Ajax（1）

# 一、概述

手册位置：W3C手册 -- 上面的JavaScript -- 左侧的Ajax

1、Ajax历史

1999年，微软在IE5上使用，但是没有被认知。

2005年，谷歌的使用（地图、gmail），使得ajax被全世界所认识，并广泛使用。

2、Ajax是什么

Ajax中的第一个A是Asynchronous [eɪˈsɪŋkrənəs]

（异步）JavaScript and XML的缩写。可以看出它不是一种技术，而是多种技术的综合体。

其中有JavaScript、有xml、有json、有html、有css、有dom、有**XMLHttpRequest**对象等等。

XMLHttpRequest对象，也叫做Ajax对象。

Ajax的工作也是基于http协议的。

3、Ajax能做什么

①、提高用户体验。

②、表单时时验证



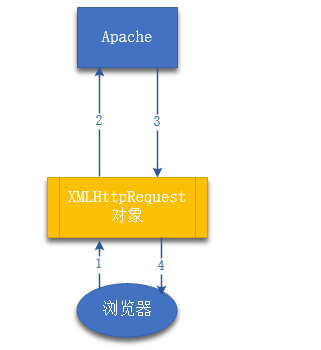
③、百度地图、谷歌地图

④、网页版的聊天室

⑤、无刷新的分页

# 二、如何使用Ajax

1、工作原理



通过ajax向服务器发送请求的时候，实际上是通过浏览器内置的XMLHttpRequest对象来发送的请求，服务器返回数据的时候，也是返回到XMLHttpRequest对象，经过XMLHttpRequest的处理，最后将用户想要的结果输出到浏览器。

