

Majestic12 Namespace

[Namespace hierarchy](#)

Classes

Class	Description
HTMLchunk	Parsed HTML token that is either text, comment, script, open or closed tag as indicated by the oType variable.
HTMLparser	Allows to parse HTML by splitting it into small token (HTMLchunks) such as tags, text, comments etc. Do NOT create multiple instances of this class - REUSE single instance Do NOT call same instance from multiple threads - it is NOT thread safe

Enumerations

Enumeration	Description
HTMLchunkType	Type of parsed HTML chunk (token), each non-null returned chunk from HTMLparser will have oType set to one of these values

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk Class

Parsed HTML token that is either text, comment, script, open or closed tag as indicated by the oType variable.

For a list of all members of this type, see [HTMLchunk Members](#).

[System.Object](#) **Majestic12.HTMLchunk**

```
public class HTMLchunk : IDisposable
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: [Majestic12](#)

Assembly: HTMLparserDotNet11 (in HTMLparserDotNet11.exe)

See Also

[HTMLchunk Members](#) | [Majestic12 Namespace](#)


[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk Members

[HTMLchunk overview](#)







Public Static Methods

 MakeSafeParamValue	Makes parameter value safe to be used in param - this will check for any conflicting quote chars, but not full entity-encoding
--	--

Public Instance Constructors

 HTMLchunk Constructor	Initialises new HTMLchunk
---	---------------------------

Public Instance Fields

 bHashMode	If true then tag params will be kept in a hash rather than in a fixed size arrays. This will be slow down parsing, but make it easier to use.
 cParamChars	Character used to quote param's value: it is taken actually from parsed HTML
 iChunkLength	Length of the chunk in bHTML data array
 iChunkOffset	Offset in bHTML data array at which this chunk starts
 iParams	Number of parameters and values stored in sParams array, OR in oParams hashtable if bHashMode is true
 oEnc	Encoder to be used for conversion of binary data into strings, Encoding.Default is used by default, but it can be

	changed if top level user of the parser detects that encoding was different
◆ oHTML	For TAGS: it stores raw HTML that was parsed to generate thus chunk will be here UNLESS HTMLparser was configured not to store it there as it can improve performance For TEXT or COMMENTS: actual text or comments - you MUST call Finalise(); first.
◆ oParams	Hashtable with tag parameters: keys are param names and values are param values. ONLY used if bHashMode is set to TRUE.
◆ oType	Chunk type showing whether its text, open or close tag, comments or script. WARNING: if type is comments or script then you have to manually call Finalise(); method in order to have actual text of comments/scripts in oHTML variable
◆ sParams	Param names will be stored here - actual number is in iParams. ONLY used if bHashMode is set to FALSE.
◆ sTag	If its open/close tag type then this is where lowercased Tag will be kept
◆ sValues	Param values will be stored here - actual number is in iParams. ONLY used if

bHashMode is set to FALSE.

Public Instance Methods

AddParam	Adds tag parameter to the chunk
Clear	Clears chunk preparing it for
ConvertParamsToHash	This function will convert parameters stored in sParams/sValues arrays into oParams hash Useful if generally parsing is done when bHashMode is FALSE. Hash operations are not the fastest, so its best not to use this function.
Dispose	
Equals (inherited from Object)	Determines whether the specified Object is equal to the current Object .
GenerateHTML	Generates HTML based on current chunk's data Note: this is not a high performance method and if you want ORIGINAL HTML that was parsed to create this chunk then use relevant HTMLparser method to obtain such HTML
GenerateParamHTML	Generates HTML for param/value pair
GenerateParamsHTML	Generates HTML for params in this chunk
GetHashCode (inherited from Object)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data

	structures like a hash table.
◆ GetParamValue	Returns value of a parameter
◆ GetType (inherited from Object)	Gets the Type of the current instance.
◆ SetEncoding	Sets encoding to be used for conversion of binary data into string
◆ ToString (inherited from Object)	Returns a String that represents the current Object .

Protected Instance Methods

◆ Finalize (inherited from Object)	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.
◆ MemberwiseClone (inherited from Object)	Creates a shallow copy of the current Object .

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk Constructor

Initialises new HTMLchunk

