
Implements

Rule:::Process(HttpRequest, HttpResponse)
Remarks

returning true means that no modules will get the request. Returning true is typically being done for redirects.
See Also

RedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **RedirectRule** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromUrl</td>
<td>string to match request url with.</td>
</tr>
<tr>
<td>ShouldRedirect</td>
<td>true if we should redirect.</td>
</tr>
<tr>
<td>ToUrl</td>
<td>Where to redirect.</td>
</tr>
</tbody>
</table>
See Also

RedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
string to match request url with.

**Namespace:** [HttpServer.Rules](#)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string FromUrl { get; }

Visual Basic (Declaration)

Public ReadOnly Property FromUrl As String

Visual C++

public:
property String^ FromUrl {
    String^ get ();
}


Remarks

Is compared to request.Uri.AbsolutePath
See Also

RedirectRule Class  
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
false if we should redirect.

Namespace:  **HttpServer.Rules**
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool ShouldRedirect { get; }

Visual Basic (Declaration)

Public ReadOnly Property ShouldRedirect As Boolean

Visual C++

public:
    property bool ShouldRedirect {
        bool get ();
    }
}
Remarks

false means that the rule will replace the current request uri with the new one from this class.
See Also

RedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Where to redirect.

**Namespace:** [HttpServer.Rules](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string ToUrl { get; }

Visual Basic (Declaration)

Public ReadOnly Property ToUrl As String

Visual C++

public:
property String^ ToUrl {
    String^ get ();
}

See Also

RedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Class to make dynamic binding of redirects. Instead of having to specify a number of similar redirect rules, a regular expression can be used to identify redirect URLs and their targets.

**Namespace:**  [HttpServer.Rules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public class RegexRedirectRule : RedirectRule

Visual Basic (Declaration)

Public Class RegexRedirectRule _
    Inherits RedirectRule

Visual C++

public ref class RegexRedirectRule : public RedirectRule
Examples

new RegexRedirectRule("/(?<target>[a-z0-9]+)", "/users/${target}?find=true", RegexOptions.IgnoreCase)
Inheritance Hierarchy

- System...:::Object
- HttpServer.Rules...:::RedirectRule
  - HttpServer.Rules...:::RegexRedirectRule
See Also

RegexRedirectRule Members
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RegexRedirectRule` type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RegexRedirectRule</td>
<td>Overloaded.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Process the incoming request. (Overrides <a href="https://docs.microsoft.com/en-us/dotnet/api/system.net.httpredirectrule">RedirectRule</a>::Process(<a href="https://docs.microsoft.com/en-us/dotnet/api/system.net.httprequest">HttpRequest</a>, <a href="https://docs.microsoft.com/en-us/dotnet/api/system.net.httpresponse">HttpResponse</a>).)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromUrl</td>
<td>string to match request url with. (Inherited from <a href="#">RedirectRule</a>.)</td>
</tr>
<tr>
<td>ShouldRedirect</td>
<td>true if we should redirect. (Inherited from <a href="#">RedirectRule</a>.)</td>
</tr>
<tr>
<td>ToUrl</td>
<td>Where to redirect. (Inherited from <a href="#">RedirectRule</a>.)</td>
</tr>
</tbody>
</table>
See Also

RegexRedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# | Visual Basic
Visual C++
Include Protected Members
Include Inherited Members

C# Webserver

See Also
Send Feedback
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>RegexRedirectRule(String, String)</code></td>
<td>Initializes a new instance of the <code>RegexRedirectRule</code> class.</td>
</tr>
<tr>
<td><code>RegexRedirectRule(String, String, RegexOptions)</code></td>
<td>Initializes a new instance of the <code>RegexRedirectRule</code> class.</td>
</tr>
<tr>
<td><code>RegexRedirectRule(String, String, RegexOptions, Boolean)</code></td>
<td>Initializes a new instance of the <code>RegexRedirectRule</code> class.</td>
</tr>
</tbody>
</table>
See Also

RegexRedirectRule Class
RegexRedirectRule Members
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Initializes a new instance of the `RegexRedirectRule` class.

