

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [**+ Page - General**](#)
- [**+ Page - Head**](#)
- [**+ Page - Body**](#)
- [**+ Simple Page**](#)
- [**+ HTML Functions**](#)
- [**+ Key Validation**](#)
- [**+ Key Value**](#)
- [**+ Error**](#)
- [**+ Other**](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

[Page - General](#)

[PageBegin](#)

[PageEnd](#)

[Page - Head](#)

[Page - Body](#)

[Simple Page](#)

[HTML Functions](#)

[Key Validation](#)

[Key Value](#)

[Error](#)

[Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [HeadHTML](#)
- [HeadMetaExpires](#)
- [HeadMetaNameContent](#)
- [HeadMetaRefresh](#)
- [Page - Body](#)
- [Simple Page](#)
- [HTML Functions](#)
- [Key Validation](#)
- [Key Value](#)
- [Error](#)
- [Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [Page - Body](#)
- [BodyBreak](#)
- [BodyFile](#)
- [BodyFileFooter](#)
- [BodyFileHeader](#)
- [BodyHRule](#)
- [BodyHTML](#)
- [BodyImageLink](#)
- [BodyList](#)
- [BodyListBegin](#)
- [BodyListEnd](#)
- [BodyTag](#)
- [BodyTextB](#)
- [BodyTextBl](#)
- [BodyTextBIU](#)
- [BodyTextFont](#)
- [BodyTextI](#)
- [BodyTextIU](#)
- [BodyTextLink](#)
- [BodyTextP](#)
- [BodyTextU](#)
- [Simple Page](#)
- [HTML Functions](#)
- [Key Validation](#)
- [Key Value](#)
- [Error](#)
- [Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [Page - Body](#)
- [Simple Page](#)
- [FileToPage](#)
- [SubmitPage](#)
- [HTML Functions](#)
- [Key Validation](#)
- [Key Value](#)
- [Error](#)
- [Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [Page - Body](#)
- [Simple Page](#)
- [HTML Functions](#)
- [GetBreak](#)
- [GetCenter](#)
- [GetFileText](#)
- [GetHRule](#)
- [GetTag](#)
- [GetTextB](#)
- [GetTextBI](#)
- [GetTextBIU](#)
- [GetTextFont](#)
- [GetTextI](#)
- [GetTextIU](#)
- [GetTextLink](#)
- [GetTextP](#)
- [GetTextQuote](#)
- [GetTextU](#)
- [Key Validation](#)
- [Key Value](#)
- [Error](#)
- [Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

[Page - General](#)

[Page - Head](#)

[Page - Body](#)

[Simple Page](#)

[HTML Functions](#)

[Key Validation](#)

[HasKeyValidDoubleRange](#)

[HasKeyValidEmail](#)

[HasKeyValidIntegerRange](#)

[HasKeyValidLongRange](#)

[HasKeyValidSingleRange](#)

[HasKeyValidStringContent](#)

[HasKeyValidStringLen](#)

[HasKeyValue](#)

[Key Value](#)

[Error](#)

[Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [Page - Body](#)
- [Simple Page](#)
- [HTML Functions](#)
- [Key Validation](#)
- [Key Value](#)
- [GetKeyDouble](#)
- [GetKeyInteger](#)
- [GetKeyLong](#)
- [GetKeySingle](#)
- [GetKeyString](#)
- [Error](#)
- [Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [Page - Body](#)
- [Simple Page](#)
- [HTML Functions](#)
- [Key Validation](#)
- [Key Value](#)
- [Error](#)
- [ErrorSubText](#)
- [ErrorPage](#)
- [Other](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

VB5HTML Object Reference

HTML object from the [VB5-CGI Objects](#) collection. Copyright © 1997, 1998 [EazyWare](#).

Reference

- [Page - General](#)
- [Page - Head](#)
- [Page - Body](#)
- [Simple Page](#)
- [HTML Functions](#)
- [Key Validation](#)
- [Key Value](#)
- [Error](#)
- [Other](#)
- [DefaultPath](#)
- [InitQueryString](#)
- [CustomDecode Event](#)

[VB5HTML Example](#)

[VB5CGI Object](#)

[VB5DB Object](#)

[VB5-CGI Objects Overview](#)

PageBegin

Begins the page with a title and other optional properties.
PageBegin has to be ended with the PageEnd method.

Syntax

Sub PageBegin(ByVal Title As String, [ByVal ColorBg As String], [ByVal ColorText As String], [ByVal ColorLink As String], [ByVal ColorVLink As String], [ByVal ColorALink As String], [ByVal ImageBgFile As String], [ByVal CenterBody As Boolean])

Parameters

Title : Page title, caption of the browser window.

ColorBg : Page background color. Format can be either "#RRGGBB" or the color name, e.g. "RED", default=browser default.

ColorText : Text color. Format can be either "#RRGGBB" or the color name, e.g. "GREEN", default=browser default.

ColorLink : Not visited link color. Format can be either "#RRGGBB" or the color name, e.g. "BLUE", default=browser default.

ColorVLink : Visited link color. Format can be either "#RRGGBB" or the color name, e.g. "BLACK", default=browser default.

ColorALink : Link color while clicked. Format can be either "#RRGGBB" or the color name, e.g. "WHITE", default=browser default.

ImageBgFile : Absolute or relative filename to a .GIF or .JPG background image file.

CenterBody : Center the page (<BODY> section), default=False.

Example

```
HTML.PageBegin "Test Page", "#FF0000", "BLUE", , , , , True
```

This example begins a centered page with the title "Test Page", a red background color and blue text.

PageEnd

Ends and submits the created HTML page.
The page has to start with a PageBegin method.