```
public HTMLchunk(  
    bool p_bHashMode  
);
```

Parameters

p_bHashMode
Sets

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#) | [bHashMode](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk Fields

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

Public Instance Fields

bHashMode	If true then tag params will be kept in a hash rather than in a fixed size arrays. This will be slow down parsing, but make it easier to use.
cParamChars	Character used to quote param's value: it is taken actually from parsed HTML
iChunkLength	Length of the chunk in bHTML data array
iChunkOffset	Offset in bHTML data array at which this chunk starts
iParams	Number of parameters and values stored in sParams array, OR in oParams hashtable if bHashMode is true
oEnc	Encoder to be used for conversion of binary data into strings, Encoding.Default is used by default, but it can be changed if top level user of the parser detects that encoding was different
oHTML	For TAGS: it stores raw HTML that was parsed to generate thus chunk will be here UNLESS HTMLparser was configured not to store it there

	as it can improve performance For TEXT or COMMENTS: actual text or comments - you MUST call Finalise(); first.
oParams	Hashtable with tag parameters: keys are param names and values are param values. ONLY used if bHashMode is set to TRUE.
oType	Chunk type showing whether its text, open or close tag, comments or script. WARNING: if type is comments or script then you have to manually call Finalise(); method in order to have actual text of comments/scripts in oHTML variable
sParams	Param names will be stored here - actual number is in iParams. ONLY used if bHashMode is set to FALSE.
sTag	If its open/close tag type then this is where lowercased Tag will be kept
sValues	Param values will be stored here - actual number is in iParams. ONLY used if bHashMode is set to FALSE.

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.bHashMode Field

If true then tag params will be kept in a hash rather than in a fixed size arrays. This will be slow down parsing, but make it easier to use.

```
public bool bHashMode;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.cParamChars Field

Character used to quote param's value: it is taken actually from parsed HTML

```
public byte[] cParamChars;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.iChunkLength Field

Length of the chunk in bHTML data array

```
public int iChunkLength;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.iChunkOffset Field

Offset in bHTML data array at which this chunk starts

```
public int iChunkOffset;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.iParams Field

Number of parameters and values stored in sParams array, OR in oParams hashtable if bHashMode is true

```
public int iParams;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.oEnc Field

Encoder to be used for conversion of binary data into strings, Encoding.Default is used by default, but it can be changed if top level user of the parser detects that encoding was different

```
public Encoding oEnc;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.oHTML Field

For TAGS: it stores raw HTML that was parsed to generate thus chunk will be here UNLESS HTMLparser was configured not to store it there as it can improve performance

For TEXT or COMMENTS: actual text or comments - you MUST call Finalise(); first.

```
public string oHTML;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.oParams Field

Hashtable with tag parameters: keys are param names and values are param values. ONLY used if bHashMode is set to TRUE.

```
public Hashtable oParams;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.oType Field

Chunk type showing whether its text, open or close tag, comments or script. WARNING: if type is comments or script then you have to manually call Finalise(); method in order to have actual text of comments/scripts in oHTML variable

```
public HTMLchunkType oType;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.sParams Field

Param names will be stored here - actual number is in iParams.
ONLY used if bHashMode is set to FALSE.

```
public string[] sParams;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.sTag Field

If its open/close tag type then this is where lowercased Tag will be kept

```
public string sTag;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.sValues Field