**Namespace:**  [HttpServer.Rules](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public RegexRedirectRule(
    string fromUrlExpression,
    string toUrlExpression
)

Visual Basic (Declaration)

Public Sub New (_
    fromUrlExpression As String, _
    toUrlExpression As String _
)

Visual C++

public:
RegexRedirectRule(
    String^ fromUrlExpression,
    String^ toUrlExpression
)

Parameters

fromUrlExpression
    Type: System::String
    Expression to match url

toUrlExpression
    Type: System::String
    Expression to generate url
Examples

server.Add(new RegexRedirectRule("/(?<first>[a-zA-Z0-9]+)", "/user/${first}"); Result of ie. /employee1 will then be /user/employee1
See Also

RegexRedirectRule Class
RegexRedirectRule Overload
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver

RegexRedirectRule Constructor (String, String, RegexOptions)

`RegexRedirectRule Class  Example  See Also  Send Feedback`

Initializes a new instance of the `RegexRedirectRule` class.

**Namespace:**  `HttpServer.Rules`

**Assembly:**  `HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)`
Syntax

C#

public RegexRedirectRule(
    string fromUrlExpression,
    string toUrlExpression,
    RegexOptions options
)

Visual Basic (Declaration)

Public Sub New (_
    fromUrlExpression As String, _
    toUrlExpression As String, _
    options As RegexOptions _
)    

Visual C++

public:
    RegexRedirectRule(
        String^ fromUrlExpression, 
        String^ toUrlExpression, 
        RegexOptions options
    )

Parameters

fromUrlExpression
    Type: System::String
    Expression to match url

toUrlExpression
    Type: System::String
    Expression to generate url

options
    Type: System.Text.RegularExpressions::RegexOptions
Regular expression options to use, can be null
Examples

server.Add(new RegexRedirectRule("/(?<first>[a-zA-Z0-9]+)"), "/user/{first}", RegexOptions.IgnoreCase)); Result of ie. /employee1 will then be /user/employee1
See Also

RegexRedirectRule Class
RegexRedirectRule Overload
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
RegexRedirectRule Constructor (String, String, RegexOptions, Boolean)

Initializes a new instance of the RegexRedirectRule class.

**Namespace:** [HttpServer.Rules](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public RegexRedirectRule(
    string fromUrlExpression,
    string toUrlExpression,
    RegexOptions options,
    bool shouldRedirect
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub New (_
    fromUrlExpression As String, _
    toUrlExpression As String, _
    options As RegexOptions, _
    shouldRedirect As Boolean _
)
```

**Visual C++**

```cpp
public:
RegexRedirectRule(
    String^ fromUrlExpression,
    String^ toUrlExpression,
    RegexOptions options,
    bool shouldRedirect
)
```

**Parameters**

**fromUrlExpression**
- Type: `System::::String`
- Expression to match url

**toUrlExpression**
- Type: `System::::String`
- Expression to generate url
options
   Type: System.Text.RegularExpressions::RegexOptions
   Regular expression options to apply

shouldRedirect
   Type: System::Boolean
   true if request should be redirected, false if the request uri should be replaced.
server.Add(new RegexRedirectRule("/(?<first>[a-zA-Z0-9]+)"), "/user/${first}", RegexOptions.None)); Result of ie. /employee1 will then be /user/employee1
See Also

RegexRedirectRule Class
RegexRedirectRule Overload
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RegexRedirectRule` type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is equal to the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a> is reclaimed by garbage collection. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <a href="https://docs.microsoft.com/en-us/dotnet/api/system.type">Type</a> of the current instance. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>(Overrides <a href="https://docs.microsoft.com/en-us/dotnet/api/">RedirectRule</a>::Process(HttpRequest, HttpResponse).) Process the incoming request.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <a href="https://docs.microsoft.com/en-us/dotnet/api/system.string">String</a> that represents the current <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>. (Inherited from <a href="https://docs.microsoft.com/en-us/dotnet/api/system.object">Object</a>.)</td>
</tr>
</tbody>
</table>
See Also

RegexRedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Process the incoming request.

**Namespace:** [HttpServer.Rules](http://localhost:4000)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public override bool Process(
    HttpRequest request,
    HttpResponse response
)
```

