

XMLHttpRequest

提供客户端同http服务器通讯的协议

Example

下面的代码是在JScript中创建一个XMLHTTP对象并从服务器请求一个XML文档。服务器返回XML文档并显示。

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
var xmlHttpReq = new ActiveXObject("MSXML2.XMLHTTP.3.0");
xmlHttpReq.open("GET", "http://localhost/books.xml", false);
xmlHttpReq.send();
alert(xmlHttpReq.responseText);
```

在非IE的浏览器中，需要用new XMLHttpRequest()来创建对象，如下：

```
var xmlHttpReq = new XMLHttpRequest();
xmlHttpReq.open("GET", "http://localhost/books.xml", false);
xmlHttpReq.send();
alert(xmlHttpReq.responseText);
```

vbscript:

```
Dim HttpReq As New MSXML2.XMLHTTP30
HttpReq.open "GET", "http://localhost/books.xml", False
HttpReq.send
MsgBox HttpReq.responseText
```

备注

客户端可以通过XmlHttp对象(MSXML2.XMLHTTP.3.0)向http服务器发送请求并使用微软XML文档对象模型Microsoft® XML Document Object Model (DOM)处理回应。

参考

[XMLHttpRequest成员](#)

Copyright © 2005 xmlhttp.cn. All rights reserved.

禁止任何形式的转载!

XMLHttpRequest成员

属性

onreadystatechange*	指定当readyState属性改变时的事件处理句柄。只写
readyState	返回当前请求的状态，只读。
responseBody	将响应信息正文以unsigned byte数组形式返回。只读
responseStream	以Ado Stream对象的形式返回响应信息。只读
responseText	将响应信息作为字符串返回。只读
responseXML	将响应信息格式化为Xml Document对象并返回，只读
status	返回当前请求的http状态码。只读
statusText	返回当前请求的响应行状态，只读

XMLHTTP对象参考

- XMLHttpRequest 对象
- XMLHttpRequest 成员
- onreadystatechange
- readyState
- responseBody
- responseStream
- responseText
- responseXML
- status
- statusText
- abort
- getAllResponseHeaders
- getResponseHeader
- open
- send
- setRequestHeader

* 表示此属性是W3C文档对象模型的扩展。

方法

abort	取消当前请求
getAllResponseHeaders	获取响应的所有http头
getResponseHeader	从响应信息中获取指定的http头
open	创建一个新的http请求，并指定此请求的方法、URL以及验证信息(用户名/密码)
send	发送请求到http服务器并接收回应
setRequestHeader	单独指定请求的某个http头

事件

无

参考

XMLHttpRequest

onreadystatechange

指定当readyState属性改变时的事件处理句柄

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
oXMLHttpRequest.onreadystatechange = funcMyHandler;
```

Example

如下的例子演示当XMLHttpRequest对象的readyState属性改变时调用HandleStateChange函数，当数据接收完毕后（readyState == 4）此页面上的一个按钮将被激活

```
var xmlhttp=null;
function PostOrder(xmlDoc)
{
    var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.5.0");
    xmlhttp.Open("POST", "http://myserver/orders/processorder.asp", false);
    xmlhttp.onreadystatechange= HandleStateChange;
    xmlhttp.Send(xmlDoc);
    myButton.disabled = true;
}
function HandleStateChange()
{
    if (xmlhttp.readyState == 4)
    {
        myButton.disabled = false;
        alert("Result = " + xmlhttp.responseXML.xml);
    }
}
```

备注

此属性只写，为W3C文档对象模型的扩展。

参考

[readyState](#) 属性

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载!

readyState

返回XMLHTTP请求的当前状态

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
lValue = oXMLHttpRequest.readyState;
```

Example

```
var XmlHttp;  
XmlHttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
  
function send() {  
    XmlHttp.onreadystatechange = doHttpRequestReadyStateChange;  
    XmlHttp.open("GET", "http://localhost/sample.xml", true);  
    XmlHttp.send();  
}  
  
function doHttpRequestReadyStateChange() {  
    if (XmlHttp.readyState == 4) {  
        alert("Done");  
    }  
}
```

备注

变量，此属性只读，状态用长度为4的整型表示.定义如下：

0 (未初始化)	对象已建立，但是尚未初始化（尚未调用open方法）
1 (初始化)	对象已建立，尚未调用send方法
2 (发送数)	send方法已调用，但是当前的状态及http头未知

据)	
3 (数据传送中)	已接收部分数据，因为响应及http头不全，这时通过responseBody和responseText获取部分数据会出现错误，
4 (完成)	数据接收完毕,此时可以通过通过responseBody和responseText获取完整的回应数据

参考

[open](#) 方法

[responseBody](#) 属性

[responseText](#) 属性

[send](#) 方法

[status](#) 属性

[statusText](#) 属性

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载！

responseBody

返回某一格式的服务器响应数据

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
strValue = oXMLHttpRequest.responseBody;
```

