Specifies the basic methods and properties of a Client object. This is an abstract class and must be inherited.

For a list of all members of this type, see Client Members.

System.Object Client

Remarks
The Client class provides an abstract base class that represents a connection to a local client and a remote server. Descendant classes further specify the protocol that is used between those two connections.

Requirements
Namespace: Org.Mentalis.Proxy Namespace
Assembly: Proxy.exe

See Also
Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
### Client Members

#### Public Instance Constructors

<table>
<thead>
<tr>
<th>Constructor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Client</strong></td>
<td>Overloaded. Initialize a new instance of the Client class.</td>
</tr>
</tbody>
</table>

#### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong></td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td>Starts communication with the local client.</td>
</tr>
<tr>
<td><strong>StartRelay</strong></td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns text information about this Client object.</td>
</tr>
</tbody>
</table>

#### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong></td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td><strong>RemoteBuffer</strong></td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

#### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong></td>
</tr>
<tr>
<td><strong>OnClientSent</strong></td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong></td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong></td>
</tr>
</tbody>
</table>

**Internal Instance Properties**

| **ClientSocket** | Gets or sets the Socket connection between the proxy server and the local client. |
| **DestinationSocket** | Gets or sets the Socket connection between the proxy server and the remote host. |

**Private Instance Fields**

| **Destroyer** | Holds the address of the method to call when this client is ready |
to be destroyed.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Buffer</td>
<td>Holds the value of the Buffer property.</td>
</tr>
<tr>
<td>m_ClientSocket</td>
<td>Holds the value of the ClientSocket property.</td>
</tr>
<tr>
<td>m_DestinationSocket</td>
<td>Holds the value of the DestinationSocket property.</td>
</tr>
<tr>
<td>m_RemoteBuffer</td>
<td>Holds the value of the RemoteBuffer property.</td>
</tr>
</tbody>
</table>

See Also

[Client Class](#) | [Org.Mentalis.Proxy Namespace](#)

Copyright © 2002, The KPD-Team
**Client**

Initializes a new instance of the Client class.

**Overload List**

Initializes a new instance of the Client class.

`public Client(Socket,DestroyDelegate);`

Initializes a new instance of the Client object.

`public Client();`

**See Also**

[Client Class](#) | [Client Members](#) | [Org.Mentalis.Proxy Namespace](#)

Copyright © 2002, The KPD-Team
Client Constructor (Socket, DestroyDelegate)

Initializes a new instance of the Client class.

**Visual Basic**

```vbnet
Overloads Public Sub New(
    ByVal ClientSocket As System.Net.Sockets.Socket,
    ByVal Destroyer As Org.Mentalis.Proxy.DestroyDelegate)
```

**C#**

```csharp
public Client(
    Socket ClientSocket,
    DestroyDelegate Destroyer)
```

**Parameters**

*ClientSocket*


*Destroyer*

The callback method to be called when this Client object disconnects from the local client and the remote server.

**See Also**

[Client Class](mailto:ClientClass)| [Client Members](mailto:ClientMembers) | [Org.Mentalis.Proxy Namespace](mailto:OrgMentalisProxyNamespace) | [Client Constructor Overload List](mailto:ClientConstructorOverloadList)

*Copyright © 2002, The KPD-Team*
**Client Constructor ()**

Initializes a new instance of the Client object.

**[Visual Basic]** Overloads Public Sub New()

**[C#]**

```
public Client();
```

**Remarks**

Both the ClientSocket property and the DestroyDelegate are initialized to null.

**See Also**

- [Client Class](#) | [Client Members](#) | [Org.Mentalis.Proxy Namespace](#) | [Client Constructor Overload List](#)

[Copyright © 2002, The KPD-Team](#)
Client Fields

The fields of the **Client** class are listed below. For a complete list of **Client** class members, see the [Client Members](#) topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destroyer</td>
<td>Holds the address of the method to call when this client is ready to be destroyed.</td>
</tr>
<tr>
<td>m_Buffer</td>
<td>Holds the value of the Buffer property.</td>
</tr>
<tr>
<td>m_ClientSocket</td>
<td>Holds the value of the ClientSocket property.</td>
</tr>
<tr>
<td>m_DestinationSocket</td>
<td>Holds the value of the DestinationSocket property.</td>
</tr>
<tr>
<td>m_RemoteBuffer</td>
<td>Holds the value of the RemoteBuffer property.</td>
</tr>
</tbody>
</table>

See Also

[Client Class](#) | [Org.Mentalis.Proxy Namespace](#)

*Copyright © 2002, The KPD-Team*
**Client.Destroyer Field**

Holds the address of the method to call when this client is ready to be destroyed.


[C#] `private DestroyDelegate Destroyer;`

See Also  
Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.m_Buffer Field

Holds the value of the Buffer property.

[Visual Basic] Private m_Buffer As Byte()

[C#] private byte[] m_Buffer;

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.m_ClientSocket Field

Holds the value of the ClientSocket property.

[Visual Basic] Private m_ClientSocket As System.Net.Sockets.Socket

[C#] private Socket m_ClientSocket;

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.m_DestinationSocket Field

Holds the value of the DestinationSocket property.

[Visual Basic] Private m_DestinationSocket As

[C#]
private Socket m_DestinationSocket;

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**Client.m_RemoteBuffer Field**

Holds the value of the RemoteBuffer property.

[Visual Basic] Private m_RemoteBuffer As Byte

[C#] private byte[] m_RemoteBuffer;

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client Properties

The properties of the Client class are listed below. For a complete list of Client class members, see the Client Members topic.

Protected Instance Properties

<table>
<thead>
<tr>
<th>Buffer</th>
<th>Gets the buffer to store all the incoming data from the local client.</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoteBuffer</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

Internal Instance Properties

<table>
<thead>
<tr>
<th>ClientSocket</th>
<th>Gets or sets the Socket connection between the proxy server and the local client.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DestinationSocket</td>
<td>Gets or sets the Socket connection between the proxy server and the remote host.</td>
</tr>
</tbody>
</table>

See Also

Client Class | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.Buffer Property

Gets the buffer to store all the incoming data from the local client.

[Visual Basic] Protected ReadOnly Property Buffer

[C#] protected byte[] Buffer {get;}

Property Value

An array of bytes that can be used to store all the incoming data from the local client.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace | RemoteBuffer

Copyright © 2002, The KPD-Team
Client.ClientSocket Property

Gets or sets the Socket connection between the proxy server and the local client.


[C#]
internal System.Net.Sockets.Socket ClientSocket

Property Value

A Socket instance defining the connection between the proxy server and the local client.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace | DestinationSocket

Copyright © 2002, The KPD-Team
Client.DestinationSocket Property

Gets or sets the Socket connection between the proxy server and the remote host.

[Visual Basic]Friend Property DestinationSocket

[C#]
internal System.Net.Sockets.Socket DestinationSocket

Property Value

A Socket instance defining the connection between the proxy server and the remote host.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace | ClientSocket

Copyright © 2002, The KPD-Team
Client.RemoteBuffer Property

Gets the buffer to store all the incoming data from the remote host.

[Visual Basic] Protected ReadOnly Property RemoteBuffer

[C#]
protected byte[] RemoteBuffer {get;}

Property Value

An array of bytes that can be used to store all the incoming data from the remote host.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace | Buffer

Copyright © 2002, The KPD-Team
## Client Methods

The methods of the **Client** class are listed below. For a complete list of **Client** class members, see the [Client Members](#) topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong></td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td>Starts communication with the local client.</td>
</tr>
<tr>
<td><strong>StartRelay</strong></td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns text information about this Client object.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong></td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong></td>
<td>Called when we have sent data to the local client.</td>
</tr>
</tbody>
</table>
When all the data has been sent, we will start receiving again from the remote host.

<table>
<thead>
<tr>
<th>OnRemoteReceive</th>
<th>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnRemoteSent</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

See Also

Client Class | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.Dispose Method

Disposes of the resources (other than memory) used by the Client.

**Visual Basic**

```vbnet
NotOverridable Public Sub Dispose()
```

**C#**

```csharp
public final void Dispose();
```

Remarks

Closes the connections with the local client and the remote host. Once Dispose has been called, this object should not be used anymore.

See Also

- Client Class | Client Members | Org.Mentalis.Proxy Namespace | System.IDisposable

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Client.OnClientReceive Method

Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.

[Visual Basic]
Protected Sub OnClientReceive( _
    ByVal ar As System.IAsyncResult _
)

[C#]
protected void OnClientReceive(  
    IAsyncResult ar
);

Parameters

ar
The result of the asynchronous operation.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.

### Client.OnClientSent Method

Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.

**[Visual Basic]**

```vbnet
Protected Sub OnClientSent(_
    ByVal ar As System.IAsyncResult _
)
```

**[C#]**

```csharp
protected void OnClientSent(_
    IAsyncResult ar
);
```

### Parameters

`ar`  
The result of the asynchronous operation.

### See Also

[Client Class] [Client Members] [Org.Mentalis.Proxy Namespace]

---

**Copyright © 2002, The KPD-Team**
Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.

**Parameters**

*ar*  
The result of the asynchronous operation.

**See Also**

- Client Class  |  Client Members  |  Org.Mentalis.Proxy Namespace

---

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Client.OnRemoteSent Method

Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.

[Visual Basic]
Protected Sub OnRemoteSent( _
   ByVal ar As System.IAsyncResult _
)

[C#]
protected void OnRemoteSent( _
   IAsyncResult ar _
);

Parameters

  ar
  The result of the asynchronous operation.

See Also

  Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.StartHandshake Method

Starts communication with the local client.

[Visual Basic] MustOverride Public Sub StartHandshake()

[C#]
public abstract void StartHandshake();

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
| Mentalis.org Proxy Documentation |
Client.StartRelay Method

Starts relaying data between the remote host and the local client.

[Visual Basic] Public Sub StartRelay()

[C#]
public void StartRelay();

Remarks

This method should only be called after all protocol specific communication has been finished.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Client.ToString Method

Returns text information about this Client object.

[Visual Basic] Overrides Public Function ToString()

[C#]
public override string ToString();

Return Value

A string representing this Client object.

See Also

Client Class | Client Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
DestroyDelegate Delegate

References the callback method to be called when the Client object disconnects from the local client and the remote server.


[C#] public delegate void DestroyDelegate(
    Client client
);

Parameters

client
    The Client that has closed its connections.

Requirements

Namespace: Org.Mentalis.Proxy Namespace
Assembly: Proxy.exe

See Also

Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ListenEntry Structure

Represents an item in a Listeners collection.

For a list of all members of this type, see ListenEntry Members.

System.Object   ValueType
ListenEntry

Requirements

Namespace: Org.Mentalis.Proxy Namespace
Assembly: Proxy.exe

See Also

ListenEntry Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
## ListenEntry Members

### Public Instance Fields

<table>
<thead>
<tr>
<th>guid</th>
<th>The Listener's ID. It must be unique throughout the Listeners collection.</th>
</tr>
</thead>
<tbody>
<tr>
<td>listener</td>
<td>The Listener object.</td>
</tr>
</tbody>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</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><strong>GetHashCode</strong> (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### See Also

- [ListenEntry Class](#) | [Org.Mentalis.Proxy Namespace](#)
Mentalis.org Proxy Documentation
The fields of the **ListenEntry** class are listed below. For a complete list of **ListenEntry** class members, see the **ListenEntry Members** topic.

### Public Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>guid</td>
<td>The Listener's ID. It must be unique throughout the Listeners collection.</td>
</tr>
<tr>
<td>listener</td>
<td>The Listener object.</td>
</tr>
</tbody>
</table>

**See Also**


*Copyright © 2002, The KPD-Team*
ListenEntry.guid Field

The Listener's ID. It must be unique throughout the Listeners collection.

[Visual Basic] Public guid As System.Guid

[C#] public Guid guid;

See Also

ListenEntry Class | ListenEntry Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
### ListenEntry.listener Field

The Listener object.

[Visual Basic]
```vbnet
Public listener As Org.Mentalis.Proxy.Listener
```

[C#]
```csharp
public Listener listener;
```

See Also

- ListenEntry Class
- ListenEntry Members
- Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ListenEntry Methods

The methods of the ListenEntry class are listed below. For a complete list of ListenEntry class members, see the ListenEntry Members topic.

Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Determines whether the specified Object is equal to the current Object.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>ToString</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finalize (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>MemberwiseClone (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

See Also

ListenEntry Class | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**ListenEntry.Equals Method**

Determines whether the specified Object is equal to the current Object.

**[Visual Basic]**

```vbnet
Overloads Public Function Equals(ByVal obj As Object) As Boolean
```

**[C#]**

```csharp
public override bool Equals(
    object obj
);
```

**Parameters**

- **obj**
  
  The Object to compare with the current Object.

**Return Value**

- True if the specified Object is equal to the current Object; otherwise, false.

**See Also**

- ListenEntry Class | ListenEntry Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener Class

Specifies the basic methods and properties of a Listener object. This is an abstract class and must be inherited.

For a list of all members of this type, see Listener Members.

System.Object Listener

[Visual Basic]
MustInherit Public Class Listener
  Implements IDisposable

[C#]
public abstract class Listener : IDisposable

Remarks

The Listener class provides an abstract base class that represents a listening socket of the proxy server. Descendant classes further specify the protocol that is used between those two connections.

Requirements

Namespace: Org.Mentalis.Proxy Namespace
Assembly: Proxy.exe

See Also

Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
# Listener Members

## Public Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetLocalExternalIP</td>
<td>Returns an external IP address of this computer, if present.</td>
</tr>
<tr>
<td>GetLocalInternalIP</td>
<td>Returns an internal IP address of this computer, if present.</td>
</tr>
</tbody>
</table>

## Protected Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsLocalIP</td>
<td>Checks whether the specified IP address is a local IP address or not.</td>
</tr>
<tr>
<td>IsRemoteIP</td>
<td>Checks whether the specified IP address is a remote IP address or not.</td>
</tr>
</tbody>
</table>

## Public Instance Constructors

<table>
<thead>
<tr>
<th>Constructor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Listener Constructor</td>
<td>Initializes a new instance of the Listener class.</td>
</tr>
</tbody>
</table>

## Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConstructString</td>
<td>Returns a string that holds all the construction information for this object.</td>
</tr>
<tr>
<td>IsDisposed</td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td>Listening</td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose</td>
<td>Disposes of the resources (other...</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong></td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong></td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming client connection waiting to be accepted.</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

**Protected Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong></td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong></td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

**Protected Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong></td>
<td>Adds the specified Client to the</td>
</tr>
</tbody>
</table>
client list.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong></td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong></td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong></td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_Address</strong></td>
<td>Holds the value of the Address property.</td>
</tr>
<tr>
<td><strong>m_Clients</strong></td>
<td>Holds the value of the Clients property.</td>
</tr>
<tr>
<td><strong>m_IsDisposed</strong></td>
<td>Holds the value of the IsDisposed property.</td>
</tr>
<tr>
<td><strong>m_ListenSocket</strong></td>
<td>Holds the value of the ListenSocket property.</td>
</tr>
<tr>
<td><strong>m_Port</strong></td>
<td>Holds the value of the Port property.</td>
</tr>
</tbody>
</table>

### See Also

- [Listener Class](#) | [Org.Mentalis.Proxy Namespace](#)

*Copyright © 2002, The KPD-Team*
Listener Constructor

Initializes a new instance of the Listener class.

[Visual Basic]
Public Sub New( _
    ByVal Port As Integer, _
    ByVal Address As System.Net.IPAddress _
)

[C#
public Listener(
    int Port,
    IPAddress Address
);

Parameters

Port
The port to listen on.

Address
The address to listen on. You can specify IPAddress.Any to listen on all installed network cards.

Remarks

For the security of your server, try to avoid to listen on every network card (IPAddress.Any). Listening on a local IP address is usually sufficient and much more secure.

See Also

Listener Class  |  Listener Members  |  Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener Fields

The fields of the **Listener** class are listed below. For a complete list of **Listener** class members, see the **Listener Members** topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Address</td>
<td>Holds the value of the Address property.</td>
</tr>
<tr>
<td>m_Clients</td>
<td>Holds the value of the Clients property.</td>
</tr>
<tr>
<td>m_IsDisposed</td>
<td>Holds the value of the IsDisposed property.</td>
</tr>
<tr>
<td>m_ListenSocket</td>
<td>Holds the value of the ListenSocket property.</td>
</tr>
<tr>
<td>m_Port</td>
<td>Holds the value of the Port property.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
Listener.m_Address Field

Holds the value of the Address property.

[Visual Basic] Private m_Address As System.Net.IPAddress

[C#]
private IPAddress m_Address;

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.m_Clients Field

Holds the value of the Clients property.

[Visual Basic] Private m_Clients As System.Collections.ArrayList

[C#] private ArrayList m_Clients;

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Listener.m_IsDisposed Field

Holds the value of the IsDisposed property.

[Visual Basic] Private m_IsDisposed As Boolean

[C#] private bool m_IsDisposed;

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.m_ListenSocket Field

Holds the value of the ListenSocket property.

[Visual Basic] Private m_ListenSocket As System.Net.Sockets.Socket

[C#] private Socket m_ListenSocket;

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.m_Port Field

Holds the value of the Port property.

[Visual Basic] Private m_Port As Integer

[C#] private int m_Port;

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener Properties

The properties of the Listener class are listed below. For a complete list of Listener class members, see the Listener Members topic.

Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConstructString</td>
<td>Returns a string that holds all the construction information for this object.</td>
</tr>
<tr>
<td>IsDisposed</td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td>Listening</td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td>Clients</td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td>ListenSocket</td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td>Port</td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

See Also

Listener Class | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.Address Property

Gets or sets the address on which to listen on.

[Visual Basic] Protected Property Address As System.Net.IPAddress

[C#]
protected System.Net.IPAddress Address {get;}

Property Value

An IPAddress instance defining the IP address to listen on.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also

[Listener Class] | [Listener Members] | [Org.Mentalis.Proxy Namespace] | [Port]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Listener.Clients Property

Gets the list of connected clients.

[Visual Basic] Protected ReadOnly Property Clients

[C#] protected System.Collections.ArrayList Clients {get;}

Property Value
An instance of the ArrayList class that's used to store all the connections.

See Also
Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**Listener.ConstructString Property**

Returns a string that holds all the construction information for this object.

**[Visual Basic]**

```vbnet
MustOverride Public ReadOnly Property ConstructString As String
```

**[C#]**

```csharp
public string ConstructString {abstract get;}
```

**Property Value**

A string that holds all the construction information for this object.

**See Also**


Copyright © 2002, The KPD-Team
**Listener.IsDisposed Property**

Gets a value indicating whether the Listener has been disposed or not.

**[Visual Basic]**

```vbnet
Public ReadOnly Property IsDisposed As Boolean
```

**[C#]**

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

**Property Value**

An boolean that specifies whether the object has been disposed or not.

**See Also**

[Listener Class] [Listener Members] [Org.Mentalis.Proxy Namespace]
Listener.Listening Property

Gets a value indicating whether the Listener is currently listening or not.

[Visual Basic] Public ReadOnly Property Listening As Boolean

[C#] public bool Listening {get;}

Property Value

A boolean that indicates whether the Listener is currently listening or not.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Listener.ListenSocket Property

Gets or sets the listening Socket.

[Visual Basic] Protected Property ListenSocket

[C#] protected System.Net.Sockets.Socket ListenSocket

Property Value

An instance of the Socket class that's used to listen for incoming connections.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SystemArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.Port Property

Gets or sets the port number on which to listen on.

[Visual Basic] Protected Property Port As Integer

[C#]
protected int Port {get; set;}

Property Value

An integer defining the port number to listen on.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentException</td>
<td>The specified value is less than or equal to zero.</td>
</tr>
</tbody>
</table>

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace | Address

Copyright © 2002, The KPD-Team
The methods of the **Listener** class are listed below. For a complete list of **Listener** class members, see the **Listener Members** topic.

### Public Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GetLocalExternalIP</strong></td>
<td>Returns an external IP address of this computer, if present.</td>
</tr>
<tr>
<td><strong>GetLocalInternalIP</strong></td>
<td>Returns an internal IP address of this computer, if present.</td>
</tr>
</tbody>
</table>

### Protected Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IsLocalIP</strong></td>
<td>Checks whether the specified IP address is a local IP address or not.</td>
</tr>
<tr>
<td><strong>IsRemoteIP</strong></td>
<td>Checks whether the specified IP address is a remote IP address or not.</td>
</tr>
</tbody>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong></td>
<td>Disposes of the resources (other than memory) used by the Listener.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong></td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong></td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>
**OnAccept**  
Called when there's an incoming client connection waiting to be accepted.

**Start**  
Starts listening on the selected IP address and port.

**ToString**  
Returns a string representation of this object.

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong></td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong></td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong></td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

**See Also**

- [Listener Class](#) | [Org.Mentalis.Proxy Namespace](#)

*Copyright © 2002, The KPD-Team*
**Listener.AddClient Method**

Adds the specified Client to the client list.

*Visual Basic*

```vbnet
Protected Sub AddClient( _
    ByVal client As Org.Mentalis.Proxy.Client
)
```

*C#*

```csharp
protected void AddClient(  
    Client client
);
```

**Parameters**

*client*

The client to add to the client list.

**Remarks**

A client will never be added twice to the list.

**See Also**


*Copyright © 2002, The KPD-Team*
Listener.Dispose Method

Disposes of the resources (other than memory) used by the Listener.

[Visual Basic] NotOverridable Public Sub Dispose()

[C#]
public final void Dispose();

Remarks

Stops listening and disposes all the client objects. Once disposed, this object should not be used anymore.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace | System.IDisposable

Copyright © 2002, The KPD-Team
Listener.Finalize Method

Finalizes the Listener.

[Visual Basic] Overrides Protected Sub Finalize()

[C#]
protected override void Finalize();

Remarks
The destructor calls the Dispose method.

See Also
Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.GetClientAt Method

Returns the requested client from the client list.

[Visual Basic] Public Function GetClientAt( ByVal Index As Integer ) As Org.Mentalis.Proxy.Client

[C#]
public Client GetClientAt( int Index );

Parameters

Index
The index of the requested client.

Return Value
The requested client.

Remarks
If the specified index is invalid, the GetClientAt method returns null.

See Also
Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.GetClientCount Method

Returns the number of clients in the client list.

[Visual Basic] Public Function GetClientCount()

[C#] public int GetClientCount();

Return Value

The number of connected clients.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Listener.GetLocalExternalIP Method

Returns an external IP address of this computer, if present.

[Visual Basic] Public Shared Function GetLocalExternalIP()

[C#] public static IPAddress GetLocalExternalIP();

Return Value

Returns an external IP address of this computer; if this computer does not have an external IP address, it returns the first local IP address it can find.

Remarks

If this computer does not have any configured IP address, this method returns the IP address 0.0.0.0.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Listener.GetLocalInternalIP Method

Returns an internal IP address of this computer, if present.

[Visual Basic] Public Shared Function GetLocalInternalIP()

[C#] public static IPAddress GetLocalInternalIP();

Return Value

Returns an internal IP address of this computer; if this computer does not have an internal IP address, it returns the first local IP address it can find.

Remarks

If this computer does not have any configured IP address, this method returns the IP address 0.0.0.0.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.IsLocalIP Method

Checks whether the specified IP address is a local IP address or not.

[Visual Basic]
Protected Shared Function IsLocalIP( ByVal IP As System.Net.IPAddress ) As Boolean

[C#]
protected static bool IsLocalIP( IPAddress IP );

Parameters

IP

The IP address to check.

Return Value

True if the specified IP address is a local address, false otherwise.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.IsRemoteIP Method

Checks whether the specified IP address is a remote IP address or not.

[Visual Basic]
Protected Shared Function IsRemoteIP(ByVal IP As System.Net.IPAddress) As Boolean

[C#]
protected static bool IsRemoteIP(IPAddress IP);

Parameters

IP
The IP address to check.

Return Value

True if the specified IP address is a remote address, false otherwise.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.OnAccept Method

Called when there's an incoming client connection waiting to be accepted.

[Visual Basic]
MustOverride Public Sub OnAccept(ByVal ar As System.IAsyncResult _
)

[C#]
public abstract void OnAccept(IAsyncResult ar);

Parameters

ar
The result of the asynchronous operation.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.RemoveClient Method

Removes the specified Client from the client list.

[Visual Basic]
Protected Sub RemoveClient(_
    ByVal client As Org.Mentalis.Proxy.Client
)

[C#]
protected void RemoveClient(
    Client client
);

Parameters

client
The client to remove from the client list.

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.Restart Method

Retarts listening on the selected IP address and port.

[Visual Basic] Protected Sub Restart()

[C#] protected void Restart();

Remarks

This method is automatically called when the listening port or the listening IP address are changed.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.Net.Sockets.SocketException</td>
<td>There was an error while creating the listening socket.</td>
</tr>
</tbody>
</table>

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.Start Method

Starts listening on the selected IP address and port.

[Visual Basic]
Public Sub Start()

[C#]
public void Start();

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.Net.Sockets.SocketException</td>
<td>There was an error while creating the listening socket.</td>
</tr>
</tbody>
</table>

See Also

Listener Class | Listener Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Listener.ToString Method

Returns a string representation of this object.

[Visual Basic] MustOverride Public Function ToString() As String

[C#]
public abstract string ToString();

Return Value

A string with information about this object.

See Also
  
  Listener Class  |  Listener Members  |  Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy Class

Defines the class that controls the settings and listener objects. For a list of all members of this type, see Proxy Members.

System.Object Proxy

[Visual Basic]
Public Class Proxy

[C#]
public class Proxy

Requirements

Namespace: Org.Mentalis.Proxy Namespace
Assembly: Proxy.exe

See Also

Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
## Proxy Members

### Public Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main</strong></td>
<td>Entry point of the application.</td>
</tr>
</tbody>
</table>

### Public Instance Constructors

<table>
<thead>
<tr>
<th>Constructor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Proxy Constructor</strong></td>
<td>Initializes a new Proxy instance.</td>
</tr>
</tbody>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddListener</strong></td>
<td>Adds a listener to the Listeners list.</td>
</tr>
<tr>
<td><strong>CreateListener</strong></td>
<td>Creates a new Listener object from a given listener name and a given listener parameter string.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Starts a new Proxy server by reading the data from the configuration file and start listening on the specified ports.</td>
</tr>
<tr>
<td><strong>Stop</strong>*</td>
<td>Stops the proxy server.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Config</strong></td>
<td>Gets or sets the configuration object for this Proxy server.</td>
</tr>
<tr>
<td><strong>Listeners</strong></td>
<td>Gets the collection that contains</td>
</tr>
<tr>
<td>all the Listener objects.</td>
<td>all the Listener objects.</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td><strong>StartTime</strong></td>
<td>Gets or sets the date when this Proxy server was first started.</td>
</tr>
</tbody>
</table>

**Protected Instance Methods**

<table>
<thead>
<tr>
<th><strong>Finalize</strong> (inherited from <strong>Object</strong>)</th>
<th>Select the method name to go to the Microsoft documentation.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ShowAddListener</strong></td>
<td>Asks the user which listener to add.</td>
</tr>
<tr>
<td><strong>ShowAddUser</strong></td>
<td>Asks the user which username to add.</td>
</tr>
<tr>
<td><strong>ShowDelListener</strong></td>
<td>Asks the user which listener to delete.</td>
</tr>
<tr>
<td><strong>ShowDelUser</strong></td>
<td>Asks the user which username to delete.</td>
</tr>
<tr>
<td><strong>ShowHelp</strong></td>
<td>Shows a list of commands in the console.</td>
</tr>
<tr>
<td><strong>ShowListeners</strong></td>
<td>Shows the Listeners list.</td>
</tr>
<tr>
<td><strong>ShowUpTime</strong></td>
<td>Shows the uptime of this proxy server.</td>
</tr>
<tr>
<td><strong>ShowUsers</strong></td>
<td>Shows a list of usernames in the console.</td>
</tr>
<tr>
<td><strong>ShowVersion</strong></td>
<td>Shows the version number of this proxy server.</td>
</tr>
</tbody>
</table>

**Internal Instance Properties**

<table>
<thead>
<tr>
<th><strong>Item</strong></th>
<th>Gets the Listener object at the specified position.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ListenerCount</strong></td>
<td>Gets the number of Listener objects.</td>
</tr>
</tbody>
</table>
Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Config</td>
<td>Holds the value of the Config property.</td>
</tr>
<tr>
<td>m_Listeners</td>
<td>Holds the value of the Listeners property.</td>
</tr>
<tr>
<td>m_StartTime</td>
<td>Holds the value of the StartTime property.</td>
</tr>
</tbody>
</table>

See Also

Proxy Class | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy Constructor

Initializes a new Proxy instance.

[Visual Basic]Public Sub New( _
    ByVal file As String _
)

[C#]
public Proxy(
    string file
);

Parameters

file
    The XML configuration file to use.

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
The fields of the **Proxy** class are listed below. For a complete list of **Proxy** class members, see the [Proxy Members](#) topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Config</td>
<td>Holds the value of the Config property.</td>
</tr>
<tr>
<td>m_Listeners</td>
<td>Holds the value of the Listeners property.</td>
</tr>
<tr>
<td>m_StartTime</td>
<td>Holds the value of the StartTime property.</td>
</tr>
</tbody>
</table>

### See Also

- [Proxy Class](#) | [Org.Mentalis.Proxy Namespace](#)

Copyright © 2002, The KPD-Team
Proxy.m_Config Field

Holds the value of the Config property.


[C#]
private ProxyConfig m_Config;

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy.m_Listeners Field

Holds the value of the Listeners property.

[Visual Basic] Private m_Listeners As System.Collections.ArrayList

[C#] private ArrayList m_Listeners;

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.m_StartTime Field

Holds the value of the StartTime property.

[Visual Basic] Private m_StartTime As Date

[C#]
private DateTime m_StartTime;

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**Proxy Properties**

The properties of the **Proxy** class are listed below. For a complete list of **Proxy** class members, see the **Proxy Members** topic.

**Protected Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Config</strong></td>
<td>Gets or sets the configuration object for this Proxy server.</td>
</tr>
<tr>
<td><strong>Listeners</strong></td>
<td>Gets the collection that contains all the Listener objects.</td>
</tr>
<tr>
<td><strong>StartTime</strong></td>
<td>Gets or sets the date when this Proxy server was first started.</td>
</tr>
</tbody>
</table>

**Internal Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Item</strong></td>
<td>Gets the Listener object at the specified position.</td>
</tr>
<tr>
<td><strong>ListenerCount</strong></td>
<td>Gets the number of Listener objects.</td>
</tr>
</tbody>
</table>

**See Also**


**Copyright © 2002, The KPD-Team**
**Proxy.Config Property**

Gets or sets the configuration object for this Proxy server.

[Visual Basic]
```vbnet
Protected Property Config As Org.Mentalis.Proxy.ProxyConfig
```

[C#]
```csharp
protected ProxyConfig Config {get; set;}
```

**Property Value**

A ProxyConfig instance that represents the configuration object for this Proxy server.

**See Also**

[Proxy Class] | [Proxy Members] | [Org.Mentalis.Proxy Namespace]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.Item Property

Gets the Listener object at the specified position.

[Visual Basic] Overridable Friend Default ReadOnly Property Item(ByVal index As Integer) As Org.Mentalis.Proxy.Listener

[C#]
internal Listener this[int index]
] {virtual get;}

Property Value

The Listener instance at position index.

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**Proxy.ListenerCount Property**

Gets the number of Listener objects.

