Ajax

# 一，预备知识

## 1，客户端与服务器

**什么是网页？**

web, www，万维网， html, css, js

**网页实际上放在什么地方？**

服务器

**客户端与服务器的本质是什么？**

**计算机**

硬件和软件

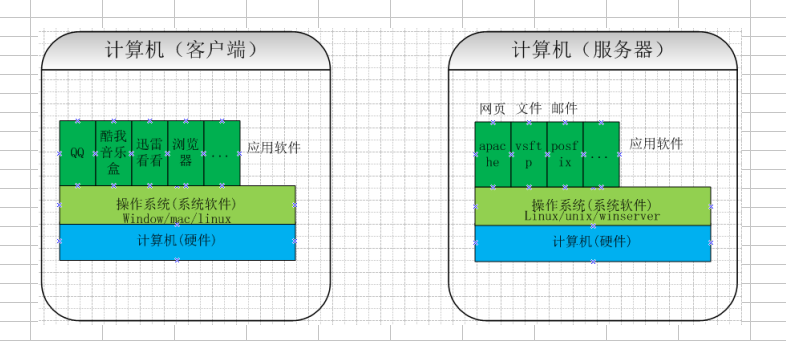
硬件：硬盘，内存条，CPU，主板，输入设备，输出设备

软件：系统软件和应用软件

系统软件：系统软件

应用软件：QQ，迅雷，浏览器，暴风，酷我音乐.....

区别如下图：



**客户端是用来干什么的？服务器是用来干什么的？**

**客户端**：平常使用应用程序，QQ，音乐播放器，浏览器....

**服务器**：对外提供服务的，网页服务（apache/nginx/tomcat），文件上传下载服务，邮件服务.....

**客户端有哪些？**

从广义上来说，可以指一台计算机，也可以指浏览器，也可以指一个应用软件。



**服务器?**



## 2，http协议

HyperText Tansfer Protocol 超文件传输协议。

**http协议有两部分组成：**

* http请求
* http响应

**http协议模型：**



**请求有几部分组成：**

* 请求行
* 请求头（消息报头）
* 请求正文

**响应有几部分组成：**

* 状态行
* 响应头（消息报头）
* 响应正文

## 3，B/S, C/S？ 它们的优缺点？

B/S: B(Browser), S(Server) 浏览器/服务器 浏览器与服务器

C/S: C(Client), S(Server) 客户端/服务器 这里的客户端指应用软件

**B/S架构的优缺点：**

优点: 更新维护方便，不需要安装各种客户端

缺点: 响应速度会慢一点

**C/S架构的优缺点：**

优点: 响应速度快

缺点: 更新维护不方便

## 5，iframe复习

iframe浮动框架， 或 内嵌框架

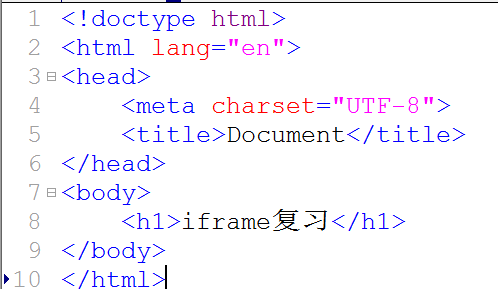
作用：嵌入第三方内容的

格式：<iframe src=””/>

使用如下：



另一个页面如下：



## 6，同步和异步

**同步：**

造成阻塞

**异步：**

非阻塞

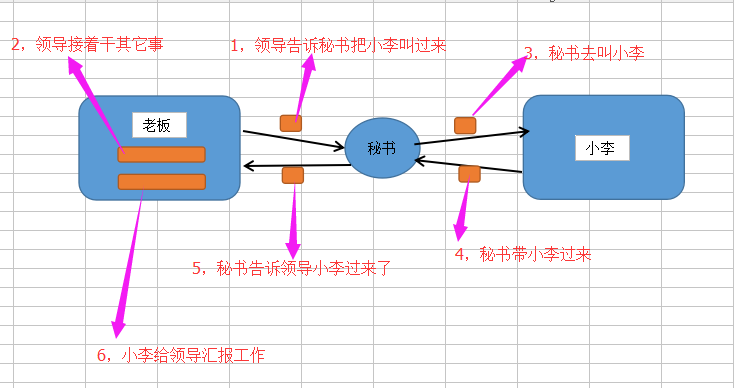
**页面加载的同步与异步：**

普通页面：有时候会出现**白屏**，相当地全屏加载，加叫页面的全屏刷新。

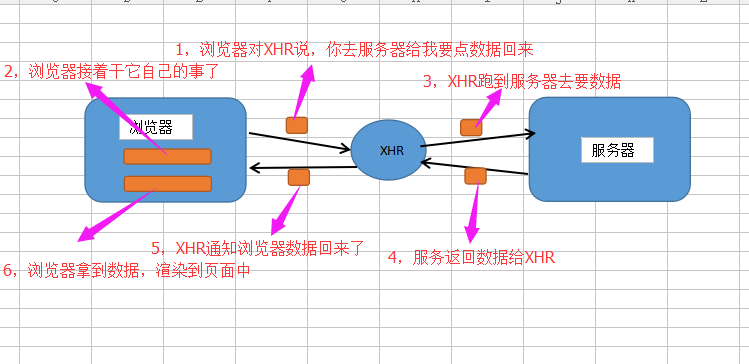
页面局部刷新：评论加载，百度

## 7，Ajax思想：

**现实生活中例子（领导找小李汇报工作）：**



**Ajax思想例子：**



# 二，初识Ajax

## 1，什么是Ajax?

**读音：**



**Ajax:** Asynchronous Javascript And XML 异步的JS和XML

AJax是多门老技术，新应用。

老技术：html, css, js, dom, bom, XMLHttpRequest..

新应用：Ajax

**为什么需要学习Ajax?**

前面说到B/S架构的缺点：响应速度慢，现在想让响应速度快一点，速度快了，那么用户体验就好了，Ajax就是为了提高用户体验的。Ajax尝试在Web应用程序实现类似于C/S架构的用户体验，就是把响应速度提上去。