Param values will be stored here - actual number is in iParams.
ONLY used if bHashMode is set to FALSE.

```
public string[] sValues;
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)


[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)







HTMLchunk Methods








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

Public Static Methods



 § MakeSafeParamValue	Makes parameter value safe to be used in param - this will check for any conflicting quote chars, but not full entity-encoding
--	--

Public Instance Methods

 AddParam	Adds tag parameter to the chunk
 Clear	Clears chunk preparing it for
 ConvertParamsToHash	This function will convert parameters stored in sParams/sValues arrays into oParams hash Useful if generally parsing is done when bHashMode is FALSE. Hash operations are not the fastest, so its best not to use this function.
 Dispose	
 Equals (inherited from Object)	Determines whether the specified Object is equal to the current Object .
 GenerateHTML	Generates HTML based on current chunk's data Note: this is not a high performance method and if you want ORIGINAL HTML that was

	parsed to create this chunk then use relevant HTMLparser method to obtain such HTML
 GenerateParamHTML	Generates HTML for param/value pair
 GenerateParamsHTML	Generates HTML for params in this chunk
 GetHashCode (inherited from Object)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.
 GetParamValue	Returns value of a parameter
 GetType (inherited from Object)	Gets the Type of the current instance.
 SetEncoding	Sets encoding to be used for conversion of binary data into string
 ToString (inherited from Object)	Returns a String that represents the current Object .

Protected Instance Methods

 Finalize (inherited from Object)	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.
 MemberwiseClone (inherited from Object)	Creates a shallow copy of the current Object .

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

Copyright (c) Alex Chudnovsky, Majestic-12 (UK)

HTMLchunk.AddParam Method

Adds tag parameter to the chunk

```
public void AddParam(  
    string sParam,  
    string sValue,  
    byte cParamChar  
);
```

Parameters

sParam

Parameter name (ie color)

sValue

Value of the parameter (ie white)

cParamChar

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.Clear Method

Clears chunk preparing it for

```
public void Clear();
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.ConvertParamsToHash Method

This function will convert parameters stored in sParams/sValues arrays into oParams hash Useful if generally parsing is done when bHashMode is FALSE. Hash operations are not the fastest, so its best not to use this function.

```
public void ConvertParamsToHash();
```

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.Dispose Method

```
public void Dispose();
```

Implements

[IDisposable.Dispose](#)

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.GenerateHTML Method

Generates HTML based on current chunk's data Note: this is not a high performance method and if you want ORIGINAL HTML that was parsed to create this chunk then use relevant HTMLparser method to obtain such HTML

```
public string GenerateHTML();
```

Return Value

HTML equivalent of this chunk

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.GenerateParamHTML Method

Generates HTML for param/value pair

```
public string GenerateParamHTML(  
    string sParam,  
    string sValue,  
    char cParamChar  
);
```

Parameters

sParam

Param

sValue

Value (empty if not specified)

cParamChar

Return Value

String with HTML

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.GenerateParamsHTML Method

Generates HTML for params in this chunk

```
public string GenerateParamsHTML();
```

Return Value

String with HTML corresponding to params

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.GetParamValue Method

Returns value of a parameter

```
public string GetParamValue(  
    string sParam  
);
```

Parameters

sParam
Parameter

Return Value

Parameter value or empty string

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.MakeSafeParamValue Method

Makes parameter value safe to be used in param - this will check for any conflicting quote chars, but not full entity-encoding

```
public static string MakeSafeParamValue(  
    string sLine,  
    char cQuoteChar  
);
```

Parameters

sLine

Line of text

cQuoteChar

Quote char used in param - any such chars in text will be entity-encoded

Return Value

Safe text to be used as param's value

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunk.SetEncoding Method

Sets encoding to be used for conversion of binary data into string

```
public void SetEncoding(  
    Encoding p_oEnc  
);
```

Parameters

p_oEnc
Encoding object

See Also

[HTMLchunk Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLchunkType Enumeration

Type of parsed HTML chunk (token), each non-null returned chunk from HTMLparser will have oType set to one of these values

```
public enum HTMLchunkType
```

Members

Member Name	Description
Text	Text data from HTML
OpenTag	Open tag, possibly with attributes
CloseTag	Closed tag (it may still have attributes)
Comment	Comment tag () depending on HTMLparser boolean flags you may have: a) nothing to oHTML variable - for faster performance, call SetRawHTML function in parser b) data BETWEEN tags (but not including comment tags themselves) - DEFAULT c) complete RAW HTML representing data between tags and tags themselves (same as you get in a) when you call SetRawHTML function)
Script	Script tag () depending on HTMLparser boolean flags a) nothing to oHTML variable - for faster performance, call SetRawHTML function in parser b) data BETWEEN tags (but not including comment tags

themselves) - DEFAULT c)
complete RAW HTML
representing data between tags
and tags themselves (same as
you get in a) when you call
SetRawHTML function)