Syntax

```
Sub PageEnd()
```

Example

```
HTML.PageEnd
```

HeadHTML

Adds an HTML string to the <HEAD> section.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub HeadHTML(ByVal HTMLText As String)

Parameters

HTMLText : Additional text, which will be added to the <HEAD> section.

Example

```
HTML.HeadHTML "<LINK REL=home HREF='MyPage.htm' TITLE='My Page'">
```

This example inserts an additional HTML tag to the <HEAD> section.

HeadMetaExpires

Adds an HTML <META HTTP-EQUIV="EXPIRES" [CONTENT="DateExpires"]> string to the <HEAD> section.

This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

```
Sub HeadMetaExpires([ByVal DateExpires As String = "0"])
```

Parameters

DateExpires : Date and/or time, when this page expires, and therefore should be loaded from the server and not from the proxy, default=0=load it now directly from the server.

Example

```
HTML.HeadMetaExpires "Sat, 11 Oct 1997 12:00:00 GMT"
```

The page will expire on Sat, 11 Oct 1997 at 12:00:00 GMT.

HeadMetaNameContent

Adds an HTML <META NAME="Name" CONTENT="Content"> string to the <HEAD> section.
Can be useful for search-engines.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub HeadMetaNameContent(ByVal Name As String, ByVal Content As String)

Parameters

Name : Meta name, e.g. "description", "author", "keywords."
Content : Meta content.

Example

```
HTML.HeadMetaNameContent "keywords" , "HTML, CGI, VB5"
```

This example inserts the keyword information "HTML, CGI, VB5" to the HTML document.

HeadMetaRefresh

Adds an HTML `<META HTTP-EQUIV="REFRESH" CONTENT="InSeconds[; URL=URL]">` string to the `<HEAD>` section.

This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Sub `HeadMetaRefresh`(ByVal `InSeconds` As Long, [ByVal `URL` As String])

Parameters

InSeconds : Number of seconds until the following URL will be accessed, default=empty=this page.

URL : Standard URL string.

Example

```
HTML.HeadMetaRefresh "5", "http://www.myotherpage.com"
```

In five seconds, `www.myotherpage.com` will be tried to access.

BodyBreak

Adds an HTML
 string to the <BODY> section.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

```
Sub BodyBreak()
```

Example

```
HTML.BodyBreak
```

This example creates a new line.

BodyFile

Adds an HTML File to the <BODY> section (if FileName starts with "\", the resulting filename will be DefaultPath + FileName).

This method has to be placed within the PageBegin .. PageEnd methods.

The file should only contain the needed HTML text (no <BODY>, <HEAD> or <HTML> tags).

Syntax

```
Sub BodyFile(ByVal FileName As String, [ByVal ShowNotExistMsg As Boolean = True])
```

Parameters

FileName : Absolute or relative filename to a HTML file.

ShowNotExistMsg : If True and the file does not exist, the error "[ERROR - File Not Found: filename]" will be shown on the resulting page. If False and the file does not exist, the method will not add any error text to the page. Default=True.

Example

```
HTML.BodyFile "\insert.htm"
```

This example inserts the file content of HTML.DefaultPath + "insert.htm".

BodyFileFooter

Returns/Sets a HTML footer file (if FileName starts with "\", the resulting filename will be DefaultPath + FileName).

The file should only contain the needed HTML text (no <BODY>, <HEAD> or <HTML> tags).

Syntax

Property BodyFileFooter As String

Example

```
HTML.BodyFileFooter = "\footer.htm"
```

This example adds the footer file HTML.DefaultPath + "\footer.htm".

BodyFileHeader

Returns/Sets a HTML header file (if FileName starts with "\", the resulting filename will be DefaultPath + FileName).
The file should only contain the needed HTML text (no <BODY>, <HEAD> or <HTML> tags).

Syntax

Property BodyFileHeader As String

Example

```
HTML.BodyFileHeader = "\header.htm"
```

This example adds the header file HTML.DefaultPath + "header.htm".

BodyHRule

Adds an HTML `<HR [NOSHADE] [SIZE=n] [WIDTH=n[%]] [ALIGN=lcr]>` string to the `<BODY>` section

This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Sub `BodyHRule`([ByVal `NoShade` As Boolean], [ByVal `Size` As Long], [ByVal `Width` As String], [ByVal `Align` As AlignEnum])

Parameters

NoShade : Do not shade the ruler (no 3D look), default=False.

Size : Thickness of the ruler, default=2.

Width : Absolute (e.g "200" for 200 pixels) or relative (e.g. "50%" for half the page width) width of the ruler, default=100%.

Align : Ruler alignment=left, center or right (`htmlAlignLeft`, `htmlAlignCenter`, `htmlAlignRight`), default=left aligned.

Example

```
HTML.BodyHRule True, 1, "50%", htmlAlignCenter
```

This example draws a no shaded, thin, halve page width and center aligned ruler.

BodyHTML

Adds an HTML string to the <BODY> section.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyHTML(ByVal HTMLText As String, [ByVal Center As Boolean])

Parameters

HTMLText : HTML code.

Center : Center the HTML code, default=False.

Example

```
HTML.BodyHTML "<IMG SRC=" & HTML.GetTextQuote(image.gif) &  
"ALIGN=BOTTOM>My Image<BR>"
```

This example adds an image "image.gif" with the text "My Image" to the page.

BodyImageLink

Adds an HTML `` string to the `<BODY>` section. This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Sub BodyImageLink(ByVal ImageFile As String, ByVal Link As String, [ByVal Center As Boolean])

Parameters

ImageFile : Filename of the hyper reference picture (if FileName starts with "\", the resulting filename will be DefaultPath + FileName).

