**AJAX**

异步 JavaScript 和 XML。

拓薪教育

刘洋

1. 初始AJAX(熟悉)
   1. AJAX介绍

AJAX = 异步 JavaScript 和 XML。

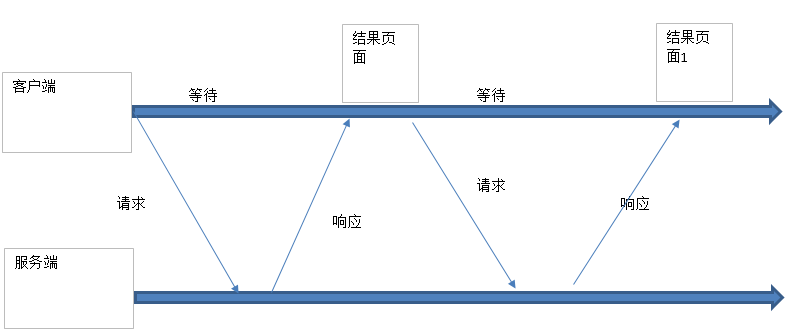
AJAX 是一种用于创建快速动态网页的技术。

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。

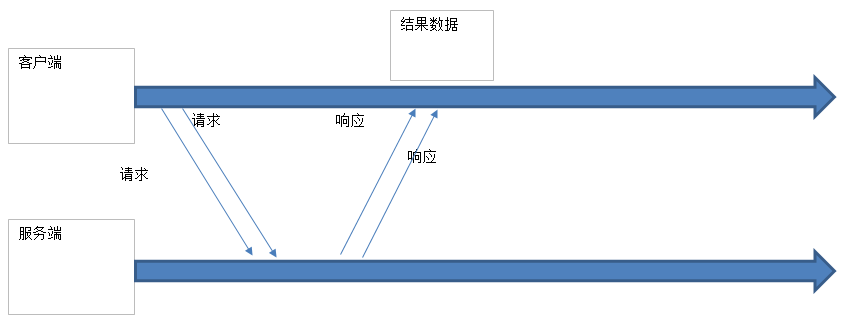
传统的网页（不使用 AJAX）如果需要更新内容，必需重载整个网页面。

目前几乎所有的网站都离不开AJAX这门技术。

传统技术:

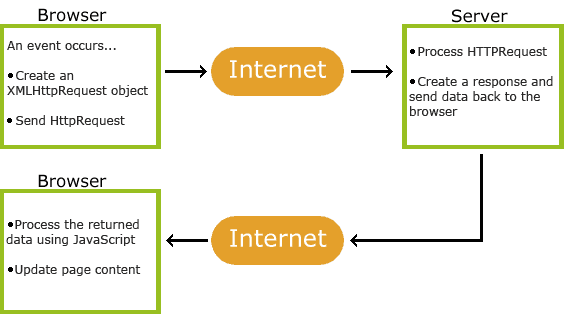


AJAX技术:



**AJAX最大特点: 页面无刷新的请求，ajax默认是异步的，速度快，用户体验比较好**

* 1. AJAX工作原理



1. AJAX使用(精通)
   1. 环境准备

使用AJAX技术我们需要用到HTML/CSS/JS/jQuery/JavaWeb相关知识,原生的JS也可以进行AJAX的操作,但是相对比较繁琐也没有必要,故此我们本节课学习使用jQuery封装后的AJAX技术来进行学习;

* 1. 创建AJAX请求

|  |
| --- |
| ***$*.*ajax*({  *url* : "ajax/getName?id="+*id*,**// ajax请求的url地址  ***type* : "get",**// 提交的方法  success **: *function*(**data**) {** // ajax请求回调后执行的方法,data表示从服务器返回的数据  *alert***(**data**);  } });** |

以上代码表示的意思为:

使用get方法向ajax/getName的URL提交id数据,并且在服务器返回后弹出服务器返回内容

* 1. AJAX参数说明
* **url**:

要求为String类型的参数，（默认为当前页地址）发送请求的地址。

* **type**:

要求为String类型的参数，请求方式（post或get）默认为get。注意其他http请求方法，例如put和delete也可以使用，但仅部分浏览器支持。

* timeout:

要求为Number类型的参数，设置请求超时时间（毫秒）。此设置将覆盖$.ajaxSetup()方法的全局设置。

* **async**:

要求为Boolean类型的参数，默认设置为**true**，所有请求均为**异步请求**。如果需要发送同步请求，请将此选项设置为false。注意，同步请求将锁住浏览器，用户其他操作必须等待请求完成才可以执行。

* cache:

要求为Boolean类型的参数，默认为true（当dataType为script时，默认为false），设置为false将不会从浏览器缓存中加载请求信息。

* **data**:

要求为Object或String类型的参数，发送到服务器的数据。如果已经不是字符串，将自动转换为字符串格式。get请求中将附加在url后。防止这种自动转换，可以查看　　processData选项。对象必须为key/value格式，例如{foo1:"bar1",foo2:"bar2"}转换为&foo1=bar1&foo2=bar2。如果是数组，JQuery将自动为不同值对应同一个名称。例如{foo:["bar1","bar2"]}转换为&foo=bar1&foo=bar2。

* **dataType**:

