

ajax怎么使用，ajax的详细使用步骤

作者：有故事的人 来源：范文网 www.wtabcd.cn/fanwen/

本文原地址：<https://www.wtabcd.cn/fanwen/zuowen/68392c650b14a2205f400eb35cee363d.html>

范文网，为你加油喝彩！

Ajax 简介

Ajax 由 HTML、JavaScript™ 技术、DHTML 和 DOM 组成，这一杰出的方法可以将笨拙的 Web 界面转成交互性的 Ajax 应用程序。它是一种构建网站的强大方法。

Ajax 尝试建立桌面应用程序的功能和交互性，与不断更新的 Web 应用程序之间的桥梁。可以使用像桌面应用程序中常见的动态用户界面和漂亮的控件，不过是在 Web 应用程序中。

Ajax 应用程序所用到的基本关于龙的诗句技术：

- 1、HTML 用于建立 Web 表单并确定应用程序其他部分使用的字段。
- 2、JavaScript 代码是运行 Ajax 应用程序的核心代码，帮助改进与服务器应用程序的通信。
- 3、DHTML 或 Dynamic HTML，用于动态更新表单。我们将使用 div、span 和其他动态 HTML 元素来标记 HTML。
- 4、文档对象模型 DOM 用于（通过 JavaScript 代码）处理 HTML 结构和（某些情况下）服务器返回的 XML。

Ajax 的定义

Ajax = Asynchronous JavaScript and XML（以及 DHTML 等）Asynchronous 异步 JS 和 XML。

XMLHttpRequest 这是一个 JavaScript 对象；
是处理所有服务器通信的对象，创建该对象很简单，如清单 1 所示。

清单 1. 创建新的 XMLHttpRequest 对象

```
<script language= " javascript " type= " text/javascript " >
```

```
var xmlHttp = new XMLHttpRequest();</script>
```

通过 XMLHttpRequest 对象与服务器进行对话的是 JavaScript 技术。这不是一般的应用程序流，这恰恰是 Ajax 的强大功能的来源。

Ajax 基本上就是把 JavaScript 技术和 XMLHttpRequest 对象放在 Web 表单和服务器之间。

得到 XMLHttpRequest 的句柄后，使用 JavaScript 代码完成以下任务：

- 1、获取表单数据：JavaScript 代码很容易从 HTML 表单中抽取数据并发送到服务器。
- 2、修改表单上的数据：更新表单也很简单，从设置字段值到迅速替换图像。
- 3、解析 HTML 和 XML：使用 JavaScript 代码操纵 DOM（请参阅下一节），处理 HTML 表单服务器返回的 XML 数据的结构

对于前两点，需要非常熟悉 getElementById() 方法，如 清单 2 所示。

清单 2. 用 JavaScript 代码捕获和设置字段值

```
//捕获字段值:

// 获得字段 " phone " 的值并用其创建一个变量phone

var phone = document.getElementById( " phone " ).value;

//设置字段值:

// 从response的数组中获得值并将其写到标签中

document.getElementById( " order " ).value = response[0];

document.getElementById( " address " ).value = response[1];
```

DOM的功能

当需要在 JavaScript 代码和服务器之间传递 XML 和改变 HTML 表单的时候，我们再深入研究 DOM。

获取 Request 对象

XMLHttpRequest 是 Ajax 应用程序的核心。

```
var xmlhttp;

if (window.XMLHttpRequest)

{ // 从 IE7+, Firefox, Chrome, Opera, Safari 中获得XMLHttpRequest对象
```

```
xmlhttp=new XMLHttpRequest();
```

```
}
```

```
else
```

```
{//从 IE6, IE5 中获得XMLHttpRequest对象
```

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

```
}
```

清单 4. 以支持多种浏览器的方式创建 XMLHttpRequest 对象

```
/* Create a new XMLHttpRequest object to talk to the Web server */
```

```
var xmlhttp = false;
```

```
/*@cc_on @*/
```

```
/*@if ( @_jscript_version >= 5)
```

```
try {
```

```
xmlhttp = new ActiveXObject( " Msxml2.XMLHTTP " );
```

```
} catch (e) {
```

```
try {
```

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

```
} catch (e2) {
```

```
xmlhttp = false;
```

```
}
```

```
}
```

```
@end @*/
```

```
if (!xmlhttp && typeof XMLHttpRequest != ' undefined ' ) {
```

```
xmlhttp = new XMLHttpRequest();
```

```
}
```

这段代码的核心分为三步：

1、建立一个变量 xmlHttp 来引用即将创建的 XMLHttpRequest 对象。

2、尝试在 Microsoft 浏览器中创建该对象：

1) 尝试使用 Msxml2.XMLHTTP 对象创建它。

如果失败，再尝试 Microsoft.XMLHTTP 对象。

3、如果仍然没有建立 xmlHttp，则以非 Microsoft 的方式创建该对象。最后，xmlHttp 应该引用一个有效的 XMLHttpRequest 对象，无论运行什么样的浏览器。