Requirements

Namespace: [Majestic12](#)

Assembly: HTMLparserDotNet11 (in HTMLparserDotNet11.exe)

See Also

[Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser Class

Allows to parse HTML by splitting it into small token (HTMLchunks) such as tags, text, comments etc. Do NOT create multiple instances of this class - REUSE single instance Do NOT call same instance from multiple threads - it is NOT thread safe

For a list of all members of this type, see [HTMLparser Members](#).

[System.Object](#) **Majestic12.HTMLparser**

```
public class HTMLparser : IDisposable
```

Thread Safety

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

Requirements

Namespace: [Majestic12](#)

Assembly: HTMLparserDotNet11 (in HTMLparserDotNet11.exe)

See Also

[HTMLparser Members](#) | [Majestic12 Namespace](#)






[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)


HTMLparser Members

[HTMLparser overview](#)


Public Static Methods

 CalculateWidth	Parses WIDTH param and calculates width
 DecodeEntities	This function will decode any entities found in a string - not fast!
 IsBiggerFont	Checks if first font is bigger than the second
 IsEqualOrBiggerFont	Checks if first font is equal or bigger than the second
 ParseFontSize	Parses font's tag size param

Public Instance Constructors

 HTMLparser	Overloaded. Initializes a new instance of the HTMLparser class.
--	---




Public Instance Fields

 bAutoExtractBetweenTagsOnly	If true (and either bAutoKeepComments or bAutoKeepScripts is true), then oHTML will be set to data BETWEEN tags excluding those tags themselves, as otherwise FULL HTML will be set, ie: " but if this is set to true then only ' comments ' will be returned
---	---







<p>bAutoKeepComments</p>	<p>If true (default) then HTML for comments tags themselves AND between them will be set to oHTML variable, otherwise it will be empty but you can always set it later</p>
<p>bAutoKeepScripts</p>	<p>If true (default: false) then HTML for script tags themselves AND between them will be set to oHTML variable, otherwise it will be empty but you can always set it later</p>
<p>bAutoMarkClosedTagsWithParamsAsOpen</p>	<p>Long winded name... by default if tag is closed BUT it has got parameters then we will consider it open tag, this is not right for proper XML parsing</p>
<p>bCompressWhiteSpaceBeforeTag</p>	<p>If true (default), then all whitespace before TAG starts will be compressed to single space char (32 or 0x20) this makes parser run a bit faster, if you need exact whitespace before tags then change this flag to FALSE</p>
<p>oHE</p>	<p>Heuristics engine used by Tag Parser to</p>

	quickly match known tags and attribute names, can be disabled or you can add more tags to it to fit your most likely cases, it is currently tuned for HTML
--	--

Public Instance Properties

 bDecodeEntities	
 bDecodeMiniEntities	
 bEnableHeuristics	If true (default) then heuristics engine will be used to match tags and attributes quicker, it is possible to add new tags to it, oHE

Public Instance Methods

 ChangeToEntities	Parses line and changes known entry characters into proper HTML entries
 CleanUp	Cleans up parser in preparation for next parsing
 Close	Closes object and releases all allocated resources
 Dispose	
 Equals (inherited from Object)	Determines whether the specified Object is equal to the current Object .
 GetHashCode (inherited from Object)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data structures like a hash table.

◆ GetType (inherited from Object)	Gets the Type of the current instance.
◆ Init	Overloaded. Initialises parses with HTML to be parsed from provided string
◆ InitMiniEntities	Inits mini-entities mode: only "nbsp" will be converted into space, all other entities will be left as is
◆ LoadFromFile	Loads HTML from file
◆ ParseNext	Parses next chunk and returns it with
◆ ParseNextTag	Returns next tag or null if end of document, text will be ignored completely
◆ Reset	Resets current parsed data to start
◆ SetChunkHashMode	Sets chunk param hash mode
◆ SetEncoding	Overloaded. Sets encoding
◆ SetRawHTML	Sets oHTML variable in a chunk to the raw HTML that was parsed for that chunk.
◆ ToString (inherited from Object)	Returns a String that represents the current Object .