**[Visual Basic]**

```vbnet
Friend ReadOnly Property ListenerCount
```

**[C#]**

```csharp
internal int ListenerCount {get;}
```

**Property Value**

An integer specifying the number of Listener objects.

**See Also**


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.Listeners Property

Gets the collection that contains all the Listener objects.

[Visual Basic]
Protected ReadOnly Property Listeners As ArrayList

[C#]
protected System.Collections.ArrayList Listeners

Property Value

An ArrayList object that contains all the Listener objects.

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**Proxy.StartTime Property**

Gets or sets the date when this Proxy server was first started.

[Visual Basic]

```vbnet
Protected Property StartTime As Date
```

[C#]

```csharp
protected System.DateTime StartTime {get; set;}
```

**Property Value**

A DateTime structure that indicates when this Proxy server was first started.

**See Also**

[Proxy Class] | [Proxy Members] | [Org.Mentalis.Proxy Namespace]

[Copyright © 2002, The KPD-Team]
# Proxy Methods

The methods of the **Proxy** class are listed below. For a complete list of **Proxy** class members, see the [Proxy Members](#) topic.

## Public Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main</strong></td>
<td>Entry point of the application.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddListener</strong></td>
<td>Adds a listener to the Listeners list.</td>
</tr>
<tr>
<td><strong>CreateListener</strong></td>
<td>Creates a new Listener object from a given listener name and a given listener parameter string.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Starts a new Proxy server by reading the data from the configuration file and start listening on the specified ports.</td>
</tr>
<tr>
<td><strong>Stop</strong></td>
<td>Stops the proxy server.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>ShowAddListener</td>
<td>Asks the user which listener to add.</td>
</tr>
<tr>
<td>ShowAddUser</td>
<td>Asks the user which username to add.</td>
</tr>
<tr>
<td>ShowDelListener</td>
<td>Asks the user which listener to delete.</td>
</tr>
<tr>
<td>ShowDelUser</td>
<td>Asks the user which username to delete.</td>
</tr>
<tr>
<td>ShowHelp</td>
<td>Shows a list of commands in the console.</td>
</tr>
<tr>
<td>ShowListeners</td>
<td>Shows the Listeners list.</td>
</tr>
<tr>
<td>ShowUpTime</td>
<td>Shows the uptime of this proxy server.</td>
</tr>
<tr>
<td>ShowUsers</td>
<td>Shows a list of usernames in the console.</td>
</tr>
<tr>
<td>ShowVersion</td>
<td>Shows the version number of this proxy server.</td>
</tr>
</tbody>
</table>

See Also

Proxy Class | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy.AddListener Method

Adds a listener to the Listeners list.

[Visual Basic] Public Sub AddListener( ByVal newItem As Org.Mentalis.Proxy.Listener )

[C#] public void AddListener( Listener newItem );

Parameters

newItem
The new Listener to add.

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy.CreateListener Method

Creates a new Listener object from a given listener name and a given listener parameter string.

**[Visual Basic]**

```vbnet
Public Function CreateListener(
    ByVal type As String, _
    ByVal cpars As String _
) As Org.Mentalis.Proxy.Listener
```

**[C#]**

```csharp
public Listener CreateListener(
    string type,  
    string cpars
);
```

**Parameters**

- `type`
- `cpars`

**Return Value**

**See Also**


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
## Proxy.Main Method

Entry point of the application.

<table>
<thead>
<tr>
<th>Visual Basic</th>
<th>Public Shared Sub Main()</th>
</tr>
</thead>
</table>

| C# | `public static void Main();` |

See Also

[Proxy Class]( #) | [Proxy Members]( #) | [Org.Mentalis.Proxy Namespace]( #)

Copyright © 2002, The KPD-Team
Proxy.ShowAddListener Method

Asks the user which listener to add.

[Visual Basic] Protected Sub ShowAddListener()

[C#] protected void ShowAddListener();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy.ShowAddUser Method

Asks the user which username to add.

[Visual Basic]
Protected Sub ShowAddUser()

[C#]
protected void ShowAddUser();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.ShowDelListener Method

Asks the user which listener to delete.

[Visual Basic] Protected Sub ShowDelListener()

[C#] protected void ShowDelListener();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy.ShowDelUser Method

Asks the user which username to delete.

[Visual Basic]
Protected Sub ShowDelUser()

[C#]
protected void ShowDelUser();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Proxy.ShowHelp Method

Shows a list of commands in the console.

[Visual Basic] Protected Sub ShowHelp()

[C#] protected void ShowHelp();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
| Mentalis.org Proxy Documentation |
Proxy.ShowListeners Method

Shows the Listeners list.

[Visual Basic] Protected Sub ShowListeners()

[C#] protected void ShowListeners();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
**Proxy.ShowUpTime Method**

Shows the uptime of this proxy server.

**[Visual Basic]**

```vbnet
Protected Sub ShowUpTime()
```

**[C#]**

```csharp
protected void ShowUpTime();
```

See Also

- [Proxy Class](#)
- [Proxy Members](#)
- [Org.Mentalis.Proxy Namespace](#)

*Copyright © 2002, The KPD-Team*
Proxy.ShowUsers Method

Shows a list of usernames in the console.

[Visual Basic] Protected Sub ShowUsers()

[C#]
protected void ShowUsers();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.ShowVersion Method

Shows the version number of this proxy server.

[Visual Basic] Protected Sub ShowVersion()

[C#] protected void ShowVersion();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.Start Method

Starts a new Proxy server by reading the data from the configuration file and start listening on the specified ports.

[Visual Basic] Public Sub Start()

[C#]
public void Start();

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Proxy.Stop Method

Stops the proxy server.

[Visual Basic]

Public Sub Stop()

[C#]

public void Stop();

Remarks

When this method is called, all listener and client objects will be disposed.

See Also

Proxy Class | Proxy Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Stores the configuration settings of this proxy server.
For a list of all members of this type, see ProxyConfig Members.

System.Object  ProxyConfig

Requirements
Namespace: Org.Mentalis.Proxy Namespace
Assembly: Proxy.exe

See Also
ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
# ProxyConfig Members

## Public Instance Constructors

<table>
<thead>
<tr>
<th>Constructor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ProxyConfig Constructor</strong></td>
<td>Initializes a new ProxyConfig instance.</td>
</tr>
</tbody>
</table>

## Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td>Gets the full path to the XML data file.</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Gets the parent object of this ProxyConfig class.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>LoadData</strong></td>
<td>Loads the data from an XML file.</td>
</tr>
<tr>
<td><strong>ReadBytes</strong></td>
<td>Reads a byte array from the settings section.</td>
</tr>
<tr>
<td><strong>ReadInt</strong></td>
<td>Overloaded. Reads an integer from the settings section.</td>
</tr>
<tr>
<td><strong>ReadString</strong></td>
<td>Overloaded. Reads a string from the settings section.</td>
</tr>
<tr>
<td><strong>RemoveUser</strong></td>
<td>Overloaded. Removes a user from the authentication list.</td>
</tr>
<tr>
<td><strong>SaveData</strong></td>
<td>Saves the data in this class to an XML file.</td>
</tr>
<tr>
<td><strong>SaveSetting</strong></td>
<td>Overloaded. Saves a string to the settings section.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>SaveUserHash</strong></td>
<td>Overloaded. Saves a username and password hash combination to the authentication list.</td>
</tr>
<tr>
<td><strong>SaveUserPass</strong></td>
<td>Overloaded. Saves a username and password combination to the authentication list.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

**Protected Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

**Internal Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UserList</strong></td>
<td>Gets the userlist.</td>
</tr>
</tbody>
</table>

**Private Instance Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_File</strong></td>
<td>Holds the value of the File property.</td>
</tr>
<tr>
<td><strong>m_Parent</strong></td>
<td>Holds the value of the Parent property.</td>
</tr>
<tr>
<td><strong>m_Settings</strong></td>
<td>Holds the value of the Settings property.</td>
</tr>
<tr>
<td><strong>m_UserList</strong></td>
<td>Holds the value of the UserList property.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Settings</strong></td>
<td>Gets the dictionary that holds the settings.</td>
</tr>
<tr>
<td><strong>LoadListeners</strong></td>
<td>Loads the listeners list from an XML file.</td>
</tr>
<tr>
<td><strong>LoadSettings</strong></td>
<td>Loads the settings from an XML file.</td>
</tr>
<tr>
<td><strong>LoadUsers</strong></td>
<td>Loads the userlist from an XML file.</td>
</tr>
<tr>
<td><strong>SaveListeners</strong></td>
<td>Saves the listeners to an XML writer.</td>
</tr>
<tr>
<td><strong>SaveSettings</strong></td>
<td>Saves the settings in this class to an XML writer.</td>
</tr>
<tr>
<td><strong>SaveUsers</strong></td>
<td>Saves the authentication list to an XML writer.</td>
</tr>
</tbody>
</table>

**See Also**

- [ProxyConfig Class](#) | [Org.Mentalis.Proxy Namespace](#)

*Copyright © 2002, The KPD-Team*
ProxyConfig Constructor

Initializes a new ProxyConfig instance.

**[Visual Basic]**
```vbnet
Public Sub New( 
    ByVal parent As Org.Mentalis.Proxy.Proxy, 
    ByVal file As String 
)
```

**[C#]**
```csharp
public ProxyConfig( 
    Proxy parent, 
    string file 
);
```

**Parameters**

- **parent**
  The parent of this ProxyConfig instance.

- **file**
  The XML file to read data from and store data to.

**Exceptions**

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>file is null -or- parent is null.</td>
</tr>
</tbody>
</table>

**See Also**

- [ProxyConfig Class](#) | [ProxyConfig Members](#) | [Org.Mentalis.Proxy Namespace](#)

**Copyright © 2002, The KPD-Team**
The fields of the **ProxyConfig** class are listed below. For a complete list of **ProxyConfig** class members, see the [ProxyConfig Members](#) topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_File</td>
<td>Holds the value of the File property.</td>
</tr>
<tr>
<td>m_Parent</td>
<td>Holds the value of the Parent property.</td>
</tr>
<tr>
<td>m_Settings</td>
<td>Holds the value of the Settings property.</td>
</tr>
<tr>
<td>m_UserList</td>
<td>Holds the value of the UserList property.</td>
</tr>
</tbody>
</table>

See Also


Copyright © 2002, The KPD-Team
**ProxyConfig.m_File Field**

Holds the value of the File property.

[Visual Basic] *Private m_File As String*

[C#] *private string m_File;*

See Also

[ProxyConfig Class] [ProxyConfig Members] [Org.Mentalis.Proxy Namespace]

[Copyright © 2002, The KPD-Team]
ProxyConfig.m_Parent Field

Holds the value of the Parent property.

[Visual Basic] Private m_Parent As Org.Mentalis.Proxy

[C#] private Proxy m_Parent;

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ProxyConfig.m_Settings Field

Holds the value of the Settings property.

[Visual Basic] Private m_Settings As System.Collections.Specialized.StringDictionary

[C#]
private StringDictionary m_Settings;

See Also
ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ProxyConfig.m_UserList Field

Holds the value of the UserList property.


[C#]
private AuthenticationList m_UserList;

See Also


Copyright © 2002, The KPD-Team
The properties of the **ProxyConfig** class are listed below. For a complete list of **ProxyConfig** class members, see the **ProxyConfig Members** topic.

### Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td>Gets the full path to the XML data file.</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Gets the parent object of this ProxyConfig class.</td>
</tr>
</tbody>
</table>

### Internal Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UserList</strong></td>
<td>Gets the userlist.</td>
</tr>
</tbody>
</table>

### Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Settings</strong></td>
<td>Gets the dictionary that holds the settings.</td>
</tr>
</tbody>
</table>

### See Also

[ProxyConfig Class] | [Org.Mentalis.Proxy Namespace]

Copyright © 2002, The KPD-Team
ProxyConfig.File Property

Gets the full path to the XML data file.

[Visual Basic] Public ReadOnly Property File As String

[C#] public string File {get;}

Property Value

A String that holds the full path to the XML data file.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ProxyConfig.Parent Property

Gets the parent object of this ProxyConfig class.

[Visual Basic] Public ReadOnly Property Parent

[C#] public Proxy Parent {get;}

Property Value

An instance of the Proxy class.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Gets the dictionary that holds the settings.

[Visual Basic] Private ReadOnly Property Settings As

[C#] private System.Collections.Specialized.StringDictionary

Property Value

An instance of the StringDictionary class that holds the settings.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.UserList Property

Gets the userlist.

[Visual Basic] Friend ReadOnly Property UserList:

[C#] internal Socks.Authentication.AuthenticationList

Property Value
An instance of the AuthenticationList class that holds all the users and their corresponding password hashes.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
The methods of the **ProxyConfig** class are listed below. For a complete list of **ProxyConfig** class members, see the **ProxyConfig Members** topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>LoadData</strong></td>
<td>Loads the data from an XML file.</td>
</tr>
<tr>
<td><strong>ReadBytes</strong></td>
<td>Reads a byte array from the settings section.</td>
</tr>
<tr>
<td><strong>ReadInt</strong></td>
<td>Overloaded. Reads an integer from the settings section.</td>
</tr>
<tr>
<td><strong>ReadString</strong></td>
<td>Overloaded. Reads a string from the settings section.</td>
</tr>
<tr>
<td><strong>RemoveUser</strong></td>
<td>Overloaded. Removes a user from the authentication list.</td>
</tr>
<tr>
<td><strong>SaveData</strong></td>
<td>Saves the data in this class to an XML file.</td>
</tr>
<tr>
<td><strong>SaveSetting</strong></td>
<td>Overloaded. Saves a string to the settings section.</td>
</tr>
<tr>
<td><strong>SaveUserHash</strong></td>
<td>Overloaded. Saves a username and password hash combination to the authentication list.</td>
</tr>
<tr>
<td><strong>SaveUserPass</strong></td>
<td>Overloaded. Saves a username and password combination to the authentication list.</td>
</tr>
<tr>
<td>Method Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

Protected Instance Methods

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

Private Instance Methods

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LoadListeners</strong></td>
<td>Loads the listeners list from an XML file.</td>
</tr>
<tr>
<td><strong>LoadSettings</strong></td>
<td>Loads the settings from an XML file.</td>
</tr>
<tr>
<td><strong>LoadUsers</strong></td>
<td>Loads the userlist from an XML file.</td>
</tr>
<tr>
<td><strong>SaveListeners</strong></td>
<td>Saves the listeners to an XML writer.</td>
</tr>
<tr>
<td><strong>SaveSettings</strong></td>
<td>Saves the settings in this class to an XML writer.</td>
</tr>
<tr>
<td><strong>SaveUsers</strong></td>
<td>Saves the authentication list to an XML writer.</td>
</tr>
</tbody>
</table>

See Also

- [ProxyConfig Class](#)  | [Org.Mentalis.Proxy Namespace](#) |

Copyright © 2002, The KPD-Team
ProxyConfig.LoadData Method

Loads the data from an XML file.

[Visual Basic] Public Sub LoadData()

[C#] public void LoadData();

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ProxyConfig.LoadListeners Method

Loads the listeners list from an XML file.

[Visual Basic] Private Sub LoadListeners( ByVal reader As System.Xml.XmlTextReader )

[C#] private void LoadListeners( XmlTextReader reader );

Parameters

reader
The XML reader to read the users from.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
## ProxyConfig.LoadSettings Method

Loads the settings from an XML file.

### [Visual Basic]

```vbnet
Private Sub LoadSettings(_
    ByVal reader As System.Xml.XmlTextReader)
```

### [C#]

```csharp
private void LoadSettings(_
    XmlTextReader reader)
```

### Parameters

**reader**

The XML reader to read the settings from.

### See Also

- [ProxyConfig Class](#) | [ProxyConfig Members](#) | [Org.Mentalis.Proxy Namespace](#)

**Copyright © 2002, The KPD-Team**
ProxyConfig.LoadUsers Method

Loads the userlist from an XML file.

[Visual Basic] Private Sub LoadUsers(ByVal reader As System.Xml.XmlTextReader)

[C#]
private void LoadUsers(XmTextReader reader);

Parameters

reader
The XML reader to read the users from.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.ReadByte Method

Reads a byte array from the settings section.

**Parameters**

*name*

The key to read from.

**Return Value**

The array of bytes that corresponds with the specified key, or `null` if the specified key was not found in the collection.

**See Also**

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.ReadInt

Reads an integer from the settings section.

Overload List

Reads an integer from the settings section.

```csharp
public int ReadInt(string);
```

Reads an integer from the settings section.

```csharp
public int ReadInt(string, int);
```

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.ReadInt Method (String)

Reads an integer from the settings section.

**[Visual Basic] Overloads Public Function ReadInt**

```vbnet
ByVal name As String _
) As Integer
```

**[C#]**

```csharp
public int ReadInt(
    string name
);
```

**Parameters**

`name`

The key to read from.

**Return Value**

The integer that corresponds with the specified key, or 0 if the specified key was not found in the collection.

**See Also**

[ProxyConfig Class] | [ProxyConfig Members] | [Org.Mentalis.Proxy Namespace] | [ProxyConfig.ReadInt Overload List]

Copyright © 2002, The KPD-Team
ProxyConfig.ReadInt Method (String, Int32)

Reads an integer from the settings section.

```visualbasic
Overloads Public Function ReadInt( _
   ByVal name As String, _
   ByVal def As Integer _
) As Integer
```

```csharp
public int ReadInt( _
   string name, _
   int def
);
```

Parameters

- **name**
  - The key to read from.

- **def**
  - The default integer to return.

Return Value

The integer that corresponds with the specified key, or `def` if the specified key was not found in the collection.

See Also


Copyright © 2002, The KPD-Team
ProxyConfig.ReadString

Reads a string from the settings section.

**Overload List**

Reads a string from the settings section.

```csharp
public string ReadString(string);
```

Reads a string from the settings section.

```csharp
public string ReadString(string,string);
```

**See Also**

- [ProxyConfig Class](#)
- [ProxyConfig Members](#)
- [Org.Mentalis.Proxy Namespace](#)

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.ReadString Method (String)

Reads a string from the settings section.

[Visual Basic] Overloads Public Function ReadString( ByVal name As String ) As String

[C#]
public string ReadString(string name);

Parameters

name
The key to read from.

Return Value

The string value that corresponds with the specified key, or an empty string if the specified key was not found in the collection.

See Also


Copyright © 2002, The KPD-Team
ProxyConfig.ReadString Method (String, String)

Reads a string from the settings section.

**[Visual Basic]**

```vbnet
Overloads Public Function ReadString(  
ByVal name As String, _  
ByVal def As String _
) As String
```

**[C#]**

```csharp
public string ReadString(  
    string name,  
    string def
);
```

**Parameters**

- **name**
  The key to read from.

- **def**
  The default string to return.

**Return Value**

The string value that corresponds with the specified key, or `def` if the specified key was not found in the collection.

**See Also**

- [ProxyConfig Class](#)
- [ProxyConfig Members](#)
- [Org.Mentalis.Proxy Namespace](#)
- [ProxyConfig.ReadString Overload List](#)

Copyright © 2002, The KPD-Team
**ProxyConfig.RemoveUser**

Removes a user from the authentication list.

**Overload List**

Removes a user from the authentication list.

```csharp
public void RemoveUser(string);
```

Removes a user from the authentication list.

```csharp
public void RemoveUser(string,bool);
```

**See Also**

- [ProxyConfig Class](#) | [ProxyConfig Members](#) | [Org.Mentalis.Proxy Namespace](#)

*Copyright © 2002, The KPD-Team*
ProxyConfig.RemoveUser Method (String)

Removes a user from the authentication list.

[Visual Basic] Overloads Public Sub RemoveUser(ByVal user As String)

[C#]
public void RemoveUser(
    string user)

Parameters

user
    The user to remove.

See Also


Copyright © 2002, The KPD-Team
ProxyConfig.RemoveUser Method (String, Boolean)

Removes a user from the authentication list.

[Visual Basic] Overloads Public Sub RemoveUser
ByVal user As String, _
ByVal save As Boolean _
)

[C#]
public void RemoveUser(
    string user, 
    bool save
);

Parameters

user
  The user to remove.

save
  True if the data has to be written to the XML file, false otherwise.

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.SaveData Method

Saves the data in this class to an XML file.

[Visual Basic] Public Sub SaveData()

[C#] public void SaveData();

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Saves the listeners to an XML writer.

```java
private void SaveListeners(
    XmlTextWriter writer
);
```

**Parameters**

- **writer**
  The XML writer to save the users to.

**See Also**

- [ProxyConfig Class](#)
- [ProxyConfig Members](#)
- [Org.Mentalis.Proxy Namespace](#)

Copyright © 2002, The KPD-Team
ProxyConfig.SaveSetting

Saves an array of bytes to the settings section.

Overload List

Saves a string to the settings section.

```csharp
public void SaveSetting(string,string);
```

Saves a string to the settings section.

```csharp
public void SaveSetting(string,string,bool);
```

Saves an integer to the settings section.

```csharp
public void SaveSetting(string,int);
```

Saves an integer to the settings section.

```csharp
public void SaveSetting(string,int,bool);
```

Saves an array of bytes to the settings section.

```csharp
public void SaveSetting(string,byte[]);
```

Saves an array of bytes to the settings section.

```csharp
public void SaveSetting(string,byte[],bool);
```

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
%ProxyConfig.SaveSetting Method (String, Byte[])%

Saves an array of bytes to the settings section.

```[Visual Basic]Overloads Public Sub SaveSetting(ByVal name As String, ByVal data As Byte())
```

```[C#]public void SaveSetting(string name, byte[] data)
```

**Parameters**

- **name**
  - The key of the setting.

- **data**
  - The byte data of the setting.

**See Also**

- ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace | ProxyConfig.SaveSetting Overload List

Copyright © 2002, The KPD-Team
ProxyConfig.SaveSetting Method (String, Byte[], Boolean)

Saves an array of bytes to the settings section.

```csharp
public void SaveSetting(
    string name,
    byte[] data,
    bool saveData
);
```

Parameters

- **name**
  - The key of the setting.

- **data**
  - The byte data of the setting.

- **saveData**
  - True if the data has to be written to the XML file, false otherwise.

See Also


Copyright © 2002, The KPD-Team
ProxyConfig.SaveSetting Method (String, Int32)

Saves an integer to the settings section.

```csharp
public void SaveSetting(
    string name,
    int data
);
```

Parameters

- `name`  
  The key of the setting.

- `data`  
  The integer data of the setting.

See Also

- ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace | ProxyConfig.SaveSetting Overload List

Copyright © 2002, The KPD-Team
Saves an integer to the settings section.

**[Visual Basic] Overloads Public Sub SaveSetting**

```vbnet
ByVal name As String, _
ByVal data As Integer, _
ByVal saveData As Boolean _
)
```

**[C#]**

```csharp
public void SaveSetting(
    string name, 
    int data, 
    bool saveData
);
```

**Parameters**

*name*

The key of the setting.

*data*

The integer data of the setting.

*saveData*

True if the data has to be written to the XML file, false otherwise.

**See Also**


Copyright © 2002, The KPD-Team
ProxyConfig.SaveSetting Method (String, String)

Saves a string to the settings section.

[Visual Basic] Overloads Public Sub SaveSetting( ByVal name As String, ByVal data As String )

[C#] public void SaveSetting( string name, string data );

Parameters

name
The key of the setting.

data
The string data of the setting.

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.SaveSetting Method (String, String, Boolean)

Saves a string to the settings section.

```visualbasic
Overloads Public Sub SaveSetting(
    ByVal name As String,
    ByVal data As String,
    ByVal saveData As Boolean
)
```

```csharp
public void SaveSetting(
    string name,
    string data,
    bool saveData
);
```

Parameters

- **name**
  The key of the setting.

- **data**
  The string data of the setting.

- **saveData**
  True if the data has to be written to the XML file, false otherwise.

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.SaveSettings Method

Saves the settings in this class to an XML writer.

[Visual Basic]
Private Sub SaveSettings( _
    ByVal writer As System.Xml.XmlTextWriter
)

[C#]
private void SaveSettings(
    XmlTextWriter writer
);

Parameters

writer
    The XML writer to save the data to.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ProxyConfig.SaveUserHash

Saves a username and password hash combination to the authentication list.

**Overload List**

Saves a username and password hash combination to the authentication list.

```csharp
public void SaveUserHash(string,string);
```

Saves a username and password hash combination to the authentication list.

```csharp
public void SaveUserHash(string,string,bool);
```

**See Also**

[ProxyConfig Class] | [ProxyConfig Members] | [Org.Mentalis.Proxy Namespace]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ProxyConfig.SaveUserHash Method (String, String)

Saves a username and password hash combination to the authentication list.

[Visual Basic] Overloads Public Sub SaveUserHash(
  ByVal username As String,
  ByVal passHash As String
)

[C#]
public void SaveUserHash(
  string username,
  string passHash
);

Parameters

username
  The username to add.

passHash
  The password hash to add.

Remarks

If the user already exists in the collection, the old password hash will be changed to the new one.

The username 'users' is invalid because it is used internally to store the usernames.

The password will not be hashed before it is stored in the authentication list. The user must make sure it is a valid MD5 hash.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SystemArgumentNullException</td>
<td>username is null -or- passHash is null.</td>
</tr>
</tbody>
</table>
System.ArgumentException | The specified username is invalid.

See Also


Copyright © 2002, The KPD-Team
ProxyConfig.SaveUserHash Method (String, String, Boolean)

Saves a username and password hash combination to the authentication list.

[Visual Basic] Overloads Public Sub SaveUserHash(
    ByVal username As String,
    ByVal passHash As String,
    ByVal saveData As Boolean
)

[C#]
public void SaveUserHash(
    string username,
    string passHash,
    bool saveData
);

Parameters

username
The username to add.

passHash
The password hash to add.

saveData
True if the data has to be written to the XML file, false otherwise.

Remarks

If the user already exists in the collection, the old password hash will be changed to the new one.

The username 'users' is invalid because it is used internally to store the usernames.

The password will not be hashed before it is stored in the authentication list. The user must make sure it is a valid MD5 hash.

Exceptions
<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.ArgumentNullException</code></td>
<td>username is null - or - passHash is null.</td>
</tr>
<tr>
<td><code>System.ArgumentException</code></td>
<td>The specified username is invalid.</td>
</tr>
</tbody>
</table>

**See Also**


[Copyright © 2002, The KPD-Team](#)
ProxyConfig.SaveUserPass

Saves a username and password combination to the authentication list.

Overload List

Saves a username and password combination to the authentication list.

    public void SaveUserPass(string,string);

Saves a username and password combination to the authentication list.

    public void SaveUserPass(string,string,bool);

See Also

    ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
ProxyConfig.SaveUserPass Method (String, String)

Saves a username and password combination to the authentication list.

[Visual Basic]
Overloads Public Sub SaveUserPass(
    ByVal username As String, _
    ByVal password As String _
)

[C#]
public void SaveUserPass(
    string username,
    string password
);

Parameters

username
The username to add.

password
The password to add.

Remarks

If the user already exists in the collection, the old password will be changed to the new one.

The username 'users' is invalid because it is used internally to store the usernames.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>username is null or password is null.</td>
</tr>
<tr>
<td>System.ArgumentException</td>
<td>The specified username is invalid.</td>
</tr>
</tbody>
</table>
See Also


Copyright © 2002, The KPD-Team
ProxyConfig.SaveUserPass Method (String, String, Boolean)

Saves a username and password combination to the authentication list.

[Visual Basic]
Overloads Public Sub SaveUserPass(
    ByVal username As String,
    ByVal password As String,
    ByVal saveData As Boolean
)

[C#]
public void SaveUserPass(
    string username,
    string password,
    bool saveData
);

Parameters

username
  The username to add.

password
  The password to add.

saveData
  True if the data has to be written to the XML file, false otherwise.

Remarks

If the user already exists in the collection, the old password will be changed to the new one.

The username 'users' is invalid because it is used internally to store the usernames.

The password will be hashed before it is stored in the authentication list.

Exceptions
<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SystemArgumentNullException</td>
<td>username is null -or- password is null.</td>
</tr>
<tr>
<td>System.ArgumentException</td>
<td>The specified username is invalid.</td>
</tr>
</tbody>
</table>

See Also


Copyright © 2002, The KPD-Team
ProxyConfig.SaveUsers Method

Saves the authentication list to an XML writer.

[Visual Basic] Private Sub SaveUsers( _
    ByVal writer As System.Xml.XmlTextWriter _
) _

[C#] 
private void SaveUsers( 
    XmlTextWriter writer 
); 

Parameters

writer  
The XML writer to save the users to.

See Also

ProxyConfig Class | ProxyConfig Members | Org.Mentalis.Proxy Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Org.Mentalis.Proxy.Ftp Namespace

Namespace hierarchy

Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FtpClient</td>
<td>Relays FTP commands between a remote host and a local client.</td>
</tr>
<tr>
<td>FtpDataConnection</td>
<td>Relays FTP data between a remote host and a local client.</td>
</tr>
<tr>
<td>FtpListener</td>
<td>Listens on a specific port on the proxy server and forwards all incoming FTP traffic to the appropriate server.</td>
</tr>
</tbody>
</table>

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpClient Class

Relays FTP commands between a remote host and a local client. For a list of all members of this type, see FtpClient Members.

System.Object  Client
FtpClient

NotInheritable Public Class FtpClient
Inherits Client
Implements IDisposable

Remarks
This class supports the 'OPEN' command, 'USER user@host:port' and 'USER user@host port'.

Requirements
Assembly: Proxy.exe

See Also
FtpClient Members  | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
# FtpClient Members

## Internal Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsValidCommand</td>
<td>Checks whether a specified command is a complete FTP command or not.</td>
</tr>
<tr>
<td>IsValidReply</td>
<td>Checks whether a specified reply is a complete FTP reply or not.</td>
</tr>
</tbody>
</table>

## Public Instance Constructors

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FtpClient Constructor</td>
<td>Initializes a new instance of the FtpClient class.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose</td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td>Equals</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>StartHandshake</td>
<td>Sends a welcome message to the client.</td>
</tr>
<tr>
<td>StartRelay</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns text information about this FtpClient object.</td>
</tr>
</tbody>
</table>

## Protected Instance Properties

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong> (inherited from <strong>Client</strong>)</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>RemoteBuffer</strong> (inherited from <strong>Client</strong>)</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th><strong>Finalize</strong> (inherited from <strong>Object</strong>)</th>
<th>Select the method name to go to the Microsoft documentation.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local host.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

### Internal Instance Properties

| **DataForward** | Gets or sets the dataconnection object used to transmit files and |
other data from and to the FTP server.