Link : URL of hyper reference.

Center : Center the image, default=False.

Example

```
HTML.BodyImageLink "\myimage.gif", "http://www.mypage.com/otherdoc.htm"
```

This example adds a left aligned image "`\myimage.gif`" to the page, with a reference to the document "`http://www.mypage.com/otherdoc.htm`".

BodyList

Adds an HTML list entry to the started list.

This method has to be placed within the PageBegin .. BodyListBegin .. BodyListEnd .. PageEnd methods.

Syntax

```
Sub BodyList(ByVal Text As String, [ByVal OLNewValue As String])
```

Parameters

Text : Text, which appears in the list entry.

OLNewValue : If it is an ordered list, assign a new list value.

Example

```
HTML.BodyListBegin True  
HTML.BodyList "This is an ordered list entry with the value 1"  
HTML.BodyList "This is an ordered list entry with the value 2"  
HTML.BodyList "This is an ordered list entry with the value 6", "6"  
HTML.BodyListEnd
```

This example shows a default decimal ordered list with three entries.

BodyListBegin

Starts an ordered or unordered HTML List.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyListBegin([ByVal OrderedList As Boolean], [ByVal ListType As String], [ByVal StartValue As String], [ByVal Center As Boolean])

Parameters

OrderedList : Ordered or unordered list, default=unordered=False.

ListType : Kind of list enumerator, can be "1", "A", "a", "I" or "i", default=decimal ordered list="1".

StartValue : List start value, default "1", "A", "a", "I" or "i".

Center : Center the list, default=False.

Example

```
HTML.BodyListBegin True, "A"  
HTML.BodyList "This is an ordered list entry with the value A"  
HTML.BodyList "This is an ordered list entry with the value B"  
HTML.BodyList "This is an ordered list entry with the value E", "E"  
HTML.BodyListEnd
```

This example shows an ordered list, with three entries, beginning with "A".

BodyListEnd

Ends the list and adds an HTML `<xL>Content</xL>` string to the `<BODY>` section. This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

```
Sub BodyListEnd()
```

Example

```
HTML.BodyListBegin  
HTML.BodyList "This is an unordered list entry"  
HTML.BodyListEnd
```

This example adds an unordered list to the page.

BodyTag

Adds an HTML <Tag>Content</Tag> string to the <BODY> section.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyTag(ByVal Tag As String, ByVal Content As String, [ByVal Center As Boolean])

Parameters

Tag : Content enclosed HTML tag.

Content : HTML tag code.

Center : Center the tag code, default=False.

Example

```
HTML.BodyTag "H2", "This is a header 2"
```

This example adds a header with the size 2 to the page.

BodyTextB

Adds an HTML `Text` string to the `<BODY>` section.
This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Sub BodyTextB(ByVal Text As String, [ByVal Center As Boolean])

Parameters

Text : Text, which will be bold.

Center : Center the text, default=False.

Example

```
HTML.BodyTextB "Bold Text"
```

This example adds a bold text to the page.

BodyTextBI

Adds an HTML `<I>Text</I>` string to the `<BODY>` section.
This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Sub BodyTextBI(ByVal Text As String, [ByVal Center As Boolean])

Parameters

Text : Text, which will be bold and italic.

Center : Center the text, default=False.

Example

```
HTML.BodyTextBI "Bold and Italic Text"
```

This example adds a bold and italic text to the page.

BodyTextBIU

Returns an HTML `<I><U>Text</U></I>` string.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyTextBIU(ByVal Text As String, [ByVal Center As Boolean])

Parameters

Text : Text, which will be bold, italic and underlined.

Center : Center the text, default=False.

Example

```
HTML.BodyTextBIU "Bold, Italic and Underlined Text"
```

This example adds a bold, italic and underlined text to the page.

BodyTextFont

Adds an HTML `Text` string to the `<BODY>` section.

This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Function `BodyTextFont`(ByVal Text As String, ByVal Size As Integer, [ByVal Color As String], [ByVal Face As String], [ByVal Center As Boolean]) As String

Parameters

Text : Text with the specified font.

Size : Absolute (1..7) or relative size (-2..+4 to the default size of 3) of the font.

Color : Font color. Format can be either "#RRGGBB" or the color name, e.g. "GREEN", default=undefined.

Face : Font face, e.g. "Courier".

Center : Center the text, default=False.

Example

```
HTML.BodyTextFont "This is a red Arial font", 6, "RED", "Arial,Helvetica", True
```

This example adds a centered red Arial font text with the size of 6 to the page. If the font Arial is not available, Helvetica will be used instead.

BodyTextI

Returns an HTML `<I>Text</I>` string.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyTextI(ByVal Text As String, [ByVal Center As Boolean])

Parameters

Text : Text, which will be italic.

Center : Center the text, default=False.

Example

```
HTML.BodyTextI "Italic Text"
```

This example adds an italic text to the page.

BodyTextIU

Returns an HTML `<I><U>Text</U></I>` string.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyTextIU(ByVal Text As String, [ByVal Center As Boolean])

Parameters

Text : Text, which will be italic and underlined.

Center : Center the text, default=False.

Example

```
HTML.BodyTextIU "Italic and Underlined Text"
```

This example adds an italic and underlined text to the page.

BodyTextLink

Adds an HTML `Text` string to the `<BODY>` section. This method has to be placed within the `PageBegin .. PageEnd` methods.

Syntax

Sub BodyTextLink(ByVal Text As String, [ByVal Link As String], [ByVal Center As Boolean])

Parameters

Text : Text, which describes the hyper reference.

Link : URL of hyper reference.

Center : Center the link text, default=False.

