## AJAX 是一种在无需重新加载整个网页的情况下，能够更新部分网页的技术。

## 什么是 AJAX ？

AJAX = 异步 JavaScript 和 XML。

AJAX 是一种用于创建快速动态网页的技术。

通过在后台与服务器进行少量数据交换，AJAX 可以使网页实现异步更新。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。

传统的网页（不使用 AJAX）如果需要更新内容，必需重载整个网页面。

有很多使用 AJAX 的应用程序案例：新浪微博、Google 地图、开心网等等。

# AJAX XHR

## XHR 创建对象

XMLHttpRequest 是 AJAX 的基础。

XHR对象是一个javascript对象，它类似于一个小线程，在不刷新网页的前提下运行并于后端交互。

### 创建语法

var xhrName = new XMLHttpRequest();//支持IE7+

## XHR 请求

向服务器发送请求有open()和send()两种方法

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

xhrName.send();

**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：只有在用"POST"，并且需要发送参数的时候，才会使用到类似这样：

xmlhttp.send("user="+username+"&password="+password)

当使用async=true时，请设置响应处于 onreadystatechange 事件中的就绪状态时执行的函数：（即一个处理服务器响应的函数）

xhrName.onreadystatechange=function(){ //执行函数

}

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

xhrName.send();

## XHR 响应

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;

### 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：

## XHR readyState

### onreadystatechange 事件

当请求被发送到服务器时，我们需要执行一些基于响应的任务。每当 readyState 改变时，就会触发 onreadystatechange 事件。readyState 属性存有 XMLHttpRequest 的状态信息。

下面是 XMLHttpRequest 对象的三个重要的属性：

**onreadystatechange** 存储函数（或函数名），每当 readyState 属性改变时，就会调用该函数。

**readyState** 存有 XMLHttpRequest 的状态。从 0 到 4 发生变化。

0: 请求未初始化

1: 服务器连接已建立

2: 请求已接收

3: 请求处理中

4: 请求已完成，且响应已就绪

**status**

200: "OK"

404: 未找到页面

当 readyState 等于 4 且状态为 200 时，表示响应已就绪,已接收到服务端返回的响应。

xmlhttp.onreadystatechange=function()

{

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

{

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

}

}

### callback函数

callback 函数是一种以参数形式传递给另一个函数的函数。

如果您的网站上存在多个 AJAX 任务，那么您应该为创建 XMLHttpRequest 对象编写一个标准的函数，并为每个 AJAX 任务调用该函数。//即把整个函数封装起来，该函数调用应该包含 URL 以及发生 onreadystatechange 事件时执行的任务

例：function myFunction()

{

loadXMLDoc("/ajax/test1.txt",function()

{

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

{

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

}

});

}

此例中callback函数为loadXMLDoc函数