Learn Discover Product documentation Development languages Topics Sign in

Filter by title

… / IServerXMLHTTPRequest/ServerXMLHTTP Examples /

Using ServerXMLHTTP Directly

Article • 10/27/2016

Using the `ServerXMLHTTP` object directly offers much greater procedural control than that of the `setProperty` method of `DOMDocument`. Instead of merely retrieving XML responses from a remote sever, the `ServerXMLHTTP` object allows developers to use the HTTP methods, `GET` and `POST`, as well as the ability to handle basic security logons.

Using ServerXMLHTTP to GET XML results from an ASP page

In the following example, `ServerXMLHTTP` retrieves an XML response from an ASP page over HTTP. By using `GET`, the example is able to send a request without actually transferring any data to the Web server.

The `responseXML` property of the `objSrvHTTP` object contains the XML response from the Web server. The example program writes this response to the browser's output by first informing the browser that the response to be received is XML (`"text/xml"`). Next, the program passes the response directly to the `Response` object for displaying onscreen.

Example

```
<%@language=JScript%>
<%
   var objSrvHTTP;
   objSrvHTTP = Server.CreateObject ("Msxml2.ServerXMLHTTP.6.0");
   objSrvHTTP.open ("GET","http://someotherserver/respond.asp", false);
   objSrvHTTP.send ();
   Response.ContentType = "text/xml";
```

App-V COM API reference
Indexing Service
Dfsncimprov Provider

```
   Response.Write (objSrvHTTP.responseXML.xml);
%>
```

Using ServerXMLHTTP to POST XML to an ASP page

In the following example, `ServerXMLHTTP` uses HTTP to send XML data to an Active Server Page (ASP) page. Unlike the preceding example that used `GET`, this example uses `POST`. The `POST` method sends data, along with the request, to the Web server.

As in the previous example, the program writes the response to the browser's output by setting the `ContentType` to "text/xml" and passing the response to the `Response` object.

Example

Copy

```
<%@language=JScript%>
<%
   var objSrvHTTP;
   var objXMLDocument;
   objSrvHTTP = Server.CreateObject ("Msxml2.ServerXMLHTTP.6.0");
   objXMLDocument = Server.CreateObject ("Msxml2.DOMDocument.6.0");

   objXMLDocument.async= false;
   objXMLDocument.loadXML ("<msg><id>1</id></msg>");

   objSrvHTTP.open ("POST","http://someotherserver/respond.asp",false);
   objSrvHTTP.send (objXMLDocument);
   Response.ContentType = "text/xml";
   Response.Write (objSrvHTTP.responseXML.xml);
%>
```

Using ServerXMLHTTP to POST and Process XML

In the following example, `ServerXMLHTTP` sends an XML document to an ASP page on a Web server and passes its XML response to a waiting `DOMDocument` for processing.

Example

Copy

```
<%@language=Jscript%><%
   var objSrvHTTP;
   var objXMLSend;
   var objXMLReceive;
   objSrvHTTP = Server.CreateObject("Msxml2.ServerXMLHTTP.6.0");
   objXMLSend = Server.CreateObject("Msxml2.DOMDocument.6.0");
   objXMLReceive = Server.CreateObject("Msxml2.DOMDocument.6.0");

   objXMLSend.async = false;
   objXMLSend.loadXML ("<msg><id>2</id></msg>");

   objSrvHTTP.open ("POST","http://someotherserver/respond.asp",false);
   objSrvHTTP.send (objXMLSend);
   objXMLReceive = objSrvHTTP.responseXML;
   Response.ContentType = "text/xml";
   Response.Write (objXMLReceive.xml);
%>
```

English (United States)   Your Privacy Choices   Theme

Manage cookies   Previous Versions   Blog   Contribute   Privacy   Terms of Use   Trademarks