2、XMLHttpRequest对象

1、创建xhr对象（ajax对象）

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



2、主要的成员属性：

readyState：表示xhr工作进行到何种状态

onreadystatechange：事件，表示readyState改变的时候触发的事件

status：http状态码

responseText：用来接收返回的文本类型的数据

responseXML：用来接收返回的是XML格式的数据

3、主要的方法：

open()：设置（定义）一个请求

send()：发送设置好的请求

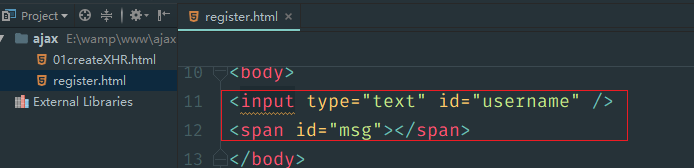
setRequestHeader()：设置ajax请求头信息

3、使用Ajax的步骤

完成验证用户名案例，要求用户名唯一，就是不能有重复。

1、创建所需文件

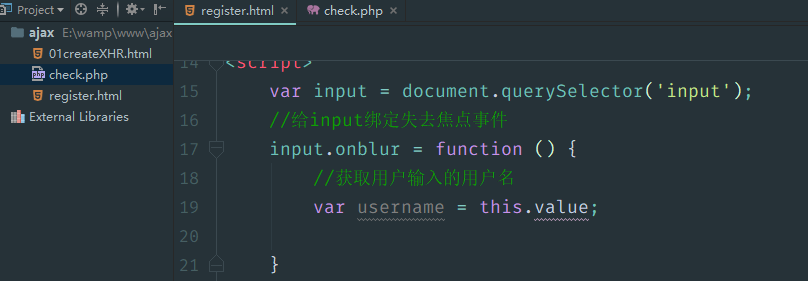
HTML：



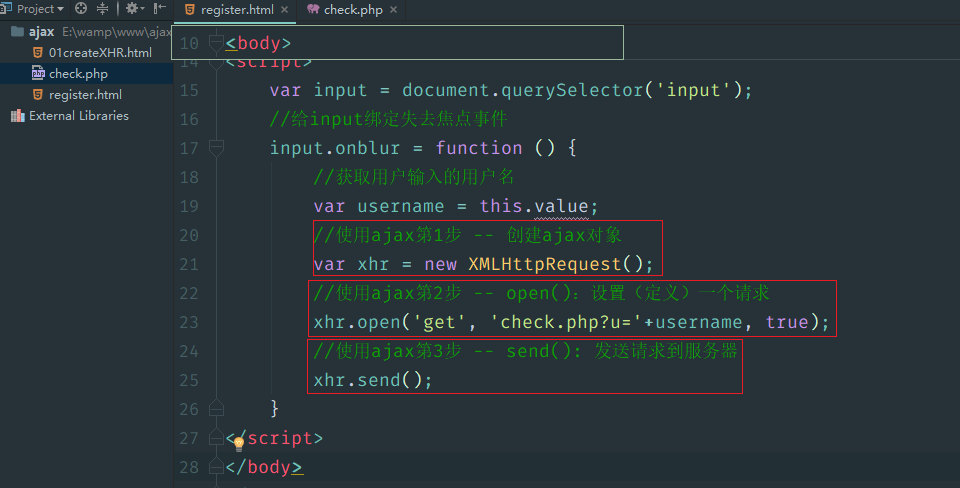
PHP文件：



2、获取input输入框，绑定失去焦点事件



3、创建ajax对象，定义请求的参数，发送请求

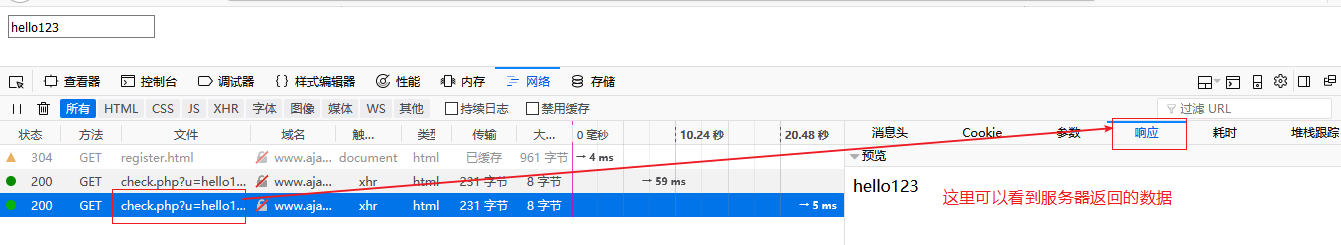


浏览器检测是否在失去焦点时发送了这个请求：

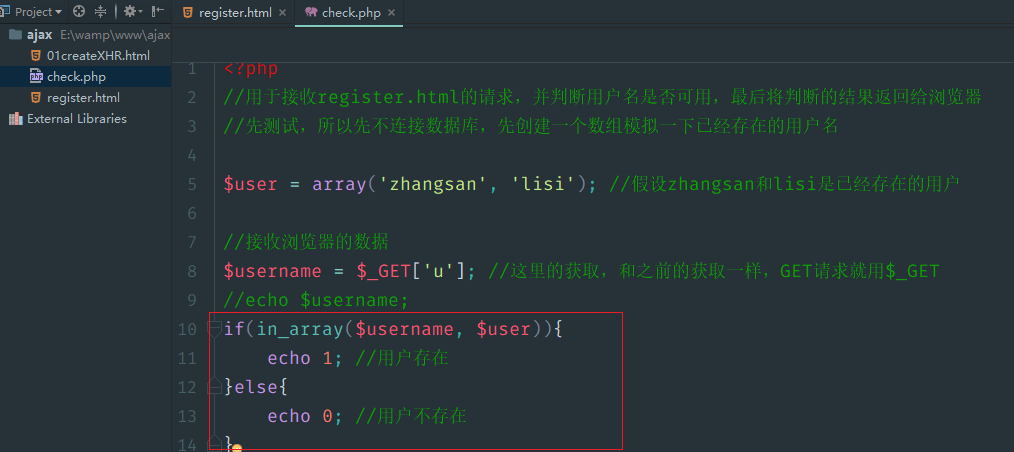


4、服务器接收数据并判断





通过上面的代码可以看到服务器确实接收到了数据，所以可以进行判断，并将判断的结果返回



5、绑定onreadystatechange事件，当readyState为4的时候，才获取服务器返回的数据

readyState的四个状态如下：

|  |  |
| --- | --- |
| 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，言外之意，open已经调用了 |
| 2 (发送数据) | send方法已调用，但是当前的状态及http头未知 |
| 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responseBody和responseText获取完整的回应数据 |