## 2，Ajax的历史

**Ajax有三个阶段：**

1，早期的Ajax, 使用iframe

2，传统Ajax

3，反向Ajax

# 三，检测用户名-传统方式（全屏刷新）

**需求：**一个注册页面，当用户输入一个用户名时，判断这个用户名是否已经存在，如果存在，提示用户这个用户名已存在，请重新输入用户名，如果不存在，提示用户，这个用户可以使用。

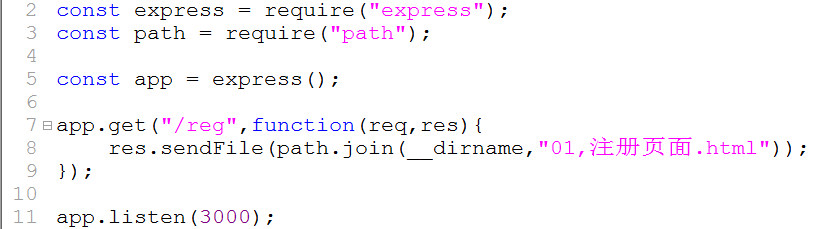
**先写一个静态页面：**





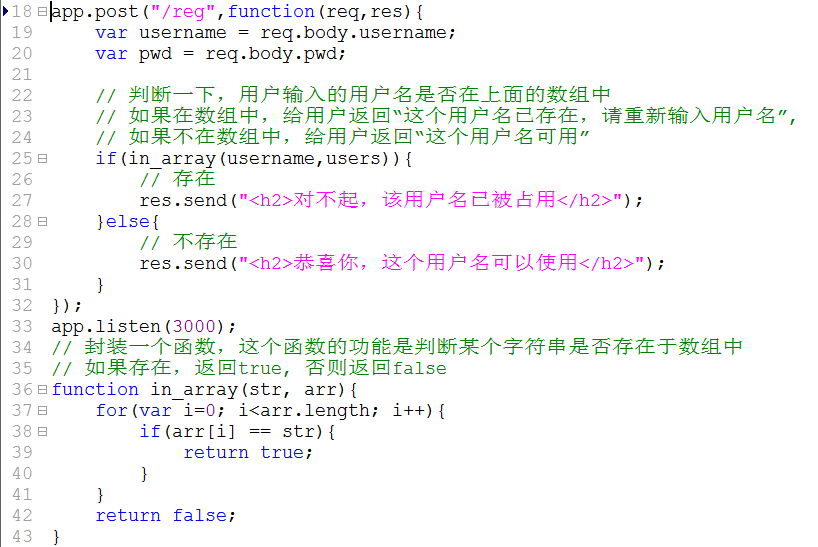
**使用express加载静态页面：**





**逻辑处理：**





# 四，早期Ajax --- iframe（局部刷新）

使用上面方式进行检测方式，不好，性能不好，用户体验也不好。需要改进，改进成当用户输入用户名后，当失去焦点，就能告诉我这个用户名是否可以。

**核心：当失去焦点时，要发送一个请求。**

**在html中哪些标签可以发送请求：**

a标签，link标签，img标签，script标签，iframe标签

**要使用上面哪一个？**

上面哪个标签能把结果返回到当前页面？ 只能使用iframe。

**什么时候发送请求？**

当失去焦点时就发送请求

**如何发送请求？**

当失去焦点时，给iframe设置一个src属性值，就可以发送请求了。

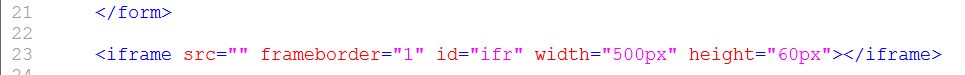
**当你发送请求时，是否空手去请求？**

不能，需要带着用户名去请求

## （1）版本一

实现步骤：

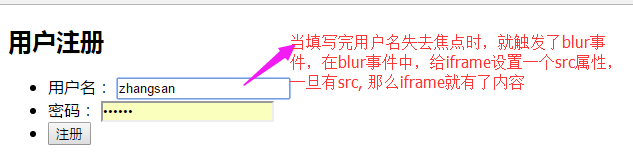
1，第一步，在form表单后面加上一个iframe：



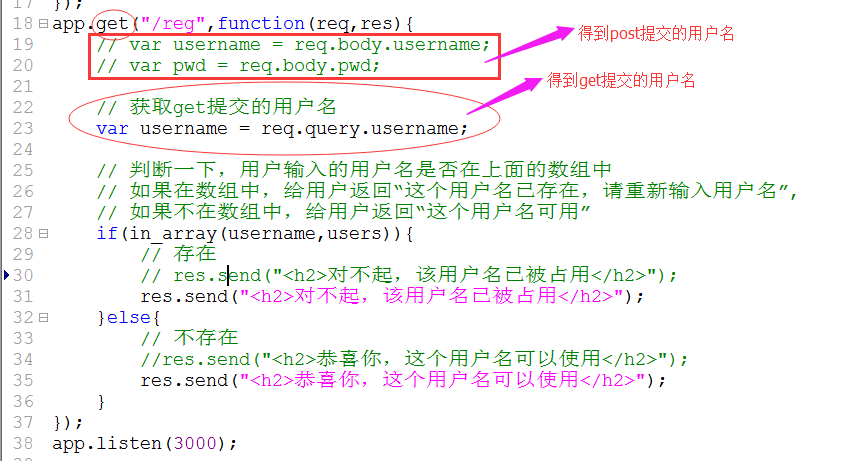
2，第二步，给文本框注册一个失去焦点事件：



3，第三步，当我们填写完用户名和密码时：



4，第四步，需要在服务端处理请求，：

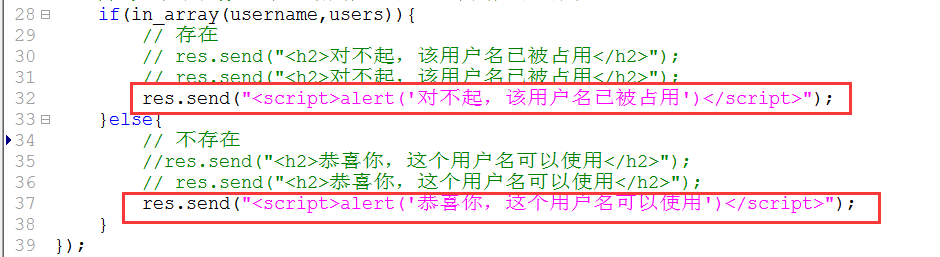


5，第五步，可以在浏览器访问了：

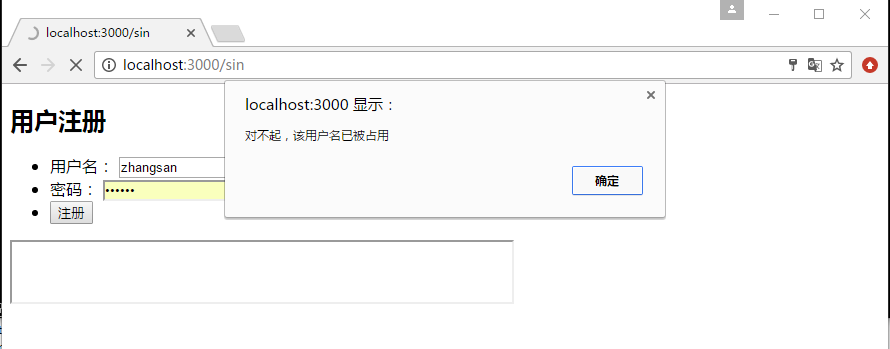


**注意问题：**

**（1）**上面服务器给客户端返回的是html代码，那么我们也可以返回js代码：



结果如下：

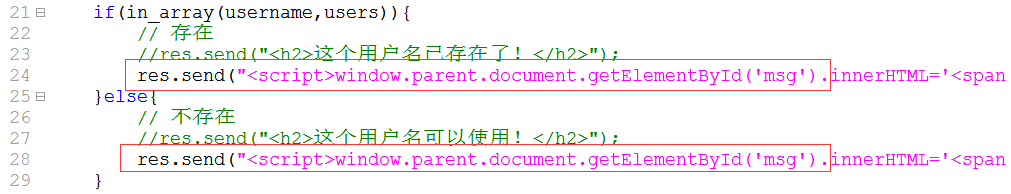


## 版本二

对于版本一，还有不太好的地方，我们一般不会放结果放到iframe中，直接放到输入框的后面。

**改进：**

1，因为结果是返回给了iframe，可以通过window.parent返回到当面页面，然后得到span标签，将结果一写在span标签中：



2，需要将iframe隐藏：



**总结：**

使用iframe实现异步刷新的有缺点：

优点：兼容性好，因为所有浏览器都支持iframe，实现也容易

缺点：功能弱

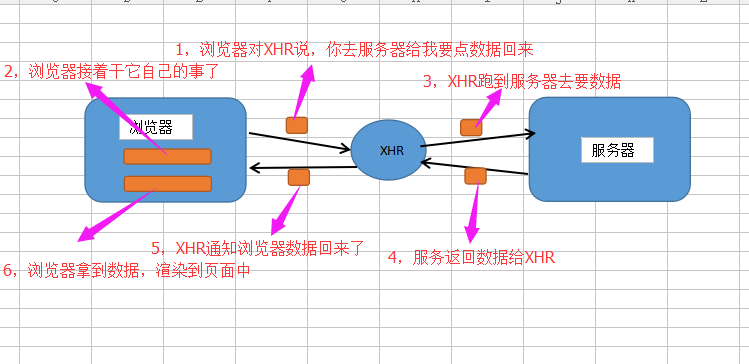
**局部刷新有什么优点：**

* 1，用户体验好
* 2，对服务器减少负担
* 3，减少带宽

对于上面iframe的缺点，解决方案：Ajax