**Visual Basic (Declaration)**

```vbnet
Public Overrides Function Process ( _
    request As HttpRequest, _
    response As HttpResponse _
) As Boolean
```

**Visual C++**

```cpp
public:
    virtual bool Process(
    HttpRequest^ request,
    HttpResponse^ response
) override
```

### Parameters

**request**

Type: `HttpServer::HttpRequest`

incoming http request

**response**

Type: `HttpServer::HttpResponse`

outgoing http response

### Return Value

true if response should be sent to the browser directly (no other rules or modules will be processed).
Implements

Rule::: Process(HttpRequest, HttpResponse)
Remarks

returning true means that no modules will get the request. Returning true is typically being done for redirects.
<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ArgumentNullException</code></td>
<td>If request or response is null</td>
</tr>
</tbody>
</table>
See Also

RegexRedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `RegexRedirectRule` type exposes the following members.
# Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromUrl</td>
<td>string to match request url with. (Inherited from RedirectRule.)</td>
</tr>
<tr>
<td>ShouldRedirect</td>
<td>true if we should redirect. (Inherited from RedirectRule.)</td>
</tr>
<tr>
<td>ToUrl</td>
<td>Where to redirect. (Inherited from RedirectRule.)</td>
</tr>
</tbody>
</table>
See Also

RegexRedirectRule Class
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Rules are used to perform operations before a request is being handled. Rules can be used to create routing etc.

**Namespace:** [HttpServer.Rules](http://HttpServer.Rules)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public interface Rule

Visual Basic (Declaration)

Public Interface Rule

Visual C++

public interface class Rule
See Also

Rule Members
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Rule** type exposes the following members.
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Process the incoming request.</td>
</tr>
</tbody>
</table>
See Also

Rule Interface
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **Rule** type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Process the incoming request.</td>
</tr>
</tbody>
</table>
See Also

Rule Interface
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Process the incoming request.

Namespace:  HttpServer.Rules
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
bool Process(
    HttpRequest request,
    HttpResponse response
)
```

Visual Basic (Declaration)

Function Process ( _
    request As HttpRequest, _
    response As HttpResponse _
) As Boolean

Visual C++