readyState表示ajax执行的状态，也就是能够表示ajax执行到哪一步了。

onreadystatechange是一个事件，用于监听readyState的状态，当readyState发生变化的时候，该事件被触发。

当readyState的值变化到4的时候，即完全接收了数据，这个时候在处理服务器返回的数据就比较准确了。



6、处理返回的数据



4、总结使用ajax的步骤

1、创建ajax对象

2、绑定onreadystatechange事件，事件中判断readyState（4）及status（200）的值，用responseText接收返回的数据

3、open，定义请求（请求的方式，url[可能有参数]，异步请求）

4、send，发送请求

# 三、细节问题

1、编码字符

encodeURI() --- 不能对“:/=&”进行编码

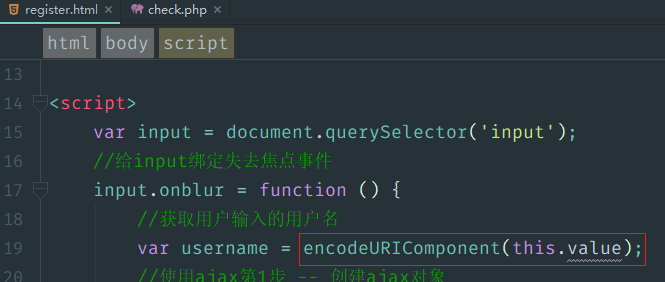
encodeURIComponent() --- 能对“:/=&”进行编码

比如我的用户名为lisi&age=20，那么请求的结果是：

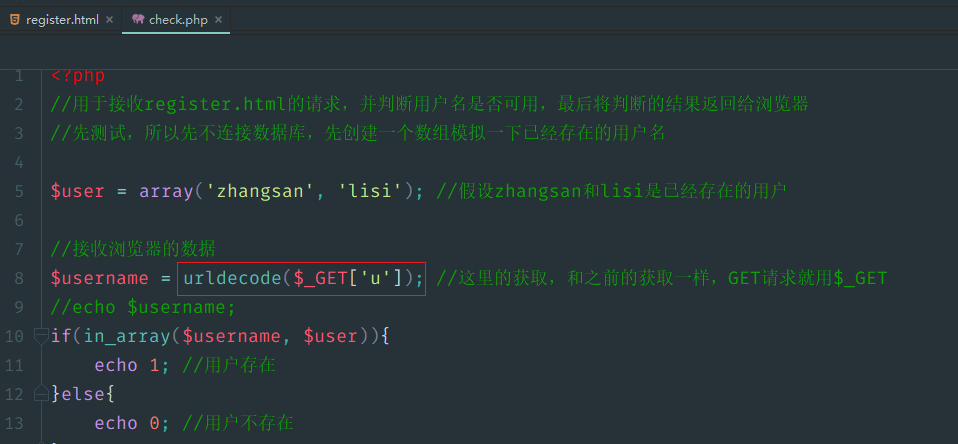


请求的时候，实际上是两个参数了，姓名仍然是lisi，所以错误，这个时候就需要对用户名进行编码处理。

此时应该用encodeURIComponent()。



服务器端应该对获取的内容进行解码操作：



2、缓存处理

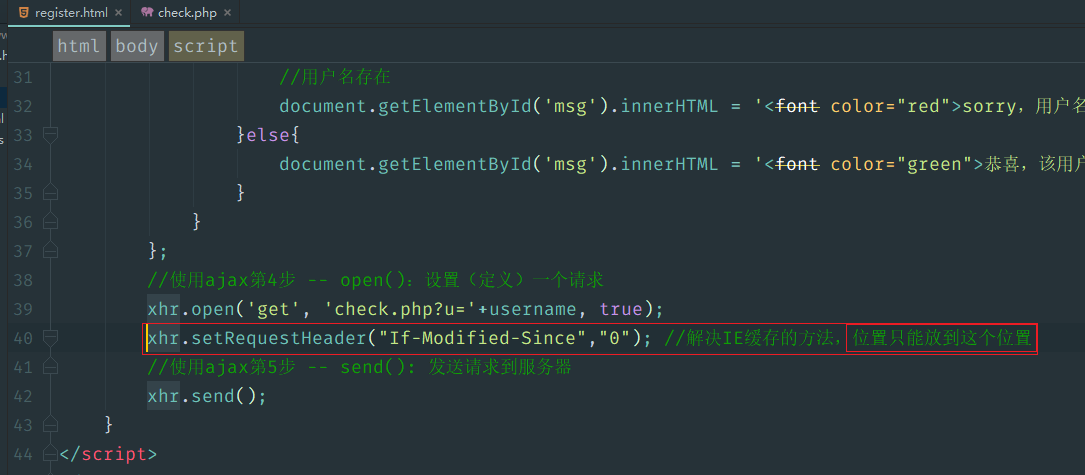
什么是Ajax缓存原理？

Ajax在发送的数据成功后，会把请求的URL和返回的响应结果保存在浏览器缓存内，当下一次调用Ajax发送相同的请求时，它会直接从缓存中把数据取出来，这是为了提高页面的响应速度和用户体验。当前这两次请求URL完全相同，包括参数。这个时候，浏览器就不会与服务器交互。

两次请求的地址完全一致（包括参数），那么IE就会从缓存中取服务器上一次返回的结果。而不会从新向服务器发送请求。

如何解决IE缓存的问题：

1、在ajax发送请求前加上 xhr.setRequestHeader("If-Modified-Since","0"); //实测可用



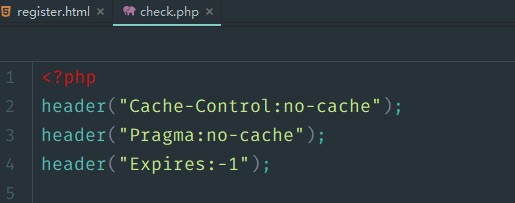
2、在服务器端加

header("Cache-Control:no-cache");

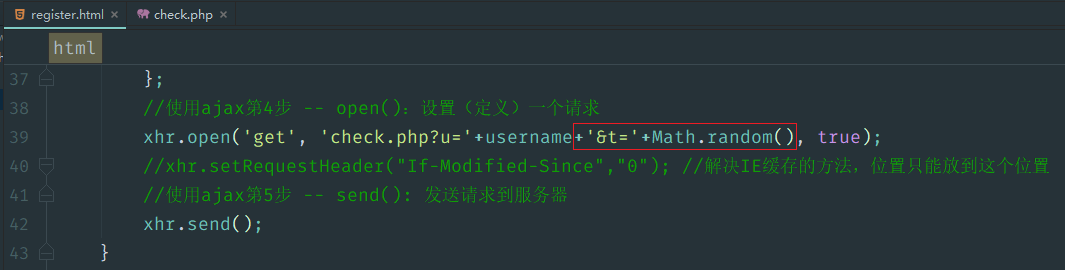
header("Pragma:no-cache");

header("Expires:-1");

加入三个header，目的是让所有的浏览器都不缓存。



3、在 Ajax 的 URL 参数后加上 "?fresh=" + Math.random(); //当然这里参数 fresh 可以任意取了



4、第种方法和第3种类似，在 URL 参数后加上 "?timestamp=" + new Date().getTime();