# 五，传统Ajax

## 1，Ajax思想



## 2，Ajax核心

上面说浏览器中有一个小兵，这个小兵就是Ajax一个核心，这个小兵有一个名字：XMLHttpRequest。

**面试题：Ajax的核心是什么？**

XMLHttpRequestc对象

既然它是一个对象，那么它肯定有属性和方式：

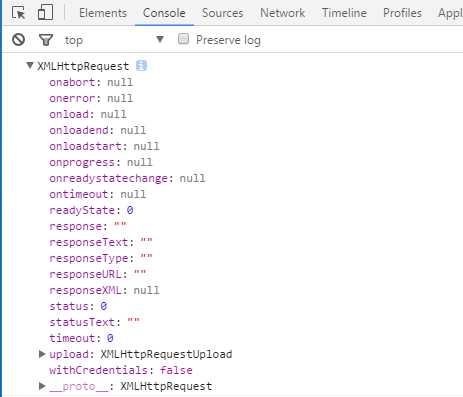




**如何去创建这个对象，如何去创建这个小兵？**



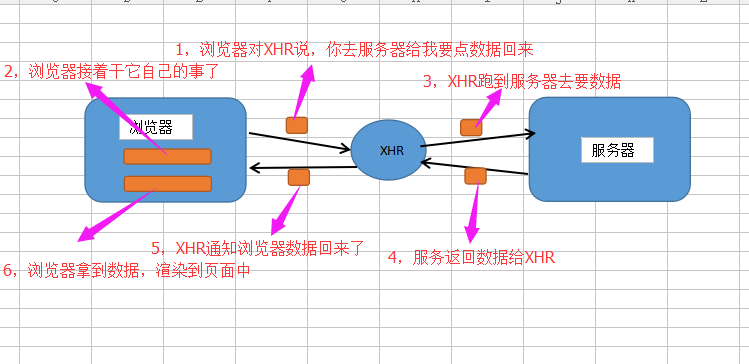
打印出这个对象，结果如下：



**Ajax到底是什么东西？**

Ajax是一堆的老技术，html, css, js, dom, bom, XMLHttpRequest.. 这里面有一个核心，核心就是XMLHttpRequest，这个核心也是一个对象，既然是对象，那么它就有属性和方法，我们可以利用这些属性和方法，向服务器发送一个请求，并且可以接收服务给你的响应数据。

**再看一个Ajax的思想：**



## 3，入门案例

案例：检验用户名是否存在。 使用Ajax

**使用Ajax的步骤：**

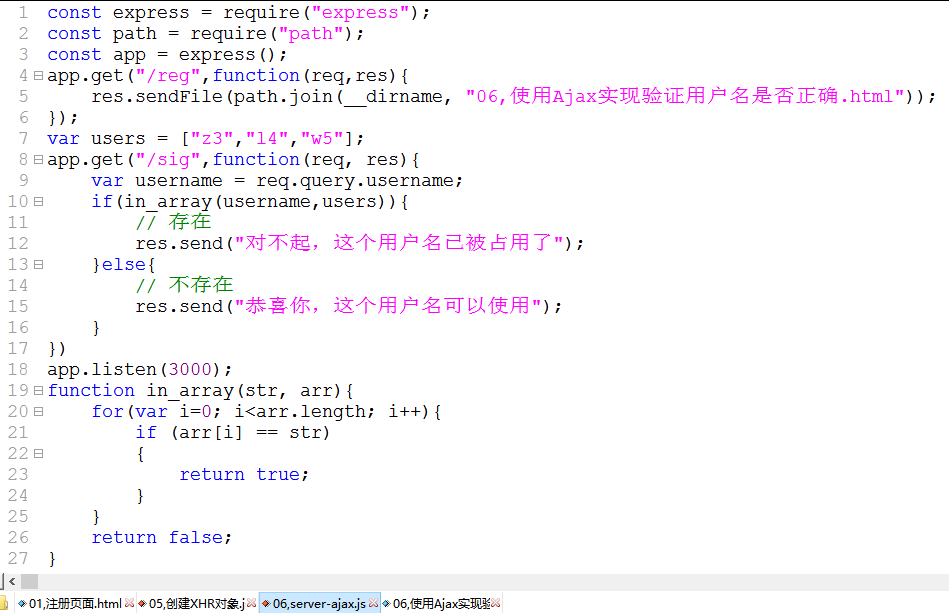
* 1，创建一个对象
* 2，打开连接
* 3，监听状态变化
* 4，发送请求

**静态页面代码如下：**





**服务端代码如下：**



## 3，XHR详解

XHR是指**X**ML**H**ttp**R**equest， 是Ajax的一个核心，也是浏览器中的一个对象。

在谷歌浏览器中打印出这个对象：



这个XMLHttpRequest在IE浏览器中不支持，IE创建方式如下：



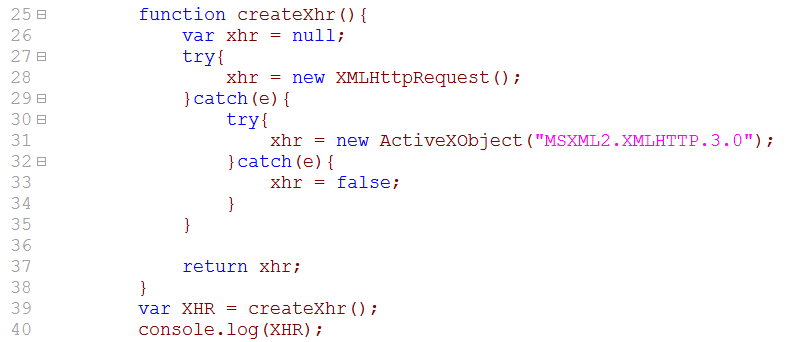
**封装兼容性的写法：**

* 1，能力检测
* 2，try-catch

## 能力检测式的兼容



## try-catch式的兼容

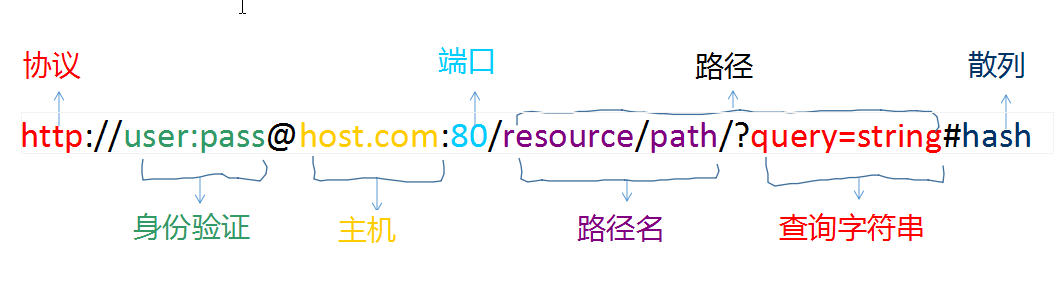


## 4，请求方式（get/post）

当我们创建出来一个XHR对象后，接下来，使用open方法与服务器之间建立连接，在建立连接过程中，我们要指定请求方法，一般有两种：get请求和post请求。

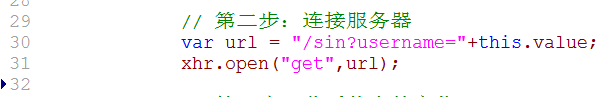
**面试题，get请求和post请求有什么区别？**

* 1，get请求数据大小有限制，一般是2k, 谷歌浏览器8k。 post是没有限制的
* 2，get请求的数据会在浏览器的地址栏中显示，post不会，所以说，get不太安全，post相对安全。
* 3，对于查询字符中，get请求中是以键盘对的形式，中间使用&进行连接。对于post是把表单中的数据以xml形式传递给服务器。

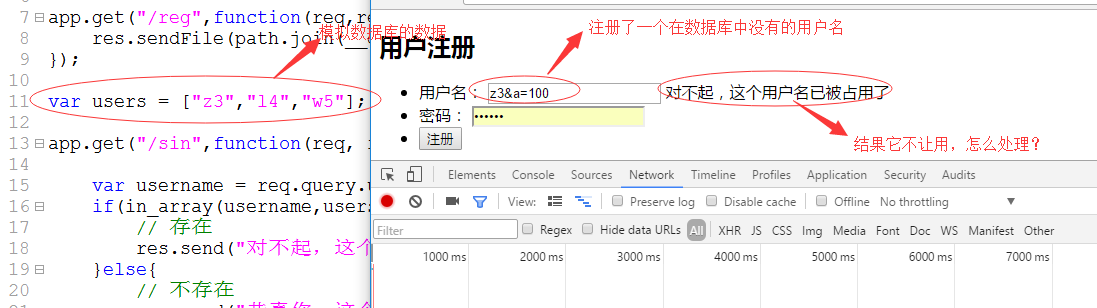


## get请求

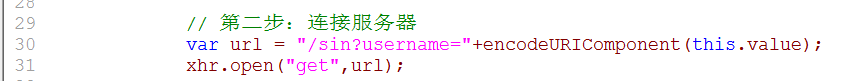
get请求只需要在open方式中写入第一个参数就可以。第一个参数后面是请求的url。 在url中有请求的数据，请求的数据是以键值对的形式存在的，中间使用&。