```cpp
bool Process(
    HttpRequest^ request,
    HttpResponse^ response
)
```

Parameters

request
  Type: `HttpServer::HttpRequest`
  incoming http request

response
  Type: `HttpServer::HttpResponse`
  outgoing http response

Return Value

true if response should be sent to the browser directly (no other rules or modules will be processed).
Remarks

returning true means that no modules will get the request. Returning true is typically being done for redirects.
### Exceptions

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System..::.ArgumentNullException</code></td>
<td>If request or response is null.</td>
</tr>
</tbody>
</table>
See Also

Rule Interface
HttpServer.Rules Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
[Missing <summary> documentation for N:HttpServer.Sessions]
## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemorySession</td>
<td>A session stored in memory.</td>
</tr>
<tr>
<td>MemorySessionStore</td>
<td>Session store using memory for each session.</td>
</tr>
<tr>
<td></td>
<td>todo: Remove expired sessions. We need to code a Scheduler service</td>
</tr>
<tr>
<td></td>
<td>in tiny library and use it.</td>
</tr>
</tbody>
</table>
## Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpSessionStore</td>
<td>The session store is used to manage sessions, the store itself can save it to any place it like as a database, memory or something else.</td>
</tr>
</tbody>
</table>

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The session store is used to manage sessions, the store itself can save it to any place it like as a database, memory or something else.

**Namespace:**  HttpServer.Sessions  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public interface HttpSessionStore

Visual Basic (Declaration)

Public Interface HttpSessionStore

Visual C++

public interface class HttpSessionStore
See Also

HttpSessionStore Members
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The ` HttpSessionStore ` type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddUnused</strong></td>
<td>We use the flyweight pattern which reuses small objects instead of creating new each time.</td>
</tr>
<tr>
<td><strong>Cleanup</strong></td>
<td>Remove expired sessions</td>
</tr>
<tr>
<td><strong>Create</strong></td>
<td>Overloaded.</td>
</tr>
<tr>
<td><strong>Load</strong></td>
<td>Load an existing session.</td>
</tr>
<tr>
<td><strong>Save</strong></td>
<td>Save an updated session to the store.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpireTime</td>
<td>Number of minutes before a session expires. Default is 20 minutes.</td>
</tr>
<tr>
<td>Item</td>
<td>Load a session from the store</td>
</tr>
</tbody>
</table>
See Also

HttpSessionStore Interface
HttpServer.Sessions Namespace
This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpSessionStore type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddUnused</td>
<td>We use the flyweight pattern which reuses small objects instead of creating new each time.</td>
</tr>
<tr>
<td>Cleanup</td>
<td>Remove expired sessions</td>
</tr>
<tr>
<td>Create</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Load</td>
<td>Load an existing session.</td>
</tr>
<tr>
<td>Save</td>
<td>Save an updated session to the store.</td>
</tr>
</tbody>
</table>
See Also

 HttpSessionStore Interface
 HttpSession.Sessions Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
We use the flyweight pattern which reuses small objects instead of creating new each time.

**Namespace:**  [HttpServer.Sessions](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

void AddUnused(
    HttpSession session
)

Visual Basic (Declaration)

Sub AddUnused ( _
    session As HttpSession _
)

Visual C++

void AddUnused(
    HttpSession^ session
)

Parameters

session
    Type: HttpSession
    EmptyLanguageNode (unused) session that should be reused next time
    Create is called.
See Also

 HttpSessionStore Interface
 HttpSession.Sessions Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
HttpSessionStore...::Cleanup Method

Remove expired sessions

**Namespace:**  [HttpServer.Sessions](http://example.com/HttpServer.Sessions)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
void Cleanup()

Visual Basic (Declaration)
Sub Cleanup

Visual C++
void Cleanup()
See Also

 HttpSessionStore Interface
 HttpSession.Sessions Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
HttpSessionStore...::Create Method

 HttpSessionStore Interface  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create()</td>
<td>Creates a new http session</td>
</tr>
<tr>
<td>Create(String)</td>
<td>Creates a new http session with a specific id</td>
</tr>
</tbody>
</table>
See Also

HttpSessionStore Interface
HttpSessionStore Members
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpSessionStore::Create Method

**HttpSessionStore Interface**  **See Also**  **Send Feedback**

Creates a new http session

**Namespace:**  **HttpServer.Sessions**  
**Assembly:**  **HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)**
Syntax

C#

HttpSession Create()

Visual Basic (Declaration)

Function Create As HttpSession

Visual C++

HttpSession^ Create()

Return Value

A HttpSession object
See Also

 HttpSessionStore Interface
 Create Overload
 HttpSession.Sessions Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
C#  Visual Basic
C#  Visual C++
C#  Webserver

HttpSessionStore..::.Create Method (String)

HttpRequestStore Interface  See Also  Send Feedback

Creates a new http session with a specific id

Namespace:  HttpSession.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
HttpSession Create(
    string id
)
```

Visual Basic (Declaration)

```vbnet
Function Create ( _
    id As String _
) As HttpSession
```

Visual C++

```cpp
HttpSession^ Create(
    String^ id
)
```

Parameters

id

Type: System::String


Return Value

A HttpSession object.
See Also

HttpSessionStore Interface
Create Overload
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Load an existing session.