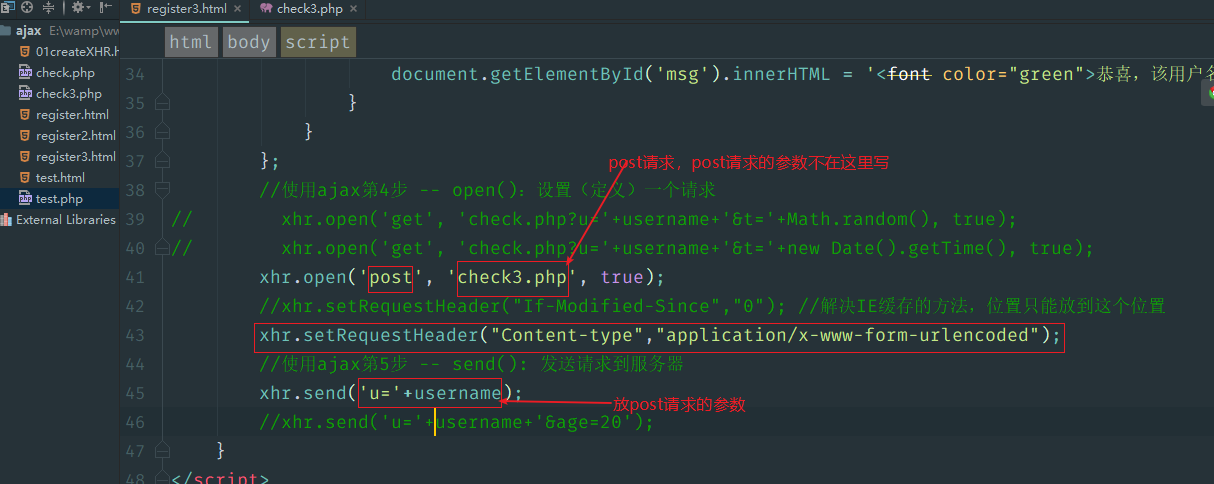
5、用POST替代GET：不推荐

以上几种方式，**比较推荐第3和第4中方法**。

3、get和post请求

使用POST请求：

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



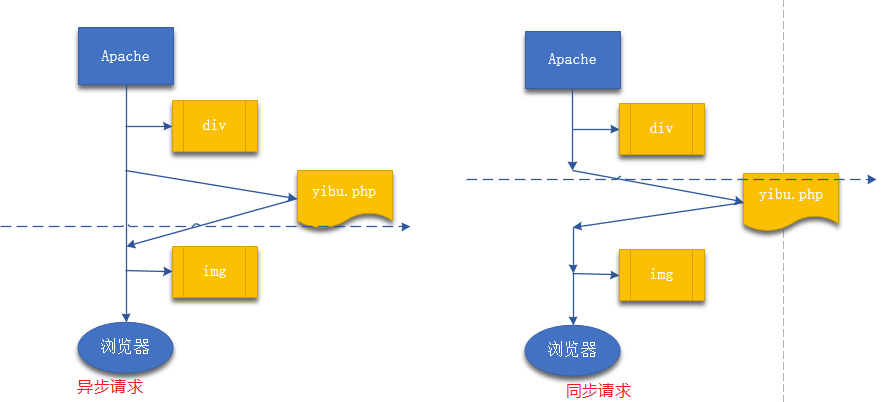
**ajax中的get和post的一些说明**：

①、用post请求，不会产生缓存。

②、用post请求的时候，能不能也用get，即post和get请求同时使用。答案是能，只不过get请求的参数用$\_GET获取，post请求的参数用$\_POST获取。

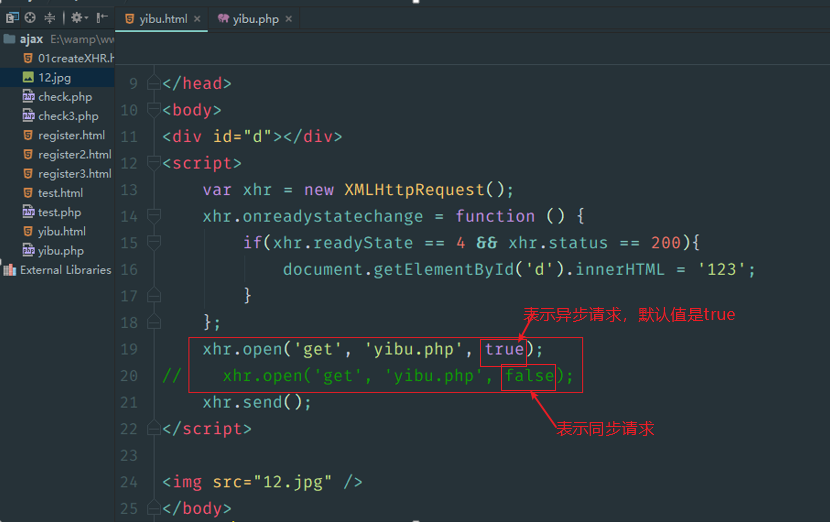
③、用$\_REQUEST获取get和post方式提交的参数，如果参数名相同，获取的是谁？获取的是post的内容

4、同步、异步

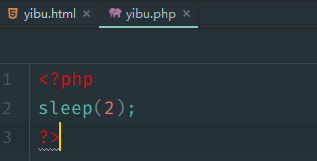


异步请求，在同一个时间点可以执行多个进程；

同步请求，在同一个时间点，只能进行一个操作，其他操作只能等待。

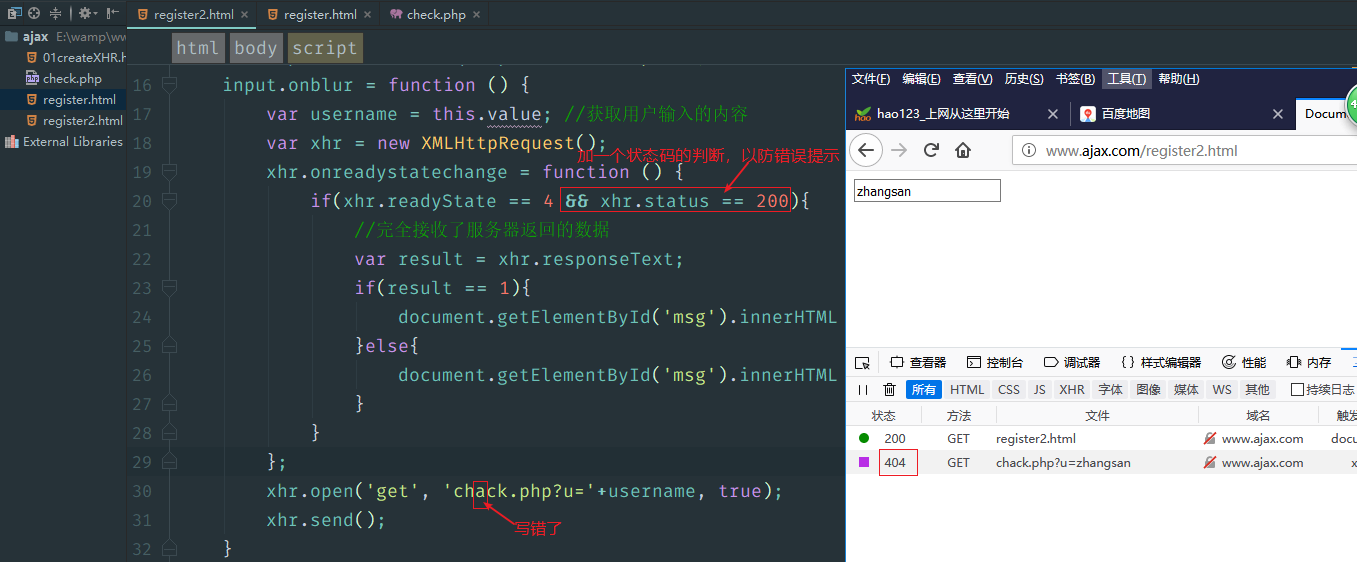


PHP代码：



5、请求的url如果出错怎么办

不但要判断readyState的值，还要判断http状态吗是否为200.