Protected Instance Methods

◆ Finalize (inherited from Object)	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.
◆ MemberwiseClone (inherited from Object)	Creates a shallow copy of the current Object .

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser Constructor

Initializes a new instance of the [HTMLparser](#) class.

Overload List

Initializes a new instance of the [HTMLparser](#) class.

[public HTMLparser\(\);](#)

Constructs parser object using provided HTML as source for parsing

[public HTMLparser\(string\);](#)

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser Constructor ()

Initializes a new instance of the [HTMLparser](#) class.

```
public HTMLparser();
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#) | [HTMLparser Constructor Overload List](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser Constructor (String)

Constructs parser object using provided HTML as source for parsing

```
public HTMLparser(  
    string p_oHTML  
);
```

Parameters

p_oHTML

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#) | [HTMLparser Constructor Overload List](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser Fields

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

Public Instance Fields

bAutoExtractBetweenTagsOnly	If true (and either <code>bAutoKeepComments</code> or <code>bAutoKeepScripts</code> is true), then <code>oHTML</code> will be set to data BETWEEN tags excluding those tags themselves, as otherwise FULL HTML will be set, ie: " but if this is set to true then only ' comments ' will be returned
bAutoKeepComments	If true (default) then HTML for comments tags themselves AND between them will be set to <code>oHTML</code> variable, otherwise it will be empty but you can always set it later
bAutoKeepScripts	If true (default: false) then HTML for script tags themselves AND between them will be set to <code>oHTML</code> variable, otherwise it will be empty but you can always set it later
Long winded name...	Long winded name...

bAutoMarkClosedTagsWithParamsAsOpen	<p>by default if tag is closed BUT it has got parameters then we will consider it open tag, this is not right for proper XML parsing</p>
bCompressWhiteSpaceBeforeTag	<p>If true (default), then all whitespace before TAG starts will be compressed to single space char (32 or 0x20) this makes parser run a bit faster, if you need exact whitespace before tags then change this flag to FALSE</p>
oHE	<p>Heuristics engine used by Tag Parser to quickly match known tags and attribute names, can be disabled or you can add more tags to it to fit your most likely cases, it is currently tuned for HTML</p>

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bAutoExtractBetweenTagsOnly Field

If true (and either bAutoKeepComments or bAutoKeepScripts is true), then oHTML will be set to data BETWEEN tags excluding those tags themselves, as otherwise FULL HTML will be set, ie: " but if this is set to true then only ' comments ' will be returned

```
public bool bAutoExtractBetweenTagsOnly;
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bAutoKeepComments Field

If true (default) then HTML for comments tags themselves AND between them will be set to oHTML variable, otherwise it will be empty but you can always set it later

```
public bool bAutoKeepComments;
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bAutoKeepScripts Field

If true (default: false) then HTML for script tags themselves AND between them will be set to oHTML variable, otherwise it will be empty but you can always set it later

```
public bool bAutoKeepScripts;
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bAutoMarkClosedTagsWithParamsAsOpen Field

Long winded name... by default if tag is closed BUT it has got parameters then we will consider it open tag, this is not right for proper XML parsing

```
public bool bAutoMarkClosedTagsWithParamsAsOpen
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bCompressWhiteSpaceBeforeTag Field

If true (default), then all whitespace before TAG starts will be compressed to single space char (32 or 0x20) this makes parser run a bit faster, if you need exact whitespace before tags then change this flag to FALSE

```
public bool bCompressWhiteSpaceBeforeTag;
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.oHE Field

Heuristics engine used by Tag Parser to quickly match known tags and attribute names, can be disabled or you can add more tags to it to fit your most likely cases, it is currently tuned for HTML

```
public HTMLheuristics oHE;
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)




[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser Properties

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

Public Instance Properties

 bDecodeEntities	
 bDecodeMiniEntities	
 bEnableHeuristics	If true (default) then heuristics engine will be used to match tags and attributes quicker, it is possible to add new tags to it, oHE

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bDecodeEntities Property