## Internal Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SendCommand</td>
<td>Sends a string to the local FTP client.</td>
</tr>
</tbody>
</table>

## Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_DataForward</td>
<td>Holds the value of the DataForward property.</td>
</tr>
<tr>
<td>m_FtpCommand</td>
<td>Holds the value of the FtpCommand property.</td>
</tr>
<tr>
<td>m_FtpReply</td>
<td>Holds the value of the FtpReply property.</td>
</tr>
<tr>
<td>m_User</td>
<td>Holds the value of the User property.</td>
</tr>
</tbody>
</table>

## Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FtpCommand</td>
<td>Gets or sets a property that can be used to store the received FTP command.</td>
</tr>
<tr>
<td>FtpReply</td>
<td>Gets or sets a property that can be used to store the received FTP reply.</td>
</tr>
<tr>
<td>User</td>
<td>Gets or sets a string containing the logged on username.</td>
</tr>
</tbody>
</table>

## Private Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectTo</td>
<td>Connects to the specified endpoint.</td>
</tr>
<tr>
<td>OnCommandSent</td>
<td>Called when an FTP command has been successfully sent to the FTP server.</td>
</tr>
<tr>
<td><strong>Function</strong></td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>OnHelloSent</strong></td>
<td>Called when the welcome message has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnIgnoreReply</strong></td>
<td>Called when we receive a reply from the FTP server that should be ignored.</td>
</tr>
<tr>
<td><strong>OnReceiveCommand</strong></td>
<td>Called when we have received some bytes from the client.</td>
</tr>
<tr>
<td><strong>OnRemoteConnected</strong></td>
<td>Called when we're connected to the remote FTP server.</td>
</tr>
<tr>
<td><strong>OnReplyReceived</strong></td>
<td>Called when we receive a reply from the FTP server.</td>
</tr>
<tr>
<td><strong>OnReplySent</strong></td>
<td>Called when the reply from the FTP server has been sent to the local FTP client.</td>
</tr>
<tr>
<td><strong>ParseIPPort</strong></td>
<td>Parses an IP address and port from a specified input string.</td>
</tr>
<tr>
<td><strong>ProcessCommand</strong></td>
<td>Processes an FTP command, sent from the client.</td>
</tr>
<tr>
<td><strong>ProcessPortCommand</strong></td>
<td>Processes a PORT command, sent from the client.</td>
</tr>
</tbody>
</table>

**See Also**

- [FtpClient Class](#)

*Copyright © 2002, The KPD-Team*
FtpClient Constructor

Initializes a new instance of the FtpClient class.

[Visual Basic] Public Sub New( _
    ByVal ClientSocket As System.Net.Sockets.Socket,
    ByVal Destroyer As Org.Mentalis.Proxy.DestroyDelegate)

[C#] public FtpClient(
    Socket ClientSocket,
    DestroyDelegate Destroyer)

Parameters

ClientSocket
The Socket connection between this proxy server and the local client.

Destroyer
The callback method to be called when this Client object disconnects from the local client and the remote server.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient Fields

The fields of the FtpClient class are listed below. For a complete list of FtpClient class members, see the FtpClient Members topic.

Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_DataForward</td>
<td>Holds the value of the DataForward property.</td>
</tr>
<tr>
<td>m_FtpCommand</td>
<td>Holds the value of the FtpCommand property.</td>
</tr>
<tr>
<td>m_FtpReply</td>
<td>Holds the value of the FtpReply property.</td>
</tr>
<tr>
<td>m_User</td>
<td>Holds the value of the User property.</td>
</tr>
</tbody>
</table>

See Also

FtpClient Class | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
**FtpClient.m_DataForward Field**

Holds the value of the DataForward property.


[C#] `private FtpDataConnection m_DataForward;`

See Also

[FtpClient Class] | [FtpClient Members] | [Org.Mentalis.Proxy.Ftp Namespace]

Copyright © 2002, The KPD-Team
FtpClient.m_FtpCommand Field

Holds the value of the FtpCommand property.

[Visual Basic] Private m_FtpCommand As String

[C#] private string m_FtpCommand;

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Holds the value of the FtpReply property.

[Visual Basic] Private m_FtpReply As String

[C#]
private string m_FtpReply;

See Also
FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
**FtpClient.m_User Field**

Holds the value of the User property.

```
[Visual Basic] Private m_User As String
```

```
[C#]
private string m_User;
```

See Also

[FtpClient Class] | [FtpClient Members] | [Org.Mentalis.Proxy.Ftp Namespace]

Copyright © 2002, The KPD-Team
FtpClient Properties

The properties of the **FtpClient** class are listed below. For a complete list of **FtpClient** class members, see the [FtpClient Members](#) topic.

**Protected Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td>RemoteBuffer</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

**Internal Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataForward</td>
<td>Gets or sets the dataconnection object used to transmit files and other data from and to the FTP server.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FtpCommand</td>
<td>Gets or sets a property that can be used to store the received FTP command.</td>
</tr>
<tr>
<td>FtpReply</td>
<td>Gets or sets a property that can be used to store the received FTP reply.</td>
</tr>
<tr>
<td>User</td>
<td>Gets or sets a string containing the logged on username.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
**FtpClient.DataForward Property**

Gets or sets the dataconnection object used to transmit files and other data from and to the FTP server.

[Visual Basic]
Friend Property DataForward As

[C#]
internal FtpDataConnection DataForward {get;}

**Property Value**

An FtpDataConnection object that's used to transmit files and other data from and to the FTP server.

**See Also**

[FtpClient Class] | [FtpClient Members] | [Org.Mentalis.Proxy.Ftp Namespace]

Copyright © 2002, The KPD-Team
FtpClient.FtpCommand Property

Gets or sets a property that can be used to store the received FTP command.

[Visual Basic] Private Property FtpCommand As

[C#]
private string FtpCommand {get; set;}

Property Value

A string that can be used to store the received FTP command.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpClient.FtpReply Property

Gets or sets a property that can be used to store the received FTP reply.

[Visual Basic]
Private Property FtpReply As String

[C#]
private string FtpReply {get; set;}

Property Value

A string that can be used to store the received FTP reply.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpClient.User Property

Gets or sets a string containing the logged on username.

[Visual Basic] Private Property User As String

[C#]
private string User {get; set;}

Property Value

A string containing the logged on username.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
The methods of the **FtpClient** class are listed below. For a complete list of **FtpClient** class members, see the **FtpClient Members** topic.

### Internal Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsValidCommand</td>
<td>Checks whether a specified command is a complete FTP command or not.</td>
</tr>
<tr>
<td>IsValidReply</td>
<td>Checks whether a specified reply is a complete FTP reply or not.</td>
</tr>
</tbody>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose</td>
<td>(inherited from <strong>Client</strong>) Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td>Equals</td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType</td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>StartHandshake</td>
<td>Sends a welcome message to the client.</td>
</tr>
<tr>
<td>StartRelay</td>
<td>(inherited from <strong>Client</strong>) Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns text information about this FtpClient object.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finalize</td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local host.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

**Internal Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SendCommand</strong></td>
<td>Sends a string to the local FTP client.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ConnectTo</strong></td>
<td>Connects to the specified endpoint.</td>
</tr>
<tr>
<td><strong>OnCommandSent</strong></td>
<td>Called when an FTP command has been successfully sent to the FTP server.</td>
</tr>
<tr>
<td><strong>OnHelloSent</strong></td>
<td>Called when the welcome message has been sent to the client.</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>OnIgnoreReply</td>
<td>Called when we receive a reply from the FTP server that should be ignored.</td>
</tr>
<tr>
<td>OnReceiveCommand</td>
<td>Called when we have received some bytes from the client.</td>
</tr>
<tr>
<td>OnRemoteConnected</td>
<td>Called when we're connected to the remote FTP server.</td>
</tr>
<tr>
<td>OnReplyReceived</td>
<td>Called when we receive a reply from the FTP server.</td>
</tr>
<tr>
<td>OnReplySent</td>
<td>Called when the reply from the FTP server has been sent to the local FTP client.</td>
</tr>
<tr>
<td>ParseIPPort</td>
<td>Parses an IP address and port from a specified input string.</td>
</tr>
<tr>
<td>ProcessCommand</td>
<td>Processes an FTP command, sent from the client.</td>
</tr>
<tr>
<td>ProcessPortCommand</td>
<td>Processes a PORT command, sent from the client.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
FtpClient.ConnectTo Method

Connects to the specified endpoint.

[Visual Basic] Private Sub ConnectTo(ByVal RemoteServer As System.Net.IPEndPoint)

[C#]
private void ConnectTo(IPEndPoint RemoteServer);

Parameters

RemoteServer
The IPEndPoint to connect to.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.Net.Sockets.SocketException</td>
<td>There was an error connecting to the specified endpoint</td>
</tr>
</tbody>
</table>

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.IsValidCommand Method

Checks whether a specified command is a complete FTP command or not.

[Visual Basic]
Friend Shared Function IsValidCommand(
    ByVal Command As String
) As Boolean

[C#]
internal static bool IsValidCommand(
    string Command
);
FtpClient.IsValidReply Method

Checks whether a specified reply is a complete FTP reply or not.

[Visual Basic] Friend Shared Function IsValidReply(ByVal Input As String) As Boolean

[C#]
internal static bool IsValidReply(string Input);

Parameters

Input
A string containing the reply to check.

Return Value

True is the reply is complete, false otherwise.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.OnCommandSent Method

Called when an FTP command has been successfully sent to the FTP server.

[Visual Basic]
Private Sub OnCommandSent( ByVal ar As System.IAsyncResult )

[C#]
private void OnCommandSent( IAsyncResult ar );

Parameters

ar
The result of the asynchronous operation.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.OnHelloSent Method

Called when the welcome message has been sent to the client.

[Visual Basic] Private Sub OnHelloSent(ByVal ar As System.IAsyncResult)

[C#] private void OnHelloSent(IAsyncResult ar);

Parameters

ar

The result of the asynchronous operation.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Called when we receive a reply from the FTP server that should be ignored.

[Visual Basic] Private Sub OnIgnoreReply( _
    ByVal ar As System.IAsyncResult _
) __

[C#] private void OnIgnoreReply( 
    IAsyncResult ar 
); __

Parameters

ar
The result of the asynchronous operation.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.OnReceiveCommand Method

Called when we have received some bytes from the client.

**[Visual Basic]**
```vbnet
Private Sub OnReceiveCommand(
    ByVal ar As System.IAsyncResult
)
```

**[C#]**
```csharp
private void OnReceiveCommand(
    IAsyncResult ar
);
```

**Parameters**

`ar`

The result of the asynchronous operation.

**See Also**

[FtpClient Class] [FtpClient Members] [Org.Mentalis.Proxy.Ftp Namespace]

Copyright © 2002, The KPD-Team
FtpClient.OnRemoteConnected Method

Called when we're connected to the remote FTP server.

[Visual Basic]
Private Sub OnRemoteConnected(ByVal ar As System.IAsyncResult)

[C#]
private void OnRemoteConnected(IAsyncResult ar);

Parameters

ar
The result of the asynchronous operation.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpClient.OnReplyReceived Method

Called when we receive a reply from the FTP server.

[Visual Basic] Private Sub OnReplyReceived( ByVal ar As System.IAsyncResult _
) 

[C#] private void OnReplyReceived( IAsyncResult ar );

Parameters

* ar

The result of the asynchronous operation.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Called when the reply from the FTP server has been sent to the local FTP client.

### [Visual Basic]

```vbnet
Private Sub OnReplySent(ByVal ar As System.IAsyncResult)
```

### [C#]

```csharp
private void OnReplySent(IAsyncResult ar);
```

#### Parameters

- `ar`
  - The result of the asynchronous operation.

#### See Also


---

**Copyright © 2002, The KPD-Team**
Mentalis.org Proxy Documentation
FtpClient.ParseIPPort Method

Parses an IP address and port from a specified input string.

**Parameters**

*Input*

The string to parse.

**Return Value**

An instance of the IPEndPoint class if successful, null otherwise.

**Remarks**

The input string is of the following form:

- **HOST:PORT**
- **HOST** PORT

**See Also**

[FtpClient Class] | [FtpClient Members] | [Org.Mentalis.Proxy.Ftp Namespace]

Copyright © 2002, The KPD-Team
FtpClient.ProcessCommand Method

Processes an FTP command, sent from the client.

[Visual Basic] Private Function ProcessCommand(ByVal Command As String) As Boolean

[C#] private bool ProcessCommand(string Command); 

Parameters

Command
   The command to process.

Return Value

True if the command may be sent to the server, false otherwise.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.ProcessPortCommand Method

Processes a PORT command, sent from the client.

```csharp
private void ProcessPortCommand(string Input);
```

Parameters

Input
The parameters of the PORT command.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.SendCommand Method

Sends a string to the local FTP client.

[Visual Basic]Friend Sub SendCommand( _
    ByVal Command As String _
)

[C#]
internal void SendCommand(
    string Command
);

Parameters

Command
The result of the asynchronous operation.

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp
Namespace

Copyright © 2002, The KPD-Team
FtpClient.StartHandshake Method

Sends a welcome message to the client.

[Visual Basic] Overrides Public Sub StartHandshake()

[C#]
public override void StartHandshake();

See Also

FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpClient.ToString Method

Returns text information about this FtpClient object.

[Visual Basic] Overrides Public Function ToString()

[C#] public override string ToString();

Return Value
A string representing this FtpClient object.

See Also
FtpClient Class | FtpClient Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection Class

Relays FTP data between a remote host and a local client.

For a list of all members of this type, see FtpDataConnection Members.

System.Object  Client  
  FtpDataConnection

[Visual Basic]
NotInheritable  Class FtpDataConnection
    Inherits Client
    Implements IDisposable

[C#]
sealed class FtpDataConnection : Client, IDisposable

Requirements

Assembly: Proxy.exe

See Also

FtpDataConnection Members  |  Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
### FtpDataConnection Members

#### Public Instance Constructors

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FtpDataConnection Constructor</strong></td>
<td>Initializes a new instance of the FtpDataConnection class.</td>
</tr>
</tbody>
</table>

#### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from Client)</td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ProcessPasv</strong></td>
<td>Called when the proxy server processes a PASV command.</td>
</tr>
<tr>
<td><strong>ProcessPort</strong></td>
<td>Initializes a new instance of the FtpDataConnection class.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td>Starts relaying data between the remote FTP server and the local FTP client.</td>
</tr>
<tr>
<td><strong>StartRelay</strong> (inherited from Client)</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from Client)</td>
<td>Returns text information about this Client object.</td>
</tr>
</tbody>
</table>

#### Protected Instance Properties

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong> (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td>RemoteBuffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>==================================================================</td>
</tr>
</tbody>
</table>

**Protected Instance Methods**

| Finalize (inherited from Object) | Select the method name to go to the Microsoft documentation. |
|---------------------------------|================================================================|
| MemberwiseClone (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| OnClientReceive (inherited from Client) | Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host. |
| OnClientSent (inherited from Client) | Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host. |
| OnRemoteReceive (inherited from Client) | Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client. |
| OnRemoteSent (inherited from Client) | Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client. |

**Internal Instance Properties**

| ExpectsReply | Gets or sets a boolean value that indicates whether the FtpDataConnection expects a reply from the remote FTP server or not. |
### Internal Instance Methods

| ProcessPasvReplyRecv | Called when the FtpClient receives a reply on the PASV command from the server. |

### Private Instance Fields

| m_ExpectsReply | Holds the value of the ExpectsReply property. |
| m_FtpReply | Holds the value of the FtpReply property. |
| m_ListenSocket | Holds the value of the ListenSocket property. |
| m_Parent | Holds the value of the Parent property. |

### Private Instance Properties

| FtpReply | Gets or sets a string that stores the reply that has been sent from the remote FTP server. |
| ListenSocket | Gets or sets the Socket that's used to listen for incoming connections. |
| Parent | Gets or sets the parent of this FtpDataConnection. |

### Private Instance Methods

| OnPasvAccept | Called when there's a connection from the local FTP client waiting to be accepted. |
| OnPasvConnected | Called when we're connected to the data port of the remote FTP server. |
| OnPortAccept | Called when there's a |
connection from the remote FTP server waiting to be accepted.

<table>
<thead>
<tr>
<th>OnPortConnected</th>
<th>Called when we're connected to the data port on the local FTP client.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParsePASVIP</td>
<td>Parses a PASV reply into an instance of the IPEndPoint class.</td>
</tr>
<tr>
<td>ProcessPASVReply</td>
<td>Processes a PASV reply from the server.</td>
</tr>
</tbody>
</table>

See Also

FtpDataConnection Class | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection Constructor

Initializes a new instance of the FtpDataConnection class.

[Visual Basic] Public Sub New()

[C#]
public FtpDataConnection();

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection Fields

The fields of the FtpDataConnection class are listed below. For a complete list of FtpDataConnection class members, see the FtpDataConnection Members topic.

Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_ExpectsReply</td>
<td>Holds the value of the ExpectsReply property.</td>
</tr>
<tr>
<td>m_FtpReply</td>
<td>Holds the value of the FtpReply property.</td>
</tr>
<tr>
<td>m_ListenSocket</td>
<td>Holds the value of the ListenSocket property.</td>
</tr>
<tr>
<td>m_Parent</td>
<td>Holds the value of the Parent property.</td>
</tr>
</tbody>
</table>

See Also

FtpDataConnection Class | Org.Mentalis.Proxy.Ftp Namespace
Copyright © 2002, The KPD-Team
FtpDataConnection.m_ExpectsReply Field

Holds the value of the ExpectsReply property.

[Visual Basic] Private m_ExpectsReply As Boolean

[C#] private bool m_ExpectsReply;

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.m_FtpReply Field

Holds the value of the FtpReply property.

[Visual Basic] Private m_FtpReply As String

[C#] private string m_FtpReply;

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Holds the value of the ListenSocket property.

**[Visual Basic]**

```vbnet
Private m_ListenSocket As System.Net.Sockets.Socket
```

**[C#]**

```csharp
private Socket m_ListenSocket;
```

See Also

[FtpDataConnection Class] | [FtpDataConnection Members] | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.m_Parent Field

Holds the value of the Parent property.


[C#]
private FtpClient m_Parent;

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
The properties of the **FtpDataConnection** class are listed below. For a complete list of **FtpDataConnection** class members, see the **FtpDataConnection Members** topic.

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong></td>
<td>(inherited from <strong>Client</strong>) Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td><strong>RemoteBuffer</strong></td>
<td>(inherited from <strong>Client</strong>) Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

### Internal Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ExpectsReply</strong></td>
<td>Gets or sets a boolean value that indicates whether the FtpDataConnection expects a reply from the remote FTP server or not.</td>
</tr>
</tbody>
</table>

### Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FtpReply</strong></td>
<td>Gets or sets a string that stores the reply that has been sent from the remote FTP server.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong></td>
<td>Gets or sets the Socket that's used to listen for incoming connections.</td>
</tr>
<tr>
<td><strong>Parent</strong></td>
<td>Gets or sets the parent of this FtpDataConnection.</td>
</tr>
</tbody>
</table>

See Also

**FtpDataConnection Class** | **Org.Mentalis.Proxy.Ftp Namespace**

Copyright © 2002, The KPD-Team
FtpDataConnection.ExpectsReply Property

Gets or sets a boolean value that indicates whether the FtpDataConnection expects a reply from the remote FTP server or not.

[Visual Basic]
Friend Property ExpectsReply As Boolean

[C#]
internal bool ExpectsReply {get; set;}

Property Value

A boolean value that indicates whether the FtpDataConnection expects a reply from the remote FTP server or not.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.FtpReply Property

Gets or sets a string that stores the reply that has been sent from the remote FTP server.

[Visual Basic] Private Property FtpReply As String

[C#] private string FtpReply {get; set;}

Property Value

A string that stores the reply that has been sent from the remote FTP server.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.ListenSocket Property

Gets or sets the Socket that's used to listen for incoming connections.

**[Visual Basic]**

Private Property ListenSocket

**[C#]**

private System.Net.Sockets.Socket ListenSocket

Property Value

A Socket that's used to listen for incoming connections.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.Parent Property

Gets or sets the parent of this FtpDataConnection.


[C#] private FtpClient Parent {get; set;}

Property Value

The FtpClient object that's the parent of this FtpDataConnection object.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
The methods of the **FtpDataConnection** class are listed below. For a complete list of **FtpDataConnection** class members, see the **FtpDataConnection Members** topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Client</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ProcessPasv</strong></td>
<td>Called when the proxy server processes a PASV command.</td>
</tr>
<tr>
<td><strong>ProcessPort</strong></td>
<td>Initializes a new instance of the FtpDataConnection class.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td>Starts relaying data between the remote FTP server and the local FTP client.</td>
</tr>
<tr>
<td><strong>StartRelay</strong> (inherited from <strong>Client</strong>)</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Client</strong>)</td>
<td>Returns text information about this Client object.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>Method Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from Client)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from Client)</td>
<td>Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from Client)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from Client)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

**Internal Instance Methods**

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ProcessPasvReplyRecv</strong></td>
<td>Called when the FtpClient receives a reply on the PASV command from the server.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnPasvAccept</strong></td>
<td>Called when there's a connection from the local FTP client waiting to be accepted.</td>
</tr>
<tr>
<td><strong>OnPasvConnected</strong></td>
<td>Called when we're connected to the data port of the remote FTP server.</td>
</tr>
<tr>
<td><strong>OnPortAccept</strong></td>
<td>Called when there's a connection from the remote FTP server waiting to be accepted.</td>
</tr>
<tr>
<td><strong>OnPortConnected</strong></td>
<td>Called when we're connected to the data port on the local FTP client.</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>ParsePaslIP</strong></td>
<td>Parses a PASV reply into an instance of the IPEndPoint class.</td>
</tr>
<tr>
<td><strong>ProcessPasvReply</strong></td>
<td>Processes a PASV reply from the server.</td>
</tr>
</tbody>
</table>

**See Also**

FtpDataConnection Class | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.OnPasvAccept Method

Called when there's a connection from the local FTP client waiting to be accepted.

[Visual Basic]Private Sub OnPasvAccept( _
   ByVal ar As System.IAsyncResult _
)

[C#]
private void OnPasvAccept(
   IAsyncResult ar
);

Parameters

   ar
   The result of the asynchronous operation.

See Also

   FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
**FtpDataConnection.OnPasvConnected Method**

Called when we're connected to the data port of the remote FTP server.

```visualbasic
Private Sub OnPasvConnected( ByVal ar As System.IAsyncResult )
```

```csharp
private void OnPasvConnected( IAsyncResult ar );
```

**Parameters**

*ar*

The result of the asynchronous operation.

**See Also**

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.OnPortAccept Method

Called when there's a connection from the remote FTP server waiting to be accepted.

[Visual Basic] Private Sub OnPortAccept( _
    ByVal ar As System.IAsyncResult _
)

[C#]
private void OnPortAccept( _
    IAsyncResult ar
);

Parameters

ar
The result of the asynchronous operation.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Called when we're connected to the data port on the local FTP client.

[Visual Basic]  Private Sub OnPortConnected(  
    ByVal ar As System.IAsyncResult  
)

[C#]  private void OnPortConnected( 
    IAsyncResult ar
);

Parameters

ar  The result of the asynchronous operation.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
**FtpDataConnection.ParsePasvIP Method**

Parses a PASV reply into an instance of the IPEndPoint class.

```visualbasic
Private Function ParsePasvIP(ByVal Reply As String) As System.Net.IPEndPoint
```

```csharp
private IPEndPoint ParsePasvIP(string Reply);
```

**Parameters**

*Reply*

The reply to parse into an IPEndPoint.

**Return Value**

An instance of the IPEndPoint class when successful, null otherwise.

**See Also**

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Called when the proxy server processes a PASV command.

**Parameters**

*Parent*  
The parent FtpClient object.

**See Also**

FtpDataConnection.ProcessPasvReply Method

Processes a PASV reply from the server.

[Visual Basic]Private Sub ProcessPasvReply(
    ByVal Reply As String _
)

[C#]
private void ProcessPasvReply(
    string Reply
);

Parameters

    Reply
        The reply to process.

See Also

    FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Called when the FtpClient receives a reply on the PASV command from the server.

[Visual Basic]Friend Function ProcessPasvReplyRecv(
    ByVal Input As String
) As Boolean

[C#]
internal bool ProcessPasvReplyRecv(
    string Input
);  

Parameters

Input
    The received reply.

Return Value

True if the input has been processed successfully, false otherwise.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpDataConnection.ProcessPort Method

Initializes a new instance of the FtpDataConnection class.

[Visual Basic] Public Function ProcessPort( ByVal RemoteAddress As System.Net.IPEndPoint ) As String

[C#] public string ProcessPort( IPEndPoint RemoteAddress );

Parameters

RemoteAddress

The address on the local FTP client to connect to.

Return Value

The PORT command string to send to the FTP server.

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpDataConnection.StartHandshake Method

Starts relaying data between the remote FTP server and the local FTP client.

[Visual Basic]
Overrides Public Sub StartHandshake()

[C#]
public override void StartHandshake();

See Also

FtpDataConnection Class | FtpDataConnection Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpListener Class

Listens on a specific port on the proxy server and forwards all incoming FTP traffic to the appropriate server.

For a list of all members of this type, see FtpListener Members.

System.Object  Listener  FtpListener

[Visual Basic]
NotInheritable Public Class FtpListener
    Inherits Listener
    Implements IDisposable

[C#]
public sealed class FtpListener : Listener, :

Requirements


Assembly: Proxy.exe

See Also

FtpListener Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
## FtpListener Members

### Public Instance Constructors

| FtpListener | Overloaded. Initialize a new instance of the FtpListener class. |

### Public Instance Properties

| ConstructString | Returns a string that holds all the construction information for this object. |
| IsDisposed (inherited from Listener) | Gets a value indicating whether the Listener has been disposed or not. |
| Listening (inherited from Listener) | Gets a value indicating whether the Listener is currently listening or not. |

### Public Instance Methods

| Dispose (inherited from Listener) | Disposes of the resources (other than memory) used by the Listener. |
| Equals (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| GetClientAt (inherited from Listener) | Returns the requested client from the client list. |
| GetClientCount (inherited from Listener) | Returns the number of clients in the client list. |
| GetHashCode (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| GetType (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| OnAccept | Called when there's an incoming |
client connection waiting to be accepted.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Listener</strong>)</td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong> (inherited from <strong>Listener</strong>)</td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

### See Also

- FtpListener Class
- Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
**FtpListener**

Initializes a new instance of the FtpListener class.

**Overload List**

Initializes a new instance of the FtpListener class.

```java
public FtpListener(int);
```

Initializes a new instance of the FtpListener class.

```java
public FtpListener(IPAddress,int);
```

**See Also**

[FtpListener Class] | [FtpListener Members] | [Org.Mentalis.Proxy.Ftp Namespace]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpListener Constructor (Int32)

Initializes a new instance of the FtpListener class.

**Parameters**

*Port*

The port to listen on.

**Remarks**

The FtpListener will start listening on all installed network cards.

**Exceptions**

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.ArgumentException</code></td>
<td>Port is not positive.</td>
</tr>
</tbody>
</table>

**See Also**

[FtpListener Class]  |  [FtpListener Members]  |  [Org.Mentalis.Proxy.Ftp Namespace]  |  [FtpListener Constructor Overload List]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
FtpListener Constructor (IPAddress, Int32)

Initializes a new instance of the FtpListener class.

[Visual Basic] Overloads Public Sub New( _
    ByVal Address As System.Net.IPAddress, _
    ByVal Port As Integer _
)

[C#]
public FtpListener( _
    IPAddress Address, _
    int Port
);

Parameters

Address
The address to listen on. You can specify IPAddress.Any to listen on all installed network cards.

Port
The port to listen on.

Remarks
For the security of your server, try to avoid to listen on every network card (IPAddress.Any). Listening on a local IP address is usually sufficient and much more secure.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Address is null.</td>
</tr>
<tr>
<td>System.ArgumentException</td>
<td>Port is not positive.</td>
</tr>
</tbody>
</table>

See Also

FtpListener Class | FtpListener Members | Org.Mentalis.Proxy.Ftp Namespace | FtpListener Constructor Overload List
Mentalis.org Proxy Documentation
FtpListener Properties

The properties of the FtpListener class are listed below. For a complete list of FtpListener class members, see the FtpListener Members topic.

Public Instance Properties

<table>
<thead>
<tr>
<th>ConstructString</th>
<th>Returns a string that holds all the construction information for this object.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsDisposed (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td>Listening (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

Protected Instance Properties

<table>
<thead>
<tr>
<th>Address (inherited from Listener)</th>
<th>Gets or sets the address on which to listen on.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clients (inherited from Listener)</td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td>ListenSocket (inherited from Listener)</td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td>Port (inherited from Listener)</td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

See Also

FtpListener Class | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpListener.ConstructString Property

Returns a string that holds all the construction information for this object.

[Visual Basic] Overrides Public ReadOnly Property

[C#]

    public string ConstructString { override get; }

Property Value

A string that holds all the construction information for this object.

See Also

FtpListener Class | FtpListener Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
### FtpListener Methods

The methods of the **FtpListener** class are listed below. For a complete list of **FtpListener** class members, see the [FtpListener Members](#) topic.

#### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Listener</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Listener.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming client connection waiting to be accepted.</td>
</tr>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

#### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Listener</strong>)</td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong> (inherited from <strong>Listener</strong>)</td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

**See Also**


*Copyright © 2002, The KPD-Team*
FtpListener.OnAccept Method

Called when there's an incoming client connection waiting to be accepted.

[Visual Basic] Overrides Public Sub OnAccept(ByVal ar As System.IAsyncResult _)

[C#] public override void OnAccept(IAsyncResult ar); 

Parameters

ar 
The result of the asynchronous operation.

See Also

FtpListener Class | FtpListener Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team
FtpListener.ToString Method

Returns a string representation of this object.

[Visual Basic] Overrides Public Function ToString()

[C#]
public override string ToString();

Return Value
A string with information about this object.

See Also
FtpListener Class | FtpListener Members | Org.Mentalis.Proxy.Ftp Namespace

Copyright © 2002, The KPD-Team

Namespace hierarchy

Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpClient</td>
<td>Relays HTTP data between a remote host and a local client.</td>
</tr>
<tr>
<td>HttpListener</td>
<td>Listens on a specific port on the proxy server and forwards all incoming HTTP traffic to the appropriate server.</td>
</tr>
</tbody>
</table>

Copyright © 2002, The KPD-Team
HttpClient Class

Relays HTTP data between a remote host and a local client.
For a list of all members of this type, see HttpClient Members.

System.Object  Client
HttpClient

[Visual Basic]
NotInheritable Public Class HttpClient
    Inherits Client
    Implements IDisposable

[C#]
public sealed class HttpClient : Client, IDisposable

Remarks
This class supports both HTTP and HTTPS.