# 四、JSON

w3c手册位置：上面的javascript – 左侧的JSON

1、json

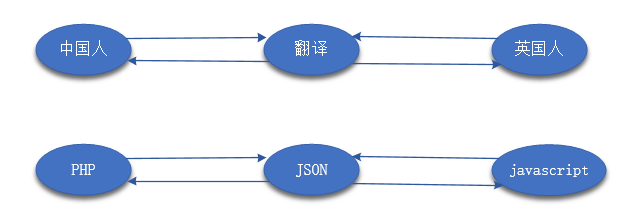
**json本质就是字符串**，只不过这个字符串的格式有要求，格式符合js中的数组和对象的格式。

举例：

①、 "['apple', 'orange', 'banana']"

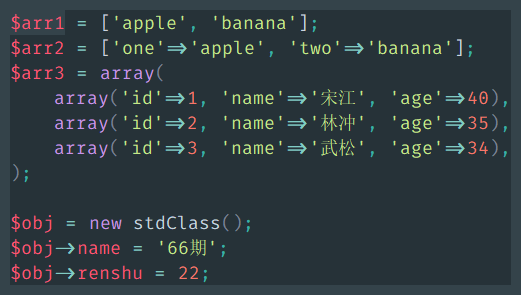
②、 "{name:'zhangsan', age:20}"

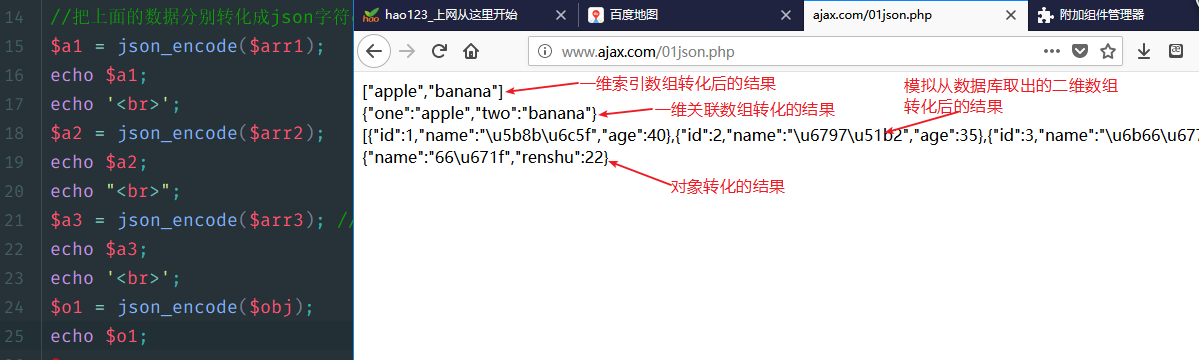
json在web开发中，它起到的作用和xml和的作用一样。具体来说，json可以当做配置文件，json文件可以存储数据，json可以当做两种编程语言交换数据的媒介。



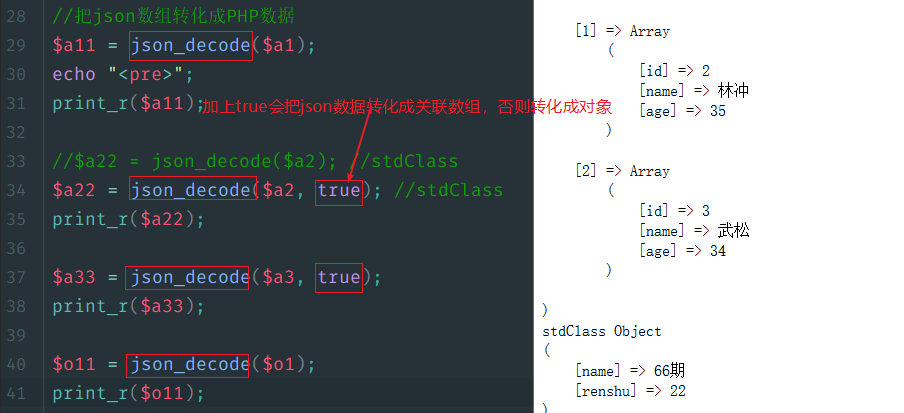
2、json和PHP进行转换

1、把PHP数据转化成json格式



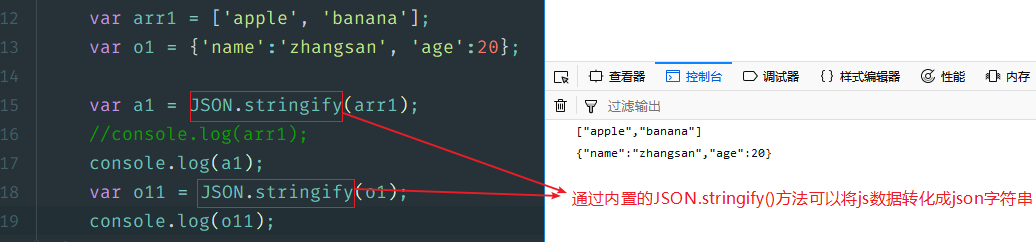


2、json数据转化成PHP数据



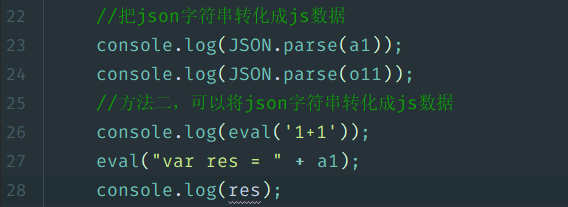
3、json和JavaScript进行转换

1、js数据转化成字符串

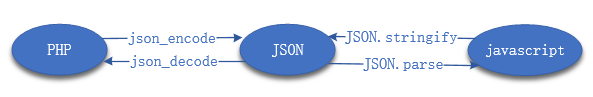


2、json字符串转化成js数据

可以使用JSON.parse()，也可以使用eval()