Example

```
HTML.BodyTextLink "This is a hyper text link",  
"http://www.mysite.com/index.htm"
```

This example adds a hypertext link to the page. The descriptive link text is "This is a hyper text link", the reference is "http://www.mysite.com/index.htm".

BodyTextP

Adds an HTML <P>Text</P> string to the <BODY> section.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyTextP(ByVal Text As String, [ByVal Align As AlignEnum])

Parameters

Text : Paragraph text.

Align : Center the paragraph, default=False.

Example

```
HTML.BodyTextP "This is a new paragraph"
```

This example adds a new paragraph to the page.

BodyTextU

Adds an HTML <U>Text</U> string to the <BODY> section.
This method has to be placed within the PageBegin .. PageEnd methods.

Syntax

Sub BodyTextU(ByVal Text As String, [ByVal Center As Boolean])

Parameters

Text : Text, which will be underlined.

Center : Center the text, default=False.

Example

```
HTML.BodyTextU "Underlined Text"
```

This example adds an underlined text to the page.

FileToPage

Submits an HTML file to a page.

Syntax

Sub FileToPage(ByVal FileName As String)

Parameters

FileName : Absolute or relative filename to a HTML file.

Example

```
HTML.FileToPage "\mypage.htm"
```

This example submits the whole HTML file HTML.DefaultPath + "\mypage.htm" to the browser.

SubmitPage

Submits a simple HTML page.

Syntax

Sub SubmitPage(ByVal Title As String, ByVal Body As String, [ByVal CenterBody As Boolean = False])

Parameters

Title : Page title, caption of the browser window.

Body : Page content (everything between <PAGE> and </PAGE>).

CenterBody : Center the page (<BODY> section), default=False.

Example

```
HTML.SubmitPage "Test Page", "Hello World!", True
```

This example creates a centered page with the title "Test Page" and the centered page text "Hello World!".

GetBreak

Returns an HTML
 string

Syntax

Function GetBreak() As String

Return Value

HTML break (new line).

Example

```
Dim str As String  
str = HTML.BodyBreak()
```

The variable str contains the HTML-string "
".

GetCenter

Returns an HTML <CENTER>HTMLText</CENTER> string.

Syntax

Function GetCenter(ByVal HTMLText As String) As String

Parameters

HTMLText : Text to center.

Return Value

HTML centered text.

Example

```
Dim str As String  
str = HTML.GetCenter("This text is centered")
```

The variable str contains the HTML-string "<CENTER>This text is centered</CENTER>".

GetFileText

Returns the content of a (HTML) text file as a string.

Syntax

Function Function GetFileText(ByVal FileName As String, [ByVal ShowNotExistMsg As Boolean = True]) As String

Parameters

FileName : Absolute or relative filename to a HTML file.

ShowNotExistMsg : If True and the file does not exist, the error "[ERROR - File Not Found: filename]" will be returned. If False and the file does not exist, the method will not return any error text. Default=True.

Return Value

String, containing the text of the file or error message.

Example

```
Dim str As String  
str = HTML.GetFileText("C:\MyDir\MyFile.htm")
```

If this file exists, the variable str contains the text content from the file"C:\MyDir\MyFile.htm".

GetHRule

Returns an HTML <HR [NOSHADE] [SIZE=n] [WIDTH=n[%]] [ALIGN=lcr]> string

Syntax

Function GetHRule([ByVal NoShade As Boolean], [ByVal Size As Long], [ByVal Width As String], [ByVal Align As AlignEnum]) As String

Parameters

NoShade : Do not shade the ruler (no 3D look), default=False.

Size : Thickness of the ruler, default=2.

Width : Absolute (e.g "200" for 200 pixels) or relative (e.g. "50%" for half the page width) width of the ruler, default=100%.

Align : Ruler alignment=left, center or right (htmlAlignLeft, htmlAlignCenter, htmlAlignRight), default=left aligned.

Return Value

String containing the HTML ruler text.

Example

```
Dim str As String  
str = HTML.BodyHRule False, 1, "100", htmlAlignRight
```

The variable str contains the HTML text for a shaded, thin, 100 pixels, right aligned ruler.

GetTag

Returns an HTML <Tag>Content</Tag> string.

Syntax

Function GetTag(ByVal Tag As String, ByVal Content As String, [ByVal Center As Boolean]) As String

Parameters

Tag : Content enclosed HTML tag.

Content : HTML tag code.

Center : Center the tag code, default=False.

Return Value

String, containing the HTML tag text.

Example

```
Dim str As String  
str = HTML.GetTag("H3", "This is a header 3")
```

The variable str contains the HTML tag text for a header with the size of 3.

GetTextB

Returns an HTML `Text` string.

Syntax

Function GetTextB(ByVal Text As String, [ByVal Center As Boolean]) As String

Parameters

Text : Text, which will be bold.

Center : Center the text, default=False.

Return Value

String, containing the HTML bold text.

Example

```
Dim str As String  
str = HTML.GetTextB("Bold Text")
```

The variable str contains the bold HTML text "Bold Text".

GetTextBI

Returns an HTML `<I>Text</I>` string.

Syntax

Function GetTextBI(ByVal Text As String, [ByVal Center As Boolean]) As String

Parameters

Text : Text, which will be bold and italic.

Center : Center the text, default=False.

Return Value

String, containing the HTML bold and italic text.

Example

```
Dim str As String  
str = HTML.GetTextBI("Bold and Italic Text")
```

The variable str contains the HTML text "Bold and Italic Text" in bold and italic.

GetTextBIU

Returns an HTML `<I><U>Text</U></I>` string.