Ajax 的请求/响应

与服务器上的 Web 应用程序打交道的是 JavaScript 技术，而不是直接提交给那个应用程序的 HTML 表单。

发出请求

如何使用 XMLHttpRequest 对象？

首先 – 需要一个能够调用 JavaScript 方法的 Web 页面。

接下来就是在所有 Ajax 应用程序中基本都雷同的流程：

- 1、从 Web 表单中获取需要的数据。
- 2、建立要连接的 URL。
- 3、打开到服务器的连接。
- 4、设置服务器在完成后要运行的函数。
- 5、发送请求。

清单 5 中的示例 Ajax 方法就是按照这个顺序组织的：

清单 5. 发出 Ajax 请求

```
function callServer() {  
  
    // 获得city和state的值
```

```
var city = document.getElementById( " city " ).value;

var state = document.getElementById( " state " ).value;

// 当它们的值任一个不存在的时候结束JS

if ((city == null) || (city == " " )) return;

if ((state == null) || (state == " " )) return;

// 创建连接的URL对象

var url = " /scripts/getZipCode.php?city=" + escape(city) + " &利比亚女孩state=" + escape(state);

// 打开一个连接服务器的连接

xmlHttp.open( " GET " , url, true);

// 设置一个方法,当请求返回的时候调用这个方法

xmlHttp.onreadystatechange = updatePage;

//xmlHttp.onreadystatechange=function()

//{{

// if (xmlHttp.readyState==4 && xmlHttp.status==200)

// {

// document.getElementById( " myDiv " ).innerHTML=xmlHttp.responseText;

// }

//}}

// 发生链接

xmlHttp.send(null);

}
```

开始的代码使用基本 JavaScript 代码获取几个表单字段的值。然后设置一个 PHP 脚本作为链接的目标。要注意脚本 URL 的指定方式，city 和 state（来自表单）使用简单的 GET 参数附加在 URL 之后。最后一个参数如果设为 true，那么将请求一个异步连接（这就是 Ajax 的由来）。如果使用 false，那么代码发出请求后将等待服务器返回的响应。如果设为

true，当服务器在后台处理请求的时候用户仍然可以使用表单（甚至调用其他 JavaScript 方法）。

onreadystatechange属性可以告诉服务器在运行完成后做什么。因为代码没有等待服务器，必须让服务器知道怎么做以便您能作出响应。

在这个示例中，如果服务器处理完了请求，一个特殊的名为 updatePage() 的方法将被触发。

最后，使用值 null 调用send()。因为已经在请求 URL 中添加了要发送给服务器的数据（city 和state），所以请求中不需要发送任何数据。这样就发出了请求，服务器按照您的要求工作。

处理响应

1.什么也不要做，直到 xmlhttp.readyState 属性的值等于 4。

2.服务器将把响应填充到 xmlhttp.responseText 属性中。贫困申请理由

其中的第一点，即就绪状态;

第二点，使用 xmlhttp.responseText 属性获得服务器的响应,清单 6中的示例方法可供服务器根据清单 5 中发送的数据调用。

清单 6. 处理服务器响应

```
function updatePage() {  
  
    if (xmlhttp.readyState == 4) { var response = xmlhttp.responseText;  
  
        document.getElementById( " zipC高中化学难学吗ode " ).value = response;  
  
    }  
  
}
```

它等待服务器调用，如果是就绪状态，则使用服务器返回的值（这里是用户输入的城市和州的 ZIP 编码）设置另一个表单字段的值。

一旦服务器返回 ZIP 编码，updatePage() 方法就用城市/州的 ZIP 编码设置那个字段的值，用户就可以改写该值。这样做有两个原因：

保持例子简单，说明有时候可能希望用户能够修改服务器返回的数据。

要记住这两点，它们对于好的用户界面设计来说很重要。

连接 Web 表单

一个 JavaScript 方法捕捉用户输入表单的信息并将其发送到服务器，另一个 JavaScript

方法监听和处理响应，并在响应返回时设置字段的值。所有这些实际上都依赖于调用 第一个 JavaScript 方法，它启动了整个过程。

利用 JavaScript 技术更新表单。

清单 7. 启动一个 Ajax 过程

```
<form>
```

```
<p>City: <input type= " text " name= " city " id= " city " size= " 25
```

```
onChange= " callServer();" /></p>
```

```
<p>State: <input type= " text " name= " state " id= " state " size= " 25
```

```
onChange= " callServer();" /></p>
```

```
<p>Zip Code: <input type= " text " name= " zipCode " id= " city " size= " 5 /></p>
```

```
</form>
```

更多 作文 请访问 https://www.wtabcd.cn/fanwen/list/92_0.html

文章生成doc功能，由[范文网](#)开发