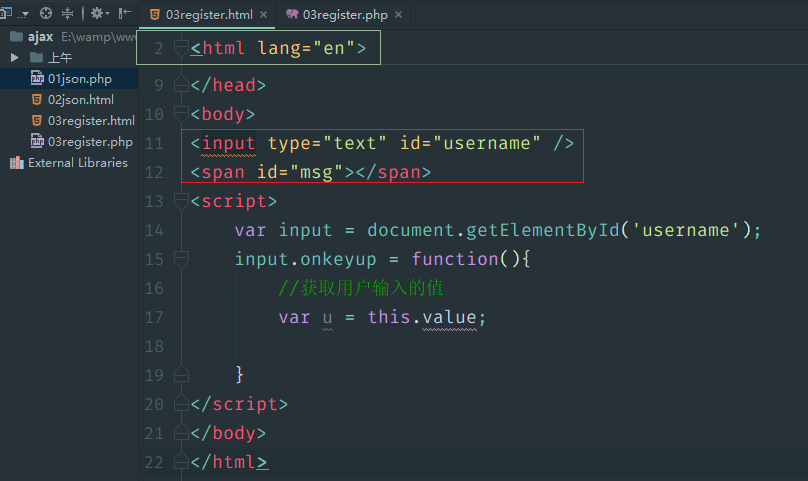


总结：

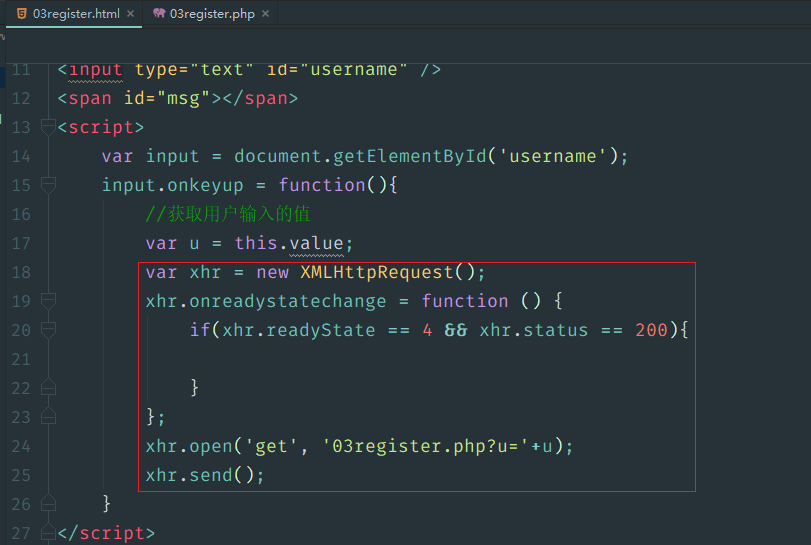


# 五、服务器返回json格式的数据

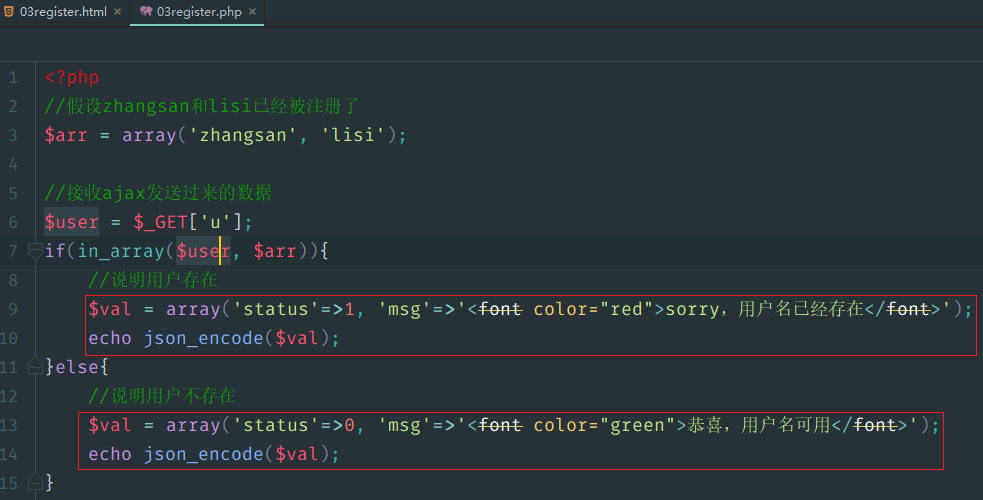
1、给input输入框绑定事件



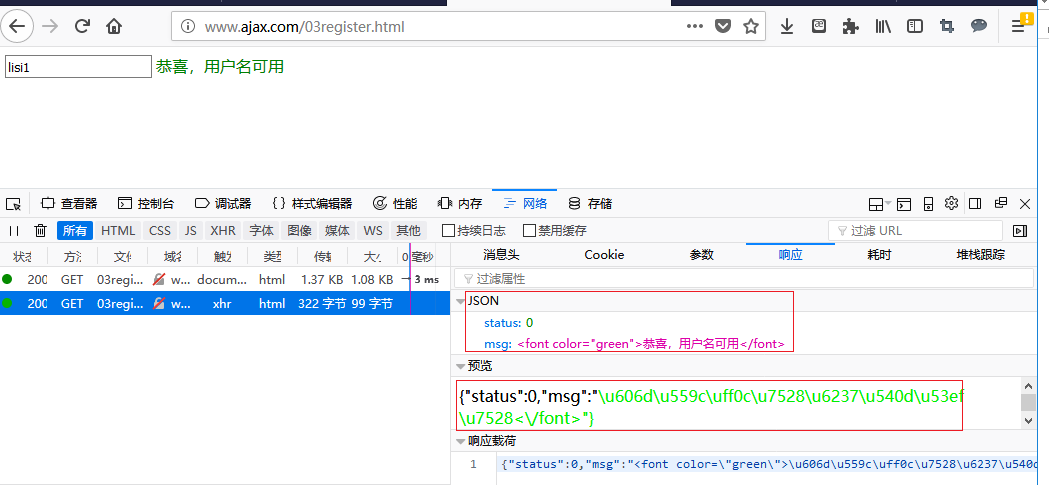
2、发送ajax请求



3、服务器端返回数据