Syntax

Function GetTextBIU(ByVal Text As String, [ByVal Center As Boolean]) As String

Parameters

Text : Text, which will be bold, italic and underlined.

Center : Center the text, default=False.

Return Value

String, containing the HTML bold, italic and underlined text.

Example

```
Dim str As String  
str = HTML.GetTextBIU("Bold, Italic and Underlined Text")
```

The variable str contains the HTML text "Bold, Italic and Underlined Text" in bold, italic and underlined.

GetTextFont

Returns an HTML Text string.

Syntax

Function GetTextFont(ByVal Text As String, ByVal Size As Integer, [ByVal Color As String], [ByVal Face As String], [ByVal Center As Boolean]) As String

Parameters

Text : Text with the specified font.

Size : Absolute (1..7) or relative size (-2..+4 to the default size of 3) of the font.

Color : Font color. Format can be either "#RRGGBB" or the color name, e.g. "GREEN", default=undefined.

Face : Font face, e.g. "Courier".

Center : Center the text, default=False.

Return Value

String containing the HTML text for the specified font.

Example

```
Dim str As String
str = HTML.GetTextBIU("This is a green font", 2, "RED", "Arial,Helvetica")
```

The variable str contains the HTML text for left aligned green Arial font text with the size of 2. If the font Arial is not available, Helvetica will be used instead.

GetTextI

Returns an HTML `<i>Text</i>` string.

Syntax

Function GetTextI(ByVal Text As String, [ByVal Center As Boolean]) As String

Parameters

Text : Text, which will be italic.

Center : Center the text, default=False.

Return Value

String, containing the HTML italic text.

Example

```
Dim str As String  
str = HTML.GetTextI("Italic Text")
```

The variable str contains the italic HTML text "Italic Text".

GetTextIU

Returns an HTML `<I><U>Text</U></I>` string.

Syntax

Function GetTextIU(ByVal Text As String, [ByVal Center As Boolean]) As String

Parameters

Text : Text, which will be italic and underlined.

Center : Center the text, default=False.

Return Value

String, containing the HTML italic and underlined text.

Example

```
Dim str As String
str = HTML.GetTextIU("Italic and Underlined Text")
```

The variable str contains the HTML text "Italic and Underlined Text" in italic and underlined.

GetTextLink

Returns an HTML Text string.

Syntax

Function GetTextLink(ByVal Text As String, [ByVal Link As String], [ByVal Center As Boolean]) As String

Parameters

Text : Text, which describes the hyper reference.

Link : URL of hyper reference.

Center : Center the link text, default=False.

Return Value

String, containing the HTML hyper reference text.

Example

```
Dim str As String
str = HTML.GetTextLink("This is a hyper text link",
"http://www.mysite.com/index.htm", True)
```

The variable str contains the HTML text of a centered hypertext link. The descriptive link text is "This is a hyper text link", the reference is "http://www.mysite.com/index.htm".

GetTextP

Returns an HTML `<P [ALIGN=lcr]>Text</P>` string.

Syntax

Function GetTextP(ByVal Text As String, [ByVal Align As AlignEnum]) As String

Parameters

Text : Paragraph text.

Align : Center the paragraph, default=False.

Return Value

String, containing the HTML paragraph text.

Example

```
Dim str As String  
str = HTML.GetTextP("New centered paragraph", True)
```

The variable str contains the HTML centered paragraph text "New centered paragraph".

GetTextQuote

Returns the string with enclosed double quotes.

Syntax

Function GetTextQuote(ByVal Text As String) As String

Parameters

Text : Text, which will be enclosed with double-quotes.

Return Value

String containing text enclosed in double-quotes.

Example

```
Dim str As String  
str = HTML.GetTextQuote("Quoted Text")
```

The variable str contains the quoted text "Quoted Text".

GetTextU

Returns an HTML <U>Text</U> string.

Syntax

Function GetTextU(ByVal Text As String, [ByVal Center As Boolean]) As String

Parameters

Text : Text, which will be underlined.

Center : Center the text, default=False.

Return Value

String, containing the HTML underlined text.

Example

```
Dim str As String  
str = HTML.GetTextU("Underlined Text")
```

The variable str contains the underlined HTML text "Underlined Text".

HasKeyValidDoubleRange

Returns True, if the query string parameter value is within the limits.

Syntax

Function HasKeyValidDoubleRange(ByVal KeyName As String, ByVal Min As Double, ByVal Max As Double) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Min : Minimum valid double-type value.

Max : Maximum valid double-type value.

Return Value

True, if the parameter key and value exists and the value is within the specified limits, otherwise False.

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValidDoubleRange("DoubleRange", 1.051, 13.256)
```

The variable valid is True, if the parameter key value from the key name "DoubleRange" is between 1.051 and 13.256.

HasKeyValidEmail

Returns True, if the query string parameter value is a valid e-mail address (valid e-mail: a@b.ch).

Syntax

Function HasKeyValidEmail(ByVal KeyName As String) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Return Value

True, if the parameter key and a valid e-mail as value exists, otherwise False.
A valid e-mail must contain a "@" and a "." character and must contain at least six characters. A valid address is "a@b.us".

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValidEmail("YourEmail")
```

The variable valid is True, if the parameter key value from the key name "YourEmail" contains a valid e-mail address.

HasKeyValidIntegerRange

Returns True, if the query string parameter value is within the limits.

Syntax

Function HasKeyValidIntegerRange(ByVal KeyName As String, ByVal Min As Integer, ByVal Max As Integer) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Min : Minimum valid integer-type value.

Max : Maximum valid integer-type value.

Return Value

True, if the parameter key and value exists and the value is within the specified limits, otherwise False.