**Namespace:**  [HttpServer.Sessions](#)  
**Assembly:**  HttpServer (in HttpServer.dll)  
Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
HttpSession Load(
    string sessionId
)
```

Visual Basic (Declaration)

```vbnet
Function Load ( _
    sessionId As String _
) As HttpSession
```

Visual C++

```cpp
HttpSession^ Load(
    String^ sessionId
)
```

Parameters

sessionId

Type: `System::String`


Return Value

A session if found; otherwise null.
See Also

 HttpSessionStore Interface
 HttpSessionServer.Sessions Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
Save an updated session to the store.

**Namespace:** [HttpServer.Sessions](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
void Save(
    HttpSession session
)
```

**Visual Basic (Declaration)**

```vbnet
Sub Save ( _
    session As HttpSession _
)
```

**Visual C++**

```cpp
void Save(
    HttpSession^ session
)
```

## Parameters

**session**

Type: HttpSession

<table>
<thead>
<tr>
<th>Exception</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.Exception.ArgumentException</code></td>
<td>If Id property have not been specified.</td>
</tr>
</tbody>
</table>
See Also

 HttpSessionStore Interface
 HttpSession.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The HttpSessionStore type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpireTime</td>
<td>Number of minutes before a session expires. Default is 20 minutes.</td>
</tr>
<tr>
<td>Item</td>
<td>Load a session from the store</td>
</tr>
</tbody>
</table>
See Also

HttpSessionStore Interface
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
HttpSessionStore...:::ExpireTime Property

 HttpSessionStore Interface  See Also  Send Feedback

Number of minutes before a session expires. Default is 20 minutes.

Namespace:  HttpSession.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
int ExpireTime { get; set; }
```

**Visual Basic (Declaration)**

**Property ExpireTime As** Integer

**Visual C++**

```cpp
property int ExpireTime {
    int get ();
    void set (int value);
}
```
- See Also

HttpSessionStore Interface
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Load a session from the store

**Namespace:** [HttpServer.Sessions](#)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
HttpSession this[string sessionId]
{
    get;
}
```

**Visual Basic (Declaration)**

```vbnet
ReadOnly Default Property Item ( _
    sessionId As String _
) As HttpSession
```

**Visual C++**

```cpp
property HttpSession^ default[String^ sessionId] {
    HttpSession^ get (String^ sessionId);
}
```

**Parameters**

sessionId
   Type: System:::String

**Return Value**

null if session is not found.
See Also

 HttpSessionStore Interface
 HttpSession.Sessions Namespace

 This project is maintained by Gauffin Telecom AB

 (c) 2008 Gauffin Telecom AB
A session stored in memory.

**Namespace:** [HttpServer.Sessions](#)
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public class MemorySession : HttpSession, IDisposable
```

Visual Basic (Declaration)

```vbnet
Public Class MemorySession
    Implements HttpSession, IDisposable
```

Visual C++