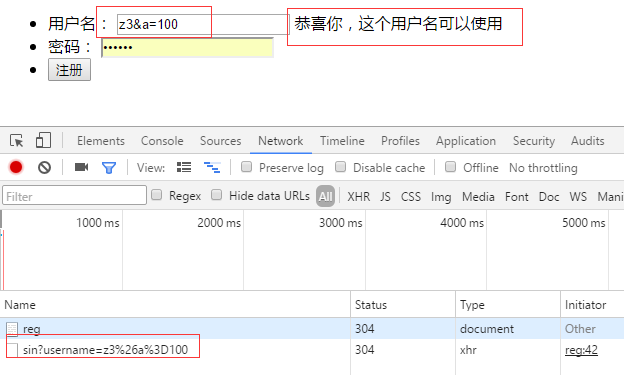
但是可能会遇到下面的问题：



针对上面的问题，需要对用户输入的内容进行一个编码处理 --- encodeURLComponent



结果：



都是以%十六进制的形式进行编码。编码后，不需要解码，因为浏览器会自动解码。

通常我们都是需要进行编码的，因为如果不编码，对于中文，在IE浏览器中是以乱码来显示的。

## post请求

将open这个方法的第一个设置为post就可以了。大小写都可以。

post请求相对于get请求变了三个地方：



说明：

1，对于post请求，需要将open方法的第一个参数设置为post

2，对于post请求，不需要查询字符串，数据是放在send方法中

3，需要设置一个头信息，从而使得以表单的方式来提交数据

4，对于中文件和特殊符号也需要进行编码。

**服务器对应的代码如下：**



**有get请求，也有post请求，那么哪种用的多，哪种用的少？**

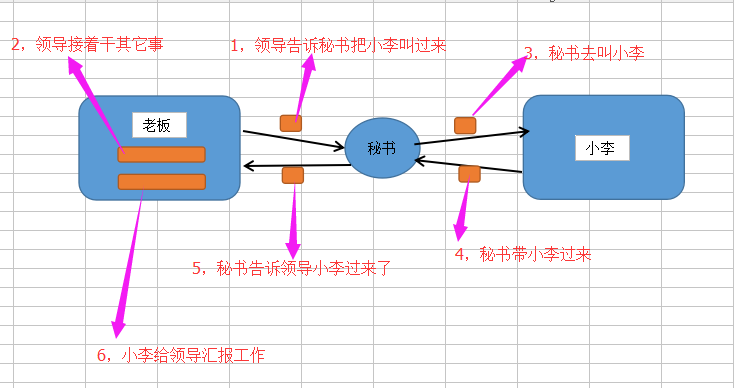
一般情况下，很少使用post请求，大部分都使用get请求。

## 5，异步原理

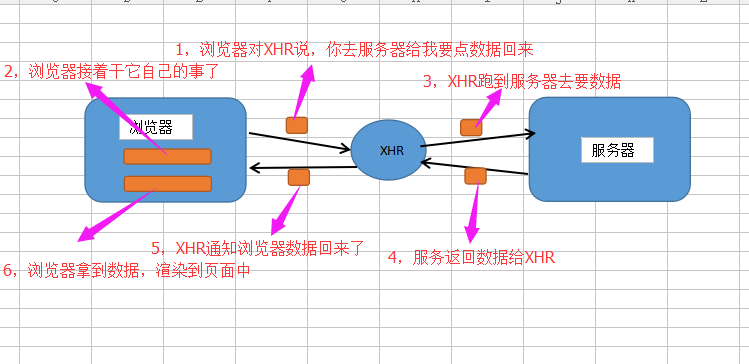
**同步**： 提交请求--->等待服务器处理---->服务器处理完毕后返回， 这个过程会造成阻塞，也就是，阻塞模式，同步，可能会一直处于阻塞模式，后面就不能干其它事。