Requirements
Assembly: Proxy.exe

See Also
HttpClient Members  |  Org.Mentalis.Proxy.Http Namespace
Copyright © 2002, The KPD-Team
# HttpClient Members

## Public Instance Constructors

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HttpClient Constructor</strong></td>
<td>Initializes a new instance of the HttpClient class.</td>
</tr>
</tbody>
</table>

## Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RequestedPath</strong></td>
<td>Gets or sets the requested path.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Client</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td>Starts receiving data from the client connection.</td>
</tr>
<tr>
<td><strong>StartRelay</strong> (inherited from <strong>Client</strong>)</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Overloaded. Returns text information about this HttpClient object.</td>
</tr>
</tbody>
</table>

## Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong> (inherited from <strong>Client</strong>)</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td><strong>RemoteBuffer</strong> (inherited from</td>
<td>Gets the buffer to store all the</td>
</tr>
</tbody>
</table>


## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

## Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_HeaderFields</strong></td>
<td>Holds the value of the HeaderFields property.</td>
</tr>
<tr>
<td><strong>m_HttpQuery</strong></td>
<td>Holds the value of the HttpQuery property.</td>
</tr>
<tr>
<td><strong>m_HttpRequestType</strong></td>
<td>Holds the value of the HttpRequestType property.</td>
</tr>
<tr>
<td><strong>m_HttpVersion</strong></td>
<td>Holds the value of the HttpVersion property.</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td><strong>m_RequestPath</strong></td>
<td>Holds the value of the RequestedPath property.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th><strong>HeaderFields</strong></th>
<th>Gets or sets a StringDictionary that stores the header fields.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HttpQuery</strong></td>
<td>Gets or sets the query string, received from the client.</td>
</tr>
<tr>
<td><strong>HttpRequestType</strong></td>
<td>Gets or sets the HTTP request type.</td>
</tr>
<tr>
<td><strong>HttpVersion</strong></td>
<td>Gets or sets the HTTP version the client uses.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th><strong>IsValidQuery</strong></th>
<th>Checks whether a specified string is a valid HTTP query string.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnConnected</strong></td>
<td>Called when we're connected to the requested remote host.</td>
</tr>
<tr>
<td><strong>OnErrorSent</strong></td>
<td>Called when the Bad Request error has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnOkSent</strong></td>
<td>Called when an OK reply has been sent to the local client.</td>
</tr>
<tr>
<td><strong>OnQuerySent</strong></td>
<td>Called when the HTTP query has been sent to the remote host.</td>
</tr>
<tr>
<td><strong>OnReceiveQuery</strong></td>
<td>Called when we received some data from the client connection.</td>
</tr>
<tr>
<td><strong>ParseQuery</strong></td>
<td>Parses a specified HTTP query into its header fields.</td>
</tr>
<tr>
<td><strong>ProcessQuery</strong></td>
<td>Processes a specified query and connects to the requested HTTP web server.</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>RebuildQuery</strong></td>
<td>Rebuilds the HTTP query, starting from the HttpRequestType, RequestedPath, HttpVersion and HeaderFields properties.</td>
</tr>
<tr>
<td><strong>SendBadRequest</strong></td>
<td>Sends a &quot;400 - Bad Request&quot; error to the client.</td>
</tr>
</tbody>
</table>

**See Also**

Mentalis.org Proxy Documentation
HttpClient Constructor

Initializes a new instance of the HttpClient class.

```visual-basic
Public Sub New( _
    ByVal ClientSocket As System.Net.Sockets.Socket,
    ByVal Destroyer As Org.Mentalis.Proxy.DestroyDelegate)
```

```csharp
public HttpClient( 
    Socket ClientSocket, 
    DestroyDelegate Destroyer
);
```

Parameters

*ClientSocket*


*Destroyer*

The callback method to be called when this Client object disconnects from the local client and the remote server.

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
**HttpClient Fields**

The fields of the **HttpClient** class are listed below. For a complete list of **HttpClient** class members, see the **HttpClient Members** topic.

**Private Instance Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_HeaderFields</td>
<td>Holds the value of the HeaderFields property.</td>
</tr>
<tr>
<td>m_HttpQuery</td>
<td>Holds the value of the HttpQuery property.</td>
</tr>
<tr>
<td>m_HttpRequestType</td>
<td>Holds the value of the HttpRequestType property.</td>
</tr>
<tr>
<td>m_HttpVersion</td>
<td>Holds the value of the HttpVersion property.</td>
</tr>
<tr>
<td>m_RequestedPath</td>
<td>Holds the value of the RequestedPath property.</td>
</tr>
</tbody>
</table>

**See Also**


*Copyright © 2002, The KPD-Team*
**HttpClient.m_HeaderFields Field**

Holds the value of the HeaderFields property.

[Visual Basic] `Private m_HeaderFields As System.Collections.Specialized.StringDictionary`

[C#] `private StringDictionary m_HeaderFields;`

See Also

[HttpClient Class] | [HttpClient Members] | [Org.Mentalis.Proxy.Http Namespace]

Copyright © 2002, The KPD-Team
**HttpClient.m_HttpQuery Field**

Holds the value of the HttpQuery property.

[Visual Basic] `Private m_HttpQuery As String`

[C#] `private string m_HttpQuery;`

**See Also**

[HttpClient Class] | [HttpClient Members] | [Org.Mentalis.Proxy.Http Namespace]

[Copyright © 2002, The KPD-Team]
HttpClient.m_HttpRequestType Field

Holds the value of the HttpRequestType property.

[Visual Basic]
Private m_HttpRequestType As String

[C#]
private string m_HttpRequestType;

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
**HttpClient.m_HttpVersion Field**

Holds the value of the HttpVersion property.

[Visual Basic] 
```vbnet
Private m_HttpVersion As String
```

[C#] 
```csharp
private string m_HttpVersion;
```

See Also

- [HttpClient Class](#) |
- [HttpClient Members](#) |

Copyright © 2002, The KPD-Team
HttpClient.m_RequestedPath Field

Holds the value of the RequestedPath property.

[Visual Basic] Private m_RequestedPath As String

[C#] private string m_RequestedPath;

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
**HttpClient Properties**

The properties of the **HttpClient** class are listed below. For a complete list of **HttpClient** class members, see the **HttpClient Members** topic.

**Public Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestedPath</td>
<td>Gets or sets the requested path.</td>
</tr>
</tbody>
</table>

**Protected Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td>RemoteBuffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HeaderFields</td>
<td>Gets or sets a StringDictionary that stores the header fields.</td>
</tr>
<tr>
<td>HttpQuery</td>
<td>Gets or sets the query string, received from the client.</td>
</tr>
<tr>
<td>HttpRequestType</td>
<td>Gets or sets the HTTP request type.</td>
</tr>
<tr>
<td>HttpVersion</td>
<td>Gets or sets the HTTP version the client uses.</td>
</tr>
</tbody>
</table>

**See Also**


*Copyright © 2002, The KPD-Team*
**HttpClient.HeaderFields Property**

Gets or sets a StringDictionary that stores the header fields.

**[Visual Basic]**

```vbnet
Private Property HeaderFields As StringDictionary
```

**[C#]**

```csharp
private System.Collections.Specialized.StringDictionary
```

**Property Value**

A StringDictionary that stores the header fields.

**See Also**


[Copyright © 2002, The KPD-Team](#)
**HttpClient.HttpQuery Property**

Gets or sets the query string, received from the client.

**[Visual Basic]**

Private Property HttpQuery As String

**[C#]**

private string HttpQuery {get; set;}

**Property Value**

A string representing the HTTP query string.

**See Also**

[HttpClient Class] | [HttpClient Members] | [Org.Mentalis.Proxy.Http Namespace]

[Copyright © 2002, The KPD-Team]
Mentalis.org Proxy Documentation
**HttpClient.HttpRequestType Property**

Gets or sets the HTTP request type.

[Visual Basic]

```vbnet
Private Property HttpRequestType
```

[C#]

```csharp
private string HttpRequestType {get; set;}
```

**Property Value**

A string representing the HTTP request type.

**Remarks**

Usually, this string is set to one of the three following values:

- GET
- POST
- CONNECT

**See Also**

Mentalis.org Proxy Documentation
**HttpClient.HttpVersion Property**

Gets or sets the HTTP version the client uses.

**[Visual Basic]**

```vbnet
Private Property HttpVersion As [SomeType]
```

**[C#]**

```csharp
private string HttpVersion {get; set;}
```

**Property Value**

A string representing the requested HTTP version.

**See Also**

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
**HttpClient-RequestedPath Property**

Gets or sets the requested path.

```plaintext
[Visual Basic]
Public Property RequestedPath

[C#]
public string RequestedPath {get; set;}
```

**Property Value**

A string representing the requested path.

**See Also**

- [HttpClient Class](#)
- [HttpClient Members](#)

**Copyright © 2002, The KPD-Team**
# HttpClient Methods

The methods of the **HttpClient** class are listed below. For a complete list of **HttpClient** class members, see the [HttpClient Members](#) topic.

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispose (inherited from Client)</td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td>Equals (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>StartHandshake</td>
<td>Starts receiving data from the client connection.</td>
</tr>
<tr>
<td>StartRelay (inherited from Client)</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td>ToString</td>
<td>Overloaded. Returns text information about this HttpClient object.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finalize (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>MemberwiseClone (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>OnClientReceive (inherited from Client)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td>OnClientSent (inherited from</td>
<td>Called when we have sent data</td>
</tr>
<tr>
<td>Client)</td>
<td>to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from Client)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from Client)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>IsValidQuery</th>
<th>Checks whether a specified string is a valid HTTP query string.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnConnected</strong></td>
<td>Called when we're connected to the requested remote host.</td>
</tr>
<tr>
<td><strong>OnErrorSent</strong></td>
<td>Called when the Bad Request error has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnOkSent</strong></td>
<td>Called when an OK reply has been sent to the local client.</td>
</tr>
<tr>
<td><strong>OnQuerySent</strong></td>
<td>Called when the HTTP query has been sent to the remote host.</td>
</tr>
<tr>
<td><strong>OnReceiveQuery</strong></td>
<td>Called when we received some data from the client connection.</td>
</tr>
<tr>
<td><strong>ParseQuery</strong></td>
<td>Parses a specified HTTP query into its header fields.</td>
</tr>
<tr>
<td><strong>ProcessQuery</strong></td>
<td>Processes a specified query and connects to the requested HTTP web server.</td>
</tr>
<tr>
<td><strong>RebuildQuery</strong></td>
<td>Rebuilds the HTTP query, starting from the HttpRequestType, RequestedPath, HttpVersion and HeaderFields properties.</td>
</tr>
<tr>
<td><strong>SendBadRequest</strong></td>
<td>Sends a &quot;400 - Bad Request&quot; error to the client.</td>
</tr>
</tbody>
</table>

**See Also**


**Copyright © 2002, The KPD-Team**
Mentalis.org Proxy Documentation
HttpResponse Class

Checks whether a specified string is a valid HTTP query string.

[Visual Basic] Private Function IsValidQuery(ByVal Query As String) As Boolean

[C#] private bool IsValidQuery(string Query);

Parameters

Query
The query to check.

Return Value
True if the specified string is a valid HTTP query, false otherwise.

See Also
HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpClient.OnConnected Method

Called when we're connected to the requested remote host.

[Visual Basic] Private Sub OnConnected( _
    ByVal ar As System.IAsyncResult _
)

[C#] private void OnConnected(IAsyncResult ar);

Parameters

ar
The result of the asynchronous operation.

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Called when the Bad Request error has been sent to the client.

[Visual Basic]  
Private Sub OnErrorSent( ByVal ar As System.IAsyncResult )

[C#]  
private void OnErrorSent( IAsyncResult ar );

Parameters

ar

The result of the asynchronous operation.

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
**HttpClient.OnOkSent Method**

Called when an OK reply has been sent to the local client.

[Visual Basic]```vbnet
Private Sub OnOkSent( _
    ByVal ar As System.IAsyncResult _
)
```

[C#]```csharp
private void OnOkSent(
    IAsyncResult ar
);
```

**Parameters**

*ar*

The result of the asynchronous operation.

**See Also**


Copyright © 2002, The KPD-Team
HttpClient.OnQuerySent Method

Called when the HTTP query has been sent to the remote host.

[Visual Basic] Private Sub OnQuerySent( ByVal ar As System.IAsyncResult )

[C#]
private void OnQuerySent( IAsyncResult ar );

Parameters

ar
The result of the asynchronous operation.

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpClient.OnReceiveQuery Method

Called when we received some data from the client connection.

**[Visual Basic]**
```vbnet
Private Sub OnReceiveQuery(
    ByVal ar As System.IAsyncResult
)
```

**[C#]**
```csharp
private void OnReceiveQuery(
    IAsyncResult ar
);
```

**Parameters**

*ar*

The result of the asynchronous operation.

**See Also**


Copyright © 2002, The KPD-Team
HttpClient.ParseQuery Method

Parses a specified HTTP query into its header fields.


[C#] private StringDictionary ParseQuery( string Query );

Parameters

Query
The HTTP query string to parse.

Return Value
A StringDictionary object containing all the header fields with their data.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified query is null.</td>
</tr>
</tbody>
</table>

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
**HttpClient.ProcessQuery Method**

Processes a specified query and connects to the requested HTTP web server.

```visualbasic
Private Sub ProcessQuery(ByVal Query As String)
```

```csharp
private void ProcessQuery(string Query);
```

**Parameters**

*Query*

A string containing the query to process.

**Remarks**

If there's an error while processing the HTTP request or when connecting to the remote server, the Proxy sends a "400 - Bad Request" error to the client.

**See Also**

Rebuilds the HTTP query, starting from the HttpRequestType, RequestedPath, HttpVersion and HeaderFields properties.

**Return Value**

A string representing the rebuilt HTTP query string.

**See Also**

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpURLConnection is used to send data to the server. Its method `sendBadRequest()` is used to send a '400 - Bad Request' error to the client.

**[Visual Basic]**

```vbnet
Private Sub SendBadRequest()
```

**[C#]**

```csharp
private void SendBadRequest();
```

See Also

 HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpClient.StartHandshake Method

Starts receiving data from the client connection.

[Visual Basic] Overrides Public Sub StartHandshake()

[C#]
public override void StartHandshake();

See Also

HttpClient Class | HttpClient Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
**HttpClient.ToString**

Returns text information about this HttpClient object.

**Overload List**

Returns text information about this HttpClient object.

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

Returns text information about this HttpClient object.

```csharp
public string ToString(bool);
```

**See Also**


Copyright © 2002, The KPD-Team
**HttpClient.ToString Method ()**

Returns text information about this HttpClient object.

**[Visual Basic]**

```vbnet
Overrides Overloads Public Function ToString()
```

**[C#]**

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

**Return Value**

A string representing this HttpClient object.

**See Also**


Copyright © 2002, The KPD-Team
**HttpClient.ToString Method (Boolean)**

Returns text information about this HttpClient object.

[Visual Basic]
```vbnet
Overloads Public Function ToString( ByVal WithUrl As Boolean _ ) As String
```

[C#]
```csharp
public string ToString( bool WithUrl
 );
```

**Parameters**

*WithUrl*

Specifies whether or not to include information about the requested URL.

**Return Value**

A string representing this HttpClient object.

**See Also**


Copyright © 2002, The KPD-Team
HttpListener Class

Listens on a specific port on the proxy server and forwards all incoming HTTP traffic to the appropriate server.

For a list of all members of this type, see HttpListener Members.

System.Object  Listener  HttpListener

[Visual Basic]
NotInheritable Public Class HttpListener
    Inherits Listener
    Implements IDisposable

[C#]
public sealed class HttpListener : Listener,

Requirements

Assembly: Proxy.exe

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
# HttpListener Members

## Public Instance Constructors

| **HttpListener** | Overloaded. Initialize a new instance of the HttpListener class. |

## Public Instance Properties

| **ConstructString** | Returns a string that holds all the construction information for this object. |
| **IsDisposed** (inherited from Listener) | Gets a value indicating whether the Listener has been disposed or not. |
| **Listening** (inherited from Listener) | Gets a value indicating whether the Listener is currently listening or not. |

## Public Instance Methods

<p>| <strong>Dispose</strong> (inherited from Listener) | Disposes of the resources (other than memory) used by the Listener. |
| <strong>Equals</strong> (inherited from <strong>Object</strong>) | Select the method name to go to the Microsoft documentation. |
| <strong>GetClientAt</strong> (inherited from Listener) | Returns the requested client from the client list. |
| <strong>GetClientCount</strong> (inherited from Listener) | Returns the number of clients in the client list. |
| <strong>GetHashCode</strong> (inherited from <strong>Object</strong>) | Select the method name to go to the Microsoft documentation. |
| <strong>GetType</strong> (inherited from <strong>Object</strong>) | Select the method name to go to the Microsoft documentation. |
| <strong>OnAccept</strong> | Called when there's an incoming |</p>
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong> (inherited from <strong>Listener</strong>)</td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Listener</strong>)</td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong> (inherited from <strong>Listener</strong>)</td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

### See Also

HttpListener

Initializes a new instance of the HttpListener class.

Overload List

Initializes a new instance of the HttpListener class.

public HttpListener(int);

Initializes a new instance of the HttpListener class.

public HttpListener(IPAddress,int);

See Also

HttpListener Class  |  HttpListener Members  |  Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpListener Constructor (Int32)

Initializes a new instance of the HttpListener class.

[Visual Basic] Overloads Public Sub New( _
    ByVal Port As Integer _
)

[C#]
public HttpListener(
    int Port
);

Parameters

Port
    The port to listen on.

Remarks

The HttpListener will start listening on all installed network cards.

See Also


Copyright © 2002, The KPD-Team
HttpListener Constructor (IPAddress, Int32)

Initializes a new instance of the HttpListener class.

[Visual Basic] Overloads Public Sub New(_ ByVal Address As System.Net.IPAddress, _ ByVal Port As Integer _)

[C#]
public HttpListener(
    IPAddress Address,
    int Port
);

Parameters

Address
The address to listen on. You can specify IPAddress.Any to listen on all installed network cards.

Port
The port to listen on.

Remarks
For the security of your server, try to avoid to listen on every network card (IPAddress.Any). Listening on a local IP address is usually sufficient and much more secure.

See Also

Copyright © 2002, The KPD-Team
HttpListener Properties

The properties of the `HttpListener` class are listed below. For a complete list of `HttpListener` class members, see the `HttpListener Members` topic.

### Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ConstructString</strong></td>
<td>Returns a string that holds all the construction information for this object.</td>
</tr>
<tr>
<td><strong>IsDisposed</strong></td>
<td>(inherited from <code>Listener</code>)  Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td><strong>Listening</strong></td>
<td>(inherited from <code>Listener</code>)  Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
<td>(inherited from <code>Listener</code>)  Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong></td>
<td>(inherited from <code>Listener</code>)  Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong></td>
<td>(inherited from <code>Listener</code>)  Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>(inherited from <code>Listener</code>)  Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

### See Also

[Internals]

Copyright © 2002, The KPD-Team
HttpListener.ConstructString Property

Returns a string that holds all the construction information for this object.

[Visual Basic] Overrides Public ReadOnly Property ConstructString As String

[C#]
public string ConstructString {override get;}

Property Value
A string that holds all the construction information for this object.

See Also

HttpListener Class | HttpListener Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpListener Methods

The methods of the **HttpListener** class are listed below. For a complete list of **HttpListener** class members, see the [HttpListener Members](#) topic.

**Public Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Listener</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Listener.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming client connection waiting to be accepted.</td>
</tr>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

**Protected Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Listener</strong>)</td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>RemoveClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong> (inherited from <strong>Listener</strong>)</td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

**See Also**

[HttpListener Class] [Org.Mentalis.Proxy.Http Namespace]

Copyright © 2002, The KPD-Team
HttpListener.OnAccept Method

Called when there's an incoming client connection waiting to be accepted.

```visual-basic
 Overrides Public Sub OnAccept(ByVal ar As System.IAsyncResult)
```

```csharp
 public override void OnAccept(IAsyncResult ar);
```

Parameters

`ar`

The result of the asynchronous operation.

See Also

HttpListener Class | HttpListener Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
HttpListener.ToString Method

Returns a string representation of this object.

[Visual Basic] Overrides Public Function ToString()

[C#]
public override string ToString();

Return Value

A string with information about this object.

See Also

HttpListener Class | HttpListener Members | Org.Mentalis.Proxy.Http Namespace

Copyright © 2002, The KPD-Team
### Org.Mentalis.Proxy.PortMap Namespace

#### Namespace hierarchy

#### Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PortMapClient</td>
<td>Relays data between a remote host and a local client.</td>
</tr>
<tr>
<td>PortMapListener</td>
<td>Listens on a specific port on the proxy server and forwards all incoming data to a specific port on another server.</td>
</tr>
</tbody>
</table>

Copyright © 2002, The KPD-Team
PortMapClient Class

Relays data between a remote host and a local client.

For a list of all members of this type, see PortMapClient Members.

System.Object  Client
PortMapClient

[Visual Basic]
NotInheritable Public Class PortMapClient
    Inherits Client
    Implements IDisposable

[C#]
public sealed class PortMapClient : Client, :

Requirements

Assembly: Proxy.exe

See Also

PortMapClient Members  |  Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
<table>
<thead>
<tr>
<th>Public Instance Constructors</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PortMapClient Constructor</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Public Instance Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
</tr>
<tr>
<td><strong>StartRelay</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Protected Instance Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td><strong>RemoteBuffer</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td>Method</td>
</tr>
<tr>
<td>---------------------------------------------</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from <strong>Client</strong>)</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from <strong>Client</strong>)</td>
</tr>
</tbody>
</table>

**Private Instance Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_MapTo</strong></td>
<td>Holds the value of the MapTo property.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MapTo</strong></td>
<td>Gets or sets the IP EndPoint to map all incoming traffic to.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnConnected</strong></td>
<td>Called when the socket is connected to the remote host.</td>
</tr>
</tbody>
</table>
PortMapClient Constructor

Initializes a new instance of the PortMapClient class.

### Parameters

**ClientSocket**


**Destroyer**

The callback method to be called when this Client object disconnects from the local client and the remote server.

**MapTo**

The IP EndPoint to send the incoming data to.

### See Also

PortMapClient Class | PortMapClient Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapClient Fields

The fields of the PortMapClient class are listed below. For a complete list of PortMapClient class members, see the PortMapClient Members topic.

Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_MapTo</td>
<td>Holds the value of the MapTo property.</td>
</tr>
</tbody>
</table>

See Also

PortMapClient Class | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapClient.m_MapTo Field

Holds the value of the MapTo property.

[Visual Basic] Private m_MapTo As System.Net.

[C#]
private IPEndPoint m_MapTo;

See Also

PortMapClient Class | PortMapClient Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapClient Properties

The properties of the PortMapClient class are listed below. For a complete list of PortMapClient class members, see the PortMapClient Members topic.

Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td>RemoteBuffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MapTo</td>
<td>Gets or sets the IP EndPoint to map all incoming traffic to.</td>
</tr>
</tbody>
</table>

See Also

PortMapClient Class | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapClient.MapTo Property

Gets or sets the IP EndPoint to map all incoming traffic to.

[Visual Basic] Private Property MapTo As System.Net.IPEndPoint

[C#] private System.Net.IPEndPoint MapTo {get; set;}

Property Value

An IPEndPoint that holds the IP address and port to use when redirecting incoming traffic.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also

PortMapClient Class | PortMapClient Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
PortMapClient Methods

The methods of the **PortMapClient** class are listed below. For a complete list of **PortMapClient** class members, see the **PortMapClient Members** topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong></td>
<td>(inherited from <strong>Client</strong>) Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td></td>
</tr>
<tr>
<td><strong>StartRelay</strong></td>
<td>(inherited from <strong>Client</strong>) Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns text information about this PortMapClient object.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong></td>
<td>(inherited from <strong>Client</strong>) Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong></td>
<td>(inherited from <strong>Object</strong>) Called when we have sent data</td>
</tr>
</tbody>
</table>
to the local client. When all the data has been sent, we will start receiving again from the remote host.

<table>
<thead>
<tr>
<th><strong>OnRemoteReceive</strong> (inherited from <strong>Client</strong>)</th>
<th>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

| **OnConnected** | Called when the socket is connected to the remote host. |

**See Also**

[PortMapClient Class] [Org.Mentalis.Proxy.PortMap Namespace]

Copyright © 2002, The KPD-Team
Called when the socket is connected to the remote host.

```Visual Basic
Private Sub OnConnected(_
    ByVal ar As System.IAsyncResult _
)
```

```C#
private void OnConnected(
    IAsyncResult ar
);
```

**Parameters**

`ar`
The result of the asynchronous operation.

**Remarks**

When the socket is connected to the remote host, the PortMapClient begins relaying traffic between the host and the client, until one of them closes the connection.

**See Also**

- PortMapClient Class | PortMapClient Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapClient.StartHandshake Method

Starts connecting to the remote host.

[Visual Basic] Overrides Public Sub StartHandshake()

[C#]
public override void StartHandshake();

See Also

PortMapClient Class | PortMapClient Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapClient.ToString Method

Returns text information about this PortMapClient object.

[Visual Basic] Overrides Public Function ToString()

[C#]
public override string ToString();

Return Value

A string representing this PortMapClient object.

See Also

PortMapClient Class | PortMapClient Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapListener Class

Listens on a specific port on the proxy server and forwards all incoming data to a specific port on another server.

For a list of all members of this type, see PortMapListener Members.

System.Object  Listener
  PortMapListener

[Visual Basic]
NotInheritable Public Class PortMapListener
  Inherits Listener
  Implements IDisposable

[C#]
public sealed class PortMapListener : Listener

Requirements


Assembly: Proxy.exe

See Also

PortMapListener Members  |  Org.Mentalis.Proxy.PortMap Namespace
Copyright © 2002, The KPD-Team
# PortMapListener Members

## Public Instance Constructors

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PortMapListener</strong></td>
<td>Overloaded. Initialize a new instance of the PortMapListener class.</td>
</tr>
</tbody>
</table>

## Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ConstructString</strong></td>
<td>Returns a string that holds all the construction information for this object.</td>
</tr>
<tr>
<td><strong>IsDisposed</strong> (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td><strong>Listening</strong> (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from Listener)</td>
<td>Disposes of the resources (other than memory) used by the Listener.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong> (inherited from Listener)</td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong> (inherited from Listener)</td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Start</strong> (inherited from Listener)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong> (inherited from Listener)</td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong> (inherited from Listener)</td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong> (inherited from Listener)</td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong> (inherited from Listener)</td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from Listener)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from Listener)</td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong> (inherited from Listener)</td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong> (inherited from Listener)</td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_MapTo</strong></td>
<td>Holds the value of the MapTo property.</td>
</tr>
</tbody>
</table>
Private Instance Properties

| MapTo   | Gets or sets the IP EndPoint to map all incoming traffic to. |

See Also

PortMapListener Class | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapListener

Initializes a new instance of the PortMapListener class.

Overload List

Initializes a new instance of the PortMapListener class.

public PortMapListener(int, IPEndPoint);

Initializes a new instance of the PortMapListener class.

public PortMapListener(IPAddress, int, IPEndPoint);

Initializes a new instance of the PortMapListener class.

public PortMapListener(IPAddress, int, IPAddress, int);

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapListener Constructor (Int32, IPEndPoint)

Initializes a new instance of the PortMapListener class.

[Visual Basic] Overloads Public Sub New(_
ByVal Port As Integer, _
ByVal MapToIP As System.Net.IPEndPoint _
)

[C#]
public PortMapListener(
    int Port,
    IPEndPoint MapToIP
);

Parameters

Port
The port to listen on.

MapToIP
The address to forward to.

Remarks
The object will listen on all network addresses on the computer.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentException</td>
<td>Port is not positive.</td>
</tr>
<tr>
<td>System.ArgumentNullException</td>
<td>MapToIP is null.</td>
</tr>
</tbody>
</table>

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace | PortMapListener Constructor Overload List

Copyright © 2002, The KPD-Team
PortMapListener Constructor (IPAddress, Int32, IPEndPoint)

Initializes a new instance of the PortMapListener class.

[Visual Basic] Overloads Public Sub New( _
    ByVal Address As System.Net.IPAddress, _
    ByVal Port As Integer, _
    ByVal MapToIP As System.Net.IPEndPoint _
)

[C#]
public PortMapListener(
    IPAddress Address,
    int Port,
    IPEndPoint MapToIP
);

Parameters

Address
  The network address to listen on.

Port
  The port to listen on.

MapToIP
  The address to forward to.

Remarks

For security reasons, Address should not be IPAddress.Any.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Address or MapToIP is null.</td>
</tr>
<tr>
<td>System.ArgumentException</td>
<td>Port is not positive.</td>
</tr>
</tbody>
</table>

See Also
PortMapListener Constructor (IPAddress, Int32, IPAddress, Int32)

Initializes a new instance of the PortMapListener class.

[Visual Basic]
Overloads Public Sub New( _
    ByVal Address As System.Net.IPAddress, _
    ByVal Port As Integer, _
    ByVal MapToAddress As System.Net.IPAddress,
    ByVal MapToPort As Integer _
)

[C#]

public PortMapListener(
    IPAddress Address,
    int Port,
    IPAddress MapToAddress,
    int MapToPort
);

Parameters

Address
    The network address to listen on.

Port
    The port to listen on.

MapToAddress
    The IP address to forward to.

MapToPort
    The port to forward to.

Remarks

For security reasons, Address should not be IPAddress.Any.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Exception Class</td>
<td>Exception Message</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>System.ArgumentNullException</td>
<td>Address or MapToAddress is null.</td>
</tr>
<tr>
<td>System.ArgumentException</td>
<td>Port or MapToPort is invalid.</td>
</tr>
</tbody>
</table>

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace | PortMapListener Constructor Overload List

Copyright © 2002, The KPD-Team
**PortMapListener Fields**

The fields of the `PortMapListener` class are listed below. For a complete list of `PortMapListener` class members, see the `PortMapListener Members` topic.

<table>
<thead>
<tr>
<th>Private Instance Fields</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>m_MapTo</code></td>
<td>Holds the value of the MapTo property.</td>
</tr>
</tbody>
</table>

**See Also**


---

Copyright © 2002, The KPD-Team
PortMapListener.m_MapTo Field

Holds the value of the MapTo property.

[Visual Basic] Private m_MapTo As System.Net.IPEndPoint

[C#] private IPEndPoint m_MapTo;

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
The properties of the PortMapListener class are listed below. For a complete list of PortMapListener class members, see the PortMapListener Members topic.

Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ConstructString</strong></td>
<td>Returns a string that holds all the construction information for this object.</td>
</tr>
<tr>
<td><strong>IsDisposed</strong> (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td><strong>Listening</strong> (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong> (inherited from Listener)</td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong> (inherited from Listener)</td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong> (inherited from Listener)</td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong> (inherited from Listener)</td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MapTo</strong></td>
<td>Gets or sets the IP EndPoint to map all incoming traffic to.</td>
</tr>
</tbody>
</table>

See Also

PortMapListener Class | Org.Mentalis.Proxy.PortMap Namespace
Copyright © 2002, The KPD-Team
PortMapListener.ConstructString Property

Returns a string that holds all the construction information for this object.

[Visual Basic]
Overrides Public ReadOnly Property

[C#]
public string ConstructString {override get;}

Property Value

A string that holds all the construction information for this object.

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapListener.MapTo Property

Gets or sets the IP EndPoint to map all incoming traffic to.

[Visual Basic] Private Property MapTo As System.Net.IPEndPoint

[C#] private System.Net.IPEndPoint MapTo {get; set;}

Property Value

An IPEndPoint that holds the IP address and port to use when redirecting incoming traffic.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
### PortMapListener Methods

The methods of the **PortMapListener** class are listed below. For a complete list of **PortMapListener** class members, see the [PortMapListener Members](#) topic.

#### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Listener</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Listener.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming client connection waiting to be accepted.</td>
</tr>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

#### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Listener</strong>)</td>
<td>Finalizes the Listener.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>RemoveClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Removes the specified Client from the client list.</td>
</tr>
<tr>
<td><strong>Restart</strong> (inherited from <strong>Listener</strong>)</td>
<td>Restarts listening on the selected IP address and port.</td>
</tr>
</tbody>
</table>

**See Also**

PortMapListener.OnAccept Method

Called when there's an incoming client connection waiting to be accepted.

[Visual Basic] Overrides Public Sub OnAccept(  
    ByVal ar As System.IAsyncResult  
)

[C#]
public override void OnAccept(  
    IAsyncResult ar  
);

Parameters

ar  
The result of the asynchronous operation.

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
PortMapListener.ToString Method

Returns a string representation of this object.

[Visual Basic] Overrides Public Function ToString()

[C#]
public override string ToString();

Return Value

A string with information about this object.

See Also

PortMapListener Class | PortMapListener Members | Org.Mentalis.Proxy.PortMap Namespace

Copyright © 2002, The KPD-Team
Org.Mentalis.Proxy.Socks Namespace

Namespace hierarchy

Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Socks4Handler</td>
<td>Implements the SOCKS4 and SOCKS4a protocols.</td>
</tr>
<tr>
<td>Socks5Handler</td>
<td>Implements the SOCKS5 protocol.</td>
</tr>
<tr>
<td>SocksClient</td>
<td>Relays data between a remote host and a local client, using the SOCKS protocols.</td>
</tr>
<tr>
<td>SocksHandler</td>
<td>Implements a specific version of the SOCKS protocol.</td>
</tr>
<tr>
<td>SocksListener</td>
<td>Listens on a specific port on the proxy server for incoming SOCKS4 and SOCKS5 requests.</td>
</tr>
</tbody>
</table>

Delegates

<table>
<thead>
<tr>
<th>Delegate</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NegotiationCompleteDelegate</td>
<td>Defines the signature of the method that's called when the SOCKS negotiation is complete.</td>
</tr>
</tbody>
</table>

Copyright © 2002, The KPD-Team
Defines the signature of the method that's called when the SOCKS negotiation is complete.