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValidIntegerRange("IntegerRange", 1, 25)
```

The variable valid is True, if the parameter key value from the key name "IntegerRange" is between 1 and 25.

HasKeyValidLongRange

Returns True, if the query string parameter value is within the limits.

Syntax

Function HasKeyValidLongRange(ByVal KeyName As String, ByVal Min As Long, ByVal Max As Long) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Min : Minimum valid long-type value.

Max : Maximum valid long-type value.

Return Value

True, if the parameter key and value exists and the value is within the specified limits, otherwise False.

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValidLongRange("LongRange", 1000, 50000)
```

The variable valid is True, if the parameter key value from the key name "LongRange" is between 1000 and 50000.

HasKeyValidSingleRange

Returns True, if the query string parameter value is within the limits.

Syntax

Function HasKeyValidSingleRange(ByVal KeyName As String, ByVal Min As Single, ByVal Max As Single) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Min : Minimum valid single-type value.

Max : Maximum valid single-type value.

Return Value

True, if the parameter key and value exists and the value is within the specified limits, otherwise False.

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValidSingleRange("SingleRange", 1.5, 3.1)
```

The variable valid is True, if the parameter key value from the key name "SingleRange" is between 1.5 and 3.1.

HasKeyValidStringContent

Returns True, if the query string parameter value only contains the selected content (enum constant)

Syntax

Function HasKeyValidStringContent(ByVal KeyName As String, ByVal Content As ContentEnum, [ByVal MinLen As Integer], [ByVal MaxLen As Integer]) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Content : One, or a combination of the following enumerated constants (ContentEnum):

- htmlDash : "-"

- htmlNumber : "0".."9"

- htmlSpace : " "

- htmlLetterUS : US upper- and lower case letters - use either htmlLetterUS or htmlLetterEU

- htmlLetterEU : htmlLetterUS + European letters (ANSI 192 .. ANSI 255) - use either htmlLetterUS or htmlLetterEU

MinLen : Minimum valid string length.

MaxLen : Maximum valid string length.

Return Value

True, if the parameter key and value exists and the value is within the specified limits, otherwise False.

Example

```
Dim valid As Boolean
valid = HTML.HasKeyValidStringContent("MyAddress", htmlLetterUS +
htmlSpace + htmlNumber, 5, 40)
```

The variable valid is True, if the parameter key value from the key name "MyAddress" contains only US-letters, spaces and numbers, and has a size between 5 and 40 characters.

HasKeyValidStringLen

Returns True, if the query string parameter value is within the limits.

Syntax

Function HasKeyValidStringLen(ByVal KeyName As String, ByVal MinLen As Integer, [ByVal MaxLen As Integer]) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

MinLen : Minimum valid string length.

MaxLen : Maximum valid string length.

Return Value

True, if the parameter key and value exists and the value is within the specified limits, otherwise False.

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValidStringLen("MyName", 2, 30)
```

The variable valid is True, if the parameter key string from the key name "MyName" contains between 2 and 30 characters.

HasKeyValue

Returns True, if a parameter in the query string has a value (leading and trailing spaces are removed).

Syntax

Function HasKeyValue(ByVal KeyName As String) As Boolean

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Return Value

True, if the parameter key and a value with at least one character exists, otherwise False.

Example

```
Dim valid As Boolean  
valid = HTML.HasKeyValue("MyParameterKey")
```

The variable valid is True, if the parameter key string from the key name "MyParameterKey" contains at least one character.

GetKeyDouble

Returns the double value of a given parameter in the query string.

Syntax

Function GetKeyDouble(ByVal KeyName As String) As Double

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Return Value

From string to double converted parameter key value.

Example

```
Dim result As Double  
result = HTML.GetKeyDouble("MyDouble")
```

The variable result contains the converted parameter key value of the parameter name "MyDouble".

GetKeyInteger

Returns the integer value of a given parameter in the query string.

Syntax

```
Function GetKeyInteger(ByVal KeyName As String) As Integer
```

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Return Value

From string to integer converted parameter key value.

Example

```
Dim result As Integer  
result = HTML.GetKeyInteger("MyInteger")
```

The variable result contains the converted parameter key value of the parameter name "MyInteger".

GetKeyLong

Returns the long value of a given parameter in the query string.

Syntax

Function GetKeyLong(ByVal KeyName As String) As Long

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Return Value

From string to long converted parameter key value.

Example

```
Dim result As Long  
result = HTML.GetKeyLong("MyLong")
```

The variable result contains the converted parameter key value of the parameter name "MyLong".

GetKeySingle

Returns the single value of a given parameter in the query string

Syntax

```
Function GetKeySingle(ByVal KeyName As String) As Single
```

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

Return Value

From string to single converted parameter key value.

Example

```
Dim result As Single  
result = HTML.GetKeySingle("MySingle")
```

The variable result contains the converted parameter key value of the parameter name "MySingle".

GetKeyString

Returns the string value of a given parameter in the query string with removed trailing and leading blanks.

Syntax

Function GetKeyString(ByVal KeyName As String, [ByVal RemoveTags As Boolean]) As String

Parameters

KeyName : Name of the parameter key name from the query string (KeyName=KeyValue), case insensitive.

RemoveTags : If True, removes all HTML tags (e.g. <H2>) from key value string. If an opening tag exists ('<') but no following closing tag can be found ('>'), all text from and with the opening tag will be removed. Default=False.

Return Value

Parameter key value string. If multiple parameters with the same name are used (e.g. multiple form checkboxes with the same name) it will return all of them, separated with a comma.