**异步**：请求通过事件触发-->服务器处理请求---> 处理完毕。 非阻塞模式。在等待处理的过程中，可以做其它事。

**现实生活中：**



**Ajax思想：**



**Ajax只能异步吗？**

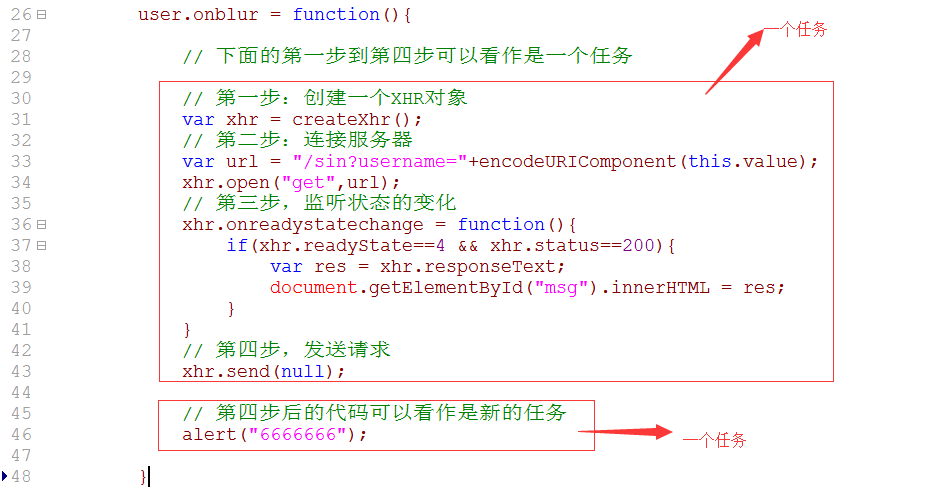
答案：不是的，也可以同步

**如何设置Ajax同步？**

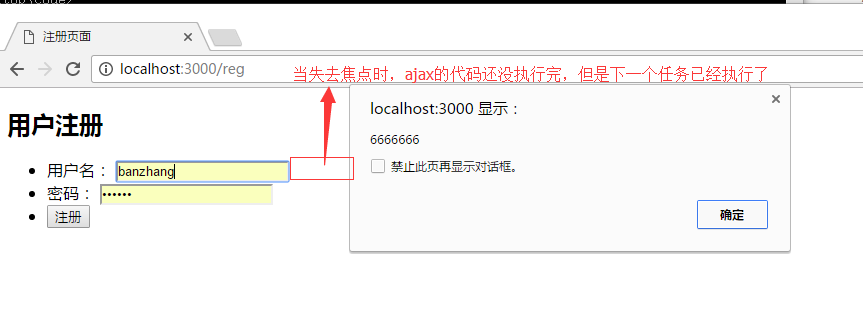
给open方法设置第三个参数，是一个布尔值：

* true 异步， 默认
* false 同步

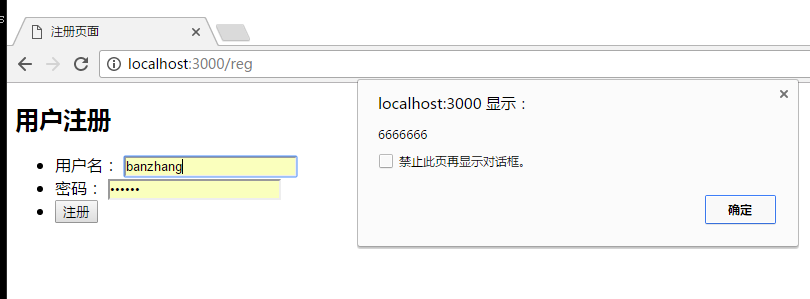
代码如下：



如果是异步：



如果是同步： 也是先执行了下一个执行

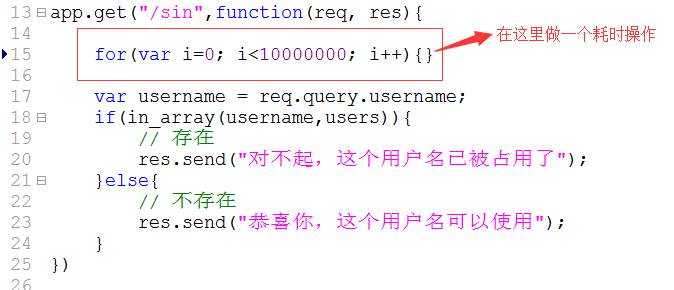


原因是谷歌浏览器的原因。

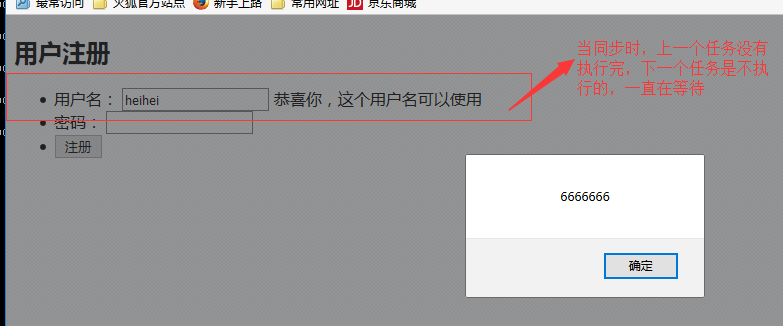
使用火狐浏览器看一下：



当失去焦点时，几乎是同时出来了，此时代码是同步的，因为速度太快了，我们想办法把速度减下来：



对于同步，上一个任务没有执行完，下一个任务不执行：



对于异步，上一个任务是否执行完毕和下一个任务是没有关系：



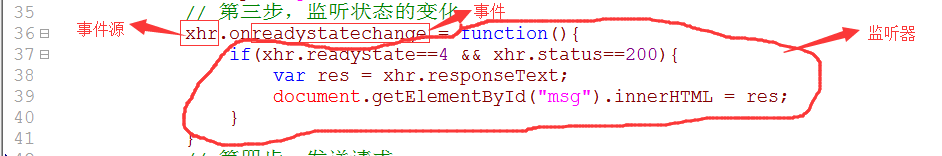
考虑一个问题：

**对于同步和异步哪一个用的多，哪一个用的少？**

在开发中，90%都是异步操作。但是也有用同步的，下一个任务必须依赖于上一下任务时，那么必须要使用同步，。

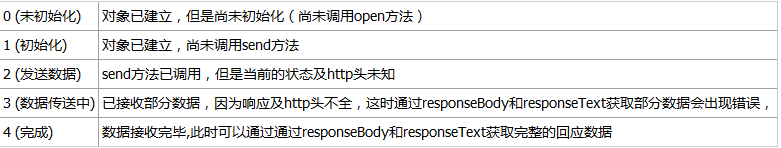
## 6，状态说明

监听状态变化：



**readyState:**

返回XMLHTTP请求的当前状态



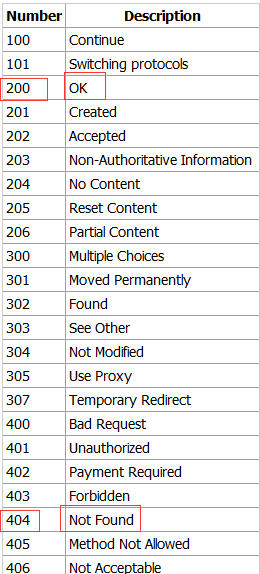
代码演示：



xhr.readyState==4 表示 数据接收完毕

**status:**

返回当前请求的http状态码



再看一个代码：



到目前为止，上面的代码中不能疑问。

## 7，XML介绍

### ，XML的基本概念

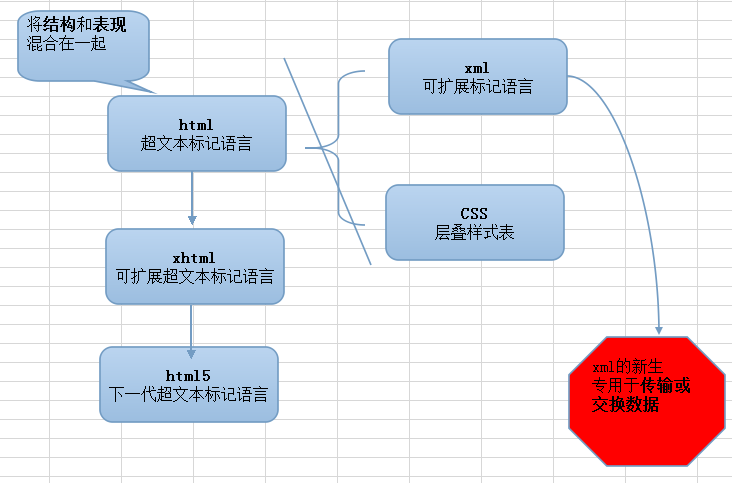
**1，什么是XML？**

e**X**tensible **M**arkup **L**anguage 可扩展的标记语言 ， 所谓的可扩展就是自己写标签。

**2，学过的标记语言有哪些？**

html, xhmtl, html5, xml

**3，XML历史？**



在XML产生初期是为了给页面提供结构的，但是后来证明，使用XML给网页提供结构，不合理，这个时候，XML没有的抛弃，它有了一个新有功能，就是用于数据的传输。

**4，XML具体是什么？**

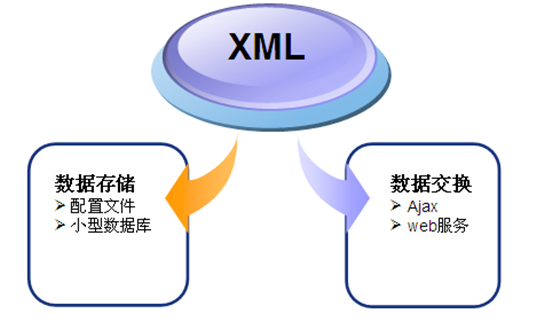
* XML 指可扩展标记语言，是独立于软件和硬件的信息传输工具。
* XML 是一种标记语言，很类似 HTML。
* XML 的设计宗旨是**传输**数据，而非显示数据
* XML 标签没有被预定义。您需要自行定义标签。
* XML 被设计为具有自我描述性。
* XML 是 W3C 的推荐标准。

### （2），XML的作用

**目前有两大作用：**

* 1，用来数据传输
* 2，用来数据存储

如下图：

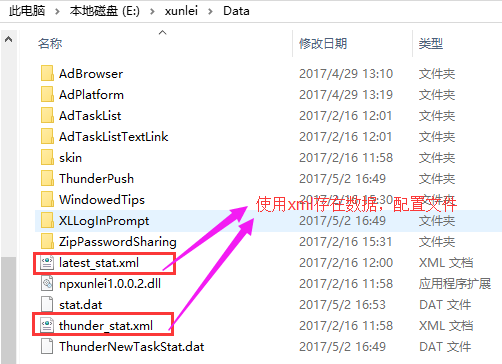


**数据存储：**

1，配置文件，如QQ, 迅雷....这些软件的配置文件通常是就xml

2，小型数据库，对于一些通常不变的数据，可以使用xml文件存储

迅雷的配置文件如下：



**数据传输：**

Ajax: 异步的JS和XML

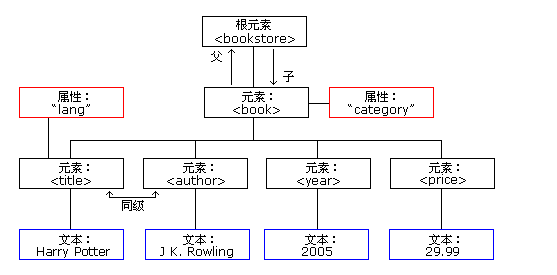
Web service: web服务

### ，XML的结构

使用xml对数据进行描述，如下：

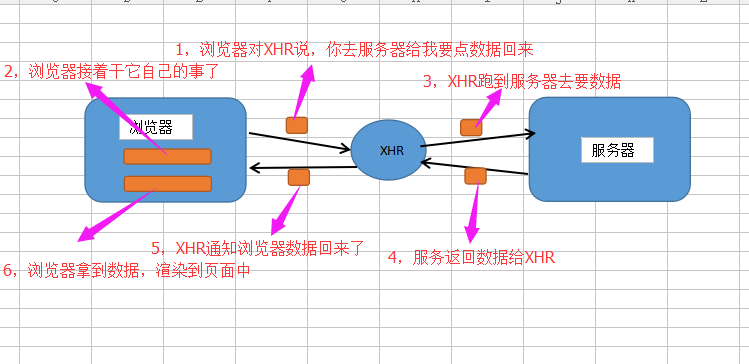


**也对应了一个树模型：**



## 8，数据返回的格式

再回顾一个Ajax的思想 ：



Ajax拿到的数据是什么样的格式的？

当数据从服务器返回，并解析完毕时，可以使用了，如何接收返回值？

**有两种形式：**

* 1，XML形式
* 2，文本形式

### XML

xml是早期的一种比较流行的**数据交换**格式，但是这种格式，处理起来比较麻烦，所以在Ajax中，几乎已经不用这种格式了，有另一个数据交换格式取代了XML, 就是JSON，JSON是轻量级的，处理起来也十分方便，有些地方还是使用XML的。