**Parameters**

*Success*
Indicates whether the negotiation was successful or not.

*Remote*
The connection with the remote server.

**Requirements**


*Assembly:* Proxy.exe

**See Also**

Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**Socks4Handler Class**

Implements the SOCKS4 and SOCKS4a protocols.

For a list of all members of this type, see [Socks4Handler Members](#).

**System.Object**  **SocksHandler**

**Socks4Handler**

[Visual Basic]

```vbnet
NotInheritable Class Socks4Handler
    Inherits SocksHandler
End Class
```

[C#]

```csharp
sealed class Socks4Handler : SocksHandler
```

**Requirements**

**Namespace:** [Org.Mentalis.Proxy.Socks Namespace](#)

**Assembly:** Proxy.exe

**See Also**


Copyright © 2002, The KPD-Team
# Socks4Handler Members

## Public Instance Constructors

<table>
<thead>
<tr>
<th>Socks4Handler Constructor</th>
<th>Initializes a new instance of the Socks4Handler class.</th>
</tr>
</thead>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartNegotiating</strong> (inherited from SocksHandler)</td>
<td>Starts accepting bytes from the client.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AcceptSocket</strong> (inherited from SocksHandler)</td>
<td>Gets or sets the socket that is used to accept incoming connections.</td>
</tr>
<tr>
<td><strong>Buffer</strong> (inherited from SocksHandler)</td>
<td>Gets a buffer that can be used when receiving bytes from the client.</td>
</tr>
<tr>
<td><strong>Bytes</strong> (inherited from SocksHandler)</td>
<td>Gets or sets a byte array that can be used to store received bytes from the client.</td>
</tr>
<tr>
<td><strong>Connection</strong> (inherited from SocksHandler)</td>
<td>Gets or sets the connection with the client.</td>
</tr>
<tr>
<td><strong>RemoteBindIP</strong> (inherited from SocksHandler)</td>
<td>Gets or sets the IP address of the requested remote server.</td>
</tr>
<tr>
<td><strong>RemoteConnection</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Gets or sets the connection with the remote host.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

**Protected Instance Methods**

| **AddBytes** (inherited from **SocksHandler**) | Adds some bytes to a byte array. |
| **Dispose** | Overloaded. Sends a reply to the client connection and disposes it afterwards. |
| **Finalize** (inherited from **Object**) | Select the method name to go to the Microsoft documentation. |
| **IsValidRequest** | Checks whether a specific request is a valid SOCKS request or not. |
| **MemberwiseClone** (inherited from **Object**) | Select the method name to go to the Microsoft documentation. |
| **OnAccept** | Called when there's an incoming connection in the AcceptSocket queue. |
| **OnDisposeBad** (inherited from **SocksHandler**) | Called when a negative reply has been sent to the client. |
| **OnDisposeGood** (inherited from **SocksHandler**) | Called when an OK reply has been sent to the client. |
| **OnReceiveBytes** (inherited from **SocksHandler**) | Called when we receive some bytes from the client. |
| **OnStartAccept** (inherited from **SocksHandler**) | Called when the AcceptSocket should start accepting incoming connections. |
| **ProcessRequest** | Processes a SOCKS request from a client. |

**Private Instance Methods**

| **OnConnected** | Called when we're successfully |
See Also

Socks4Handler Class | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks4Handler Constructor

Initializes a new instance of the Socks4Handler class.

```visualbasic
Public Sub New( _
   ByVal ClientConnection As System.Net.Sockets.Socket,
```

```csharp
public Socks4Handler(
   Socket ClientConnection,
   NegotiationCompleteDelegate Callback
);
```

Parameters

*ClientConnection*

The connection with the client.

*Callback*

The method to call when the SOCKS negotiation is complete.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Callback is null.</td>
</tr>
</tbody>
</table>

See Also

[Socks4Handler Class][1] | [Socks4Handler Members][2] | [Org.Mentalis.Proxy.Socks Namespace][3]

Copyright © 2002, The KPD-Team

---

[1]: #/Socks4HandlerClass
[2]: #/Socks4HandlerMembers
Mentalis.org Proxy Documentation
Socks4Handler Methods

The methods of the **Socks4Handler** class are listed below. For a complete list of **Socks4Handler** class members, see the **Socks4Handler Members** topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartNegotiating</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Starts accepting bytes from the client.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Adds some bytes to a byte array.</td>
</tr>
<tr>
<td><strong>Dispose</strong></td>
<td>Overloaded. Sends a reply to the client connection and disposes it afterwards.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>IsValidRequest</strong></td>
<td>Checks whether a specific request is a valid SOCKS request or not.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming connection in the AcceptSocket</td>
</tr>
</tbody>
</table>
queue.

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Inherited From</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnDisposeBad</strong> (inherited from SocksHandler)</td>
<td>Called when a negative reply has been sent to the client.</td>
<td></td>
</tr>
<tr>
<td><strong>OnDisposeGood</strong> (inherited from SocksHandler)</td>
<td>Called when an OK reply has been sent to the client.</td>
<td></td>
</tr>
<tr>
<td><strong>OnReceiveBytes</strong> (inherited from SocksHandler)</td>
<td>Called when we receive some bytes from the client.</td>
<td></td>
</tr>
<tr>
<td><strong>OnStartAccept</strong> (inherited from SocksHandler)</td>
<td>Called when the AcceptSocket should start accepting incoming connections.</td>
<td></td>
</tr>
<tr>
<td><strong>ProcessRequest</strong></td>
<td>Processes a SOCKS request from a client.</td>
<td></td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OnConnected</strong></td>
<td>Called when we're successfully connected to the remote host.</td>
</tr>
</tbody>
</table>

**See Also**

[Socks4Handler Class] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
Sends a reply to the client connection and disposes it afterwards.

**[Visual Basic]**

```vbnet
Overrides Overloads Protected Sub Dispose(ByVal Value As Byte)
```

**[C#]**

```csharp
protected override void Dispose(byte Value);
```

### Parameters

**Value**
A byte that contains the reply code to send to the client.

### See Also

[Socks4Handler Class] | [Socks4Handler Members] | [Org.Mentalis.Proxy.Socks Namespace] | [Socks4Handler.Dispose Overload List]

Copyright © 2002, The KPD-Team
Socks4Handler.IsValidRequest Method

Checks whether a specific request is a valid SOCKS request or not.

[Visual Basic]
Overrides Protected Function Is:
ByVal Request As Byte() _
) As Boolean

[C#]
protected override bool IsValidRequest(
byte[] Request
);

Parameters

Request
The request array to check.

Return Value

True is the specified request is valid, false otherwise

See Also

Socks4Handler Class | Socks4Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks4Handler.OnAccept Method

Called when there's an incoming connection in the AcceptSocket queue.

[Visual Basic] Overrides Protected Sub OnAccept(ByVal ar As System.IAsyncResult)  

[C#]  
protected override void OnAccept(IAsyncResult ar);  

Parameters

ar  
The result of the asynchronous operation.

See Also

Socks4Handler Class | Socks4Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks4Handler.OnConnected Method

Called when we're successfully connected to the remote host.

```vbnet
Private Sub OnConnected(ByVal ar As System.IAsyncResult)
```

```csharp
private void OnConnected(IAsyncResult ar);
```

Parameters

**ar**

The result of the asynchronous operation.

See Also

Socks4Handler Class | Socks4Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Socks4Handler.ProcessRequest Method

Processes a SOCKS request from a client.

[Visual Basic] Overrides Protected Sub ProcessRequest(ByVal Request As Byte() _
)

[C#]
protected override void ProcessRequest(byte[] Request
);

Parameters

Request
The request to process.

See Also

Socks4Handler Class | Socks4Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Socks5Handler Class

Implements the SOCKS5 protocol.
For a list of all members of this type, see Socks5Handler Members.

[SocksHandler](#)  Socks5Handler

**[Visual Basic]**
```
NotInheritable Class Socks5Handler
  Inherits SocksHandler
```

**[C#]**
```
sealed class Socks5Handler : SocksHandler
```

Requirements

**Namespace:** Org.Mentalis.Proxy.Socks Namespace

**Assembly:** Proxy.exe

See Also

Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
## Socks5Handler Members

### Public Instance Constructors

<table>
<thead>
<tr>
<th>Socks5Handler</th>
<th>Overloaded. Initialize a new instance of the Socks5Handler class.</th>
</tr>
</thead>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>StartNegotiating</td>
<td>Starts accepting bytes from the client.</td>
</tr>
<tr>
<td>ToString</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcceptSocket</td>
<td>Gets or sets the socket that is used to accept incoming connections.</td>
</tr>
<tr>
<td>Buffer</td>
<td>Gets a buffer that can be used when receiving bytes from the client.</td>
</tr>
<tr>
<td>Bytes</td>
<td>Gets or sets a byte array that can be used to store received bytes from the client.</td>
</tr>
<tr>
<td>Connection</td>
<td>Gets or sets the connection with the client.</td>
</tr>
<tr>
<td>RemoteBindIP</td>
<td>Gets or sets the IP address of the requested remote server.</td>
</tr>
<tr>
<td><strong>RemoteConnection</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Gets or sets the connection with the remote host.</td>
</tr>
<tr>
<td>------------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th><strong>AddBytes</strong> (inherited from <strong>SocksHandler</strong>)</th>
<th>Adds some bytes to a byte array.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong></td>
<td>Overloaded. Sends a reply to the client connection and disposes it afterwards.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>IsValidRequest</strong></td>
<td>Checks whether a specific request is a valid SOCKS request or not.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming connection in the AcceptSocket queue.</td>
</tr>
<tr>
<td><strong>OnDisposeBad</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Called when a negative reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnDisposeGood</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Called when an OK reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnReceiveBytes</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Called when we receive some bytes from the client.</td>
</tr>
<tr>
<td><strong>OnStartAccept</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Called when the AcceptSocket should start accepting incoming connections.</td>
</tr>
<tr>
<td><strong>ProcessRequest</strong></td>
<td>Processes a SOCKS request from a client and selects an authentication method.</td>
</tr>
</tbody>
</table>

### Private Instance Fields
### m_AuthList
Holds the value of the AuthList property.

### m_AuthMethod
Holds the value of the AuthMethod property.

#### Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthList</td>
<td>Gets or sets the AuthenticationList object to use when trying to authenticate the SOCKS client.</td>
</tr>
<tr>
<td>AuthMethod</td>
<td>Gets or sets the AuthBase object to use when trying to authenticate the SOCKS client.</td>
</tr>
</tbody>
</table>

#### Private Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsValidQuery</td>
<td>Checks whether a specified query is a valid query or not.</td>
</tr>
<tr>
<td>OnAuthenticationComplete</td>
<td>Called when the authentication is complete.</td>
</tr>
<tr>
<td>OnAuthSent</td>
<td>Called when client has been notified of the selected authentication method.</td>
</tr>
<tr>
<td>OnConnected</td>
<td>Called when we're successfully connected to the remote host.</td>
</tr>
<tr>
<td>OnRecvRequest</td>
<td>Called when we received the request of the client.</td>
</tr>
<tr>
<td>ProcessQuery</td>
<td>Processes a received query.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
Socks5Handler

Initializes a new instance of the Socks5Handler class.

Overload List

Initializes a new instance of the Socks5Handler class.

   public Socks5Handler(Socket, NegotiationCompleteDelegate, Auther

Initializes a new instance of the Socks5Handler class.

   public Socks5Handler(Socket, NegotiationCompleteDelegate);

See Also

Socks5Handler Class  |  Socks5Handler Members  |  Org.Mentalis.Proxy.Socks Namespace
Socks5Handler Constructor (Socket, NegotiationCompleteDelegate, AuthenticationList)

Initializes a new instance of the Socks5Handler class.

[Visual Basic] Overloads Public Sub New( _
    ByVal ClientConnection As System.Net.Sockets.Socket,
    ByVal Callback As Org.Mentalis.Proxy.Socks.NegotiationCompleteDelegate,
)

[C#]
public Socks5Handler(
    Socket ClientConnection,
    NegotiationCompleteDelegate Callback,
    AuthenticationList AuthList
);

Parameters

ClientConnection
The connection with the client.

Callback
The method to call when the SOCKS negotiation is complete.

AuthList
The authentication list to use when clients connect.

Remarks

If the AuthList parameter is null, no authentication will be required when a client connects to the proxy server.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Callback is null.</td>
</tr>
</tbody>
</table>

See Also
Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace | Socks5Handler Constructor Overload List

Copyright © 2002, The KPD-Team
Socks5Handler Constructor (Socket, NegotiationCompleteDelegate)

Initializes a new instance of the Socks5Handler class.

[Visual Basic]
Overloads Public Sub New( _
    ByVal ClientConnection As System.Net.Sockets.Socket,

[C#]
public Socks5Handler(
    Socket ClientConnection,
    NegotiationCompleteDelegate Callback
);

Parameters

ClientConnection
The connection with the client.

Callback
The method to call when the SOCKS negotiation is complete.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Callback is null.</td>
</tr>
</tbody>
</table>

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace | Socks5Handler Constructor Overload List

Copyright © 2002, The KPD-Team
The fields of the **Socks5Handler** class are listed below. For a complete list of **Socks5Handler** class members, see the [Socks5Handler Members](#) topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AuthList</td>
<td>Holds the value of the AuthList property.</td>
</tr>
<tr>
<td>m_AuthMethod</td>
<td>Holds the value of the AuthMethod property.</td>
</tr>
</tbody>
</table>

**See Also**

Mentalis.org Proxy Documentation
### Socks5Handler.m_AuthList Field

Holds the value of the AuthList property.

```visual-basic
```

```csharp
private AuthenticationList m_AuthList;
```

See Also

[Socks5Handler Class] | [Socks5Handler Members] | [Org.Mentalis.Proxy.Socks Namespace]

[Copyright © 2002, The KPD-Team]
**Socks5Handler.m_AuthMethod Field**

Holds the value of the AuthMethod property.


[C#] `private AuthBase m_AuthMethod;`

See Also

[Socks5Handler Class] | [Socks5Handler Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
## Socks5Handler Properties

The properties of the Socks5Handler class are listed below. For a complete list of Socks5Handler class members, see the Socks5Handler Members topic.

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AcceptSocket</code></td>
<td>(inherited from SocksHandler) Gets or sets the socket that is used to accept incoming connections.</td>
</tr>
<tr>
<td><code>Buffer</code></td>
<td>(inherited from SocksHandler) Gets a buffer that can be used when receiving bytes from the client.</td>
</tr>
<tr>
<td><code>Bytes</code></td>
<td>(inherited from SocksHandler) Gets or sets a byte array that can be used to store received bytes from the client.</td>
</tr>
<tr>
<td><code>Connection</code></td>
<td>(inherited from SocksHandler) Gets or sets the connection with the client.</td>
</tr>
<tr>
<td><code>RemoteBindIP</code></td>
<td>(inherited from SocksHandler) Gets or sets the IP address of the requested remote server.</td>
</tr>
<tr>
<td><code>RemoteConnection</code></td>
<td>(inherited from SocksHandler) Gets or sets the connection with the remote host.</td>
</tr>
</tbody>
</table>

### Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AuthList</code></td>
<td>Gets or sets the AuthenticationList object to use when trying to authenticate the SOCKS client.</td>
</tr>
<tr>
<td><code>AuthMethod</code></td>
<td>Gets or sets the the AuthBase object to use when trying to authenticate the SOCKS client.</td>
</tr>
</tbody>
</table>

### See Also

Socks5Handler.AuthList Property

Gets or sets the AuthenticationList object to use when trying to authenticate the SOCKS client.


[C#] private Authentication.AuthenticationList AuthList {get;set;}

Property Value

The AuthenticationList object to use when trying to authenticate the SOCKS client.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks5Handler.AuthMethod Property

Gets or sets the AuthBase object to use when trying to authenticate the SOCKS client.

[Visual Basic]
Private Property AuthMethod As Authentication.AuthBase

[C#]
private Authentication.AuthBase AuthMethod { get; set; }

Property Value

The AuthBase object to use when trying to authenticate the SOCKS client.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
The methods of the **Socks5Handler** class are listed below. For a complete list of **Socks5Handler** class members, see the [Socks5Handler Members] topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartNegotiating</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Starts accepting bytes from the client.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong> (inherited from <strong>SocksHandler</strong>)</td>
<td>Adds some bytes to a byte array.</td>
</tr>
<tr>
<td><strong>Dispose</strong></td>
<td>Overloaded. Sends a reply to the client connection and disposes it afterwards.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>IsValidRequest</strong></td>
<td>Checks whether a specific request is a valid SOCKS request or not.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming connection in the AcceptSocket</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>OnDisposeBad</strong> (inherited from SocksHandler)</td>
<td>Called when a negative reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnDisposeGood</strong> (inherited from SocksHandler)</td>
<td>Called when an OK reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnReceiveBytes</strong> (inherited from SocksHandler)</td>
<td>Called when we receive some bytes from the client.</td>
</tr>
<tr>
<td><strong>OnStartAccept</strong> (inherited from SocksHandler)</td>
<td>Called when the AcceptSocket should start accepting incoming connections.</td>
</tr>
<tr>
<td><strong>ProcessRequest</strong></td>
<td>Processes a SOCKS request from a client and selects an authentication method.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IsValidQuery</strong></td>
<td>Checks whether a specified query is a valid query or not.</td>
</tr>
<tr>
<td><strong>OnAuthenticationComplete</strong></td>
<td>Called when the authentication is complete.</td>
</tr>
<tr>
<td><strong>OnAuthSent</strong></td>
<td>Called when client has been notified of the selected authentication method.</td>
</tr>
<tr>
<td><strong>OnConnected</strong></td>
<td>Called when we're successfully connected to the remote host.</td>
</tr>
<tr>
<td><strong>OnRecvRequest</strong></td>
<td>Called when we received the request of the client.</td>
</tr>
<tr>
<td><strong>ProcessQuery</strong></td>
<td>Processes a received query.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
Sends a reply to the client connection and disposes it afterwards.

**Parameters**

- **Value**
  A byte that contains the reply code to send to the client.

**See Also**

- Socks5Handler Class |
- Socks5Handler Members |
- Socks5Handler.Dispose

Copyright © 2002, The KPD-Team
Socks5Handler.IsValidQuery Method

Checks whether a specified query is a valid query or not.

[Visual Basic]
Private Function IsValidQuery(ByVal Query As Byte()) As Boolean

[C#]
private bool IsValidQuery(byte[] Query);

Parameters

Query
The query to check.

Return Value

True if the query is valid, false otherwise.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks5Handler.IsValidRequest Method

Checks whether a specific request is a valid SOCKS request or not.

[Visual Basic] Overrides Protected Function Is_valid_request(ByVal Request As Byte()_) As Boolean

[C#] protected override bool IsValidRequest(byte[] Request);

Parameters

Request
The request array to check.

Return Value

True is the specified request is valid, false otherwise

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks5Handler.OnAccept Method

Called when there's an incoming connection in the AcceptSocket queue.

[Visual Basic] Overrides Protected Sub OnAccept(ByVal ar As System.IAsyncResult)
)

[C#]
protected override void OnAccept(IAsyncResult ar);

Parameters

ar
The result of the asynchronous operation.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks5Handler.OnAuthenticationComplete Method

Called when the authentication is complete.

[Visual Basic]
Private Sub OnAuthenticationComplete(ByVal Success As Boolean)
)

[C#]
private void OnAuthenticationComplete(
bool Success
);

Parameters

Success
    Indicates whether the authentication was successful or not.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks5Handler.OnAuthSent Method

Called when client has been notified of the selected authentication method.

[Visual Basic]
Private Sub OnAuthSent(_
    ByVal ar As System.IAsyncResult _
)

[C#]
private void OnAuthSent(
    IAsyncResult ar
);

Parameters

ar

The result of the asynchronous operation.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Socks5Handler.OnConnected Method

Called when we're successfully connected to the remote host.

[Visual Basic] Private Sub OnConnected( ByVal ar As System.IAsyncResult )

[C#] private void OnConnected( IAsyncResult ar );

Parameters

ar
The result of the asynchronous operation.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Called when we received the request of the client.

[Visual Basic] Private Sub OnRecvRequest( ByVal ar As System.IAsyncResult )

[C#] private void OnRecvRequest( IAsyncResult ar );

Parameters

*ar*  
The result of the asynchronous operation.

See Also

[Socks5Handler Class] [Socks5Handler Members] [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Processes a received query.

### Parameters

**Query**
- The query to process.

### See Also

*Copyright © 2002, The KPD-Team*
Socks5Handler.ProcessRequest Method

Processes a SOCKS request from a client and selects an authentication method.

[Visual Basic] Overrides Protected Sub ProcessRequest(ByVal Request As Byte() _)

[C#] protected override void ProcessRequest(byte[] Request);

Parameters

Request
The request to process.

See Also

Socks5Handler Class | Socks5Handler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
SocksClient Class

Relays data between a remote host and a local client, using the SOCKS protocols.

For a list of all members of this type, see SocksClient Members.

System.Object  Client  SocksClient

[Visual Basic]
NotInheritable Public Class SocksClient
    Inherits Client
    Implements IDisposable

[C#]
public sealed class SocksClient : Client, IDisposable

Remarks

This class implements the SOCKS4, SOCKS4a and SOCKS5 protocols. If the MustAuthenticate property is set, only SOCKS5 connections are allowed and the AuthList parameter of the constructor should not be null.

Requirements


Assembly: Proxy.exe

See Also

SocksClient Members  |  Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
## SocksClient Members

### Public Instance Constructors

<table>
<thead>
<tr>
<th>SocksClient Constructor</th>
<th>Initializes a new instance of the SocksClient class.</th>
</tr>
</thead>
</table>

### Public Instance Properties

<table>
<thead>
<tr>
<th>MustAuthenticate</th>
<th>Gets or sets the SOCKS handler to be used when communicating with the client.</th>
</tr>
</thead>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Dispose (inherited from Client)</th>
<th>Disposes of the resources (other than memory) used by the Client.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>StartHandshake</td>
<td>Starts communication with the client.</td>
</tr>
<tr>
<td>StartRelay (inherited from Client)</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td>ToString</td>
<td>Returns text information about this SocksClient object.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Buffer (inherited from Client)</th>
<th>Gets the buffer to store all the incoming data from the local client.</th>
</tr>
</thead>
</table>
**RemoteBuffer** (inherited from **Client**) | Gets the buffer to store all the incoming data from the remote host.
--- | ---

**Protected Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

**Internal Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Handler</strong></td>
<td>Gets or sets the SOCKS handler to be used when communicating with the client.</td>
</tr>
</tbody>
</table>

**Private Instance Fields**

---
### Private Instance Properties

| **m_AuthList** | Holds the value of the AuthList property. |
| **m_Handler**  | Holds the value of the Handler property.  |
| **m_MustAuthenticate** | Holds the value of the MustAuthenticate property. |

### Private Instance Methods

| **OnEndSocksProtocol** | Called when the SOCKS protocol has ended. We can no start relaying data, if the SOCKS authentication was successful. |
| **OnStartSocksProtocol** | Called when we have received some data from the client. |

### See Also


---

Copyright © 2002, The KPD-Team
SocksClient Constructor

Initializes a new instance of the SocksClient class.

[Visual Basic]
Public Sub New( _
    ByVal ClientSocket As System.Net.Sockets.Socket,
    ByVal Destroyer As Org.Mentalis.Proxy.DestroyDelegate,
)

[C#]
public SocksClient(
    Socket ClientSocket,
    DestroyDelegate Destroyer,
    AuthenticationList AuthList
);

Parameters

ClientSocket  The Socket connection between this proxy server and the local client.

Destroyer    The method to be called when this SocksClient object disconnects from the local client and the remote server.

AuthList        The list with valid username/password combinations.

Remarks

If the AuthList is non-null, every client has to authenticate before he can use this proxy server to relay data. If it is null, the clients don't have to authenticate.

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace
The fields of the **SocksClient** class are listed below. For a complete list of **SocksClient** class members, see the **SocksClient Members** topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>m_AuthList</code></td>
<td>Holds the value of the AuthList property.</td>
</tr>
<tr>
<td><code>m_Handler</code></td>
<td>Holds the value of the Handler property.</td>
</tr>
<tr>
<td><code>m_MustAuthenticate</code></td>
<td>Holds the value of the MustAuthenticate property.</td>
</tr>
</tbody>
</table>

**See Also**

[SocksClient Class] | [Org.Mentalis.Proxy.Socks Namespace]

[Copyright © 2002, The KPD-Team]
SocksClient.m_AuthList Field

Holds the value of the AuthList property.


[C#] private AuthenticationList m_AuthList;

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksClient.m_Handler Field

Holds the value of the Handler property.


[C#] private SocksHandler m_Handler;

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksClient.m_MustAuthenticate Field

Holds the value of the MustAuthenticate property.

[Visual Basic] Private m_MustAuthenticate As !

[C#] private bool m_MustAuthenticate;

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksClient Properties

The properties of the SocksClient class are listed below. For a complete list of SocksClient class members, see the SocksClient Members topic.

Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MustAuthenticate</td>
<td>Gets or sets the SOCKS handler to be used when communicating with the client.</td>
</tr>
</tbody>
</table>

Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the local client.</td>
</tr>
<tr>
<td>RemoteBuffer (inherited from Client)</td>
<td>Gets the buffer to store all the incoming data from the remote host.</td>
</tr>
</tbody>
</table>

Internal Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handler</td>
<td>Gets or sets the SOCKS handler to be used when communicating with the client.</td>
</tr>
</tbody>
</table>

Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthList</td>
<td>Gets or sets the AuthenticationList to use when a computer tries to authenticate on the proxy server.</td>
</tr>
</tbody>
</table>

See Also

SocksClient Class | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksClient.AuthList Property**

Gets or sets the AuthenticationList to use when a computer tries to authenticate on the proxy server.

[Visual Basic]
```vbnet
```

[C#]
```csharp
private Authentication.AuthenticationList AuthList {get;set;}
```

**Property Value**

An instance of the AuthenticationList class that contains all the valid username/password combinations.

**See Also**

[SocksClient Class] | [SocksClient Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
SocksClient.Handler Property

Gets or sets the SOCKS handler to be used when communicating with the client.


[C#]
internal SocksHandler Handler {get; set;}

Property Value

The SocksHandler to be used when communicating with the client.

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
SocksClient.MustAuthenticate Property

gets or sets the SOCKS handler to be used when communicating with the client.

[Visual Basic] Public Property MustAuthenticate

[C#] public bool MustAuthenticate {get; set;}

Property Value

The SocksHandler to be used when communicating with the client.

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
# SocksClient Methods

The methods of the **SocksClient** class are listed below. For a complete list of **SocksClient** class members, see the [SocksClient Members](#) topic.

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Client</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Client.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartHandshake</strong></td>
<td>Starts communication with the client.</td>
</tr>
<tr>
<td><strong>StartRelay</strong> (inherited from <strong>Client</strong>)</td>
<td>Starts relaying data between the remote host and the local client.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns text information about this SocksClient object.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnClientReceive</strong> (inherited from <strong>Client</strong>)</td>
<td>Called when we have received data from the local client. Incoming data will immediately be forwarded to the remote host.</td>
</tr>
<tr>
<td><strong>OnClientSent</strong> (inherited from <strong>Object</strong>)</td>
<td>Called when we have sent data</td>
</tr>
<tr>
<td>Client)</td>
<td>to the local client. When all the data has been sent, we will start receiving again from the remote host.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>OnRemoteReceive</strong> (inherited from Client)</td>
<td>Called when we have received data from the remote host. Incoming data will immediately be forwarded to the local client.</td>
</tr>
<tr>
<td><strong>OnRemoteSent</strong> (inherited from Client)</td>
<td>Called when we have sent data to the remote host. When all the data has been sent, we will start receiving again from the local client.</td>
</tr>
</tbody>
</table>

### Private Instance Methods

| **OnEndSocksProtocol** | Called when the SOCKS protocol has ended. We can no start relaying data, if the SOCKS authentication was successful. |
| **OnStartSocksProtocol** | Called when we have received some data from the client. |

### See Also


*Copyright © 2002, The KPD-Team*
Mentalis.org Proxy Documentation
Called when the SOCKS protocol has ended. We can no start relaying data, if the SOCKS authentication was successful.

[C#]
private void OnEndSocksProtocol(
    bool Success,
    Socket Remote
);

Parameters

Success
   Specifies whether the SOCKS negotiation was successful or not.

Remote
   The connection with the remote server.

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Called when we have received some data from the client.

**Visual Basic**

```vbnet
Private Sub OnStartSocksProtocol(ByVal ar As System.IAsyncResult)
)
```

**C#**

```csharp
private void OnStartSocksProtocol(
    IAsyncResult ar
);
```

**Parameters**

- `ar`  
  The result of the asynchronous operation.

**See Also**


Copyright © 2002, The KPD-Team
SocksClient.StartHandshake Method

Starts communication with the client.

[Visual Basic] Overrides Public Sub StartHandshake()

[C#]
public override void StartHandshake();

See Also

SocksClient Class | SocksClient Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksClient.ToString Method**

Returns text information about this SocksClient object.

[Visual Basic] **Overrides Public Function ToString**

[C#]  
```csharp
public override string ToString();
```

**Return Value**

A string representing this SocksClient object.

**See Also**


*Copyright © 2002, The KPD-Team*
Mentalis.org Proxy Documentation
SocksHandler Class

Implements a specific version of the SOCKS protocol.
For a list of all members of this type, see SocksHandler Members.

System.Object SocksHandler

[Visual Basic]
MustInherit Class SocksHandler

[C#]
abstract class SocksHandler

Requirements

Assembly: Proxy.exe

See Also

SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
## SocksHandler Members

### Public Instance Constructors

<table>
<thead>
<tr>
<th>SocksHandler Constructor</th>
<th>Initializes a new instance of the SocksHandler class.</th>
</tr>
</thead>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>StartNegotiating</strong></td>
<td>Starts accepting bytes from the client.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AcceptSocket</strong></td>
<td>Gets or sets the socket that is used to accept incoming connections.</td>
</tr>
<tr>
<td><strong>Buffer</strong></td>
<td>Gets a buffer that can be used when receiving bytes from the client.</td>
</tr>
<tr>
<td><strong>Bytes</strong></td>
<td>Gets or sets a byte array that can be used to store received bytes from the client.</td>
</tr>
<tr>
<td><strong>Connection</strong></td>
<td>Gets or sets the connection with the client.</td>
</tr>
<tr>
<td><strong>RemoteBindIP</strong></td>
<td>Gets or sets the IP address of the requested remote server.</td>
</tr>
</tbody>
</table>
### RemoteConnection

Gets or sets the connection with the remote host.