Example

```
var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/books.xml", false);  
xmlhttp.send();  
alert(xmlhttp.responseBody);
```

备注

变量，此属性只读，以**unsigned array**格式表示直接从服务器返回的未经解码的二进制数据。

参考

[responseStream](#) 属性

[responseText](#) 属性

[responseXML](#) 属性

responseStream

以Ado Stream对象的形式返回响应信息

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
strValue = oXMLHttpRequest.responseStream;
```

备注

变量，此属性只读，以Ado Stream对象的形式返回响应信息。

参考

[responseBody](#) 属性

[responseText](#) 属性

[responseXML](#) 属性

responseText

将响应信息作为字符串返回

语法



responseStream

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
strValue = oXMLHttpRequest.responseText;
```

Example

```
var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/books.xml", false);  
xmlhttp.send();  
alert(xmlhttp.responseText);
```

备注

变量，此属性只读，将响应信息作为字符串返回。

XMLHTTP尝试将响应信息解码为Unicode字符串，XMLHTTP默认将响应数据的编码定为UTF-8，如果服务器返回的数据带BOM(byte-order mark)，XMLHTTP可以解码任何UCS-2 (big or little endian)或者UCS-4 数据。注意，如果服务器返回的是xml文档，此属性并不处理xml文档中的编码声明。你需要使用[responseXML](#)来处理。

参考

[responseBody](#) 属性

[responseText](#) 属性

[responseXML](#) 属性

responseXML

将响应信息格式化为Xml Document对象并返回

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
var objDispatch = oXMLHttpRequest.responseXML;
```

Example

```
var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/books.xml", false);  
xmlhttp.send();  
alert(xmlhttp.responseXML.xml);
```

备注

变量，此属性只读，将响应信息格式化为Xml Document对象并返回。如果响应数据不是有效的XML文档，此属性本身不返回XMLDOMParseError，可以通过处理过的DOMDocument对象获取错误信息。

参考

[responseBody 属性](#)

[responseStream 属性](#)

[responseText 属性](#)

status

返回当前请求的http状态码

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
lValue = oXMLHttpRequest.status;
```

Example

```
var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/books.xml", false);  
xmlhttp.send();  
alert(xmlhttp.status);
```

返回值

长整形标准http状态码，定义如下：

Number	Description
100	Continue
101	Switching protocols
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently

302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request-URI Too Long
415	Unsupported Media Type
416	Requested Range Not Suitable
417	Expectation Failed
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version Not Supported

备注

长整形，此属性只读，返回当前请求的http状态码,此属性仅当数据发送并接收完毕后才可获取。

参考

[statusText 属性](#)

[send 方法](#)

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载！

statusText

返回当前请求的响应行状态

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
strValue = oXMLHttpRequest.statusText;
```

Example

```
var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/books.xml", false);  
xmlhttp.send();  
alert(xmlhttp.statusText);
```

备注

字符串，此属性只读，以BSTR返回当前请求的响应行状态,此属性仅当数据发送并接收完毕后才可获取。

参考

[status](#) 属性

[send](#) 方法

abort

取消当前请求

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
oXMLHttpRequest.abort();
```

备注

调用此方法后，当前请求返回UNINITIALIZED 状态。

参考

[readyState](#) 属性

[open](#) 方法

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载！

getAllResponseHeaders

获取响应的所有http头

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
strValue = oXMLHttpRequest.getAllResponseHeaders();
```

Example

```
var xmlhttp = new ActiveXObject("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/sample.xml", false);  
xmlhttp.send();  
alert(xmlhttp.getAllResponseHeaders());
```

输出由web服务器返回的http头信息， example:

```
Server:Microsoft-IIS/5.1  
X-Powered-By:ASP.NET  
Date:Sat, 07 Jun 2003 23:23:06 GMT  
Content-Type:text/xml  
Accept-Ranges:bytes  
Last Modified:Sat, 06 Jun 2003 17:19:04 GMT  
ETag:"a0e2eeba4f2cc31:97f"  
Content-Length:9
```

备注

每个http头名称和值用冒号分割，并以\r\n结束。当send方法完成后才可调用该方法。

参考

[send 方法](#)

[getResponseHeader](#) 方法
[setRequestHeader](#) 方法

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载!

getResponseHeader

从响应信息中获取指定的http头

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
strValue = oXMLHttpRequest.getResponseHeader(bstrHeader);
```