前面我们可以通过，拿到服务器给我返回的文本形式的数据，如何拿到服务器给我返回的XML形式的数据，可以通过：xhr.responseXML

我们学习重点放在文本形式的数据上面。

### 文本格式

我们可以通过xhr.responseText接收服务器给我返回的文本格式的数据，这些文本格式的数据又分成了三个情况：

* 1，纯文本
* 2，HTML文本
* 3，JSON

**纯文本: 直接返回一个字符串**



**HTML文本**



对于上面的纯文本和HTML文本，在客户端可以使用innerHTML来处理：



**JSON格式的字符串**

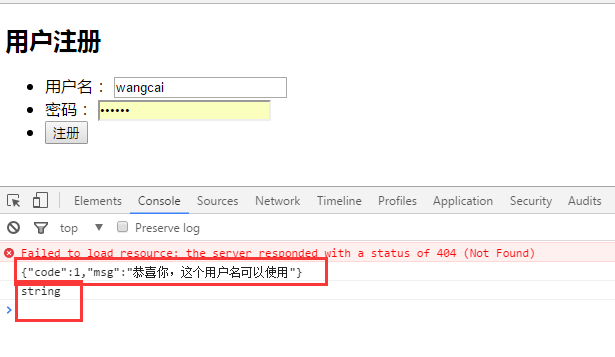
在服务器端编写如下代码：



在浏览器端看一个服务给我们返回的数据，以及它的类型：

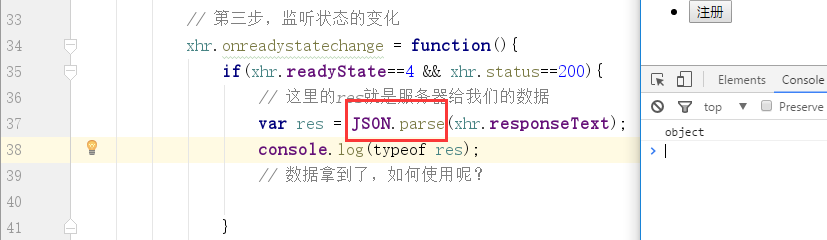


结果：

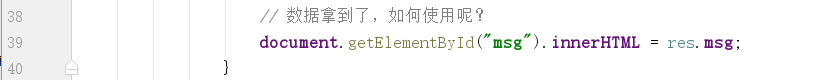


如何去使用：

1，需要将res转换成对象：



2，转换成对象后，可以通过点语法来使用



注意点：服务器可以直接通过res.json给我们返回一个json串，在浏览器端拿到的依然是字符串，也需要转换成对象。



# 六，JQ中的Ajax

对于Ajax的操作，JQ也封装了一套自己的方法，让开发人员使用。

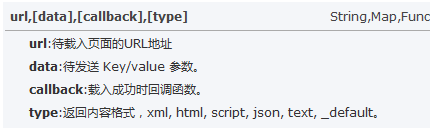
JQ中的$就是一个JQuery对象，这个对象中有很多属性和方法。

## $.get()

$.get()说明：



参数的含义：



代码演示$.get的使用：

**第一步，引入JQ:**

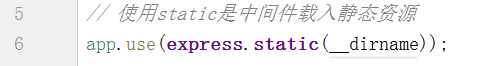


**第二步，表单：**



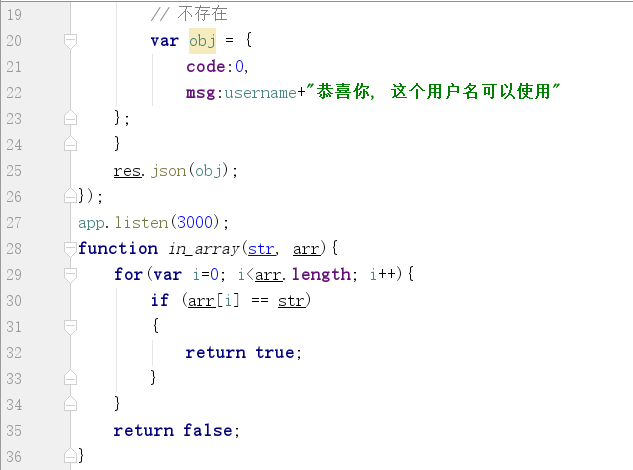
**第三步，创建一个http server服务**

注意点，需要加载静态资源，否则JQ不能用：

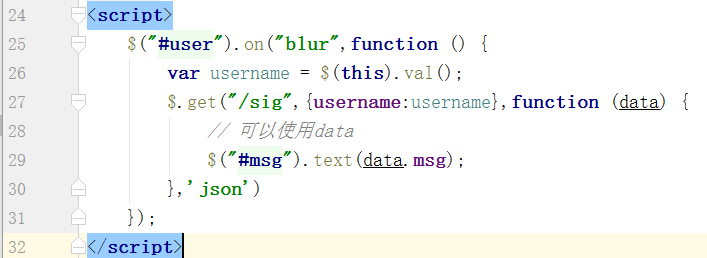


完整代码如下：





**第四步，使用$.get()方法如下：**

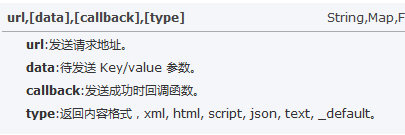


## $.post()

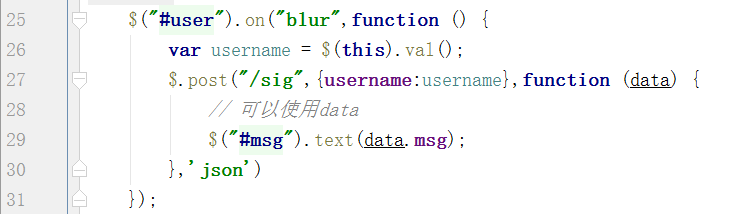
$.post()和$.get()类似，使用如下：



参数说明如下：



代码使用：



**注意细节：**

1，$.get()和$.post都是异步的，没有同步。

2，和原生的相比，不需要设置头信息。

3，原生的需要进行编码处理，jq不需要编码处理

4，原生的对于XHR还需要做兼容性处理，jq也不需要

## $.ajax()

尽管$.get()或$.post()已经很强大了，有时候，仍然不能满足，比如我要用同步，这个时候就要用到$.ajax()。

用法：



settings是以对象的方式进行设置的。

**参数说明：**

* url，请求的目标URL，默认为当前页面
* async，是否是异步请求，默认为true，异步
* **type,请求类型，可以为POST或GET,默认为GET**
* data，发送到服务器的数据，可以是字符串或对象类型
* **dataType，指定返回数据类型xml/html/script/json/text/jsonp**
* success，指定请求成功后执行的回调函数。同时携带参数，表示返回的内容
* error,请求失败时执行的回调函数
* timeout，设置请求超时的毫秒值