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong></td>
<td>Adds some bytes to a byte array.</td>
</tr>
<tr>
<td><strong>Dispose</strong></td>
<td>Overloaded. Sends a reply to the client connection and disposes it afterwards.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>IsValidRequest</strong></td>
<td>Checks whether a specific request is a valid SOCKS request or not.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming connection in the AcceptSocket queue.</td>
</tr>
<tr>
<td><strong>OnDisposeBad</strong></td>
<td>Called when a negative reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnDisposeGood</strong></td>
<td>Called when an OK reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnReceiveBytes</strong></td>
<td>Called when we receive some bytes from the client.</td>
</tr>
<tr>
<td><strong>OnStartAccept</strong></td>
<td>Called when the AcceptSocket should start accepting incoming connections.</td>
</tr>
<tr>
<td><strong>ProcessRequest</strong></td>
<td>Processes a SOCKS request from a client.</td>
</tr>
</tbody>
</table>

### Internal Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Username</strong></td>
<td>Gets or sets the username of the SOCKS user.</td>
</tr>
</tbody>
</table>
Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AcceptSocket</td>
<td>Holds the value of the AcceptSocket property.</td>
</tr>
<tr>
<td>m_Buffer</td>
<td>Holds the value of the Buffer property.</td>
</tr>
<tr>
<td>m_Bytes</td>
<td>Holds the value of the Bytes property.</td>
</tr>
<tr>
<td>m_Connection</td>
<td>Holds the value of the Connection property.</td>
</tr>
<tr>
<td>m_RemoteBindIP</td>
<td>Holds the value of the RemoteBindIP property.</td>
</tr>
<tr>
<td>m_RemoteConnection</td>
<td>Holds the value of the RemoteConnection property.</td>
</tr>
<tr>
<td>m_Username</td>
<td>Holds the value of the Username property.</td>
</tr>
<tr>
<td>Signaler</td>
<td>Holds the address of the method to call when the SOCKS negotiation is complete.</td>
</tr>
</tbody>
</table>

See Also

[SocksHandler Class] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
SocksHandler Constructor

Initializes a new instance of the SocksHandler class.

[Visual Basic] Public Sub New( _
    ByVal ClientConnection As System.Net.Sockets.Socket,

[C#] public SocksHandler(
    Socket ClientConnection,
    NegotiationCompleteDelegate Callback);

Parameters

ClientConnection
  The connection with the client.

Callback
  The method to call when the SOCKS negotiation is complete.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Callback is null.</td>
</tr>
</tbody>
</table>

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler Fields

The fields of the SocksHandler class are listed below. For a complete list of SocksHandler class members, see the SocksHandler Members topic.

Private Instance Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AcceptSocket</td>
<td>Holds the value of the AcceptSocket property.</td>
</tr>
<tr>
<td>m_Buffer</td>
<td>Holds the value of the Buffer property.</td>
</tr>
<tr>
<td>m_Bytes</td>
<td>Holds the value of the Bytes property.</td>
</tr>
<tr>
<td>m_Connection</td>
<td>Holds the value of the Connection property.</td>
</tr>
<tr>
<td>m_RemoteBindIP</td>
<td>Holds the value of the RemoteBindIP property.</td>
</tr>
<tr>
<td>m_RemoteConnection</td>
<td>Holds the value of the RemoteConnection property.</td>
</tr>
<tr>
<td>m_Username</td>
<td>Holds the value of the Username property.</td>
</tr>
<tr>
<td>Signaler</td>
<td>Holds the address of the method to call when the SOCKS negotiation is complete.</td>
</tr>
</tbody>
</table>

See Also

SocksHandler Class | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
SocksHandler.m_AcceptSocket Field

Holds the value of the AcceptSocket property.

[Visual Basic] Private m_AcceptSocket As System.Net.Sockets.Socket

[C#] private Socket m_AcceptSocket;

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksHandler.m_Buffer Field**

Holds the value of the Buffer property.

- **Visual Basic**
  ```vbs
  Private m_Buffer As Byte()
  ```

- **C#**
  ```csharp
  private byte[] m_Buffer;
  ```

**See Also**

- [SocksHandler Class](#)
- [SocksHandler Members](#)

**Copyright © 2002, The KPD-Team**
SocksHandler.m_Bytes Field

Holds the value of the Bytes property.

**[Visual Basic]**
```
Private m_Bytes As Byte()
```

**[C#]**
```
private byte[] m_Bytes;
```

See Also

[SocksHandler Class]  |  [SocksHandler Members]  |  [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
SocksHandler.m_Connection Field

Holds the value of the Connection property.

[Visual Basic] Private m_Connection As System.Net.Sockets.Socket

[C#] private Socket m_Connection;

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.m_RemoteBindIP Field

Holds the value of the RemoteBindIP property.

[Visual Basic] Private m_RemoteBindIP As System.Net.IPAddress

[C#] private IPAddress m_RemoteBindIP;

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksHandler.m_RemoteConnection Field**

Holds the value of the RemoteConnection property.

- **[Visual Basic]**
  ```vbnet
  Private m_RemoteConnection As System.Net.Sockets.Socket
  ```

- **[C#]**
  ```csharp
  private Socket m_RemoteConnection;
  ```

**See Also**


*Copyright © 2002, The KPD-Team*
SocksHandler.m_Username Field

Holds the value of the Username property.

[Visual Basic] Private m_Username As String

[C#] private string m_Username;

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.Signaler Field

Holds the address of the method to call when the SOCKS negotiation is complete.

[Visual Basic]

[C#]
private NegotiationCompleteDelegate Signaler;

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler Properties

The properties of the SocksHandler class are listed below. For a complete list of SocksHandler class members, see the SocksHandler Members topic.

Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcceptSocket</td>
<td>Gets or sets the socket that is used to accept incoming connections.</td>
</tr>
<tr>
<td>Buffer</td>
<td>Gets a buffer that can be used when receiving bytes from the client.</td>
</tr>
<tr>
<td>Bytes</td>
<td>Gets or sets a byte array that can be used to store received bytes from the client.</td>
</tr>
<tr>
<td>Connection</td>
<td>Gets or sets the connection with the client.</td>
</tr>
<tr>
<td>RemoteBindIP</td>
<td>Gets or sets the IP address of the requested remote server.</td>
</tr>
<tr>
<td>RemoteConnection</td>
<td>Gets or sets the connection with the remote host.</td>
</tr>
</tbody>
</table>

Internal Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Gets or sets the username of the SOCKS user.</td>
</tr>
</tbody>
</table>

See Also

SocksHandler Class | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.AcceptSocket Property

Gets or sets the socket that is used to accept incoming connections.

[Visual Basic] Protected Property AcceptSocket

[C#] protected System.Net.Sockets.Socket AcceptSocket

Property Value

A Socket that is used to accept incoming connections.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.Buffer Property

Gets a buffer that can be used when receiving bytes from the client.

[Visual Basic] Protected ReadOnly Property Buffer

[C#]
protected byte[] Buffer {get;}

Property Value

A byte array that can be used when receiving bytes from the client.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.Bytes Property

Gets or sets a byte array that can be used to store received bytes from the client.

[Visual Basic] Protected Property Bytes As Byte()

[C#]
protected byte[] Bytes {get; set;}

Property Value
A byte array that can be used to store bytes from the client.

See Also
SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.Connection Property

Gets or sets the connection with the client.

[Visual Basic]
Protected Property Connection As

[C#]

Property Value

A Socket representing the connection between the proxy server and the SOCKS client.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.RemoteBindIP Property

Gets or sets the IP address of the requested remote server.

[Visual Basic] Protected Property RemoteBindIP

[C#] protected System.Net.IPAddress RemoteBindIP

Property Value

An IPAddress object specifying the address of the requested remote server.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.RemoteConnection Property

Gets or sets the connection with the remote host.

[Visual Basic] Protected Property RemoteConnection

[C#]
protected System.Net.Sockets.Socket RemoteConnection {get; set;}

Property Value
A Socket representing the connection between the proxy server and the remote host.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

See Also
- SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksHandler.Username Property**

Gets or sets the username of the SOCKS user.

**[Visual Basic]**

```
Friend Property Username As String
```

**[C#]**

```
internal string Username {get; set;}
```

**Property Value**

A String representing the username of the logged on user.

**Exceptions**

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>System.ArgumentNullException</code></td>
<td>The specified value is null.</td>
</tr>
</tbody>
</table>

**See Also**

[SocksHandler Class] | [SocksHandler Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
The methods of the **SocksHandler** class are listed below. For a complete list of **SocksHandler** class members, see the **SocksHandler Members** topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Documentation Link</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
<tr>
<td><strong>StartNegotiating</strong></td>
<td>Starts accepting bytes from the client.</td>
<td></td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Documentation Link</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong></td>
<td>Adds some bytes to a byte array.</td>
<td></td>
</tr>
<tr>
<td><strong>Dispose</strong></td>
<td>Overloaded. Sends a reply to the client connection and disposes it afterwards.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
<tr>
<td><strong>IsValidRequest</strong></td>
<td>Checks whether a specific request is a valid SOCKS request or not.</td>
<td></td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td><a href="https://docs.microsoft.com">Go to documentation</a></td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming connection in the AcceptSocket queue.</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>OnDisposeBad</td>
<td>Called when a negative reply has been sent to the client.</td>
<td></td>
</tr>
<tr>
<td>OnDisposeGood</td>
<td>Called when an OK reply has been sent to the client.</td>
<td></td>
</tr>
<tr>
<td>OnReceiveBytes</td>
<td>Called when we receive some bytes from the client.</td>
<td></td>
</tr>
<tr>
<td>OnStartAccept</td>
<td>Called when the AcceptSocket should start accepting incoming connections.</td>
<td></td>
</tr>
<tr>
<td>ProcessRequest</td>
<td>Processes a SOCKS request from a client.</td>
<td></td>
</tr>
</tbody>
</table>

See Also

SocksHandler Class | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.AddBytes Method

Adds some bytes to a byte array.

[Visual Basic]
Protected Sub AddBytes(
    ByVal NewBytes As Byte(), _
    ByVal Cnt As Integer
)

[C#]
protected void AddBytes(
    byte[] NewBytes,
    int Cnt
);

Parameters

NewBytes
The new bytes to add.

Cnt
The number of bytes to add.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
SocksHandler.Dispose

Closes the listening socket if present, and signals the parent object that SOCKS negotiation is complete.

Overload List

Sends a reply to the client connection and disposes it afterwards.

protected abstract void Dispose(byte);

Closes the listening socket if present, and signals the parent object that SOCKS negotiation is complete.

protected void Dispose(bool);

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Closes the listening socket if present, and signals the parent object that SOCKS negotiation is complete.

```visual-basic
Overloads Protected Sub Dispose(
    ByVal Success As Boolean _
)
```

```csharp
protected void Dispose(
    bool Success
);
```

**Parameters**

*Success*

Indicates whether the SOCKS negotiation was successful or not.

**See Also**

[SocksHandler Class] | [SocksHandler Members] | [Org.Mentalis.Proxy.Socks Namespace] | [SocksHandler.Dispose Overload List]

Copyright © 2002, The KPD-Team
Sends a reply to the client connection and disposes it afterwards.

**Parameters**

*Value*

A byte that contains the reply code to send to the client.

**See Also**


Copyright © 2002, The KPD-Team
**SocksHandler.IsValidRequest Method**

Checks whether a specific request is a valid SOCKS request or not.

**[Visual Basic]**

```vbnet
MustOverride Protected Function IsValidRequest(ByVal Request As Byte() _) As Boolean
```

**[C#]**

```csharp
protected abstract bool IsValidRequest(byte[] Request);
```

**Parameters**

*Request*

The request array to check.

**Return Value**

True is the specified request is valid, false otherwise

**See Also**

[SocksHandler Class] | [SocksHandler Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
| Mentalis.org Proxy Documentation |
SocksHandler.OnAccept Method

Called when there's an incoming connection in the AcceptSocket queue.

[Visual Basic] MustOverride Protected Sub OnAccept(ByVal ar As System.IAsyncResult)

[C#] protected abstract void OnAccept(IAsyncResult ar);

Parameters

ar
The result of the asynchronous operation.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksHandler.OnDisposeBad Method**

Called when a negative reply has been sent to the client.

```csharp
protected void OnDisposeBad(IAsyncResult ar);
```

**Parameters**

`ar`

The result of the asynchronous operation.

**See Also**

[SocksHandler Class] | [SocksHandler Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
SocksHandler.OnDisposeGood Method

Called when an OK reply has been sent to the client.

[Visual Basic]
Protected Sub OnDisposeGood(
    ByVal ar As System.IAsyncResult
)

[C#]
protected void OnDisposeGood(
    IAsyncResult ar
);

Parameters

ar
The result of the asynchronous operation.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.OnReceiveBytes Method

Called when we receive some bytes from the client.

[Visual Basic] Protected Sub OnReceiveBytes(_
    ByVal ar As System.IAsyncResult _)

[C#]
protected void OnReceiveBytes(_
    IAsyncResult ar
);

Parameters

ar
  The result of the asynchronous operation.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.OnStartAccept Method

Called when the AcceptSocket should start accepting incoming connections.

[Visual Basic] Protected Sub OnStartAccept( ByVal ar As System.IAsyncResult )

[C#] protected void OnStartAccept( IAsyncResult ar );

Parameters

ar The result of the asynchronous operation.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
Processes a SOCKS request from a client.

[Visual Basic]
MustOverride Protected Sub ProcessRequest(ByVal Request As Byte())
"

[C#]
protected abstract void ProcessRequest(byte[] Request);

Parameters

Request
The request to process.

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksHandler.StartNegotiating Method

Starts accepting bytes from the client.

[Visual Basic] Public Sub StartNegotiating()

[C#]
public void StartNegotiating();

See Also

SocksHandler Class | SocksHandler Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksListener Class

Listens on a specific port on the proxy server for incoming SOCKS4 and SOCKS5 requests.

For a list of all members of this type, see SocksListener Members.

System.Object  Listener  SocksListener

[Visual Basic]
NotInheritable Public Class SocksListener
  Inherits Listener
  Implements IDisposable

[C#]
public sealed class SocksListener : Listener

Remarks

This class also implements the SOCKS4a protocol.

Requirements

Assembly: Proxy.exe

See Also

SocksListener Members  Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
# SocksListener Members

## Public Instance Constructors

<table>
<thead>
<tr>
<th>SocksListener</th>
<th>Overloaded. Initialize a new instance of the SocksListener class.</th>
</tr>
</thead>
</table>

## Public Instance Properties

<table>
<thead>
<tr>
<th>ConstructString</th>
<th>Returns a string that holds all the construction information for this object.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsDisposed (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td>Listening (inherited from Listener)</td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Dispose (inherited from Listener)</th>
<th>Disposes of the resources (other than memory) used by the Listener.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetClientAt (inherited from Listener)</td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td>GetClientCount (inherited from Listener)</td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td>GetHashCode (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>OnAccept</td>
<td>Called when there's an incoming</td>
</tr>
<tr>
<td>client connection waiting to be accepted.</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td></td>
</tr>
<tr>
<td>Starts listening on the selected IP address and port.</td>
<td></td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td></td>
</tr>
<tr>
<td>Returns a string representation of this object.</td>
<td></td>
</tr>
</tbody>
</table>

### Protected Instance Properties

| **Address** (inherited from **Listener**) |
| Gets or sets the address on which to listen on. |
| **Clients** (inherited from **Listener**) |
| Gets the list of connected clients. |
| **ListenSocket** (inherited from **Listener**) |
| Gets or sets the listening Socket. |
| **Port** (inherited from **Listener**) |
| Gets or sets the port number on which to listen on. |

### Protected Instance Methods

| **AddClient** (inherited from **Listener**) |
| Adds the specified Client to the client list. |
| **Finalize** (inherited from **Listener**) |
| Finalizes the Listener. |
| **MemberwiseClone** (inherited from **Object**) |
| Select the method name to go to the Microsoft documentation. |
| **RemoveClient** (inherited from **Listener**) |
| Removes the specified Client from the client list. |
| **Restart** (inherited from **Listener**) |
| Restarts listening on the selected IP address and port. |

### Private Instance Fields

| **m_AuthList** |
| Holds the value of the AuthList property. |
### Private Instance Properties

| AuthList         | Gets or sets the AuthenticationList to be used when a SOCKS5 client connects. |

### See Also

[SocksListener Class] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
SocksListener

Initializes a new instance of the SocksListener class.

Overload List

Initializes a new instance of the SocksListener class.

`public SocksListener(int);`

Initializes a new instance of the SocksListener class.

`public SocksListener(IPAddress,int);`

Initializes a new instance of the SocksListener class.

`public SocksListener(int,AuthenticationList);`

Initializes a new instance of the SocksListener class.

`public SocksListener(IPAddress,int,AuthenticationList);`

See Also

[SocksListener Class] | [SocksListener Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
SocksListener Constructor (Int32)

Initializes a new instance of the SocksListener class.

[Visual Basic]
Overloads Public Sub New( _
    ByVal Port As Integer _
)

[C#]
public SocksListener(
    int Port
);

Parameters

Port
The port to listen on.

Remarks
The SocksListener will listen on all available network cards and it will not use an AuthenticationList.

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
SocksListener Constructor (IPAddress, Int32)

Initializes a new instance of the SocksListener class.

[Visual Basic] Overloads Public Sub New( _
  ByVal Address As System.Net.IPAddress, _
  ByVal Port As Integer _
)

[C#]
public SocksListener(
  IPAddress Address,
  int Port
);

Parameters

Address
  The address to listen on. You can specify IPAddress.Any to listen on all installed network cards.

Port
  The port to listen on.

Remarks

For the security of your server, try to avoid to listen on every network card (IPAddress.Any). Listening on a local IP address is usually sufficient and much more secure. The SocksListener object will not use an AuthenticationList.

See Also


Copyright © 2002, The KPD-Team
SocksListener Constructor (Int32, AuthenticationList)

Initializes a new instance of the SocksListener class.

```[Visual Basic]Overloads Public Sub New( _
    ByVal Port As Integer, _
```

```[C#]
public SocksListener(
    int Port,
    AuthenticationList AuthList
);
```

Parameters

**Port**
- The port to listen on.

**AuthList**
- The list of valid login/password combinations. If you do not need password authentication, set this parameter to null.

Remarks

The SocksListener will listen on all available network cards.

See Also


Copyright © 2002, The KPD-Team
SocksListener Constructor (IPAddress, Int32, AuthenticationList)

Initializes a new instance of the SocksListener class.

[Visual Basic] Overloads Public Sub New( _
  ByVal Address As System.Net.IPAddress, _
  ByVal Port As Integer, _
  ByVal AuthList As Org.Mentalis.Proxy.Socks)

[C#]
public SocksListener(
  IPAddress Address,
  int Port,
  AuthenticationList AuthList
);

Parameters

Address
  The address to listen on. You can specify IPAddress.Any to listen on all installed network cards.

Port
  The port to listen on.

AuthList
  The list of valid login/password combinations. If you do not need password authentication, set this parameter to null.

Remarks

For the security of your server, try to avoid to listen on every network card (IPAddress.Any). Listening on a local IP address is usually sufficient and much more secure.

See Also

SocksListener Class | SocksListener Members | Org.Mentalis.Proxy.Socks Namespace | SocksListener Constructor
The fields of the SocksListener class are listed below. For a complete list of SocksListener class members, see the SocksListener Members topic.

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AuthList</td>
<td>Holds the value of the AuthList property.</td>
</tr>
</tbody>
</table>

See Also

SocksListener Class | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
SocksListener.m_AuthList Field

Holds the value of the AuthList property.


[C#] private AuthenticationList m_AuthList;

See Also

SocksListener Class | SocksListener Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksListener Properties**

The properties of the **SocksListener** class are listed below. For a complete list of **SocksListener** class members, see the **SocksListener Members** topic.

**Public Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ConstructString</strong></td>
<td>Returns a string that holds all the construction information for this object.</td>
</tr>
<tr>
<td><strong>IsDisposed</strong> <em>(inherited from Listener)</em></td>
<td>Gets a value indicating whether the Listener has been disposed or not.</td>
</tr>
<tr>
<td><strong>Listening</strong> <em>(inherited from Listener)</em></td>
<td>Gets a value indicating whether the Listener is currently listening or not.</td>
</tr>
</tbody>
</table>

**Protected Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong> <em>(inherited from Listener)</em></td>
<td>Gets or sets the address on which to listen on.</td>
</tr>
<tr>
<td><strong>Clients</strong> <em>(inherited from Listener)</em></td>
<td>Gets the list of connected clients.</td>
</tr>
<tr>
<td><strong>ListenSocket</strong> <em>(inherited from Listener)</em></td>
<td>Gets or sets the listening Socket.</td>
</tr>
<tr>
<td><strong>Port</strong> <em>(inherited from Listener)</em></td>
<td>Gets or sets the port number on which to listen on.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AuthList</strong></td>
<td>Gets or sets the AuthenticationList to be used when a SOCKS5 client connects.</td>
</tr>
</tbody>
</table>

**See Also**
SocksListener.AuthList Property

Gets or sets the AuthenticationList to be used when a SOCKS5 client connects.


[C#] private Authentication.AuthenticationList AuthList {get; set;}

Property Value

An AuthenticationList that is to be used when a SOCKS5 client connects.

Remarks

This value can be null.

See Also

SocksListener Class | SocksListener Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team
**SocksListener.ConstructString Property**

Returns a string that holds all the construction information for this object.

[Visual Basic] **Overrides Public ReadOnly Property**

[C#]

```csharp
public string ConstructString {override get;}
```

**Property Value**

A string that holds all the construction information for this object.

**See Also**

[SocksListener Class] | [SocksListener Members] | [Org.Mentalis.Proxy.Socks Namespace]

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
The methods of the **SocksListener** class are listed below. For a complete list of **SocksListener** class members, see the **SocksListener Members** topic.

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dispose</strong> (inherited from <strong>Listener</strong>)</td>
<td>Disposes of the resources (other than memory) used by the Listener.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetClientAt</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the requested client from the client list.</td>
</tr>
<tr>
<td><strong>GetClientCount</strong> (inherited from <strong>Listener</strong>)</td>
<td>Returns the number of clients in the client list.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>OnAccept</strong></td>
<td>Called when there's an incoming client connection waiting to be accepted.</td>
</tr>
<tr>
<td><strong>Start</strong> (inherited from <strong>Listener</strong>)</td>
<td>Starts listening on the selected IP address and port.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>Returns a string representation of this object.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddClient</strong> (inherited from <strong>Listener</strong>)</td>
<td>Adds the specified Client to the client list.</td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Listener</strong>)</td>
<td>Finalizes the Listener.</td>
</tr>
</tbody>
</table>
### MemberwiseClone (inherited from **Object**)
Select the method name to go to the Microsoft documentation.

### RemoveClient (inherited from **Listener**)
Removes the specified Client from the client list.

### Restart (inherited from **Listener**)
Restarts listening on the selected IP address and port.

### See Also
- [SocksListener Class](#)

*Copyright © 2002, The KPD-Team*
Called when there's an incoming client connection waiting to be accepted.

**[Visual Basic]**

```vbnet
Overrides Public Sub OnAccept(ByVal ar As System.IAsyncResult)
```

**[C#]**

```csharp
public override void OnAccept(IAsyncResult ar);
```

**Parameters**

*ar*

The result of the asynchronous operation.

**See Also**

[SocksListener Class] | [SocksListener Members] | [Org.Mentalis.Proxy.Socks Namespace]
SocksListener.ToString Method

Returns a string representation of this object.

[Visual Basic]
Overrides Public Function ToString()

[C#]
public override string ToString();

Return Value

A string with information about this object.

See Also

SocksListener Class | SocksListener Members | Org.Mentalis.Proxy.Socks Namespace

Copyright © 2002, The KPD-Team

Namespace hierarchy

Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthBase</td>
<td>Authenticates a user on a SOCKS5 server according to the implemented subprotocol.</td>
</tr>
<tr>
<td>AuthenticationList</td>
<td>Stores a dictionary with username/password combinations.</td>
</tr>
<tr>
<td>AuthNone</td>
<td>Authenticates a user on a SOCKS5 server according to the 'No Authentication' subprotocol.</td>
</tr>
<tr>
<td>AuthUserPass</td>
<td>Authenticates a user on a SOCKS5 server according to the username/password authentication subprotocol.</td>
</tr>
</tbody>
</table>

Delegates

<table>
<thead>
<tr>
<th>Delegate</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthenticationCompleteDelegate</td>
<td>Defines the signature of the method to be called when the authentication is complete.</td>
</tr>
</tbody>
</table>

Copyright © 2002, The KPD-Team
AuthBase Class

Authenticates a user on a SOCKS5 server according to the implemented subprotocol.

For a list of all members of this type, see AuthBase Members.

System.Object AuthBase

[Visual Basic]
MustInherit Class AuthBase

[C#]
abstract class AuthBase

Remarks

This is an abstract class. The subprotocol that’s used to authenticate a user is specified in the subclasses of this base class.

Requirements


Assembly: Proxy.exe

See Also


Copyright © 2002, The KPD-Team
## AuthBase Members

### Public Instance Constructors

<table>
<thead>
<tr>
<th>AuthBase Constructor</th>
<th>Initializes a new instance of the AuthBase class.</th>
</tr>
</thead>
</table>

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Callback</td>
<td>The method to call when the authentication is complete.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer</td>
<td>Gets a buffer that can be used to receive data from the client connection.</td>
</tr>
<tr>
<td>Bytes</td>
<td>Gets or sets an array of bytes that can be used to store all received data.</td>
</tr>
<tr>
<td>Connection</td>
<td>Gets or sets the Socket connection between the proxy server and the SOCKS client.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddBytes</td>
<td>Adds bytes to the array returned</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>Finalize (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>MemberwiseClone (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

**Internal Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartAuthentication</td>
<td>Starts the authentication process.</td>
</tr>
</tbody>
</table>

**Private Instance Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Buffer</td>
<td>Holds the value of the Buffer property.</td>
</tr>
<tr>
<td>m_Bytes</td>
<td>Holds the value of the Bytes property.</td>
</tr>
<tr>
<td>m_Connection</td>
<td>Holds the value of the Connection property.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
AuthBase Constructor

Initializes a new instance of the AuthBase class.

[Visual Basic] Public Sub New()

[C#]
public AuthBase();

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
AuthBase Fields

The fields of the **AuthBase** class are listed below. For a complete list of **AuthBase** class members, see the **AuthBase Members** topic.

### Protected Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Callback</strong></td>
<td>The method to call when the authentication is complete.</td>
</tr>
</tbody>
</table>

### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_Buffer</strong></td>
<td>Holds the value of the Buffer property.</td>
</tr>
<tr>
<td><strong>m_Bytes</strong></td>
<td>Holds the value of the Bytes property.</td>
</tr>
<tr>
<td><strong>m_Connection</strong></td>
<td>Holds the value of the Connection property.</td>
</tr>
</tbody>
</table>

See Also


Copyright © 2002, The KPD-Team
The method to call when the authentication is complete.


[C#] **protected** `AuthenticationCompleteDelegate` Cal]

See Also

[AuthBase Class] [AuthBase Members] [Org.Mentalis.Proxy.Socks.Authentication Namespace]

Copyright © 2002, The KPD-Team
AuthBase.m_Buffer Field

Holds the value of the Buffer property.

[Visual Basic]
Private m_Buffer As Byte()

[C#]
private byte[] m_Buffer;

See Also


Copyright © 2002, The KPD-Team
### AuthBase.m_Bytes Field

Holds the value of the Bytes property.

[Visual Basic] `Private m_Bytes As Byte()`

[C#] `private byte[] m_Bytes;`

**See Also**

[AuthBase Class] | [AuthBase Members] | [Org.Mentalis.Proxy.Socks.Authentication Namespace]

Copyright © 2002, The KPD-Team
AuthBase.m_Connection Field

Holds the value of the Connection property.

[Visual Basic] Private m_Connection As System.Net.Sockets.Socket

[C#] private Socket m_Connection;

See Also


Copyright © 2002, The KPD-Team
### AuthBase Properties

The properties of the AuthBase class are listed below. For a complete list of AuthBase class members, see the AuthBase Members topic.

#### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong></td>
<td>Gets a buffer that can be used to receive data from the client connection.</td>
</tr>
<tr>
<td><strong>Bytes</strong></td>
<td>Gets or sets an array of bytes that can be used to store all received data.</td>
</tr>
<tr>
<td><strong>Connection</strong></td>
<td>Gets or sets the Socket connection between the proxy server and the SOCKS client.</td>
</tr>
</tbody>
</table>

#### See Also


*Copyright © 2002, The KPD-Team*
AuthBase.Buffer Property

Gets a buffer that can be used to receive data from the client connection.

[Visual Basic]
Protected Readonly Property Buffer As Byte() As Byte[]

[C#]
protected byte[] Buffer {get;}

Property Value

An array of bytes that can be used to receive data from the client connection.

See Also


Copyright © 2002, The KPD-Team
AuthBase.Bytes Property

Gets or sets an array of bytes that can be used to store all received data.

[Visual Basic] Protected Property Bytes As Byte()

[C#]
protected byte[] Bytes {get; set;}

Property Value

An array of bytes that can be used to store all received data.

See Also


Copyright © 2002, The KPD-Team
AuthBase.Connection Property

Gets or sets the Socket connection between the proxy server and the SOCKS client.

[Visual Basic] Protected Property Connection

[C#]

Property Value
A Socket instance defining the connection between the proxy server and the local client.

See Also


Copyright © 2002, The KPD-Team
## AuthBase Methods

The methods of the **AuthBase** class are listed below. For a complete list of **AuthBase** class members, see the [AuthBase Members](#) topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong></td>
<td>Adds bytes to the array returned by the Bytes property.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Internal Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>StartAuthentication</strong></td>
<td>Starts the authentication process.</td>
</tr>
</tbody>
</table>

### See Also

Mentalis.org Proxy Documentation
AuthBase.AddBytes Method

Adds bytes to the array returned by the Bytes property.

[Visual Basic]
Protected Sub AddBytes( _
    ByVal NewBytes As Byte(), _
    ByVal Cnt As Integer _)
)

[C#]
protected void AddBytes(
    byte[] NewBytes,
    int Cnt
);

Parameters

NewBytes
  The bytes to add.

Cnt
  The number of bytes to add.

See Also


Copyright © 2002, The KPD-Team
AuthBase.StartAuthentication Method

Starts the authentication process.


[C#] internal abstract void StartAuthentication(Socket Connection, AuthenticationCompleteDelegate Callback);

Parameters

Connection
  The connection with the SOCKS client.

Callback
  The method to call when the authentication is complete.

Remarks

This abstract method must be implemented in the subclasses, according to the selected subprotocol.

See Also


Copyright © 2002, The KPD-Team
AuthenticationCompleteDelegate Delegate

Defines the signature of the method to be called when the authentication is complete.

[Visual Basic] Delegate Sub AuthenticationCompleteDelegate(ByVal Success As Boolean )

[C#]
delegate void AuthenticationCompleteDelegate(bool Success );

Parameters

Success
Specifies whether the authentication was successful or not.

Requirements

Assembly: Proxy.exe

See Also


Copyright © 2002, The KPD-Team
Stores a dictionary with username/password combinations.
For a list of all members of this type, see AuthenticationList Members.

System.Object AuthenticationList

Remarks
This class can be used by a SOCKS5 listener. This class uses an MD5 has to store the passwords in a secure manner. The username is treated in a case-insensitive manner, the password is treated case-sensitive.