当用户输入的时候，浏览器查看效果：

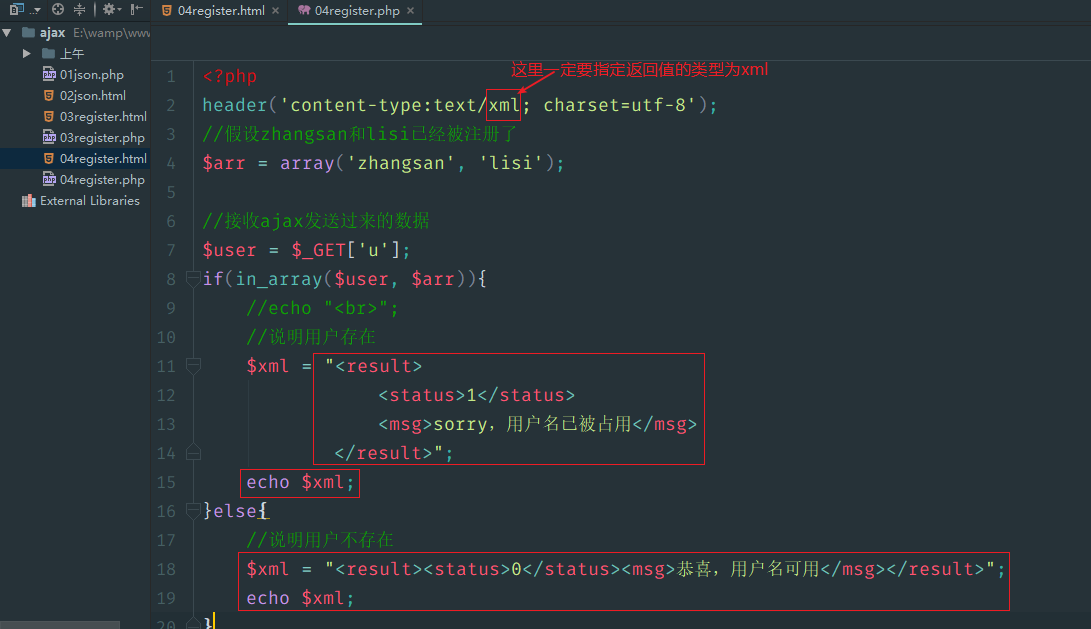


4、Ajax对象接收返回的数据并处理

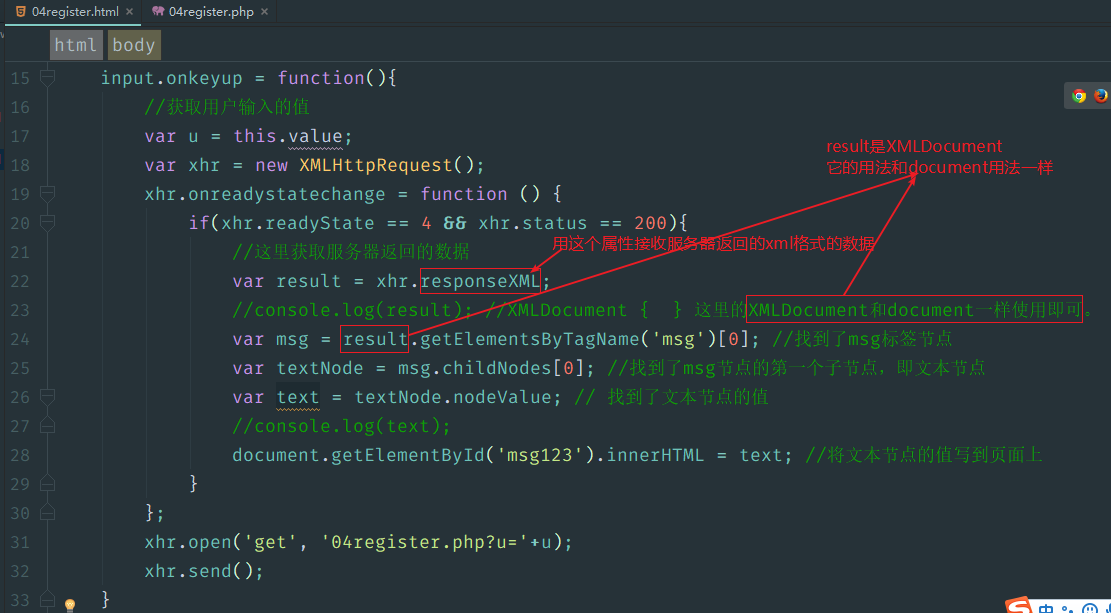


# 六、服务器返回xml类型的数据

服务器端必须声明返回的数据是xml：



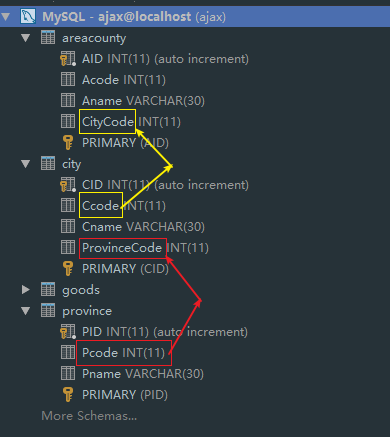
必须使用ajax（xhr）对象中的responseXML来获取服务器返回的数据：