代码演示：



注意细节：

1，$.get()或$.post也是在$.ajax()的基础上又进行了一次封装。

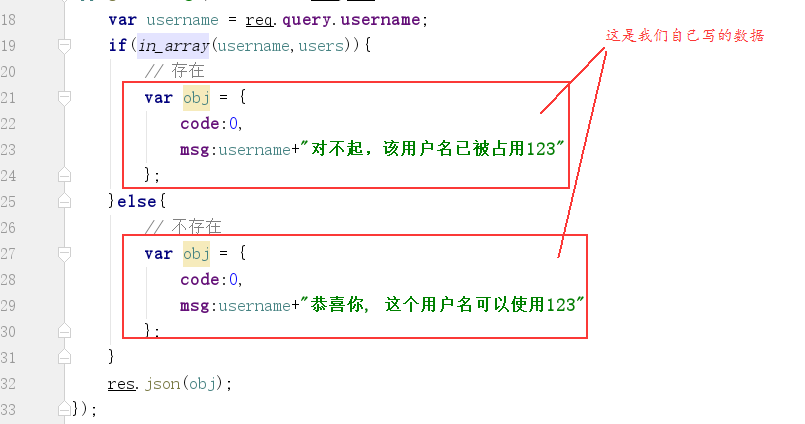
# 七，案例练习

## 1，手机归属地查询

### 极速数据

地址：<https://www.jisuapi.com/>

我们在前面，数据是我们自己写的，思考，别人有没有写的数据，我们拿过来直接用。



有别人写的数据，我们直接用，比如可以使用极速数据这个网站提供的数据，怎么用？

* 1，注册
* 2，登录
* 3，申请数据
* 4，使用

**注意细节：**

1，当注册成功后，有一个appkey， 这个appkey是有用。

2，申请数据时，会送1000，如下：



3，对于接口的理解，发送一个请求，服务器给你一个响应，响应的内容我们就可以使用了：



### 跨域

**什么是跨域？**

比如: 假设我访问a站点，在a站点得到一个页面，在a站点要去访问b站点的资源，这个时候就会产生跨域，所谓的跨域，就是跨域名访问。

对于跨域，浏览器是有安全限制，默认情况下我们不能跨域。

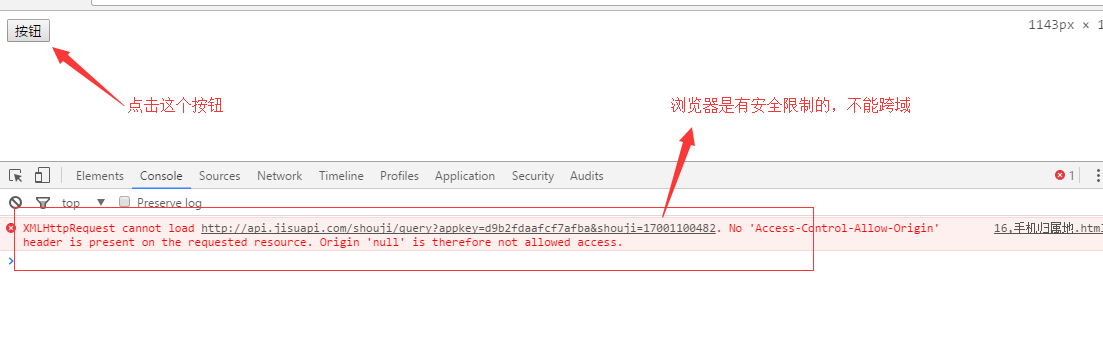
写代码演示：

我写一个页面，去访问另极速数据这个网站中的数据：

用原生的方式，发出请求，此时这个请求是跨域的：

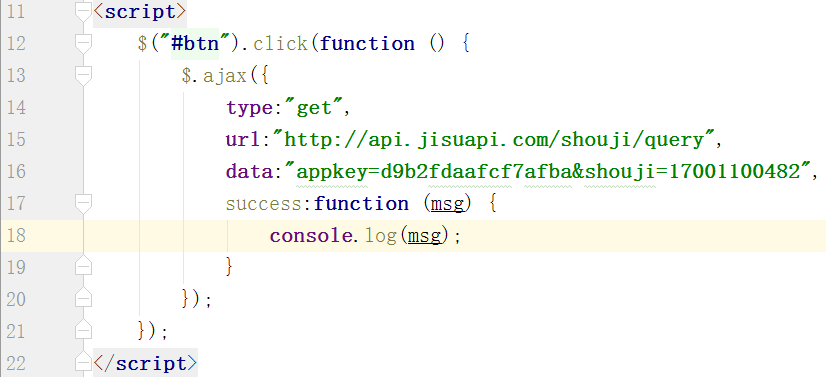


结果：

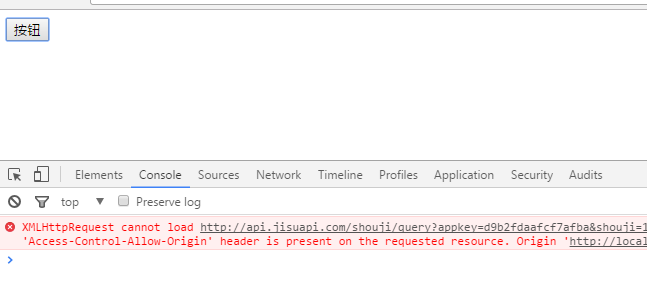


### 跨域的解决方案---JSONP

我们使用JQ封装好的Ajax, 代码如下：



也不能实现跨域，结果如下：



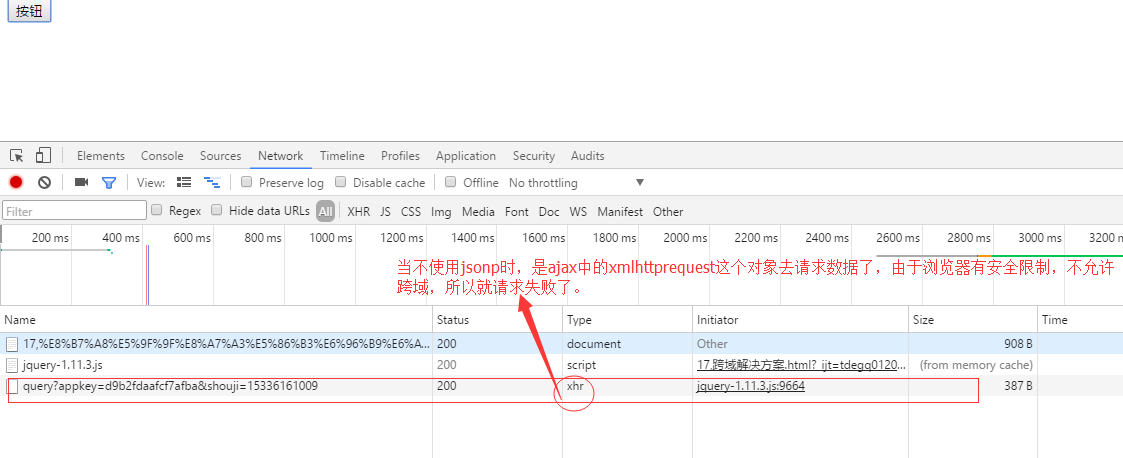
**怎么让它跨域呢？ 使用 jsonp**



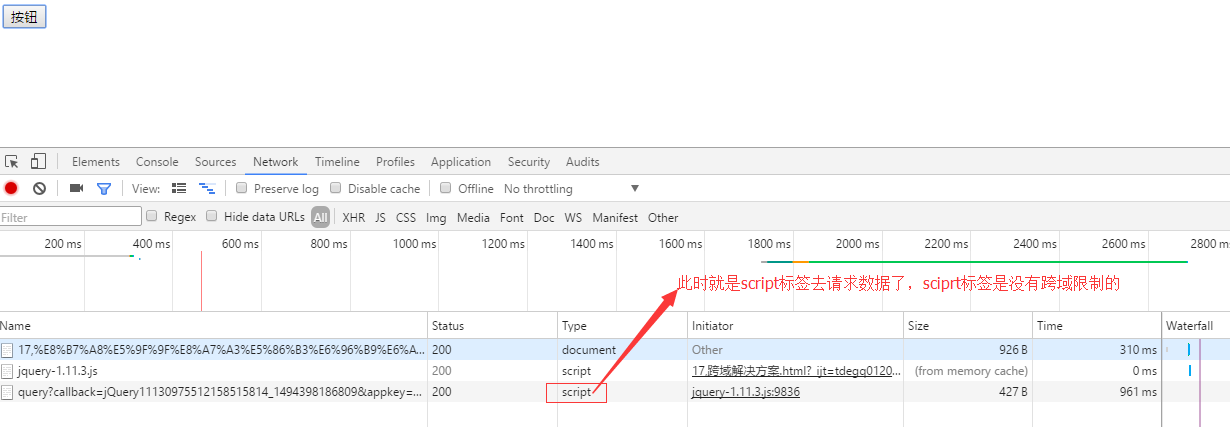
这样我们就可以使用人家给我们返回的数据了。

**JSONP是什么东西？**

当我们不使用jsonp是，是ajax去请求数据了：



当使用jsonp时，是script去请求数据了：



jsonp的本质就是通过script标签去请求数据，和ajax就没关系了。

**使用jsonp只能解决get请求的跨域，因为script标签中的src请求就是get请求。也就是说，通过JQ只能解决get请求的跨域。post请求的跨域，需要在服务器进行设置。**

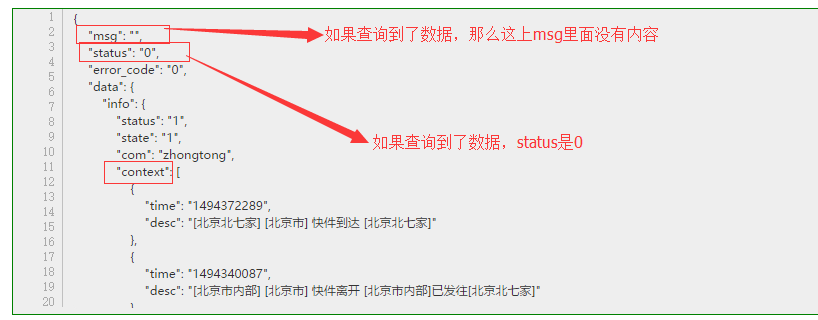