Requirements
Assembly: Proxy.exe

See Also

Copyright © 2002, The KPD-Team
## AuthenticationList Members

### Public Instance Constructors

<table>
<thead>
<tr>
<th>AuthenticationList Constructor</th>
<th>Initializes a new instance of the AuthenticationList class.</th>
</tr>
</thead>
</table>

### Public Instance Properties

| Hashes | Gets an array with all the hashes in the authentication list. |
| Keys   | Gets an array with all the keys in the authentication list. |

### Public Instance Methods

| AddHash | Adds an item to the list. |
| AddItem | Adds an item to the list. |
| Clear   | Clears the authentication list. |
| Equals  | Select the method name to go to the Microsoft documentation. (inherited from Object) |
| GetHashCode  | Select the method name to go to the Microsoft documentation. (inherited from Object) |
| GetType  | Select the method name to go to the Microsoft documentation. (inherited from Object) |
| IsHashPresent  | Checks whether a user/passhash combination is present in the collection or not. |
| IsItemPresent  | Checks whether a user/pass combination is present in the collection or not. |
| IsUserPresent  | Checks whether a username is present in the collection or not. |
| RemoveItem  | Removes an item from the list. |
**ToString** (inherited from **Object**) | Select the method name to go to the Microsoft documentation.

**Protected Instance Properties**

**Listing**

Gets the StringDictionary that's used to store the user/pass combinations.

**Protected Instance Methods**

**Finalize** (inherited from **Object**) | Select the method name to go to the Microsoft documentation.

**MemberwiseClone** (inherited from **Object**) | Select the method name to go to the Microsoft documentation.

**Private Instance Fields**

**m_Listing**

Holds the value of the Listing property.

**See Also**


*Copyright © 2002, The KPD-Team*
AuthenticationList Constructor

Initializes a new instance of theAuthenticationList class.

[Visual Basic] Public Sub New()

[C#]
public AuthenticationList();

See Also


Copyright © 2002, The KPD-Team
### AuthenticationList Fields

The fields of the **AuthenticationList** class are listed below. For a complete list of **AuthenticationList** class members, see the **AuthenticationList Members** topic.

#### Private Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Listing</td>
<td>Holds the value of the Listing property.</td>
</tr>
</tbody>
</table>

#### See Also


*Copyright © 2002, The KPD-Team*
Mentalis.org Proxy Documentation
**AuthenticationList.m_Listing Field**

Holds the value of the Listing property.

**[Visual Basic]**
Private m_Listing As System.Collections.Specialized.StringDictionary

**[C#]**
private StringDictionary m_Listing;

See Also

## AuthenticationList Properties

The properties of the AuthenticationList class are listed below. For a complete list of AuthenticationList class members, see the [AuthenticationList Members](#) topic.

### Public Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hashes</strong></td>
<td>Gets an array with all the hashes in the authentication list.</td>
</tr>
<tr>
<td><strong>Keys</strong></td>
<td>Gets an array with all the keys in the authentication list.</td>
</tr>
</tbody>
</table>

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Listing</strong></td>
<td>Gets the StringDictionary that's used to store the user/pass combinations.</td>
</tr>
</tbody>
</table>

### See Also


[Copyright © 2002, The KPD-Team](#)
**AuthenticationList.Hashes Property**

Gets an array with all the hashes in the authentication list.

```vbnet
Public Readonly Property Hashes
```

```csharp
public string[] Hashes {get;}
```

**Property Value**

An array of strings containing all the hashes in the authentication list.

**See Also**

- [AuthenticationList Class](#)
- [AuthenticationList Members](#)

Copyright © 2002, The KPD-Team
**AuthenticationList.Keys Property**

Gets an array with all the keys in the authentication list.

[Visual Basic] `Public ReadOnly Property Keys As String()`

[C#] `public string[] Keys {get;}`

**Property Value**

An array of strings containing all the keys in the authentication list.

**See Also**


Copyright © 2002, The KPD-Team
**AuthenticationList.Listing Property**

Gets the StringDictionary that's used to store the user/pass combinations.

**[Visual Basic]**

```vbnet
Protected ReadOnly Property Listing As StringDictionary
```

**[C#]**

```csharp
protected System.Collections.Specialized.StringDictionary Listing
```

**Property Value**

A StringDictionary object that's used to store the user/pass combinations.

**See Also**

- [AuthenticationList Class](#)
- [AuthenticationList Members](#)

*Copyright © 2002, The KPD-Team*
# AuthenticationList Methods

The methods of the `AuthenticationList` class are listed below. For a complete list of `AuthenticationList` class members, see the [AuthenticationList Members](#) topic.

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddHash</strong></td>
<td>Adds an item to the list.</td>
</tr>
<tr>
<td><strong>AddItem</strong></td>
<td>Adds an item to the list.</td>
</tr>
<tr>
<td><strong>Clear</strong></td>
<td>Clears the authentication list.</td>
</tr>
<tr>
<td><strong>Equals</strong> (inherited from <code>Object</code>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from <code>Object</code>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from <code>Object</code>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>IsHashPresent</strong></td>
<td>Checks whether a user/passhash combination is present in the collection or not.</td>
</tr>
<tr>
<td><strong>IsItemPresent</strong></td>
<td>Checks whether a user/pass combination is present in the collection or not.</td>
</tr>
<tr>
<td><strong>IsUserPresent</strong></td>
<td>Checks whether a username is present in the collection or not.</td>
</tr>
<tr>
<td><strong>RemoveItem</strong></td>
<td>Removes an item from the list.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from <code>Object</code>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <code>Object</code>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited)</td>
<td>Select the method name to go to</td>
</tr>
</tbody>
</table>
See Also


Copyright © 2002, The KPD-Team
AuthenticationList.AddHash Method

Adds an item to the list.

[Visual Basic]Public Sub AddHash( _
    ByVal Username As String, _
    ByVal PassHash As String _
)

[C#]
public void AddHash(
    string Username,
    string PassHash
);

Parameters

Username
The username to add.

PassHash
The hashed password to add.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SystemArgumentNullException</td>
<td>Either Username or Password is null.</td>
</tr>
</tbody>
</table>

See Also


Copyright © 2002, The KPD-Team
AuthenticationList.AddItem Method

Adds an item to the list.

[Visual Basic]
Public Sub AddItem( _
    ByVal Username As String, _
    ByVal Password As String _
)

[C#]
public void AddItem(
    string Username,
    string Password
);

Parameters

Username
The username to add.

Password
The corresponding password to add.

Exceptions

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Either Username or Password is null.</td>
</tr>
</tbody>
</table>

See Also


Copyright © 2002, The KPD-Team
AuthenticationList.Clear Method

Clears the authentication list.

[Visual Basic] Public Sub Clear()

[C#]
public void Clear();

See Also


Copyright © 2002, The KPD-Team
Checks whether a user/passhash combination is present in the collection or not.

**Parameters**

*Username*

The username to search for.

*PassHash*

The corresponding password hash to search for.

**Return Value**

True when the user/passhash combination is present in the collection, false otherwise.

**See Also**


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
Checks whether a user/pass combination is present in the collection or not.

**[Visual Basic]**
```
Public Function IsItemPresent(ByVal Username As String, ByVal Password As String) As Boolean
```

**[C#]**
```
public bool IsItemPresent(string Username, string Password);
```

**Parameters**

*Username*
- The username to search for.

*Password*
- The corresponding password to search for.

**Return Value**
- True when the user/pass combination is present in the collection, false otherwise.

**See Also**

Copyright © 2002, The KPD-Team
Checks whether a username is present in the collection or not.

[Visual Basic] Public Function IsUserPresent( ByVal Username As String ) As Boolean

[C#]
public bool IsUserPresent(
    string Username
);

Parameters

**Username**

The username to search for.

Return Value

True when the username is present in the collection, false otherwise.

See Also


Copyright © 2002, The KPD-Team
AuthenticationList.RemoveItem Method

Removes an item from the list.

**Parameters**

*Username*

The username to remove.

**Exceptions**

<table>
<thead>
<tr>
<th>Exception Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>System.ArgumentNullException</td>
<td>Username is null.</td>
</tr>
</tbody>
</table>

**See Also**


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
AuthNone Class

Authenticates a user on a SOCKS5 server according to the 'No Authentication' subprotocol.

For a list of all members of this type, see AuthNone Members.

System.Object AuthBase
AuthNone

[Visual Basic]
NotInheritable Class AuthNone
Inherits AuthBase

[C#]
sealed class AuthNone : AuthBase

Requirements

Assembly: Proxy.exe

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
# AuthNone Members

## Public Instance Constructors

| AuthNone Constructor | Initializes a new instance of the AuthNone class. |

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>ToString</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Protected Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Callback</td>
<td></td>
</tr>
</tbody>
</table>

## Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer</td>
<td>Gets a buffer that can be used to receive data from the client connection.</td>
</tr>
<tr>
<td>Bytes</td>
<td>Gets or sets an array of bytes that can be used to store all received data.</td>
</tr>
<tr>
<td>Connection</td>
<td>Gets or sets the Socket connection between the proxy server and the SOCKS client.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddBytes</td>
<td>Adds bytes to the array returned by the Bytes property.</td>
</tr>
</tbody>
</table>
### Internal Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>StartAuthentication</code></td>
<td>Calls the parent class to inform it authentication is complete.</td>
</tr>
</tbody>
</table>

### See Also


Copyright © 2002, The KPD-Team
**AuthNone Constructor**

Initializes a new instance of the AuthNone class.

[Visual Basic] `Public Sub New()`

[C#] `public AuthNone();`

See Also

- [AuthNone Class](#)
- [AuthNone Members](#)

Copyright © 2002, The KPD-Team
AuthNone Fields

The fields of the AuthNone class are listed below. For a complete list of AuthNone class members, see the AuthNone Members topic.

Protected Instance Fields

<table>
<thead>
<tr>
<th>Callback</th>
</tr>
</thead>
</table>

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
AuthNone.Callback Field


[C#] protected AuthenticationCompleteDelegate Callback;

See Also

AuthNone Class | AuthNone Members | Org.Mentalis.Proxy.Socks.Authentication Namespace

Copyright © 2002, The KPD-Team
AuthNone Methods

The methods of the **AuthNone** class are listed below. For a complete list of **AuthNone** class members, see the **AuthNone Members** topic.

### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong></td>
<td>(inherited from <strong>AuthBase</strong>) Adds bytes to the array returned by the Bytes property.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### Internal Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>StartAuthentication</strong></td>
<td>Calls the parent class to inform it authentication is complete.</td>
</tr>
</tbody>
</table>

### See Also


---

**Copyright © 2002, The KPD-Team**
AuthNone.StartAuthentication Method

Calls the parent class to inform it authentication is complete.

[Visual Basic] Overrides Friend Sub StartAuthentication(
    ByVal Connection As System.Net.Sockets.Socket,

[C#] internal override void StartAuthentication(
    Socket Connection,
    AuthenticationCompleteDelegate Callback);

Parameters

Connection
The connection with the SOCKS client.

Callback
The method to call when the authentication is complete.

See Also

AuthNone Class | AuthNone Members | Org.Mentalis.Proxy.Socks.Authentication Namespace

Copyright © 2002, The KPD-Team
AuthUserPass Class

Authenticates a user on a SOCKS5 server according to the username/password authentication subprotocol.

For a list of all members of this type, see AuthUserPass Members.

System.Object  AuthBase  AuthUserPass

[Visual Basic]
NotInheritable  Class AuthUserPass
Inherits AuthBase

[C#]
sealed class AuthUserPass : AuthBase

Requirements

Assembly: Proxy.exe

See Also


Copyright © 2002, The KPD-Team
### AuthUserPass Members

#### Public Instance Constructors

| AuthUserPass Constructor | Initializes a new instance of the AuthUserPass class. |

#### Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Inheritance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals</td>
<td>(inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>(inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType</td>
<td>(inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>ToString</td>
<td>(inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

#### Protected Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Callback</td>
<td></td>
</tr>
</tbody>
</table>

#### Protected Instance Properties

<table>
<thead>
<tr>
<th>Field</th>
<th>Inheritance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer</td>
<td>(inherited from AuthBase)</td>
<td>Gets a buffer that can be used to receive data from the client connection.</td>
</tr>
<tr>
<td>Bytes</td>
<td>(inherited from AuthBase)</td>
<td>Gets or sets an array of bytes that can be used to store all received data.</td>
</tr>
<tr>
<td>Connection</td>
<td>(inherited from AuthBase)</td>
<td>Gets or sets the Socket connection between the proxy server and the SOCKS client.</td>
</tr>
</tbody>
</table>

#### Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Inheritance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddBytes</td>
<td>(inherited from AuthBase)</td>
<td>Adds bytes to the array returned by the Bytes property.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Finalize</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td></td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
<td></td>
</tr>
</tbody>
</table>

**Internal Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>StartAuthentication</strong></td>
<td>Starts the authentication process.</td>
</tr>
</tbody>
</table>

**Private Instance Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_AuthList</strong></td>
<td>Holds the value of the AuthList property.</td>
</tr>
</tbody>
</table>

**Private Instance Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AuthList</strong></td>
<td>Gets or sets the AuthenticationList to use when a computer tries to authenticate on the proxy server.</td>
</tr>
</tbody>
</table>

**Private Instance Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IsValidQuery</strong></td>
<td>Checks whether the specified authentication query is a valid one.</td>
</tr>
<tr>
<td><strong>OnOkSent</strong></td>
<td>Called when an OK reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>OnRecvRequest</strong></td>
<td>Called when we have received the initial authentication data from the SOCKS client.</td>
</tr>
<tr>
<td><strong>OnUhohSent</strong></td>
<td>Called when a negatiev reply has been sent to the client.</td>
</tr>
<tr>
<td><strong>ProcessQuery</strong></td>
<td>Processes an authentication query.</td>
</tr>
</tbody>
</table>
See Also


Copyright © 2002, The KPD-Team
AuthUserPass Constructor

Initializes a new instance of the AuthUserPass class.


[C#]
public AuthUserPass( AuthenticationList AuthList )

Parameters

AuthList
An AuthenticationList object that contains the list of all valid username/password combinations.

Remarks

If the AuthList parameter is null, any username/password combination will be accepted.

See Also


Copyright © 2002, The KPD-Team
The fields of the AuthUserPass class are listed below. For a complete list of AuthUserPass class members, see the AuthUserPass Members topic.

Protected Instance Fields

| Callback |

Private Instance Fields

| m_AuthList | Holds the value of the AuthList property. |

See Also


Copyright © 2002, The KPD-Team

[C#] protected AuthenticationCompleteDelegate Callback;

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
AuthUserPass.m_AuthList Field

Holds the value of the AuthList property.


[C#]
private AuthenticationList m_AuthList;

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
## AuthUserPass Properties

The properties of the AuthUserPass class are listed below. For a complete list of AuthUserPass class members, see the AuthUserPass Members topic.

### Protected Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Buffer</strong></td>
<td>Gets a buffer that can be used to receive data from the client connection.</td>
</tr>
<tr>
<td><strong>Bytes</strong></td>
<td>Gets or sets an array of bytes that can be used to store all received data.</td>
</tr>
<tr>
<td><strong>Connection</strong></td>
<td>Gets or sets the Socket connection between the proxy server and the SOCKS client.</td>
</tr>
</tbody>
</table>

### Private Instance Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AuthList</strong></td>
<td>Gets or sets the AuthenticationList to use when a computer tries to authenticate on the proxy server.</td>
</tr>
</tbody>
</table>

### See Also


**Copyright © 2002, The KPD-Team**
AuthUserPass.AuthList Property

Gets or sets the AuthenticationList to use when a computer tries to authenticate on the proxy server.


[C#] private AuthenticationList AuthList {get; set;}

Property Value

An instance of the AuthenticationList class that contains all the valid username/password combinations.

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
# AuthUserPass Methods

The methods of the **AuthUserPass** class are listed below. For a complete list of **AuthUserPass** class members, see the **AuthUserPass Members** topic.

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ToString</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddBytes</strong></td>
<td>(inherited from <strong>AuthBase</strong>) Adds bytes to the array returned by the Bytes property.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong></td>
<td>(inherited from <strong>Object</strong>) Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Internal Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>StartAuthentication</strong></td>
<td>Starts the authentication process.</td>
</tr>
</tbody>
</table>

## Private Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IsValidQuery</strong></td>
<td>Checks whether the specified authentication query is a valid one.</td>
</tr>
<tr>
<td><strong>OnOkSent</strong></td>
<td>Called when an OK reply has</td>
</tr>
</tbody>
</table>
been sent to the client.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnRecvRequest</td>
<td>Called when we have received the initial authentication data from the SOCKS client.</td>
</tr>
<tr>
<td>OnUhohSent</td>
<td>Called when a negative reply has been sent to the client.</td>
</tr>
<tr>
<td>ProcessQuery</td>
<td>Processes an authentication query.</td>
</tr>
</tbody>
</table>

See Also


Copyright © 2002, The KPD-Team
AuthUserPass.IsValidQuery Method

Checks whether the specified authentication query is a valid one.

[Visual Basic] Private Function IsValidQuery(ByVal Query As Byte()) As Boolean

[C#] private bool IsValidQuery(byte[] Query);

Parameters

Query
The query to check.

Return Value
True if the query is a valid authentication query, false otherwise.

See Also

Copyright © 2002, The KPD-Team
AuthUserPass.OnOkSent Method

Called when an OK reply has been sent to the client.

[Visual Basic] Private Sub OnOkSent( ByVal ar As System.IAsyncResult _ )

[C#] private void OnOkSent( IAsyncResult ar )

Parameters

ar

The result of the asynchronous operation.

See Also


Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
AuthUserPass.OnRecvRequest Method

Called when we have received the initial authentication data from the SOCKS client.

[Visual Basic] Private Sub OnRecvRequest( ByVal ar As System.IAsyncResult )

[C#]
private void OnRecvRequest( IAsyncResult ar );

Parameters

ar
The result of the asynchronous operation.

See Also


Copyright © 2002, The KPD-Team
AuthUserPass.OnUhohSent Method

Called when a negative reply has been sent to the client.

[Visual Basic] Private Sub OnUhohSent( _
    ByVal ar As System.IAsyncResult _
) _

[C#] private void OnUhohSent(
    IAsyncResult ar
); _

Parameters

ar
The result of the asynchronous operation.

See Also


Copyright © 2002, The KPD-Team
AuthUserPass.ProcessQuery Method

Processes an authentication query.

[Visual Basic] Private Sub ProcessQuery( _
  ByVal Query As Byte() _
)

[C#] private void ProcessQuery(
  byte[] Query
);

Parameters

Query
  The query to process.

See Also


Copyright © 2002, The KPD-Team
AuthUserPass.StartAuthentication Method

Starts the authentication process.


[C#] internal override void StartAuthentication(Socket Connection, AuthenticationCompleteDelegate Callback);

Parameters

Connection
The connection with the SOCKS client.

Callback
The method to call when the authentication is complete.

See Also


Copyright © 2002, The KPD-Team
### Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConsoleAttributes</td>
<td>The ConsoleAttributes class can change several attributes of your console window.</td>
</tr>
</tbody>
</table>

### Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSOLE_CURSOR_INFO</td>
<td>The CONSOLE_CURSOR_INFO structure contains information about the console cursor.</td>
</tr>
<tr>
<td>CONSOLE_SCREEN_BUFFER_INFO</td>
<td>The CONSOLE_SCREEN_BUFFER_INFO structure contains information about the console screen buffer.</td>
</tr>
<tr>
<td>COORD</td>
<td>The COORD structure defines the coordinates of a character cell in a console screen buffer. The origin of the coordinate system (0,0) is the top, left cell of the buffer.</td>
</tr>
<tr>
<td>SMALL_RECT</td>
<td>The SMALL_RECT structure defines the coordinates of the upper left and lower right corners of a rectangle.</td>
</tr>
</tbody>
</table>

### Enumerations

<table>
<thead>
<tr>
<th>Enumeration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConsoleColor</td>
<td>Enumerates all available colors for the forecolor or the backcolor of the console.</td>
</tr>
</tbody>
</table>
The CONSOLE_CURSOR_INFO structure contains information about the console cursor.

For a list of all members of this type, see CONSOLE_CURSOR_INFO Members.

System.Object  ValueType
CONSOLE_CURSOR_INFO

Requirements
Assembly: Proxy.exe

See Also
CONSOLE_CURSOR_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
CONSOLE_CURSOR_INFO Members

Public Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bVisible</td>
<td>Specifies the visibility of the cursor. If the cursor is visible, this member is TRUE (nonzero).</td>
</tr>
<tr>
<td>dwSize</td>
<td>Specifies a number between 1 and 100, indicating the percentage of the character cell that is filled by the cursor. The cursor appearance varies, ranging from completely filling the cell to showing up as a horizontal line at the bottom of the cell.</td>
</tr>
</tbody>
</table>

Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>ToString (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finalize (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>MemberwiseClone (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

See Also
CONSOLE_CURSOR_INFO Class
Org.Mentalis.Utilities.ConsoleAttributes Namespace
Copyright © 2002, The KPD-Team
The fields of the CONSOLE_CURSOR_INFO class are listed below. For a complete list of CONSOLE_CURSOR_INFO class members, see the CONSOLE_CURSOR_INFO Members topic.

### Public Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bVisible</td>
<td>Specifies the visibility of the cursor. If the cursor is visible, this member is TRUE (nonzero).</td>
</tr>
<tr>
<td>dwSize</td>
<td>Specifies a number between 1 and 100, indicating the percentage of the character cell that is filled by the cursor. The cursor appearance varies, ranging from completely filling the cell to showing up as a horizontal line at the bottom of the cell.</td>
</tr>
</tbody>
</table>

**See Also**

- [CONSOLE_CURSOR_INFO Class](#)

**Copyright © 2002, The KPD-Team**
CONSOLE_CURSOR_INFO.bVisible Field

Specifies the visibility of the cursor. If the cursor is visible, this member is TRUE (nonzero).

[Visual Basic] Public bVisible As Integer

[C#] public int bVisible;

See Also

CONSOLE_CURSOR_INFO Class | CONSOLE_CURSOR_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
CONSOLE_CURSOR_INFO.dwSize Field

Specifies a number between 1 and 100, indicating the percentage of the character cell that is filled by the cursor. The cursor appearance varies, ranging from completely filling the cell to showing up as a horizontal line at the bottom of the cell.

[Visual Basic] Public dwSize As Integer

[C#] public int dwSize;

See Also

CONSOLE_CURSOR_INFO Class | CONSOLE_CURSOR_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The CONSOLE_SCREEN_BUFFER_INFO structure contains information about a console screen buffer.

For a list of all members of this type, see CONSOLE_SCREEN_BUFFER_INFO Members.

System.Object  ValueType
CONSOLE_SCREEN_BUFFER_INFO

Requirements
Assembly: Proxy.exe

See Also
CONSOLE_SCREEN_BUFFER_INFO Members |
Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
## CONSOLE_SCREEN_BUFFER_INFO Members

### Public Instance Fields

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>dwCursorPosition</strong></td>
<td>Specifies the column and row coordinates of the cursor in the screen buffer.</td>
</tr>
<tr>
<td><strong>dwMaximumWindowSize</strong></td>
<td>Specifies the maximum size of the console window, given the current screen buffer size and font and the screen size.</td>
</tr>
<tr>
<td><strong>dwSize</strong></td>
<td>Specifies the size, in character columns and rows, of the screen buffer.</td>
</tr>
<tr>
<td><strong>srWindow</strong></td>
<td>Specifies a SMALL_RECT structure that contains the screen buffer coordinates of the upper-left and lower-right corners of the display window.</td>
</tr>
<tr>
<td><strong>wAttributes</strong></td>
<td>Specifies the foreground (text) and background color attributes to be used for characters that are written to a screen buffer by the WriteFile and WriteConsole functions, or echoed to a screen buffer by the ReadFile and ReadConsole functions. The attribute values are some combination of the following values: FOREGROUND_BLUE, FOREGROUND_GREEN, FOREGROUND_RED, FOREGROUND_INTENSITY, BACKGROUND_BLUE, BACKGROUND_GREEN, BACKGROUND_RED, and BACKGROUND_INTENSITY.</td>
</tr>
</tbody>
</table>
Public Instance Methods

- **Equals** (inherited from ValueType)
  - Select the method name to go to the Microsoft documentation.
- **GetHashCode** (inherited from ValueType)
  - Select the method name to go to the Microsoft documentation.
- **GetType** (inherited from Object)
  - Select the method name to go to the Microsoft documentation.
- **ToString** (inherited from ValueType)
  - Select the method name to go to the Microsoft documentation.

Protected Instance Methods

- **Finalize** (inherited from Object)
  - Select the method name to go to the Microsoft documentation.
- **MemberwiseClone** (inherited from Object)
  - Select the method name to go to the Microsoft documentation.

See Also

- [CONSOLE_SCREEN_BUFFER_INFO Class](#)

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
The fields of the `CONSOLE_SCREEN_BUFFER_INFO` class are listed below. For a complete list of `CONSOLE_SCREEN_BUFFER_INFO` class members, see the `CONSOLE_SCREEN_BUFFER_INFO Members` topic.

### Public Instance Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dwCursorPosition</code></td>
<td>Specifies the column and row coordinates of the cursor in the screen buffer.</td>
</tr>
<tr>
<td><code>dwMaximumWindowSize</code></td>
<td>Specifies the maximum size of the console window, given the current screen buffer size and font and the screen size.</td>
</tr>
<tr>
<td><code>dwSize</code></td>
<td>Specifies the size, in character columns and rows, of the screen buffer.</td>
</tr>
<tr>
<td><code>srWindow</code></td>
<td>Specifies a <code>SMALL_RECT</code> structure that contains the screen buffer coordinates of the upper-left and lower-right corners of the display window.</td>
</tr>
<tr>
<td><code>wAttributes</code></td>
<td>Specifies the foreground (text) and background color attributes to be used for characters that are written to a screen buffer by the WriteFile and WriteConsole functions, or echoed to a screen buffer by the ReadFile and ReadConsole functions. The attribute values are some combination of the following values: FOREGROUND_BLUE, FOREGROUND_GREEN, FOREGROUND_RED,</td>
</tr>
<tr>
<td>FOREGROUND_INTENSITY, BACKGROUND_BLUE, BACKGROUND_GREEN, BACKGROUND_RED, and BACKGROUND_INTENSITY.</td>
<td></td>
</tr>
</tbody>
</table>

See Also

CONSOLE_SCREEN_BUFFER_INFO Class | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
CONSOLE_SCREEN_BUFFER_INFO.dwCursorPosition Field

Specifies the column and row coordinates of the cursor in the screen buffer.


[C#] public COORD dwCursorPosition;

See Also

CONSOLE_SCREEN_BUFFER_INFO Class | CONSOLE_SCREEN_BUFFER_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
CONSOLE_SCREEN_BUFFER_INFO.dwMaximumWindowSize Field

Specifies the maximum size of the console window, given the current screen buffer size and font and the screen size.


[C#]
public COORD dwMaximumWindowSize;

See Also

CONSOLE_SCREEN_BUFFER_INFO Class | CONSOLE_SCREEN_BUFFER_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
CONSOLE_SCREEN_BUFFER_INFO.dwSize Field

Specifies the size, in character columns and rows, of the screen buffer.


[C#] public COORD dwSize;

See Also

CONSOLE_SCREEN_BUFFER_INFO Class | CONSOLE_SCREEN_BUFFER_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
CONSOLE_SCREEN_BUFFER_INFO.srWindow Field

Specifies a SMALL_RECT structure that contains the screen buffer coordinates of the upper-left and lower-right corners of the display window.


[C#]
public SMALL_RECT srWindow;

See Also

CONSOLE_SCREEN_BUFFER_INFO Class | CONSOLE_SCREEN_BUFFER_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Specifies the foreground (text) and background color attributes to be used for characters that are written to a screen buffer by the WriteFile and WriteConsole functions, or echoed to a screen buffer by the ReadFile and ReadConsole functions. The attribute values are some combination of the following values: FOREGROUND_BLUE, FOREGROUND_GREEN, FOREGROUND_RED, FOREGROUND_INTENSITY, BACKGROUND_BLUE, BACKGROUND_GREEN, BACKGROUND_RED, and BACKGROUND_INTENSITY.

[Visual Basic] Public wAttributes As Short

[C#] public short wAttributes;

See Also

CONSOLE_SCREEN_BUFFER_INFO Class | CONSOLE_SCREEN_BUFFER_INFO Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The ConsoleAttributes class can change several attributes of your console window.

For a list of all members of this type, see `ConsoleAttributes Members`.

```
System.Object ConsoleAttributes
```

Example

The following example will change the foreground color of the console, disable 'EchoInput', ask for a string and show that string.

```
ConsoleAttributes.ForeColor = ConsoleColor.White;
Console.Write("Please enter your password:
" + ThePass);
Console.WriteLine("Press enter to exit...");
```

Requirements

- **Namespace**: `Org.Mentalis.Utilities.ConsoleAttributes Namespace`
- **Assembly**: Proxy.exe

See Also

- `ConsoleAttributes Members` | `Org.Mentalis.Utilities.ConsoleAttributes Namespace`
## ConsoleAttributes Members

### Public Static (Shared) Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>BackColor</code></td>
<td>Gets or sets the color of the console background.</td>
</tr>
<tr>
<td><code>Caption</code></td>
<td>Gets or sets the caption of the console.</td>
</tr>
<tr>
<td><code>CursorVisible</code></td>
<td>Gets or sets whether the cursor is visible or not.</td>
</tr>
<tr>
<td><code>CursorX</code></td>
<td>Gets or sets the current cursor position on the x axis in the console.</td>
</tr>
<tr>
<td><code>CursorY</code></td>
<td>Gets or sets the current cursor position on the y axis in the console.</td>
</tr>
<tr>
<td><code>EchoInput</code></td>
<td>Gets or sets whether the console must echo the input or not.</td>
</tr>
<tr>
<td><code>ForeColor</code></td>
<td>Gets or sets the color of the console font.</td>
</tr>
<tr>
<td><code>OvrMode</code></td>
<td>Gets or sets whether the cursor is in overwrite-mode or not.</td>
</tr>
<tr>
<td><code>WindowHeight</code></td>
<td>Gets the height (in characters) of the console window.</td>
</tr>
<tr>
<td><code>WindowWidth</code></td>
<td>Gets the width (in characters) of the console window.</td>
</tr>
</tbody>
</table>

### Public Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>MoveCursor</code></td>
<td>Moves the cursor to the specified location.</td>
</tr>
</tbody>
</table>

### Internal Static (Shared) Methods

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function</strong></td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>GetConsoleMode</td>
</tr>
<tr>
<td>GetConsoleScreenBufferInfo</td>
</tr>
<tr>
<td>GetConsoleTitle</td>
</tr>
<tr>
<td>GetStdHandle</td>
</tr>
<tr>
<td>SetConsoleCursorInfo</td>
</tr>
<tr>
<td>SetConsoleCursorPosition</td>
</tr>
<tr>
<td>SetConsoleMode</td>
</tr>
<tr>
<td>SetConsoleTextAttribute</td>
</tr>
</tbody>
</table>
WriteFile or WriteConsole function, or echoed by the ReadFile or ReadConsole function. This function affects only text written after the function call.

