Ajax即异步的JavaScript和XMl，异步即延迟加载

1. **XMLHttpRequest 是 AJAX 的基础。**

**XMLHttpRequest 对象**

所有现代浏览器均支持 XMLHttpRequest 对象（IE5 和 IE6 使用 ActiveXObject）。

XMLHttpRequest 用于在后台与服务器交换数据。这意味着可以在不重新加载整个网页的情况下，对网页的某部分进行更新。

**创建 XMLHttpRequest 对象**

所有现代浏览器（IE7+、Firefox、Chrome、Safari 以及 Opera）均内建 XMLHttpRequest 对象。

**创建 XMLHttpRequest 对象的语法：**

variable=new XMLHttpRequest();

**老版本的 Internet Explorer （IE5 和 IE6）使用 ActiveX 对象：**

variable=new ActiveXObject("Microsoft.XMLHTTP");

为了应对所有的现代浏览器，包括 IE5 和 IE6，请检查浏览器是否支持 XMLHttpRequest 对象。如果支持，则创建 XMLHttpRequest 对象。如果不支持，则创建 ActiveXObject ：

var xmlhttp;

if (window.XMLHttpRequest)

{// code for IE7+, Firefox, Chrome, Opera, Safari

xmlhttp=new XMLHttpRequest();

}

else

{// code for IE6, IE5

xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

}

2. **XMLHttpRequest 对象用于和服务器交换数据。**

**向服务器发送请求**

如需将请求发送到服务器，我们使用 XMLHttpRequest 对象的 open() 和 send() 方法：

xmlhttp.open("GET","test1.txt",true);

xmlhttp.send();

|  |  |
| --- | --- |
| **方法** | **描述** |
| open(*method*,*url*,*async*) | 规定请求的类型、URL 以及是否异步处理请求。   * *method*：请求的类型；GET 或 POST * *url*：文件在服务器上的位置 * *async*：true（异步）或 false（同步） |
| send(*string*) | 将请求发送到服务器。   * *string*：仅用于 POST 请求 |

**GET 还是 POST？**

与 POST 相比，GET 更简单也更快，并且在大部分情况下都能用。

然而，在以下情况中，请使用 POST 请求：

* 无法使用缓存文件（更新服务器上的文件或数据库）
* 向服务器发送大量数据（POST 没有数据量限制）
* 发送包含未知字符的用户输入时，POST 比 GET 更稳定也更可靠

**GET 请求**

一个简单的 GET 请求：

xmlhttp.open("GET","demo\_get.asp",true);

xmlhttp.send();

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_get)

在上面的例子中，您可能得到的是缓存的结果。

为了避免这种情况，请向 URL 添加一个唯一的 ID：

xmlhttp.open("GET","demo\_get.asp?t=" + Math.random(),true);

xmlhttp.send();

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_get_unique)

如果您希望通过 GET 方法发送信息，请向 URL 添加信息：

xmlhttp.open("GET","demo\_get2.asp?fname=Bill&lname=Gates",true);

xmlhttp.send();

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_get2)

**POST 请求**

一个简单 POST 请求：

xmlhttp.open("POST","demo\_post.asp",true);

xmlhttp.send();

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_post)

如果需要像 HTML 表单那样 POST 数据，请使用 setRequestHeader() 来添加 HTTP 头。然后在 send() 方法中规定您希望发送的数据：

xmlhttp.open("POST","ajax\_test.asp",true);

xmlhttp.setRequestHeader("Content-type","application/x-www-form-urlencoded");

xmlhttp.send("fname=Bill&lname=Gates");

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_post2)

|  |  |
| --- | --- |
| **方法** | **描述** |
| setRequestHeader(*header*,*value*) | 向请求添加 HTTP 头。   * *header*: 规定头的名称 * *value*: 规定头的值 |

**url - 服务器上的文件**

open() 方法的 *url* 参数是服务器上文件的地址：

xmlhttp.open("GET","ajax\_test.asp",true);