要求为String类型的参数，预期服务器返回的数据类型。如果不指定，JQuery将自动根据http包mime信息返回responseXML或responseText，并作为回调函数参数传递。可用的类型如下：

xml：返回XML文档，可用JQuery处理。

html：返回纯文本HTML信息；包含的script标签会在插入DOM时执行。

script：返回纯文本JavaScript代码。不会自动缓存结果。除非设置了cache参数。注意在远程请求时（不在同一个域下），所有post请求都将转为get请求。

**json**：返回JSON数据。

jsonp：JSONP格式。使用SONP形式调用函数时，例如myurl?callback=?，JQuery将自动替换后一个“?”为正确的函数名，以执行回调函数。

text：返回纯文本字符串。

* beforeSend：

要求为Function类型的参数，发送请求前可以修改XMLHttpRequest对象的函数，例如添加自定义HTTP头。在beforeSend中如果返回false可以取消本次ajax请求。XMLHttpRequest对象是惟一的参数。

function(XMLHttpRequest){

this; //调用本次ajax请求时传递的options参数

}

* complete：

要求为Function类型的参数，请求完成后调用的回调函数（请求成功或失败时均调用）。参数：XMLHttpRequest对象和一个描述成功请求类型的字符串。

function(XMLHttpRequest, textStatus){

this; //调用本次ajax请求时传递的options参数

}

* **success**：

要求为Function类型的参数，请求成功后调用的回调函数，有两个参数。

(1)由服务器返回，并根据dataType参数进行处理后的数据。

(2)描述状态的字符串。

function(data, textStatus){

//data可能是xmlDoc、jsonObj、html、text等等

this; //调用本次ajax请求时传递的options参数

}

* error:

要求为Function类型的参数，请求失败时被调用的函数。该函数有3个参数，即XMLHttpRequest对象、错误信息、捕获的错误对象(可选)。ajax事件函数如下：

function(XMLHttpRequest, textStatus, errorThrown){

//通常情况下textStatus和errorThrown只有其中一个包含信息

this; //调用本次ajax请求时传递的options参数

}

* **contentType**：

要求为String类型的参数，当发送信息至服务器时，内容编码类型默认为"application/x-www-form-urlencoded","json"。该默认值适合大多数应用场合。

* dataFilter：

要求为Function类型的参数，给Ajax返回的原始数据进行预处理的函数。提供data和type两个参数。data是Ajax返回的原始数据，type是调用jQuery.ajax时提供的dataType参数。函数返回的值将由jQuery进一步处理。

function(data, type){

//返回处理后的数据

return data;

}

* global：

要求为Boolean类型的参数，默认为true。表示是否触发全局ajax事件。设置为false将不会触发全局ajax事件，ajaxStart或ajaxStop可用于控制各种ajax事件。

* ifModified：

要求为Boolean类型的参数，默认为false。仅在服务器数据改变时获取新数据。服务器数据改变判断的依据是Last-Modified头信息。默认值是false，即忽略头信息。

* jsonp：

要求为String类型的参数，在一个jsonp请求中重写回调函数的名字。该值用来替代在"callback=?"这种GET或POST请求中URL参数里的"callback"部分，例如{jsonp:'onJsonPLoad'}会导致将"onJsonPLoad=?"传给服务器。

* username：

要求为String类型的参数，用于响应HTTP访问认证请求的用户名。

* password：

要求为String类型的参数，用于响应HTTP访问认证请求的密码。

* processData：

要求为Boolean类型的参数，默认为true。默认情况下，发送的数据将被转换为对象（从技术角度来讲并非字符串）以配合默认内容类型"application/x-www-form-urlencoded"。如果要发送DOM树信息或者其他不希望转换的信息，请设置为false。

* scriptCharset：

要求为String类型的参数，只有当请求时dataType为"jsonp"或者"script"，并且type是GET时才会用于强制修改字符集(charset)。通常在本地和远程的内容编码不同时使用。

1. AJAX练习

表单验证用户名是否存在

前台页面:

|  |
| --- |
| **<*head*>  <*meta* charset="UTF-8">  <*title*>**AJAX测试页**</*title*>** <!--引入jQuery的js-->  **<*script* type="text/javascript" src="js/jquery-1.8.3.js"></*script*> </*head*> <*body*>** ID:**<*input* type="text" id="username" onblur="***sendAjax***()"/><*br*>  <*span* id="msg"></*span*><*br*> </*body*> <*script* type="text/javascript">** // 发送ajax的方法  ***function*** *sendAjax***(){** // 获取当前输入框的id  ***var*** username **= *$*("#username").**val**();** // 调用jquery的ajax方法发送请求  ***$*.*ajax*({  *url* : "ajax/getName?username="+**username**,  *type* : "post",** success **: *function*(**data**) {** // 获取服务器返回内容后显示在页面上  ***$*("#msg").*html*(**data**);  }  });  } </*script*> </*html*>** |

Servlet代码:

|  |
| --- |
| ***public class AJAXServlet extends HttpServlet* {  *protected void* doPost(**HttpServletRequest request**,** HttpServletResponse response**) *throws ServletException*, *IOException* {  *String username* =** request**.getParameter("username");  *String result* = "";  *if*(*username*!=*null* && *username*.equals("zhangsan")){  *result* = "is exist";  }*else*{  *result* = "not exist";  }** response**.getWriter().write(*result*);  }   *protected void* doGet(**HttpServletRequest request**,** HttpServletResponse response**) *throws ServletException*, *IOException* {  doPost(**request**,**response**);  } }** |