Example

```
var xmlhttp = new ActiveXObject("MSXML2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/sample.xml", false);  
xmlhttp.send();  
alert(xmlhttp.getResponseHeader("Server"));
```

输出http头中的server列：当前web服务器的版本及名称。

备注

当send方法成功后才可调用该方法。如果服务器返回的文档类型为"text/xml", 则这句话xmlhttp.getResponseHeader("Content-Type");将返回字符串"text/xml"。可以使用getAllResponseHeaders方法获取完整的http头信息。

参考

[send 方法](#)

[getAllResponseHeaders 方法](#)

[setRequestHeader 方法](#)

open

创建一个新的http请求，并指定此请求的方法、URL以及验证信息

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
oXMLHttpRequest.open(bstrMethod, bstrUrl, varAsync, bstrUser, bstrPassword).
```

参数

bstrMethod

http方法，例如：POST、GET、PUT及PROPFIND。大小写不敏感。

bstrUrl

请求的URL地址，可以为绝对地址也可以为相对地址。

varAsync[可选]

布尔型，指定此请求是否为异步方式，默认为true。如果为真，当状态改变时会调用onreadystatechange属性指定的回调函数。

bstrUser[可选]

如果服务器需要验证，此处指定用户名，如果未指定，当服务器需要验证时，会弹出验证窗口。

bstrPassword[可选]

验证信息中的密码部分，如果用户名为空，则此值将被忽略。

Example

下面的例子演示从服务器请求book.xml,并显示其中的book字段。

```
var xmlhttp = new XMLHttpRequest("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/books.xml", false);  
xmlhttp.send();
```

```
var book = xmlhttp.responseXML.selectSingleNode("//book[@id='bk101']");  
alert(book.xml);
```

备注

调用此方法后，可以调用send方法向服务器发送数据。

参考

[abort](#) 方法

[onreadystatechange](#) 属性

[XMLHttpRequest](#) 对象

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载！

send

发送请求到http服务器并接收回应

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
oXMLHttpRequest.send( varBody );
```

参数

varBody

欲通过此请求发送的数据。

Example

```
xmlhttp = new XMLHttpRequest("Msxml2.XMLHTTP.3.0");  
xmlhttp.open("GET", "http://localhost/sample.xml", false);  
xmlhttp.send();  
alert(xmlhttp.responseXML.xml);
```

备注

此方法的同步或异步方式取决于open方法中的bAsync参数，如果bAsync == False，此方法将会等待请求完成或者超时时才会返回，如果bAsync == True，此方法将立即返回。

This method takes one optional parameter, which is the requestBody to use. The acceptable VARIANT input types are BSTR, SAFEARRAY of UI1 (unsigned bytes), IDispatch to an XML Document Object Model (DOM) object, and IStream *. You can use only chunked encoding (for sending) when sending IStream * input types. The component automatically sets the Content-Length header for all but IStream * input types.

如果发送的数据为BSTR，则回应被编码为utf-8，必须在适当位置设置一个包含charset的文档类型头。

If the input type is a SAFEARRAY of UI1, the response is sent as is without additional encoding. The

caller must set a Content-Type header with the appropriate content type.

如果发送的数据为XML DOM object，则回应将被编码为在xml文档中声明的编码，如果在xml文档中没有声明编码，则使用默认的UTF-8。

If the input type is an IStream *, the response is sent as is without additional encoding. The caller must set a Content-Type header with the appropriate content type.

参考

[open](#) 方法

[XMLHttpRequest](#) 对象

Copyright © 2005 [xmlhttp.cn](#). All rights reserved.

禁止任何形式的转载！

setRequestHeader

单独指定请求的某个http头

语法

XMLHTTP对象参考
XMLHttpRequest 对象
XMLHttpRequest 成员
onreadystatechange
readyState
responseBody
responseStream
responseText
responseXML
status
statusText
abort
getAllResponseHeaders
getResponseHeader
open
send
setRequestHeader

```
oXMLHttpRequest.setRequestHeader(bstrHeader, bstrValue);
```

参数

bstrHeader

字符串，头名称。

bstrValue

字符串，值。

备注

如果已经存在已此名称命名的http头，则覆盖之。此方法必须在open方法后调用。

参考

[getAllResponseHeaders](#) 方法

[XMLHttpRequest](#) 对象