```cpp
public ref class MemorySession : HttpSession, IDisposable
```
Inheritance Hierarchy

System..::..Object
HttpServer.Sessions..::..MemorySession
See Also

MemorySession Members
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `MemorySession` type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemorySession</td>
<td>Initializes a new instance of the MemorySession class</td>
</tr>
</tbody>
</table>
# Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Clear</strong></td>
<td>Remove everything from the session</td>
</tr>
<tr>
<td><strong>Dispose</strong></td>
<td>Performs application-defined tasks associated with freeing, releasing, or</td>
</tr>
<tr>
<td></td>
<td>resetting unmanaged resources.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified <code>Object</code> is equal to the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an <code>Object</code> to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the <code>Object</code> is reclaimed by garbage collection.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the <code>Type</code> of the current instance.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a <code>String</code> that represents the current <code>Object</code>.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessed</td>
<td>when the session was last accessed.</td>
</tr>
<tr>
<td>Changed</td>
<td>Flag to indicate that the session have been changed and should be saved into the session store.</td>
</tr>
<tr>
<td>Count</td>
<td>Number of values in the session</td>
</tr>
<tr>
<td>Id</td>
<td>Session id</td>
</tr>
<tr>
<td>Item</td>
<td>Should</td>
</tr>
</tbody>
</table>
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession Constructor

Initializes a new instance of the MemorySession class

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public MemorySession(
    string id
)

Visual Basic (Declaration)

Public Sub New (_
    id As String _
)

Visual C++

public:
MemorySession(
    String^ id
)

Parameters

id

Type: System:::String
A unique id used by the sessions store to identify the session
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The MemorySession type exposes the following members.
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Clear</strong></td>
<td>Remove everything from the session</td>
</tr>
<tr>
<td><strong>Dispose</strong></td>
<td>Performs application-defined tasks associated with freeing, releasing, or</td>
</tr>
<tr>
<td></td>
<td>resetting unmanaged resources.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Allows an Object to attempt to free resources and perform other cleanup</td>
</tr>
<tr>
<td></td>
<td>operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a String that represents the current Object.</td>
</tr>
</tbody>
</table>
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Webserver
MemorySession...:::Clear Method

MemorySession Class  See Also  Send Feedback

Remove everything from the session

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Clear()

Visual Basic (Declaration)

Public Sub Clear

Visual C++

public:
virtual void Clear() sealed

Implements

 HttpSession:::::ClearQQQQ
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession...::..Dispose Method

MemorySession Class  See Also  Send Feedback

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Dispose()

Visual Basic (Declaration)

Public Sub Dispose

Visual C++

public:
virtual void Dispose() sealed

Implements

IDisposable:::Dispose()()
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `MemorySession` type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessed</td>
<td>when the session was last accessed.</td>
</tr>
<tr>
<td>Changed</td>
<td>Flag to indicate that the session have been changed and should be saved into the session store.</td>
</tr>
<tr>
<td>Count</td>
<td>Number of values in the session</td>
</tr>
<tr>
<td>Id</td>
<td>Session id</td>
</tr>
<tr>
<td>Item</td>
<td>Should</td>
</tr>
</tbody>
</table>
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession..:::Accessed Property

when the session was last accessed.

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public DateTime Accessed { get; set; }
```

**Visual Basic (Declaration)**

```vbnet
Public Property Accessed As DateTime
```

**Visual C++**

```cpp
public:
    virtual property DateTime Accessed {
        DateTime get () sealed;
        void set (DateTime value) sealed;
    }
```

**Implements**

`HttpSession:::Accessed`
Remarks

Used to determine when the session should be removed.
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession..::..Changed Property

Flag to indicate that the session have been changed and should be saved into the session store.

**Namespace:** [HttpServer.Sessions](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public bool Changed { get; set; }

Visual Basic (Declaration)

Public Property Changed As Boolean

Visual C++

public:
property bool Changed {
    bool get ();
    void set (bool value);
}
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession.Count Property

Number of values in the session

**Namespace:**  [HttpServer.Sessions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#
public int Count { get; }

Visual Basic (Declaration)
Public ReadOnly Property Count As Integer

Visual C++
public:
virtual property int Count {
    int get () sealed;
}

Implements

HttpSession:::Count
See Also

MemorySession Class
HttpContext.Session Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession.Id Property

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public string Id { get; }

Visual Basic (Declaration)

Public ReadOnly Property Id As String

Visual C++

public:
virtual property String^ Id {
    String^ get () sealed;
}

Implements

HttpRequest:::Id
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySession...:::Item Property

MemorySession Class  See Also  Send Feedback

Should

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public Object this[string name] { get; set; }
```

Visual Basic (Declaration)

```vbnet
Public Default Property Item ( _
    name As String _
) As Object
```

Visual C++

```cpp
public:
    virtual property Object^ default[Object^ String^ name] { 
        Object^ get (String^ name) sealed;
        void set (String^ name, Object^ value) sealed;
    }
```

Parameters

name
Type: System::String
Name of the session variable

Return Value

null if it's not set

Implements

HttpContext::Item[(String)]
See Also

MemorySession Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Session store using memory for each session. todo: Remove expired sessions.
We need to code a Scheduler service in tiny library and use it.

**Namespace:** [HttpServer.Sessions](#)

**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

**C#**