### 手机归属地查询



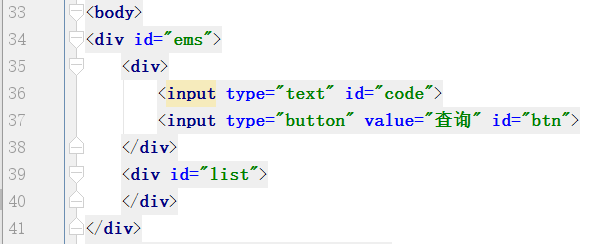
## 2，快递查询

对于服务器给我返回的信息，如下 ：

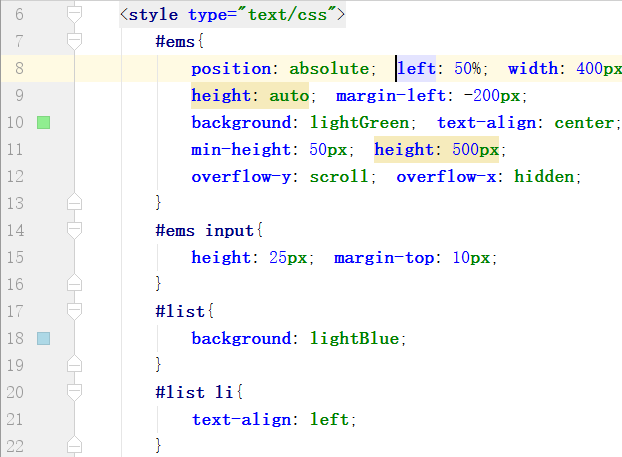




### （1）结构



### 样式

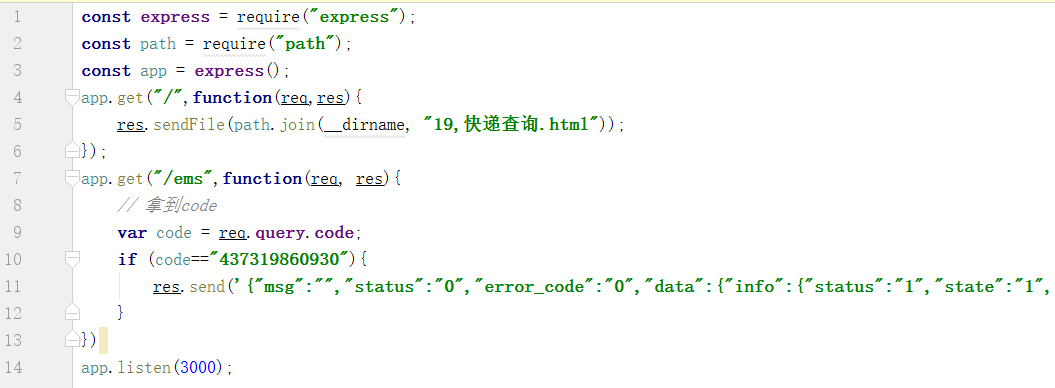


### Ajax





### 服务器



## 4，省市区三级联动

实现目标：



**实现步骤：**

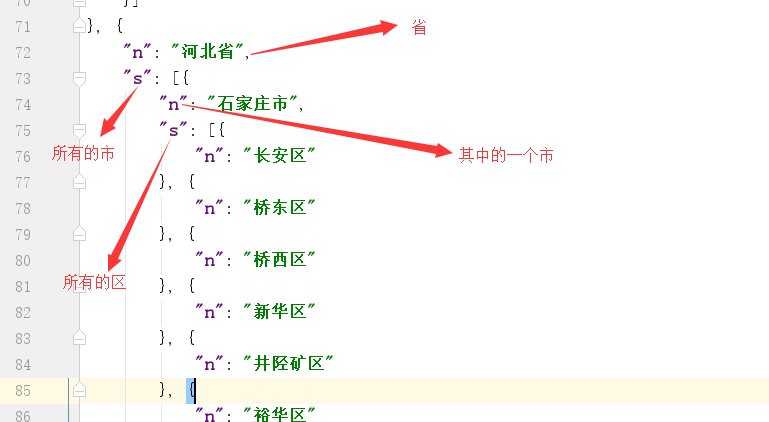
分析：

1，当选择时，整个页面没有刷新，它是异步的，我们使用ajax实现。

2，什么时候发送请求？ 在change事件中发送

3，很多省和市都是数据，那么数据存储在什么？ 数据库可以，json文件，也可以。就用json文件存在。

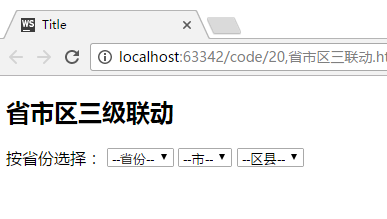
4，我们在josn文件中，写入了我们需要的数据，是一个数组，在这个数组中，放了很多对象，



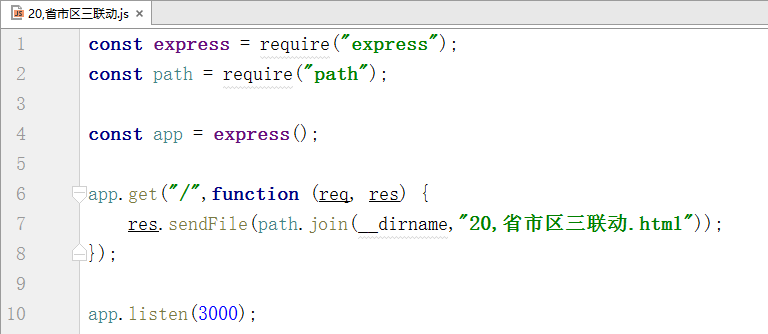
### 第一步，创建一个html页面，如下：



**结果：**



### 第二步，创建http对象，提供web服务



### 第三步，在html页面，通过ajax获取所有的省份



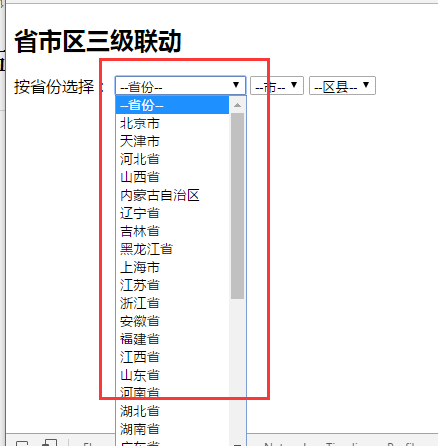
### 第四步，在js中，实现/province路由，然后读取json文件，并处理，返回结果



### 第五步，回到第三步，尚未处理完的地方



**结果：**



### 第六步，当选择一个省份时，需要获取对应的市(区)

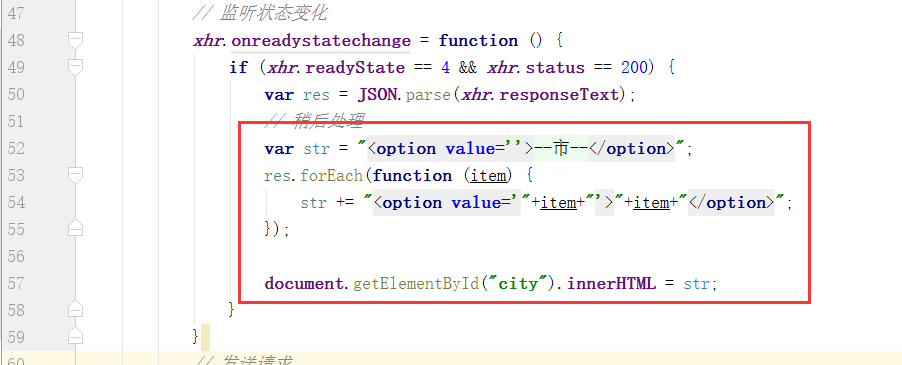
这里肯定要用到事件，用change事件，在chang事件中发送ajax请求。



### 第七步，编写服务端的代码，获取指定省份的市区并返回



### 第八步，在客户端，将返回的市区添加到对应的select中



### 第九步，对市区的下拉列表注册change事件，发起ajax请求



### 第十步，编写代码，获取指定市区下面的所有区县



### 第十一步，将返回的区县数据，添加到对应的下拉框中