Example

Assuming you created the following HTML form:

```
<HTML>
<HEAD><TITLE>Test Form</TITLE></HEAD>

<BODY>
  <FORM ACTION="Test.exe" METHOD="GET">
    Please enter your text:
    <INPUT TYPE="Text" NAME="MyString">
    <INPUT TYPE="Submit">
  </FORM>
</BODY>
</HTML>
```

If the user presses the [Submit] button, the following compiled Test.exe script will be called:

```
Dim HTML As New VB5HTML.clsHTML

Sub Main()
  HTML.InitQueryString
  HTML.SubmitPage "Test", "The input field text with removed tags is: "
  & _
  HTML.GetKeyString("MyString", True)
End Sub
```

The resulting page, produced by this script, will show the input form field entry with removed tags.

ErrorSubText

Returns/Sets the error page sub text (e.g. e-mail address).

Syntax

Property ErrorSubText As String

Example

```
HTML.ErrorSubText = "Please, send comments to " &  
HTML.GetTextLink("MyEmail", "mailto:me@my.com")
```

If an error occurs, or submitting an error message (ErrorPage), this example adds an additional and personalized error text to the error message.

ErrorPage

Submits an error page. After that, no further page can be sent within the same script.

Syntax

Sub ErrorPage(ByVal ErrorMessage As String, [ByVal PageTitle As String], [ByVal Center As Boolean = True])

Parameters

ErrorMessage : A specified error message. If a sub text was specified (ErrorSubText), it will appear below the ErrorMessage.

PageTitle : Page title, caption of the browser window, default="CGI Application Error".

Center : Center the error message, default=True.

Example

```
HTML.ErrorPage = "Error: No Query String to evaluate"
```

This example submits the centered error message "Error: No Query String to evaluate" with the default title "CGI Application Error". After that, while the script is running, no further error page can be submitted.

DefaultPath

Returns/Sets the default HTML path and will be used, if a path starts with "\".

Syntax

Property DefaultPath As String

Example

```
HTML.DefaultPath = "C:\MyDataPath\New"
```

This example defines the path "C:\MyDataPath\New" as the default path. This path will be used as a relative path for the VB5HTML object, in case a filename parameter starts with a backslash.

InitQueryString

Sets and URL-decodes the query string and returns True, if not empty.

Syntax

Function InitQueryString([ByVal AllowOnlyPOST As Boolean], [ByVal MaxInputLen As Long = 2048], [ByVal UseCustomDecodeEvent As Boolean]) As Boolean

Parameters

AllowOnlyPOST : If True, only the POST method will be accepted, default=False (GET and POST are allowed).

Note: If AllowOnlyPOST=True and a GET method was invoked, the function result will be False and the query string will be empty.

MaxInputLen : Defines the maximum input length, of either the environment variable QUERY_STRING (GET method) or the standard input file (POST method), default=2048 characters.

UseCustomDecodeEvent : If True, instead of internal URL-decoding, an event will allow the developer to program his own decoding algorithm, default=False.

Note: The length of the (URL-encoded) input can be longer than the decoded query string. If the query string is longer than MaxInputLen, only the left most part of the query string will be available.

Return Value

True, if the (URL-encoded) query string contains any data.

Example

```
If HTML.InitQueryString(True, 100) Then
    If HTML.GetKeyString("MyPassword") = "Secret" Then
        HTML.SubmitPage "OK", "Password is correct"
    Else
        HTML.SubmitPage "Not OK", "Wrong password"
    End If
End If
```

This example assumes, that you:

1. send the password from a HTML form, containing a text input field with the name "MyPassword",
2. using the POST method and
3. the (URL-encoded) query string is no longer than 100 characters.

If the case sensitive password "Secret" was entered in the text field and submitted, a page with the title "OK" and the content "Password is correct" will appear.

CustomDecode Event

Occurs when using InitQueryString and parameter 'UseCustomDecodeEvent = true'.

Syntax

Event CustomDecode(TextToDecode As String)

Parameters

TextToDecode : URL-encoded text, which needs to be custom decoded.

Return Value

The event procedure has to return the decoded string in the 'TextToDecode' parameter.

Example

To be able to receive an event from an ActiveX-DLL, you have to create an additional class module. The following code shows the standard module, which is needed to create an .EXE file:

```
Private StartClass As New clsScript    'Create instance of the main script
class
Sub Main()
    StartClass.Start                    'Invoke the CGI script
End Sub
```

The startup procedure Main() of this standard module calls the Start method of the class module clsScript:

```
Private CGI As New VB5CGI.clsCGI        'Create instance of the
VB5CGI object
Private WithEvents HTML As VB5HTML.clsHTML    'Enable events for custom
decoding

'Start routine, which was invoked from the Sub Main
Public Sub Start()
    Set HTML = New VB5HTML.clsHTML        'Create instance of the
VB5HTML object
    With HTML                            'We use the VB5HTML object
        'Here comes your main script
    End With
End Sub

'This is the custom decode event procedure
Private Sub HTML_CustomDecode(TextToDecode As String)
    'Here comes your custom decode algorith
End Sub
```


VB5HTML Example

The following Visual Basic source code (demoHTML.bas) shows a simple CGI script, which mainly uses the VB5HTML object. This example was tested with the Microsoft Personal Web Server version 1.0a and 2.0 and the IIS3.

If not submitting a query string (e.g. <http://localhost/scripts/demoHTML.exe>), or using the GET method, an error page will be shown.