```
public bool bDecodeEntities {get; set;}
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bDecodeMiniEntities Property

```
public bool bDecodeMiniEntities {get; set;}
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.bEnableHeuristics Property

If true (default) then heuristics engine will be used to match tags and attributes quicker, it is possible to add new tags to it, [oHE](#)

```
public bool bEnableHeuristics {get; set;}
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)






[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)







HTMLparser Methods

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

Public Static Methods

 CalculateWidth	Parses WIDTH param and calculates width
 DecodeEntities	This function will decode any entities found in a string - not fast!
 IsBiggerFont	Checks if first font is bigger than the second
 IsEqualOrBiggerFont	Checks if first font is equal or bigger than the second
 ParseFontSize	Parses font's tag size param


Public Instance Methods

 ChangeToEntities	Parses line and changes known entry characters into proper HTML entries
 CleanUp	Cleans up parser in preparation for next parsing
 Close	Closes object and releases all allocated resources
 Dispose	
 Equals (inherited from Object)	Determines whether the specified Object is equal to the current Object .
 GetHashCode (inherited from Object)	Serves as a hash function for a particular type, suitable for use in hashing algorithms and data

	structures like a hash table.
◆ GetType (inherited from Object)	Gets the Type of the current instance.
◆ Init	Overloaded. Initialises parses with HTML to be parsed from provided string
◆ InitMiniEntities	Inits mini-entities mode: only "nbsp" will be converted into space, all other entities will be left as is
◆ LoadFromFile	Loads HTML from file
◆ ParseNext	Parses next chunk and returns it with
◆ ParseNextTag	Returns next tag or null if end of document, text will be ignored completely
◆ Reset	Resets current parsed data to start
◆ SetChunkHashMode	Sets chunk param hash mode
◆ SetEncoding	Overloaded. Sets encoding
◆ SetRawHTML	Sets oHTML variable in a chunk to the raw HTML that was parsed for that chunk.
◆ ToString (inherited from Object)	Returns a String that represents the current Object .

Protected Instance Methods

◆ Finalize (inherited from Object)	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection.

 [MemberwiseClone](#) (inherited from **Object**)

Creates a shallow copy of the current [Object](#).

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.CalculateWidth Method

Parses WIDTH param and calculates width

```
public static int CalculateWidth(  
    string sWidth,  
    int iAvailWidth,  
    ref bool bRelative  
);
```

Parameters

sWidth

WIDTH param from tag

iAvailWidth

Currently available width for relative calculations, if negative width will be returned as is

bRelative

Flag that will be set to true if width was relative

Return Value

Width in pixels

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.ChangeToEntities Method

Parses line and changes known entry characters into proper HTML entities

```
public string ChangeToEntities(  
    string sLine  
);
```

Parameters

sLine
Line of text

Return Value

Line of text with proper HTML entities

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.CleanUp Method

Cleans up parser in preparation for next parsing

```
public void CleanUp();
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.Close Method

Closes object and releases all allocated resources

```
public void Close();
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.DecodeEntities Method

This function will decode any entities found in a string - not fast!

```
public static string DecodeEntities(  
    string sData  
);
```

Return Value

Possibly decoded string

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.Dispose Method

```
public void Dispose();
```

Implements

[IDisposable.Dispose](#)

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.Init Method

Initialises parses with HTML to be parsed from provided data buffer: this is best in terms of correctness of parsing of various encodings that can be used in HTML

Overload List

Initialises parses with HTML to be parsed from provided data buffer: this is best in terms of correctness of parsing of various encodings that can be used in HTML

[public void Init\(byte\[\]\):](#)

Initialises parses with HTML to be parsed from provided string

[public void Init\(string\):](#)

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.Init Method (Byte[])

Initialises parser with HTML to be parsed from provided data buffer: this is best in terms of correctness of parsing of various encodings that can be used in HTML

```
public void Init(  
    byte[] p_bHTML  
);
```

