Using the HTTP Control

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscHTTPOverviewC"}

The Hyper Text Transfer Protocol is the protocol commonly used to send and receive HTML documents. As indicated by its name, the protocol is intended to be used for the transmission of many types of data which are used in Hyper Text documents

Possible Uses

The **HTTP** control can be used in the following scenarios:

- To create a text only Web browser.
- To create an application that regularly searches HTML documents for keywords of interest, and later retrieval.
- To create an application that downloads HTML text for storage in a database.

Scenario: Sending a Request for a Document to a Server

This sample session demonstrates how to use the HTTP ActiveX ™ Control to retrieve a HTTP document, parse it and process the contents of the document. The steps involved in the process are as follows:

1. Use the **PerformRequest** method to send a request
2. Use the DocOutput event to route the data.

Setup

The Visual Basic code below uses the following objects:

- **HTTP** control named "HTTP1."
- **CommandButton** "cmdPerformRequest."
- **TextBox** control named "txtLog."

Use the PerformRequest Method to Send a Request

To request a document from an HTTP server, you must use the **PerformRequest** method. Before invoking the method, however, you must also set the **Method** property, which determines the action to be requested. To retrieve a document, the method must be set to 1 (**prcGet**) to get a document.

The **PeformRequest** method requires two arguments, *hostName* and *document*. By default, the control uses the **RemoteHost** property if the *hostName* is not specified, and the **Document** property if *document* isn't specified. The code below sets the **Method**, **Document** and **RemoteHost** properties prior to invoking the **PeformRequest** method.

```
Private Sub cmdPerformRequest_Click()
   HTTP1.Method = prcGet   ' Use the Get method.
   HTTP1.Document = "c:\pickup.txt"
   HTTP1.RemoteHost = "companyServer.com"
   HTTP1.PerformRequest
End Sub
```

Use the DocOutput Event to Route the Data

In response to the **PerformRequest** method, the DocOutput event occurs. You can then use the reference to the **DocOutput** object to parse the data, and route it to the right destination. The code below first tests to see if the returned data is of the type "data". If so, the code then tests to see if the string "text/" is found within the data. The string "text/" is a defined MIME(Multipurpose Internet Mail Extensions) header that signifies the data is text. If the data is of another type, the code returns the size of the data in bytes.

```
Private Sub HTTP1_DocOutput(ByVal DocOutputAs docOutput)

If DocOutput.State = icDocData Then _
   contentType = DocOutput.DocHeaders. _
   Item("content-type").Value

   Dim vtData As Variant
   If InStr$(1, contentType, "text/", 1) then
      ' Retrieve text data.
      docOutput.GetData vtData, vbString
      txtOutput.Text = txtOutput.Text & vtData
      txtLog = txtLog & _
      "received html data of length = " & _
      str$(Len(data)) & vbCrLf
   Else
      ' GIF, WAV or other non-html data.
      DocOutput.GetData(data, vbArray + vbByte)
      txtLog = txtLog & _
      "received binary data of length = " & _
      Str$(UBound(data)) & vbCrLf
   End if
End If
End Sub
```

Scenario 2: Use the GetDoc Method to Route Data to a File

The following code shows how to request a document using the **GetDoc** method, and writing the data directly to a file. The process has these steps:

1. Set the **DocOutput** object's **FileName** property.
2. Invoke the **GetDoc** method supplying a valid URL.
3. Process any errors that may occur using the Error event.

Set the DocOutput Object's FileName Property

To automatically stream data into a file, you must first set the name of the new file. When this is done, there is no need to further process the DocOutput event. The code below simply sets the file name:

```
   HTTP1.DocOutput.FileName = "file.tmp"
```

Consequently, you must invoke the **GetDoc** method.

```
Private Sub GetDoc_Click()
   HTTP1.GetDoc "http://www.microsoft.com/"
End Sub
```

Process Errors Using the Error Event

It is not uncommon for errors to occur during the transfer of data from an external server. The following code can be used to process any errors that might occur.

```
Private Sub Httpct1_Error(Number As Integer, Description As String, Scode
As Long, Source As String, HelpFile As String, HelpContext As Long,
CancelDisplay As Boolean)
   CancelDisplay = True
   errLog = errLog & "Error " & Str$(Number) & ": " _
   & Description & vbCrLf
End Sub
```

HTTP Client Control

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbobjHTTPClientControlC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbobjHTTPClientControlX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbobjHTTPClientControlP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbobjHTTPClientControlM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbobjHTTPClientControlE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbobjHTTPClientControlS"}

The **HTTP** Client control implements the HyperText Transfer Protocol (HTTP), based on the HTTP specification.

Remarks

The **HTTP** control lets you directly retrieve HTTP documents if no browsing or image processing is necessary. Use the HTTP control to create HTML browsers or other applications that need access to HTTP.

The **HTTP** control uses a number of methods to retrieve or send (post) a document. It can retrieve MIME (Multipurpose Internet Mail Extensions) information about the document from the **DocHeaders** collection property.

Document Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDocumentPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDocumentPropertyX"} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDocumentPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDocumentPropertyS"}

Returns or sets the target document. The **Document** property can be used with **RemoteHost** to identify the URL. It can also be used instead of URL. Read/Write and available at run time and design time.

Syntax

Development Tool	Syntax
Microsoft Visual Basic and Microsoft Access	*object*.**Document** [= *String*]
Microsoft Visual FoxPro	[*Form*.]*Object*.Document[= *cDocumentName*]
Microsoft Visual C++	**CString GetDocument**( ); **void SetDocument**(**LPCTSTR** *lpszNewValue*);

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Data Type

String.

PerformRequest Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthPerformRequestMethodC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthPerformRequestMethodX"} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthPerformRequestMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthPerformRequestMethodS"}

Initiates a request method to retrieve a document. If no parameters are specified, **Document**, **HostName**, **RemotePort**, and **Method** properties are used for the retrieval. This method is similar to **GetDoc**, except that it uses a different set of arguments.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Visual Basic and Microsoft Access	*object*.**PerformRequest** [*HostName*,] [*Document*,] [*Method*,] [*RemotePort*]
Microsoft Visual FoxPro	*Object*.PerformRequest([*cHostName*] [, *cDocument*] [, *nMethod*] [, *nRemotePort*]

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Arguments

Microsoft Access and Visual Basic

Argument	Default	Data Type	Description
HostName	None	STRING	Optional. Host name to use for the request. This value overrides the **RemoteHost** property.
Document	None	STRING	Optional. Document to be retrieved. This value overrides the **Document** property.
Method	None	Integer	Optional. Method to be used for the retrieval. This value overrides the **Method** property.
RemotePort	None	Long	Optional. Remote port number to use when connecting. This value overrides the **RemotePort** property.

Visual FoxPro

Argument	Default	Data Type	Description
cHostName	None	Character expression	Optional. Host name to use for the request. This value overrides the **RemoteHost** property.
cDocument	None	Character expression	Optional. Document to be retrieved. This value overrides the **Document**

			property.
nMethod	None	Numeric	Optional. Method to be used for the retrieval. This value overrides the **Method** property.
nRemotePort	None	Numeric	Optional. Remote port number to use when connecting. This value overrides the **RemotePort** property.

Remarks

The DocOutput event occurs when the data is available.