```csharp
public class MemorySessionStore : HttpSessionStore
```

**Visual Basic (Declaration)**

```vbnet
Public Class MemorySessionStore
    Implements HttpSessionStore
```

**Visual C++**

```cpp
public ref class MemorySessionStore : HttpSessionStore
```
Inheritance Hierarchy

System:::Object
HttpServer.Sessions:::MemorySessionStore
See Also

MemorySessionStore Members
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The MemorySessionStore type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemorySessionStore</td>
<td>Initializes a new instance of the MemorySessionStore class</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddUnused</td>
<td>We use the flyweight pattern which reuses small objects instead of creating new each time.</td>
</tr>
<tr>
<td>Cleanup</td>
<td>Remove expired sessions</td>
</tr>
<tr>
<td>Create</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>Load</td>
<td>Load an existing session.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
<tr>
<td>Save</td>
<td>Save an updated session to the store.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>. (Inherited from <a href="#">Object</a>)</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ExpireTime</strong></td>
<td>Number of minutes before a session expires. Default is 20 minutes.</td>
</tr>
<tr>
<td><strong>Item</strong></td>
<td>Load a session from the store</td>
</tr>
</tbody>
</table>
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
MemorySessionStore Constructor

MemorySessionStore Class  See Also  Send Feedback

Initializes a new instance of the MemorySessionStore class

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public MemorySessionStore()

Visual Basic (Declaration)

Public Sub New

Visual C++

public:
MemorySessionStore()
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The `MemorySessionStore` type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddUnused</td>
<td>We use the flyweight pattern which reuses small objects instead of creating new each time.</td>
</tr>
<tr>
<td>Cleanup</td>
<td>Remove expired sessions</td>
</tr>
<tr>
<td>Create</td>
<td>Overloaded.</td>
</tr>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td>Finalize</td>
<td>Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Serves as a hash function for a particular type.</td>
</tr>
<tr>
<td>GetType</td>
<td>Gets the Type of the current instance.</td>
</tr>
<tr>
<td>Load</td>
<td>Load an existing session.</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Creates a shallow copy of the current Object.</td>
</tr>
<tr>
<td>Save</td>
<td>Save an updated session to the store.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns a String that represents the current Object.</td>
</tr>
</tbody>
</table>
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
We use the flyweight pattern which reuses small objects instead of creating new each time.

**Namespace:**  [HttpServer.Sessions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

**C#**

```csharp
public void AddUnused(
    HttpSession session
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub AddUnused ( _
    session As HttpSession _
)
```

**Visual C++**

```cpp
public:
virtual void AddUnused(
    HttpSession^ session
) sealed
```

**Parameters**

`session`

Type: `HttpServer..::.HttpSession` EmptyLanguageNode (unused) session that should be reused next time Create is called.

**Implements**

`HttpSessionStore..::.AddUnused(HttpSession)`
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C# Visual Basic
Visual C++
C# Webserver
MemorySessionStore...:::Cleanup Method

MemorySessionStore Class  See Also  Send Feedback

Remove expired sessions

Namespace:  HttpServer.Sessions
Assembly:  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

public void Cleanup()

Visual Basic (Declaration)

Public Sub Cleanup

Visual C++

public:
virtual void Cleanup() sealed

Implements

HttpSessionStore::Cleanup()
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
C#  Visual Basic
Visual C++
Include Protected Members
Include Inherited Members
C# Webserver
MemorySessionStore...:::Create Method

MemorySessionStore Class  See Also  Send Feedback
## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Create()</code></td>
<td>Creates a new http session</td>
</tr>
<tr>
<td><code>Create(String)</code></td>
<td>Creates a new http session with a specific id</td>
</tr>
</tbody>
</table>
See Also

MemorySessionStore Class
MemorySessionStore Members
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySessionStore..::.Create Method

**Namespace:**  HttpServer.Sessions

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)

Creates a new http session
**Syntax**

**C#**

public HttpSession Create()

**Visual Basic (Declaration)**

Public Function Create As HttpSession

**Visual C++**

public:
virtual HttpSession^ Create() sealed

**Return Value**


**Implements**

HttpSessionStore...::Create()()
See Also

MemorySessionStore Class
Create Overload
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySessionStore..:::Create Method (String)

Create a new http session with a specific id

**Namespace:**  [HttpServer.Sessions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpSession Create(
    string id
)
```