Otherwise, the script validates the form field values "FirstName", "LastName" and "Age". Then it creates a page with a header and footer file. This page contains an alphabetically ordered list with different values and a hit counter. The page title shows the script name including path.

```

Private CGI As New VB5CGI.clsCGI      'Instance the VB5CGI Object (needs
VB5CGI.DLL)
Private HTML As New VB5HTML.clsHTML  'Instance the VB5HTML Object
(needs VB5HTML.DLL)

Sub Main()
    With HTML
        'Add an additional text (e.g. your e-mail address) to an occurring
error message
        .ErrorSubText = "Please, send your comments to "
& .GetTextLink("EasyWare", "mailto:EazyWare@yahoo.com")
        .BodyFileHeader = "\demoHEAD.htm"          'Use a
HTML header file
        .BodyFileFooter = "\demoFOOT.htm"         'Use a
HTML footer file
        'Retrieve and check the query string (allow only POST-method and
max 1024 not encoded characters)
        If .InitQueryString(True, 1024) Then
            'Validate the FirstName entry. Allow only US letters, Spaces,
min 3 to max 25 characters
            If Not .HasKeyValidStringContent("FirstName", htmlLetterUS +
htmlSpace, 3, 25) Then
                'If the entry was not valid, submit a simple information
page
                .SubmitPage "Validation", "Please enter your first name
(US letters, Spaces, min 3 to max 25 characters)", True
                End                                'And
end the script
            End If
            If Not .HasKeyValidStringLength("LastName", 2, 25) Then
                'Validate the LastName entry
                .SubmitPage "Validation", "Please enter your last name (2
to 25 characters)", True
                End
            End If
            If Not .HasKeyValidIntegerRange("Age", 1, 150) Then
                'Validate the Age entry
                .SubmitPage "Validation", "Please enter your age (1 to 150
years)", True
                End
            End If
            'Start the sample page with a title and background color
            .PageBegin "VB5-CGI Objects demonstration [" &
CGI.GetScriptName(True) & "]", "#FFFFCC"
            .HeadMetaNameContent "Author", "James Bond"          'Add a
meta tag
            .BodyTag "H2", "Dear " & .GetKeyString("FirstName") & " " & _
                .GetKeyString("LastName") & ", welcome to the
VB5HTML demo!"
            .BodyTextBI "Some demo information:"
            .BodyListBegin True, "A"                              'Begin
a new ordered list
            .BodyList "First Name: " & .GetKeyString("FirstName")  'List
the FirstName
            .BodyList "Last Name: " & .GetKeyString("LastName")    'List
the LastName
            .BodyList "Age: " & .GetKeyInteger("Age")              'List

```

```

the Age (as integer)
    .BodyList "Default Path: " & .DefaultPath()           'List
the default path
    .BodyList "Script Filename: " & CGI.GetScriptName()   'List
the script name
    .BodyList "Browser: " & CGI.EnvHTTPUserAgent         'List
the browser type (from environment variable)
    .BodyListEnd                                         'End
the list
    .BodyHRule True, 2, "20%", htmlAlignCenter
    'Show and increment the hit counter, but only for this page
    .BodyTextFont "This page has been accessed " &
CGI.GetHitCounterInc(App.EXENAME) & _
    " times!", 1, "#0000FF", "Arial, Helvetica",
True
    .BodyBreak
    .PageEnd                                             'End
and submit the complete page
    Else
    'Occurs, if other than a POST-method was used, or no query
string was submitted
    .ErrorPage "No POST method used, or no Query String received!"
    End If
    End With
End Sub

```

The following HTML text was used (demoHTML.htm) to invoke the script:

```

<HTML>
<HEAD>
  <TITLE>VB5-CGI Objects demo: VB5HTML object</TITLE>
</HEAD>

<BODY BGCOLOR="#FFFFFF">
<H2>VB5-CGI Objects demo: VB5HTML object</H2>
<B>CGI script: demoHTML.exe</B>
<BR>Remark: Please copy the demoHEAD.htm and demoFOOT.htm to your Web
server root directory.
<HR>
Please, enter your personal information:<P>
<FORM METHOD="POST" ACTION="http://localhost/scripts/demoHTML.exe">
  First Name: <INPUT TYPE="text" NAME="FirstName" SIZE=40 MAXLENGTH=60><P>
  Last Name: <INPUT TYPE="text" NAME="LastName" SIZE=40 MAXLENGTH=60><P>
  Age: <INPUT TYPE="text" NAME="Age" SIZE=5 MAXLENGTH=10><P>
  <INPUT TYPE="reset">
  <INPUT TYPE="submit" VALUE="send with POST method">
</FORM>
</BODY>
</HTML>

```

ActiveX DLL

An in-process Dynamic Link Library, which runs in the same process as the client. It provides the fastest way of accessing objects, because property and method calls don't have to be marshaled across process boundaries.

DLL

A Dynamic Link Library is a program module, with functionality for one or multiple programs. When loaded in the memory it can be used simultaneously by multiple threads.

EazyWare

Internet and software solutions, Switzerland

VB5-CGI Objects:

Author: Stephan Schmid

Web: <http://www.eazyware.com/vb5-cgi>

E-mail: tools@eazyware.com

relative filename

A filename parameter can be either absolute or relative.

Starting the filename with a back slash (\) indicates a relative file path. In this case, the base path to this relative file will be retrieved from the DefaultPath property, which is by default the servers document root path.

VB5-CGI Objects

A collection of powerful and easy to use Visual Basic 5 objects to develop CGI scripts on any CGI capable web server under Windows 95/NT4.0.

VB5CGI

An object from the VB5-CGI Objects collection, which handles most of the low-level CGI functionality like standard file input/output.

VB5DB

An object from the VB5-CGI Objects collection, which handles the most used database functionality.

VB5HTML

An object from the VB5-CGI Objects collection, which handles most of the high-level HTML functionality like validating query string parameter values and submitting pages.