# 七、案例—省市县三级联动

1、创建各个数据表

导入SQL，并确定各个表之间的关系。



在省（province）表中，Pcode表示省的代码，Pname表示省的名字。

在市（city）表中，Ccode表示市的代码，Cname表示市的名字，ProvinceCode表示该市是哪个省的。

在县（areacounty）表，Acode表示县的代码，Aname表示县的名字，CityCode表示该县是哪个市的。

比如查询所有的省：select \* from province;

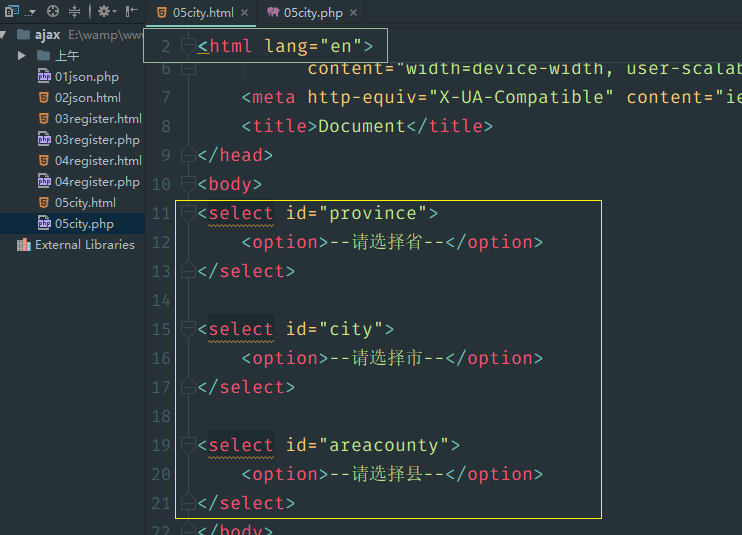
比如查询河北省的市(已知河北省代码为130000)：select \* from city where ProvinceCode = '130000';

比如查询张家口的县(已知张家口市的代码为130700)：select \* from areacounty where CityCode = '130700';

2、创建city.html和city.php

city.html用于界面，显示省市县。

city.php用于从数据库查询数据，并返回给city.html



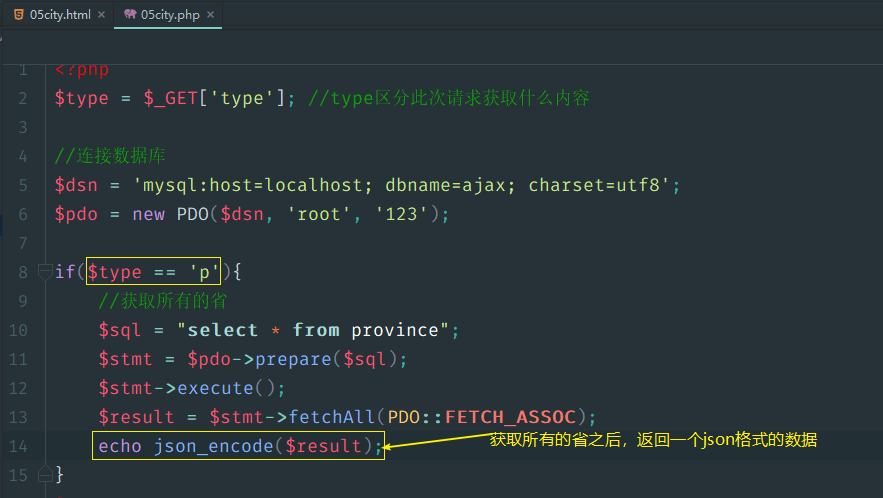
3、页面加载完毕，获取所有的省

1、发送ajax请求



2、服务器处理

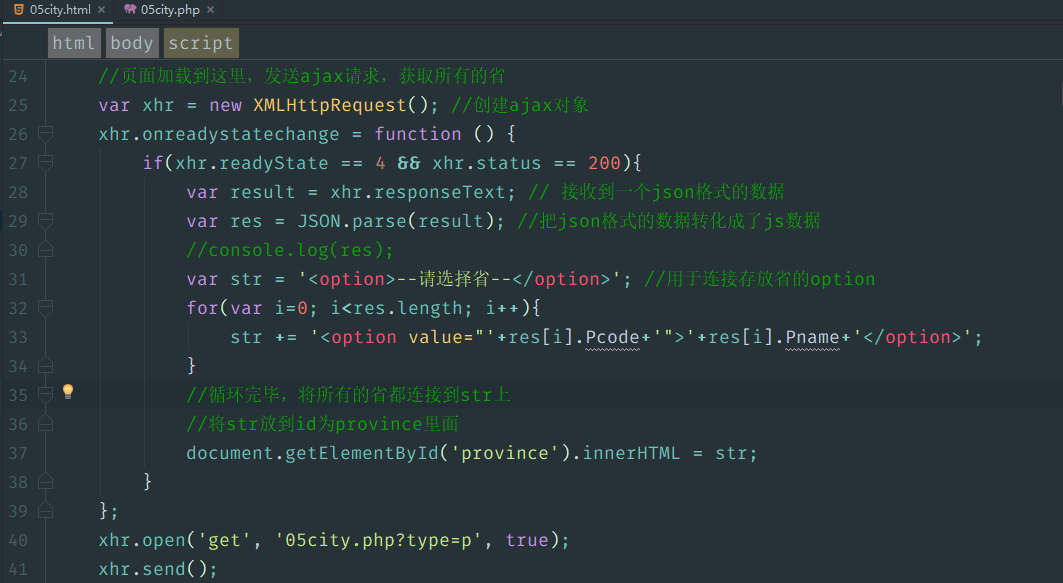
服务器端接收type参数，连接数据库，写SQL，获取所有的省，转化成json，返回给浏览器。



浏览器查看效果：



3、浏览器接收返回的数据并处理



4、切换省的时候，获取对应的市

复制代码，修改一下即可。

HTML中的js代码：



PHP代码：