<table>
<thead>
<tr>
<th>SetConsoleTitle</th>
<th>The SetConsoleTitle function sets the title bar string for the current console window.</th>
</tr>
</thead>
</table>

**Private Static (Shared) Fields**

<table>
<thead>
<tr>
<th>BacgroundColors</th>
<th>Lists all the possible background color values.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ENABLE_ECHO_INPUT</strong></td>
<td>Characters read by the ReadFile or ReadConsole function are written to the active screen buffer as they are read. This mode can be used only if the ENABLE_LINE_INPUT mode is also enabled.</td>
</tr>
<tr>
<td>ForegroundColors</td>
<td>Lists all the possible foreground color values.</td>
</tr>
<tr>
<td><strong>m_BackColor</strong></td>
<td>Holds the backcolor of the console window.</td>
</tr>
<tr>
<td><strong>m_CursorVisible</strong></td>
<td>Holds the value of the CursorVisible property.</td>
</tr>
<tr>
<td><strong>m_EchoInput</strong></td>
<td>Holds the value of the EchoInput property.</td>
</tr>
<tr>
<td><strong>m_ForeColor</strong></td>
<td>Holds the forecolor of the console window.</td>
</tr>
<tr>
<td><strong>m_OvrMode</strong></td>
<td>Holds the value of the OvrMode property.</td>
</tr>
<tr>
<td><strong>STD_INPUT_HANDLE</strong></td>
<td>Standard input handle.</td>
</tr>
<tr>
<td>STD_OUTPUT_HANDLE</td>
<td>Standard output handle.</td>
</tr>
</tbody>
</table>

Private Static (Shared) Methods

| ChangeCursor | Applies the current cursor settings. |

Public Instance Constructors

| ConsoleAttributes Constructor |

Public Instance Methods

| Equals (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| GetHashCode (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| GetType (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| ToString (inherited from Object) | Select the method name to go to the Microsoft documentation. |

Protected Instance Methods

| Finalize (inherited from Object) | Select the method name to go to the Microsoft documentation. |
| MemberwiseClone (inherited from Object) | Select the method name to go to the Microsoft documentation. |

See Also

- ConsoleAttributes Class | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes Constructor

[Visual Basic] Public Sub New()

[C#]
public ConsoleAttributes();

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The fields of the **ConsoleAttributes** class are listed below. For a complete list of **ConsoleAttributes** class members, see the **ConsoleAttributes Members** topic.

### Private Static (Shared) Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BacgroundColors</strong></td>
<td>Lists all the possible background color values.</td>
</tr>
<tr>
<td><strong>ENABLE_ECHO_INPUT</strong></td>
<td>Characters read by the ReadFile or ReadConsole function are written to the active screen buffer as they are read. This mode can be used only if the ENABLE_LINE_INPUT mode is also enabled.</td>
</tr>
<tr>
<td><strong>ForegroundColors</strong></td>
<td>Lists all the possible foreground color values.</td>
</tr>
<tr>
<td><strong>m_BackColor</strong></td>
<td>Holds the backcolor of the console window.</td>
</tr>
<tr>
<td><strong>m_CursorVisible</strong></td>
<td>Holds the value of the CursorVisible property.</td>
</tr>
<tr>
<td><strong>m_EchoInput</strong></td>
<td>Holds the value of the EchoInput property.</td>
</tr>
<tr>
<td><strong>m_ForeColor</strong></td>
<td>Holds the forecolor of the console window.</td>
</tr>
<tr>
<td><strong>m_OvrMode</strong></td>
<td>Holds the value of the OvrMode property.</td>
</tr>
<tr>
<td><strong>STD_INPUT_HANDLE</strong></td>
<td>Standard input handle.</td>
</tr>
<tr>
<td><strong>STD_OUTPUT_HANDLE</strong></td>
<td>Standard output handle.</td>
</tr>
</tbody>
</table>

### See Also

[ConsoleAttributes Class](#) | [Org.Mentalis.Utilities.ConsoleAttributes](#)
Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ConsoleAttributes.BacgroundColors Field

Lists all the possible background color values.

[Visual Basic] Private Shared BacgroundColors

[C#]
private static int[] BacgroundColors;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Characters read by the ReadFile or ReadConsole function are written to the active screen buffer as they are read. This mode can be used only if the ENABLE_LINE_INPUT mode is also enabled.

[Visual Basic] Private Shared ENABLE_ECHO_INPUT

[C#] private static _int_ ENABLE_ECHO_INPUT;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.ForegroundColors Field

Lists all the possible foreground color values.

[Visual Basic] Private Shared ForegroundColors

[C#]
private static int[] ForegroundColors;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.m_BackColor Field

Holds the backcolor of the console window.

[Visual Basic] Private Shared m_BackColor As

[C#]
private static ConsoleColor m_BackColor;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.m_CursorVisible Field

Holds the value of the CursorVisible property.

[Visual Basic] Private Shared m_CursorVisible

[C#]
private static bool m_CursorVisible;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
**ConsoleAttributes.m_EchoInput Field**

Holds the value of the EchoInput property.

**[Visual Basic]**

```vbnet
Private Shared m_EchoInput As ![Visual Basic]
```

**[C#]**

```csharp
private static bool m_EchoInput;
```

**See Also**

[ConsoleAttributes Class] | [ConsoleAttributes Members] | [Org.Mentalis.Utilities.ConsoleAttributes Namespace]

[Copyright © 2002, The KPD-Team]
**ConsoleAttributes.m_ForeColor Field**

Holds the forecolor of the console window.

**[Visual Basic]**

```vbnet
Private Shared m_ForeColor As !
```

**[C#]**

```csharp
private static ConsoleColor m_ForeColor;
```

**See Also**

[ConsoleAttributes Class](#) | [ConsoleAttributes Members](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)

[Copyright © 2002, The KPD-Team](#)
ConsoleAttributes.m_OvrMode Field

Holds the value of the OvrMode property.

[Visual Basic] Private Shared m_OvrMode As Boolean

[C#] private static bool m_OvrMode;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.STD_INPUT_HANDLE Field

Standard input handle.

[Visual Basic]
Private Shared STD_INPUT_HANDLE

[C#]
private static int STD_INPUT_HANDLE;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.STD_OUTPUT_HANDLE Field

Standard output handle.

[Visual Basic] Private Shared STD_OUTPUT_HANDLE

[C#] private static int STD_OUTPUT_HANDLE;

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The properties of the **ConsoleAttributes** class are listed below. For a complete list of **ConsoleAttributes** class members, see the **ConsoleAttributes Members** topic.

### Public Static (Shared) Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BackColor</strong></td>
<td>Gets or sets the color of the console background.</td>
</tr>
<tr>
<td><strong>Caption</strong></td>
<td>Gets or sets the caption of the console.</td>
</tr>
<tr>
<td><strong>CursorVisible</strong></td>
<td>Gets or sets whether the cursor is visible or not.</td>
</tr>
<tr>
<td><strong>CursorX</strong></td>
<td>Gets or sets the current cursor position on the x axis in the console.</td>
</tr>
<tr>
<td><strong>CursorY</strong></td>
<td>Gets or sets the current cursor position on the y axis in the console.</td>
</tr>
<tr>
<td><strong>EchoInput</strong></td>
<td>Gets or sets whether the console must echo the input or not.</td>
</tr>
<tr>
<td><strong>ForeColor</strong></td>
<td>Gets or sets the color of the console font.</td>
</tr>
<tr>
<td><strong>OvrMode</strong></td>
<td>Gets or sets whether the cursor is in overwrite-mode or not.</td>
</tr>
<tr>
<td><strong>WindowHeight</strong></td>
<td>Gets the height (in characters) of the console window.</td>
</tr>
<tr>
<td><strong>WindowWidth</strong></td>
<td>Gets the width (in characters) of the console window.</td>
</tr>
</tbody>
</table>

**See Also**

[ConsoleAttributes Class](#) | [Org.Mentalis.Utilities.ConsoleAttributes](#)
Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ConsoleAttributes.BackColor Property

Gets or sets the color of the console background.

[Visual Basic] Public Shared Property BackColor

[C#] public static ConsoleColor BackColor {get; set;}

Property Value

A value of the ConsoleColor enum that specifies the color of the console background.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.Caption Property

Gets or sets the caption of the console.

[Visual Basic] Public Shared Property Caption

[C#]
public static string Caption {get; set;}

Property Value

A String that specifies the caption of the console.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.CursorVisible Property

Gets or sets whether the cursor is visible or not.

[Visual Basic] Public Shared Property CursorVisible

[C#]
public static bool CursorVisible {get; set;}

Property Value

A boolean value that specifies the visibility of the cursor.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.CursorX Property

Gets or sets the current cursos position on the x axis in the console.

[Visual Basic] Public Shared Property CursorX

[C#]
public static short CursorX {get; set;}

Property Value

A short that specifies the current cursos position on the x axis in the console.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
**ConsoleAttributes.CursorY Property**

Gets or sets the current cursos position on the y axis in the console.

**[Visual Basic]**

```vbnet
Public Shared Property CursorY
```

**[C#]**

```csharp
public static short CursorY {get; set;}
```

**Property Value**

A short value that specifies the current cursos position on the y axis in the console.

**See Also**

- [ConsoleAttributes Class](#) | [ConsoleAttributes Members](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)

*Copyright © 2002, The KPD-Team*
ConsoleAttributes.EchoInput Property

Gets or sets whether the console must echo the input or not.

[Visual Basic]
Public Shared Property EchoInput

[C#]
public static bool EchoInput {get; set;}

Property Value

A boolean value that specifies the console must echo the input or not.

Remarks

EchoInput is often turned off when the program asks the user to type in a password.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes_FORE_COLOR Property

Gets or sets the color of the console font.

[Visual Basic] Public Shared Property ForeColor

[C#]
public static ConsoleColor ForeColor {get; set;}

Property Value

A value of the ConsoleColor enum that specifies the color of the console font.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
ConsoleAttributes.OvrMode Property

Gets or sets whether the cursor is in overwrite-mode or not.

[Visual Basic] Public Shared Property OvrMode

[C#]
public static bool OvrMode {get; set;}

Property Value
A boolean value that specifies the mode of the cursor.

Remarks
In overwrite mode, the cursor size will be 50% of the character space instead of 25% in normal mode.

See Also
ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ConsoleAttributes.WindowHeight Property

Gets the height (in characters) of the console window.

[Visual Basic] Public Shared Readonly Property

[C#]
public static int WindowHeight {get;}

Property Value

An integer that holds the height of the console window in characters.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
ConsoleAttributes.WindowWidth Property

Gets the width (in characters) of the console window.

[Visual Basic] Public Shared ReadOnly Property

[C#]
public static int WindowWidth {get;}

Property Value

An integer that holds the width of the console window in characters.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The methods of the `ConsoleAttributes` class are listed below. For a complete list of `ConsoleAttributes` class members, see the `ConsoleAttributes Members` topic.

**Public Static (Shared) Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MoveCursor</strong></td>
<td>Moves the cursor to the specified location.</td>
</tr>
</tbody>
</table>

**Internal Static (Shared) Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GetConsoleMode</strong></td>
<td>The <code>GetConsoleMode</code> function reports the current input mode of a console's input buffer or the current output mode of a console screen buffer.</td>
</tr>
<tr>
<td><strong>GetConsoleScreenBufferInfo</strong></td>
<td>The <code>GetConsoleScreenBufferInfo</code> function retrieves information about the specified console screen buffer.</td>
</tr>
<tr>
<td><strong>GetConsoleTitle</strong></td>
<td>The <code>GetConsoleTitle</code> function retrieves the title bar string for the current console window.</td>
</tr>
<tr>
<td><strong>GetStdHandle</strong></td>
<td>The <code>GetStdHandle</code> function returns a handle for the standard input, standard output, or standard error device.</td>
</tr>
<tr>
<td><strong>SetConsoleCursorInfo</strong></td>
<td>The <code>SetConsoleCursorInfo</code> function sets the size and visibility of the cursor for the specified console screen buffer.</td>
</tr>
<tr>
<td><strong>SetConsoleCursorPosition</strong></td>
<td>The <code>SetConsoleCursorPosition</code> function sets the cursor position in the specified console screen</td>
</tr>
</tbody>
</table>
SetConsoleMode  
The SetConsoleMode function sets the input mode of a console's input buffer or the output mode of a console screen buffer.

SetconsoleTextAttribute  
The SetConsoleTextAttribute function sets the foreground (text) and background color attributes of characters written to the screen buffer by the WriteFile or WriteConsole function, or echoed by the ReadFile or ReadConsole function. This function affects only text written after the function call.

SetconsoleTitle  
The SetConsoleTitle function sets the title bar string for the current console window.

Private Static (Shared) Methods

ChangeCursor  
Applies the current cursor settings.

Public Instance Methods

Equals (inherited from Object)  
Select the method name to go to the Microsoft documentation.

GetHashCode (inherited from Object)  
Select the method name to go to the Microsoft documentation.

GetType (inherited from Object)  
Select the method name to go to the Microsoft documentation.

ToString (inherited from Object)  
Select the method name to go to the Microsoft documentation.
## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from <strong>Object</strong>)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

### See Also

- [ConsoleAttributes Class](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)

**Copyright © 2002, The KPD-Team**
**ConsoleAttributes.ChangeCursor Method**

Applies the current cursor settings.

**Visual Basic**

```vbnet
Private Shared Sub ChangeCursor()
```

**C#**

```csharp
private static void ChangeCursor();
```

**Remarks**

This method applies changes in the CursorVisible and OvrMode properties.

**See Also**

- [ConsoleAttributes Class](#) | [ConsoleAttributes Members](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)

*Copyright © 2002, The KPD-Team*
The `GetConsoleMode` function reports the current input mode of a console's input buffer or the current output mode of a console screen buffer.

**Parameters**

*hConsoleHandle*

Handle to a console input buffer or a screen buffer. The handle must have GENERIC_READ access.

*lpConsoleCursorInfo*

Pointer to a 32-bit variable that indicates the current mode of the specified buffer. If the hConsoleHandle parameter is an input handle, the mode can be a combination of the following values. When a console is created, all input modes except ENABLE_WINDOW_INPUT are enabled by default.

- **ENABLE_LINE_INPUT** The ReadFile or ReadConsole function returns only when a carriage return character is read. If this mode is disabled, the functions return when one or more characters are available.

- **ENABLE_ECHO_INPUT** Characters read by the ReadFile or ReadConsole function are written to the active screen buffer as they are read. This mode can be used only if the ENABLE_LINE_INPUT mode is also...
enabled.

- **ENABLE_PROCESSED_INPUT**
  
  Ctrl+c is processed by the system and is not placed in the input buffer. If the input buffer is being read by ReadFile or ReadConsole, other control keys are processed by the system and are not returned in the ReadFile or ReadConsole buffer. If the ENABLE_LINE_INPUT mode is also enabled, backspace, carriage return, and linefeed characters are handled by the system.

- **ENABLE_WINDOW_INPUT**
  
  User interactions that change the size of the console screen buffer are reported in the console’s input buffer. Information about these events can be read from the input buffer by applications using the ReadConsoleInput function, but not by those using ReadFile or ReadConsole.

- **ENABLE_MOUSE_INPUT**
  
  If the mouse pointer is within the borders of the console window and the window has the keyboard focus, mouse events generated by mouse movement and button presses are placed in the input buffer. These events are discarded by ReadFile or ReadConsole, even when this mode is enabled.

If the hConsoleHandle parameter is a screen buffer handle, the mode can be a combination of the following values. When a screen buffer is created, both output modes are enabled by default.

- **ENABLE_PROCESSED_OUTPUT**
  
  Characters written by the WriteFile or WriteConsole function or echoed by the ReadFile or ReadConsole function are parsed for ASCII control sequences, and the correct action is performed. Backspace, tab, bell, carriage return, and linefeed characters are processed.

- **ENABLE_WRAP_AT_EOL_OUTPUT**
  
  When writing with WriteFile or WriteConsole or echoing with ReadFile or ReadConsole, the cursor moves to the beginning of the next row when it reaches the end of the current row. This causes the rows displayed in the console window to
scroll up automatically when the cursor advances beyond
the last row in the window. It also causes the contents of
the screen buffer to scroll up (discarding the top row of
the screen buffer) when the cursor advances beyond the
last row in the screen buffer. If this mode is disabled, the
last character in the row is overwritten with any
subsequent characters.

Return Value

If the function succeeds, the return value is nonzero.

If the function fails, the return value is zero. To get extended error
information, call GetLastError.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The GetConsoleScreenBufferInfo function retrieves information about the specified console screen buffer.

[Visual Basic]
Friend Shared Function GetConsoleScreenBufferInfo( _
    ByVal hConsoleOutput As Integer, _

[C#]
internal static int GetConsoleScreenBufferInfo( _
    int hConsoleOutput, _
    ref ConsoleAttributes.CONSOLE_SCREEN_BUFFER_INFO lpConsoleScreenBufferInfo);

Parameters

hConsoleOutput
Handle to a console screen buffer. The handle must have GENERIC_READ access.

lpConsoleScreenBufferInfo
Pointer to a CONSOLE_SCREEN_BUFFER_INFO structure in which the screen buffer information is returned.

Return Value
If the function succeeds, the return value is nonzero.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

See Also
ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
The GetConsoleTitle function retrieves the title bar string for the current console window.

**Parameters**

*lpConsoleTitle*  
Pointer to a buffer that receives a null-terminated string containing the text that appears in the title bar of the console window.

*nSize*  
Specifies the size, in characters, of the buffer pointed to by the *lpConsoleTitle* parameter.

**Return Value**

If the function succeeds, the return value is the length, in characters, of the string copied to the buffer.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

**See Also**

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The GetStdHandle function returns a handle for the standard input, standard output, or standard error device.

```
[Visual Basic]
Friend Shared Function GetStdHandle(
    ByVal nStdHandle As Integer
) As Integer

[C#]
internal static int GetStdHandle(
    int nStdHandle
);
```

Parameters

*nStdHandle*
  Specifies the device for which to return the handle. This parameter can have one of the following values:
  - **STD_INPUT_HANDLE** Standard input handle.
  - **STD_OUTPUT_HANDLE** Standard output handle.
  - **STD_ERROR_HANDLE** Standard error handle.

Return Value

If the function succeeds, the return value is a handle to the specified device.

If the function fails, the return value is the INVALID_HANDLE_VALUE flag. To get extended error information, call GetLastError.

See Also

  - [ConsoleAttributes Class](#) | [ConsoleAttributes Members](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)

Copyright © 2002, The KPD-Team
ConsoleAttributes.MoveCursor Method

Moves the cursor to the specified location.

[Visual Basic] Public Shared Sub MoveCursor(  
ByVal x As Short,  _  
ByVal y As Short _  
)

[C#]
public static void MoveCursor(  
    short x,  
    short y  
);

Parameters

x  
Specifies the x value of the new location.

y  
Specifies the y value of the new location.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The SetConsoleCursorInfo function sets the size and visibility of the cursor for the specified console screen buffer.

**Parameters**

- **hConsoleOutput**
  Handle to a console screen buffer. The handle must have GENERIC_WRITE access.

- **lpConsoleCursorInfo**
  Pointer to a CONSOLE_CURSOR_INFO structure containing the new specifications for the screen buffer's cursor.

**Return Value**

If the function succeeds, the return value is nonzero.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

**See Also**

[ConsoleAttributes Class](#) | [ConsoleAttributes Members](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)
The SetConsoleCursorPosition function sets the cursor position in the specified console screen buffer.

### Parameters

**hConsoleOutput**
- Handle to a console screen buffer. The handle must have GENERIC_WRITE access.

**dwCursorPosition**
- Specifies a COORD structure containing the new cursor position. The coordinates are the column and row of a screen buffer character cell. The coordinates must be within the boundaries of the screen buffer.

### Return Value

If the function succeeds, the return value is nonzero.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

### See Also

- [ConsoleAttributes Class](#) | [ConsoleAttributes Members](#) | [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)

Copyright © 2002, The KPD-Team
The SetConsoleMode function sets the input mode of a console's input buffer or the output mode of a console screen buffer.

**[Visual Basic]**

```
Friend Shared Function SetConsoleMode(ByVal hConsoleHandle As Integer, ByVal lpConsoleCursorInfo As Integer) As Integer
```

**[C#]**

```
internal static int SetConsoleMode(int hConsoleHandle, int lpConsoleCursorInfo);
```

**Parameters**

* `hConsoleHandle`  
  Handle to a console input buffer or a screen buffer. The handle must have GENERIC_WRITE access.

* `lpConsoleCursorInfo`  
  Pointer to a 32-bit variable that indicates the current mode of the specified buffer. If the hConsoleHandle parameter is an input handle, the mode can be a combination of the following values. When a console is created, all input modes except ENABLE_WINDOW_INPUT are enabled by default.

- **ENABLE_LINE_INPUT**  
  The ReadFile or ReadConsole function returns only when a carriage return character is read. If this mode is disabled, the functions return when one or more characters are available.

- **ENABLE_ECHO_INPUT**  
  Characters read by the ReadFile or ReadConsole function are written to the active screen buffer as they are read. This mode can be used only if the ENABLE_LINE_INPUT mode is also enabled.
- **ENABLE_PROCESSED_INPUT**
  
  `ctrl+c` is processed by the system and is not placed in the input buffer. If the input buffer is being read by `ReadFile` or `ReadConsole`, other control keys are processed by the system and are not returned in the `ReadFile` or `ReadConsole` buffer. If the **ENABLE_LINE_INPUT** mode is also enabled, backspace, carriage return, and linefeed characters are handled by the system.

- **ENABLE_WINDOW_INPUT**
  
  User interactions that change the size of the console screen buffer are reported in the console's input buffer. Information about these events can be read from the input buffer by applications using the `ReadConsoleInput` function, but not by those using `ReadFile` or `ReadConsole`.

- **ENABLE_MOUSE_INPUT**
  
  If the mouse pointer is within the borders of the console window and the window has the keyboard focus, mouse events generated by mouse movement and button presses are placed in the input buffer. These events are discarded by `ReadFile` or `ReadConsole`, even when this mode is enabled.

If the `hConsoleHandle` parameter is a screen buffer handle, the mode can be a combination of the following values. When a screen buffer is created, both output modes are enabled by default.

- **ENABLE_PROCESSED_OUTPUT**
  
  Characters written by the `WriteFile` or `WriteConsole` function or echoed by the `ReadFile` or `ReadConsole` function are parsed for ASCII control sequences, and the correct action is performed. Backspace, tab, bell, carriage return, and linefeed characters are processed.

- **ENABLE_WRAP_AT_EOL_OUTPUT**
  
  When writing with `WriteFile` or `WriteConsole` or echoing with `ReadFile` or `ReadConsole`, the cursor moves to the beginning of the next row when it reaches the end of the current row. This causes the rows displayed in the console window to scroll up automatically when the cursor advances beyond
the last row in the window. It also causes the contents of the screen buffer to scroll up (discarding the top row of the screen buffer) when the cursor advances beyond the last row in the screen buffer. If this mode is disabled, the last character in the row is overwritten with any subsequent characters.

Return Value

If the function succeeds, the return value is nonzero.

If the function fails, the return value is zero. To get extended error information, call GetLastError.

See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
The `SetConsoleTextAttribute` function sets the foreground (text) and background color attributes of characters written to the screen buffer by the `WriteFile` or `WriteConsole` function, or echoed by the `ReadFile` or `ReadConsole` function. This function affects only text written after the function call.

**Parameters**

- **hConsoleOutput**
  Handle to a console screen buffer. The handle must have GENERIC_READ access.

- **wAttributes**
  Specifies the foreground and background color attributes. Any combination of the following values can be specified:
  - FOREGROUND_BLUE
  - FOREGROUND_GREEN
  - FOREGROUND_RED
  - FOREGROUND_INTENSITY
  - BACKGROUND_BLUE
  - BACKGROUND_GREEN
  - BACKGROUND_RED
  - BACKGROUND_INTENSITY.

**Return Value**

- If the function succeeds, the return value is nonzero.
- If the function fails, the return value is zero. To get extended error information, call GetLastError.
See Also

ConsoleAttributes Class | ConsoleAttributes Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The `SetConsoleTitle` function sets the title bar string for the current console window.

**Parameters**

`lpConsoleTitle`  
Pointer to a null-terminated string that contains the string to appear in the title bar of the console window.

**Return Value**

If the function succeeds, the return value is nonzero.

If the function fails, the return value is zero. To get extended error information, call `GetLastError`.

**See Also**

- [ConsoleAttributes Class](#)  
- [ConsoleAttributes Members](#)  

---

**ConsoleAttributes.SetConsoleTitle Method**

```vbnet
Friend Shared Function SetConsoleTitle(  
    ByVal lpConsoleTitle As String  
) As Integer
```

```cs
internal static int SetConsoleTitle(  
    string lpConsoleTitle  
);
```
Mentalis.org Proxy Documentation
**ConsoleColor Enumeration**

Enumerates all available colors for the forecolor or the backcolor of the console.

**Visual Basic**

```vbnet
Public Enum ConsoleColor
```

**C#**

```csharp
public enum ConsoleColor
```

### Members

<table>
<thead>
<tr>
<th>Member Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black</td>
<td>Black</td>
</tr>
<tr>
<td>Red</td>
<td>Red</td>
</tr>
<tr>
<td>LightRed</td>
<td>Light red</td>
</tr>
<tr>
<td>Green</td>
<td>Green</td>
</tr>
<tr>
<td>LightGreen</td>
<td>Light green</td>
</tr>
<tr>
<td>Blue</td>
<td>Blue</td>
</tr>
<tr>
<td>LightBlue</td>
<td>Light blue</td>
</tr>
<tr>
<td>Gold</td>
<td>Gold</td>
</tr>
<tr>
<td>Yellow</td>
<td>Yellow</td>
</tr>
<tr>
<td>Cyan</td>
<td>Cyan</td>
</tr>
<tr>
<td>LightCyan</td>
<td>Light cyan</td>
</tr>
<tr>
<td>Purple</td>
<td>Purple</td>
</tr>
<tr>
<td>LightPurple</td>
<td>Light purple</td>
</tr>
<tr>
<td>Gray</td>
<td>Gray</td>
</tr>
<tr>
<td>White</td>
<td>White</td>
</tr>
</tbody>
</table>

### Requirements

**Namespace:** [Org.Mentalis.Utilities.ConsoleAttributes Namespace](#)
Assembly: Proxy.exe

See Also

Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
The COORD structure defines the coordinates of a character cell in a console screen buffer. The origin of the coordinate system (0,0) is at the top, left cell of the buffer.

For a list of all members of this type, see COORD Members.

Requirements

Assembly: Proxy.exe

See Also

COORD Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
COORD Members

Public Instance Fields

<table>
<thead>
<tr>
<th>x</th>
<th>Horizontal or column value.</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>Vertical or row value.</td>
</tr>
</tbody>
</table>

Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Documentation Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetHashCode (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>GetType (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>ToString (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Documentation Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finalize (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td>MemberwiseClone (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

See Also

COORD Class | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
COORD Fields

The fields of the COORD class are listed below. For a complete list of COORD class members, see the COORD Members topic.

Public Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Horizontal or column value.</td>
</tr>
<tr>
<td>y</td>
<td>Vertical or row value.</td>
</tr>
</tbody>
</table>

See Also

COORD Class | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation
**COORD.x Field**

Horizontal or column value.

```[Visual Basic]Public x As Short

[C#]public short x;
```

See Also

COORD Class | COORD Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
COORD.y Field

Vertical or row value.

[Visual Basic] Public y As Short

[C#]
public short y;

See Also

COORD Class | COORD Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
SMALL_RECT Structure

The SMALL_RECT structure defines the coordinates of the upper left and lower right corners of a rectangle.

For a list of all members of this type, see SMALL_RECT Members.

System.Object  ValueType
   SMALL_RECT

[Visual Basic]
Structure SMALL_RECT

[C#]
struct SMALL_RECT

Requirements

Namespace: Org.MentalisUtilities.ConsoleAttributes Namespace

Assembly: Proxy.exe

See Also

   SMALL_RECT Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
# SMALL_RECT Members

## Public Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bottom</strong></td>
<td>Specifies the y-coordinate of the lower right corner of the rectangle.</td>
</tr>
<tr>
<td><strong>Left</strong></td>
<td>Specifies the x-coordinate of the upper left corner of the rectangle.</td>
</tr>
<tr>
<td><strong>Right</strong></td>
<td>Specifies the x-coordinate of the lower right corner of the rectangle.</td>
</tr>
<tr>
<td><strong>Top</strong></td>
<td>Specifies the y-coordinate of the upper left corner of the rectangle.</td>
</tr>
</tbody>
</table>

## Public Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Equals</strong> (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetHashCode</strong> (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>GetType</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>ToString</strong> (inherited from ValueType)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## Protected Instance Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalize</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
<tr>
<td><strong>MemberwiseClone</strong> (inherited from Object)</td>
<td>Select the method name to go to the Microsoft documentation.</td>
</tr>
</tbody>
</table>

## See Also

- SMALL_RECT Members
SMALL_RECT Fields

The fields of the SMALL_RECT class are listed below. For a complete list of SMALL_RECT class members, see the SMALL_RECT Members topic.

Public Instance Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bottom</td>
<td>Specifies the y-coordinate of the lower right corner of the rectangle.</td>
</tr>
<tr>
<td>Left</td>
<td>Specifies the x-coordinate of the upper left corner of the rectangle.</td>
</tr>
<tr>
<td>Right</td>
<td>Specifies the x-coordinate of the lower right corner of the rectangle.</td>
</tr>
<tr>
<td>Top</td>
<td>Specifies the y-coordinate of the upper left corner of the rectangle.</td>
</tr>
</tbody>
</table>

See Also

SMALL_RECT Class | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
SMALL_RECT.Bottom Field

Specifies the y-coordinate of the lower right corner of the rectangle.

[Visual Basic] Public Bottom As Short

[C#]
public short Bottom;

See Also

SMALL_RECT Class | SMALL_RECT Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
**SMALL_RECT.Left Field**

Specifies the x-coordinate of the upper left corner of the rectangle.

[Visual Basic]
```vbnet
Public Left As Short
```

[C#]
```csharp
public short Left;
```

See Also

[SMLRECT Class] [SMALL_RECT Members] [Org.Mentalis.Utilities.ConsoleAttributes Namespace]

Copyright © 2002, The KPD-Team
SMALL_RECT.Right Field

Specifies the x-coordinate of the lower right corner of the rectangle.

[Visual Basic] Public Right As Short

[C#] public short Right;

See Also

SMALL_RECT Class | SMALL_RECT Members | Org.Mentalis.Utilities.ConsoleAttributes Namespace

Copyright © 2002, The KPD-Team
**SMALL_RECT.Top Field**

Specifies the y-coordinate of the upper left corner of the rectangle.

[Visual Basic] **Public Top As Short**

[C#]  
**public short Top;**

See Also


Copyright © 2002, The KPD-Team
Org.Mentalis.Proxy namespace documentation
Copyright © 2002, The KPD-Team
All rights reserved.
http://www.mentalis.org/

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Neither the name of the KPD-Team, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
Mentalis.org Proxy Documentation

- System.Object
  - FtpClient
  - FtpDataConnection
  - HttpClient
  - PortMapClient
  - SocksClient
- Listener
  - FtpListener
  - HttpListener
  - PortMapListener
  - SocksListener

Copyright © 2002, The KPD-Team
Org.Mentalis.Proxy.PortMap Hierarchy

-System.Object
  -Client
    -FtpClient
    -FtpDataConnection
    -HttpClient
    -PortMapClient
    -SocksClient
  -Listener
    -FtpListener
    -HttpListener
    -PortMapListener
    -SocksListener

Copyright © 2002, The KPD-Team
Org.Mentalis.Proxy.Socks Hierarchy

System.Object - Client
  - FtpClient
  - FtpDataConnection
  - HttpClient
  - PortMapClient
  - SocksClient
  - Listener
    - FtpListener
    - HttpListener
    - PortMapListener
    - SocksListener
    - SocksHandler
      - Socks4Handler
      - Socks5Handler

Copyright © 2002, The KPD-Team
Mentalis.org Proxy Documentation

System.Object - AuthBase
- AuthNone
- AuthUserPass
- AuthenticationList

Copyright © 2002, The KPD-Team

System.Object - ConsoleAttributes
    - ValueType

Copyright © 2002, The KPD-Team