该文件可以是任何类型的文件，比如 .txt 和 .xml，或者服务器脚本文件，比如 .asp 和 .php （在传回响应之前，能够在服务器上执行任务）。

**异步 - True 或 False？**

AJAX 指的是异步 JavaScript 和 XML（Asynchronous JavaScript and XML）。

XMLHttpRequest 对象如果要用于 AJAX 的话，其 open() 方法的 async 参数必须设置为 true：

xmlhttp.open("GET","ajax\_test.asp",true);

对于 web 开发人员来说，发送异步请求是一个巨大的进步。很多在服务器执行的任务都相当费时。AJAX 出现之前，这可能会引起应用程序挂起或停止。

通过 AJAX，JavaScript 无需等待服务器的响应，而是：

* 在等待服务器响应时执行其他脚本
* 当响应就绪后对响应进行处理

**Async = true**

当使用 async=true 时，请规定在响应处于 onreadystatechange 事件中的就绪状态时执行的函数：

xmlhttp.onreadystatechange=function()

{

if (xmlhttp.readyState==4 && xmlhttp.status==200)

{

document.getElementById("myDiv").innerHTML=xmlhttp.responseText;

}

}

xmlhttp.open("GET","test1.txt",true);

xmlhttp.send();

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_async_true)

您将在稍后的章节学习更多有关 onreadystatechange 的内容。

**Async = false**

如需使用 async=false，请将 open() 方法中的第三个参数改为 false：

xmlhttp.open("GET","test1.txt",false);

我们不推荐使用 async=false，但是对于一些小型的请求，也是可以的。

请记住，JavaScript 会等到服务器响应就绪才继续执行。如果服务器繁忙或缓慢，应用程序会挂起或停止。

注释：当您使用 async=false 时，请不要编写 onreadystatechange 函数 - 把代码放到 send() 语句后面即可：

xmlhttp.open("GET","test1.txt",false);

xmlhttp.send();

document.getElementById("myDiv").innerHTML=xmlhttp.responseText;

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_async_false)

## 3. 服务器响应

如需获得来自服务器的响应，请使用 XMLHttpRequest 对象的 responseText 或 responseXML 属性。

|  |  |
| --- | --- |
| **属性** | **描述** |
| responseText | 获得字符串形式的响应数据。 |
| responseXML | 获得 XML 形式的响应数据。 |

## responseText 属性

如果来自服务器的响应并非 XML，请使用 responseText 属性。

responseText 属性返回字符串形式的响应，因此您可以这样使用：

document.getElementById("myDiv").innerHTML=xmlhttp.responseText;

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_async_true)

## responseXML 属性

如果来自服务器的响应是 XML，而且需要作为 XML 对象进行解析，请使用 responseXML 属性：

请求 [books.xml](http://www.w3school.com.cn/example/xmle/books.xml) 文件，并解析响应：

xmlDoc=xmlhttp.responseXML;

txt="";

x=xmlDoc.getElementsByTagName("ARTIST");

for (i=0;i<x.length;i++)

{

txt=txt + x[i].childNodes[0].nodeValue + "<br />";

}

document.getElementById("myDiv").innerHTML**=txt**;

[亲自试一试](http://www.w3school.com.cn/tiy/t.asp?f=ajax_responsexml)

<html>

<head>

<script type="text/javascript">

function loadXMLDoc()

{

var xmlhttp;

if (window.XMLHttpRequest)

{// code for IE7+, Firefox, Chrome, Opera, Safari

xmlhttp=new XMLHttpRequest();

}

else

{// code for IE6, IE5

xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

}

xmlhttp.onreadystatechange=function()

{

if (xmlhttp.readyState==4 && xmlhttp.status==200)

{

document.getElementById("myDiv").innerHTML=xmlhttp.responseText;

}

}

xmlhttp.open("GET","/ajax/test1.txt",true);

xmlhttp.send();

}

</script>

</head>

<body>

<div id="myDiv"><h2>Let AJAX change this text</h2></div>

<button type="button" onclick="loadXMLDoc()">通过 AJAX 改变内容</button>

</body>

</html>

4.