Parameters

p_bHTML
Data buffer with HTML in it

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#) | [HTMLparser.Init Overload List](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.Init Method (String)

Initialises parser with HTML to be parsed from provided string

```
public void Init(  
    string p_oHTML  
);
```

Parameters

p_oHTML
String with HTML in it

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#) | [HTMLparser.Init Overload List](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.InitMiniEntities Method

Inits mini-entities mode: only "nbsp" will be converted into space, all other entities will be left as is

```
public void InitMiniEntities();
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.IsBiggerFont Method

Checks if first font is bigger than the second

```
public static bool IsBiggerFont(  
    FontSize oFont1,  
    FontSize oFont2  
);
```

Parameters

oFont1
Font #1

oFont2
Font #2

Return Value

True if Font #1 bigger than the second, false otherwise

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.IsEqualOrBiggerFont Method

Checks if first font is equal or bigger than the second

```
public static bool IsEqualOrBiggerFont(  
    FontSize oFont1,  
    FontSize oFont2  
);
```

Parameters

oFont1
Font #1

oFont2
Font #2

Return Value

True if Font #1 equal or bigger than the second, false otherwise

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.LoadFromFile Method

Loads HTML from file

```
public void LoadFromFile(  
    string sFileName  
);
```

Parameters

sFileName

Full filename

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.ParseFontSize Method

Parses font's tag size param

```
public static FontSize ParseFontSize(  
    string sSize,  
    FontSize oCurSize  
);
```

Parameters

sSize

String value of the size param

oCurSize

Return Value

Relative size of the font size or Unknown if it was not determined

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.ParseNext Method

Parses next chunk and returns it with

```
public HTMLchunk ParseNext();
```

Return Value

HTMLchunk or null if end of data reached

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.ParseNextTag Method

Returns next tag or null if end of document, text will be ignored completely

```
public HTMLchunk ParseNextTag( );
```

Return Value

Tag chunk or null

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.Reset Method

Resets current parsed data to start

```
public void Reset();
```

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.SetChunkHashMode Method

Sets chunk param hash mode

```
public void SetChunkHashMode(  
    bool bHashMode  
);
```

Parameters

bHashMode

If true then tag's params will be kept in Chunk's hashtable (slower), otherwise kept in arrays (sParams/sValues)

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.SetEncoding Method

Sets current encoding in format used in HTTP headers and HTML META tags

Overload List

Sets current encoding in format used in HTTP headers and HTML META tags

[public bool SetEncoding\(string\);](#)

Sets encoding

[public void SetEncoding\(Encoding\);](#)

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.SetEncoding Method (String)

Sets current encoding in format used in HTTP headers and HTML META tags

```
public bool SetEncoding(  
    string sCharSet  
);
```

Parameters

sCharSet
Charset as

Return Value

True if encoding was set, false otherwise (in which case Default encoding will be used)

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#) |
[HTMLparser.SetEncoding Overload List](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.SetEncoding Method (Encoding)

Sets encoding

```
public void SetEncoding(  
    Encoding p_oEnc  
);
```

Parameters

p_oEnc
Encoding object

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#) |
[HTMLparser.SetEncoding Overload List](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

HTMLparser.SetRawHTML Method

Sets oHTML variable in a chunk to the raw HTML that was parsed for that chunk.

```
public void SetRawHTML(  
    HTMLchunk oChunk  
);
```

Parameters

oChunk

Chunk returned by ParseNext function, it must belong to the same HTMLparser that was initiated with the same HTML data that this chunk belongs to

See Also

[HTMLparser Class](#) | [Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)

Majestic12 Hierarchy

[System.Object](#)

[Majestic12.HTMLchunk](#) ---- [System.IDisposable](#)

[Majestic12.HTMLparser](#) ---- [System.IDisposable](#)

[System.ValueType](#)

[System.Enum](#) ---- [System.IComparable](#), [System.IConvertible](#),
[System.IFormattable](#)

[Majestic12.HTMLchunkType](#)

See Also

[Majestic12 Namespace](#)

[Send comments on this topic.](#)

[Copyright \(c\) Alex Chudnovsky, Majestic-12 \(UK\)](#)