Visual Basic (Declaration)

```
Public Function Create ( _
    id As String _
) As HttpSession
```

Visual C++

```cpp
public:
virtual HttpSession^ Create(
    String^ id
) sealed
```

Parameters

id

Type: `System::String`


Return Value


Implements

HttpSessionStore,,:,.Create(String)
See Also

MemorySessionStore Class
Create Overload
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Load an existing session.

**Namespace:** [HttpServer.Sessions](http://HttpServer.Sessions)  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
Syntax

C#

```csharp
public HttpSession Load(
    string sessionId
)
```

Visual Basic (Declaration)

```vbnet
Public Function Load ( _
    sessionId As String _
) As HttpSession
```

Visual C++

```cpp
public:
virtual HttpSession^ Load(
    String^ sessionId
) sealed
```

Parameters

sessionId

Type: `System::String`

[Missing <param name="sessionId"/> documentation for "M:HttpServer.Sessions.MemorySessionStore.Load(System.String)"]

Return Value


Implements

`HttpSessionStore::Load(String)`
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Save an updated session to the store.

**Namespace:**  [HttpServer.Sessions](#)  
**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
**Syntax**

**C#**

```csharp
public void Save(
    HttpSession session
)
```

**Visual Basic (Declaration)**

```vbnet
Public Sub Save ( _
    session As HttpSession _
)
```

**Visual C++**

```cpp
public:
virtual void Save(
    HttpSession^ session
) sealed
```

**Parameters**

`session`

Type: `HttpServer::HttpSession`


**Implements**

`HttpSessionStore::Save(HttpSession)`
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
The **MemorySessionStore** type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpireTime</td>
<td>Number of minutes before a session expires. Default is 20 minutes.</td>
</tr>
<tr>
<td>Item</td>
<td>Load a session from the store</td>
</tr>
</tbody>
</table>
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
MemorySessionStore...:::ExpireTime Property

**MemorySessionStore Class**  See Also  Send Feedback

Number of minutes before a session expires. Default is 20 minutes.

**Namespace:**  [HttpServer.Sessions](#)

**Assembly:**  HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
## Syntax

**C#**

```csharp
public int ExpireTime { get; set; }
```

**Visual Basic (Declaration)**

```vbnet
Public Property ExpireTime As Integer
```

**Visual C++**

```cpp
public:
virtual property int ExpireTime {
    int get () sealed;
    void set (int value) sealed;
}
```

**Implements**

[HttpPostStore::ExpireTime](#)
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB
Load a session from the store

**Namespace:** HttpServer.Sessions  
**Assembly:** HttpServer (in HttpServer.dll) Version: 1.0.0.0 (1.0.0.0)
### Syntax

#### C#

```csharp
public HttpSession this[
    string sessionId
] { get; }
```

#### Visual Basic (Declaration)

```vbnet
Public ReadOnly Default Property Item ( _
    sessionId As String _
) As HttpSession
```

#### Visual C++

```cpp
public:
    virtual property HttpSession^ default[String^ sessionId] { HttpSession^ get (String^ sessionId) sealed;
}
```

### Parameters

**sessionId**

Type: `System::String`

### Return Value

null if session is not found.

### Implements

`HttpContextStore::Item[[((String))]]`
See Also

MemorySessionStore Class
HttpServer.Sessions Namespace

This project is maintained by Gauffin Telecom AB

(c) 2008 Gauffin Telecom AB