Ajax（2）

# 一、昨天内容回顾

1、Ajax对象的创建

var xhr = window.XMLHttpRequest ? new XMLHttpRequest() : new ActiveXObject('Microsoft.XMLHTTP');

2、使用Ajax的使用步骤

1、创建ajax对象

2、绑定监听事件（监听ajax执行的状态，当readyState等于4的时候，用responseText来获取文本内容，用responseXML获取XML类型的数据）

xhr.onreadystatechange = function(){

if(xhr.readyState == 4 && xhr.status == 200){

var result = xhr.responseText;

}

}

3、定义ajax请求（请求方式，请求地址，请求的参数，是否是异步）

xhr.open('get', '1.php?id=1', true);

4、发送请求（如果是post请求，这里指定请求参数）

xhr.send(null); // 如果是post请求，xhr.send(请求的参数);

3、如何处理缓存

让每次请求的地址不同。

xhr.open('get', '1.php?t='+Math.random(), true);  
xhr.open('get', '1.php?t='+new Date().getTime(), true);

4、如何对请求的参数进行编码

encodeURIComponent('lisi&age=20');

5、处理XML格式数据

1、服务器端要指定header('content-type:text/xml; charset=utf-8);

2、浏览器端接收xml类型的数据用responseXML

3、浏览器端接收到的数据可以看做是document来使用。

6、处理json数据

1、服务器端，返回的数据要用json\_encode();

2、浏览器端用responseText来接收

3、浏览器端接收完数据要用JSON.parse()转化成js的数组或对象。

# 二、优化省市县三级联动案例

1、优化代码

1、优化Ajax请求代码

首先封装一个用于ajax请求的get方法。该方法只能处理json类型的数据。



应用get方法，优化案例代码：

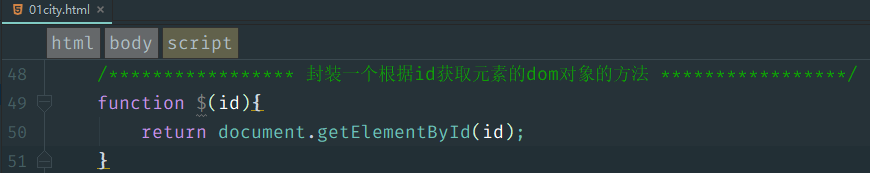




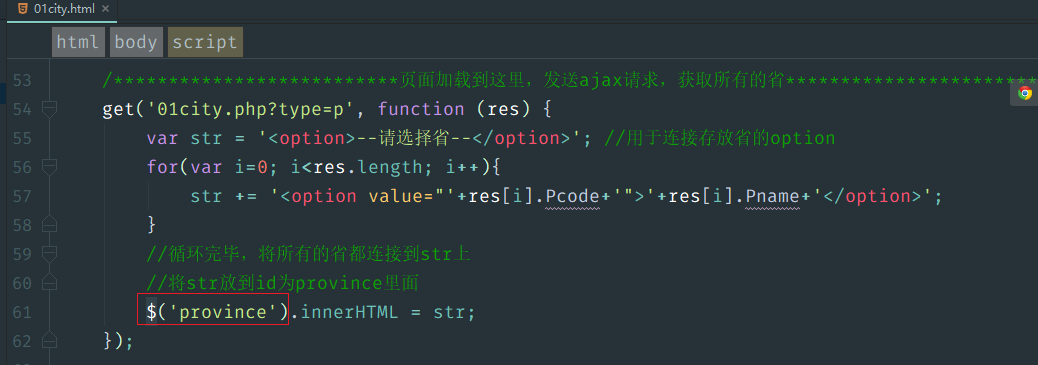


2、优化获取dom对象的代码

封装一个根据id获取元素的dom对象的方法



案例中能够使用$函数的位置，应用一下：



# 三、今日目标

1、知道FormData的应用场景

2、掌握如何使用FormData获取表单数据

3、能够使用XMLHttpReqest对象中的upload属性完成文件上传时的进度条效果。

4、知道什么是跨域请求

5、能够通过代理方式解决跨域问题

6、能够通过JSONP技术完成跨域请求

7、能够通过CORS方式完成跨域请求

# 四、FormData

1、使用FormData获取数据

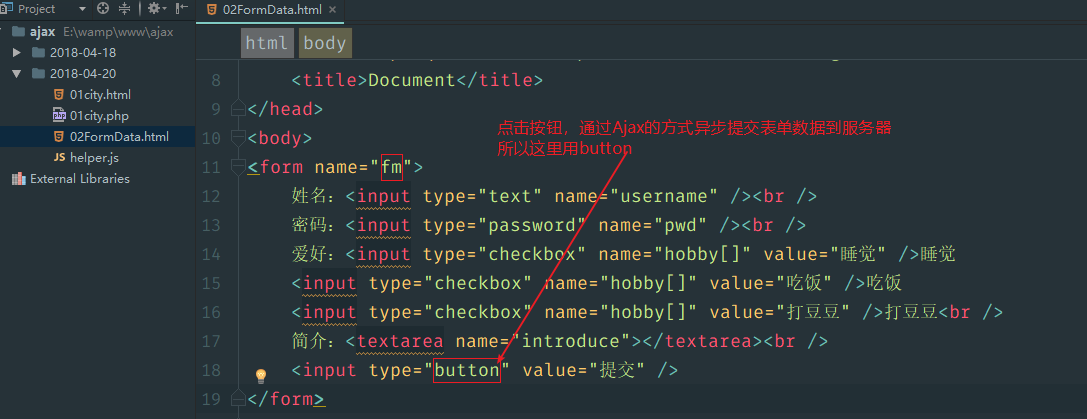
本节课我们将学习以下内容：

①、使用FormData完成表单数据的收集

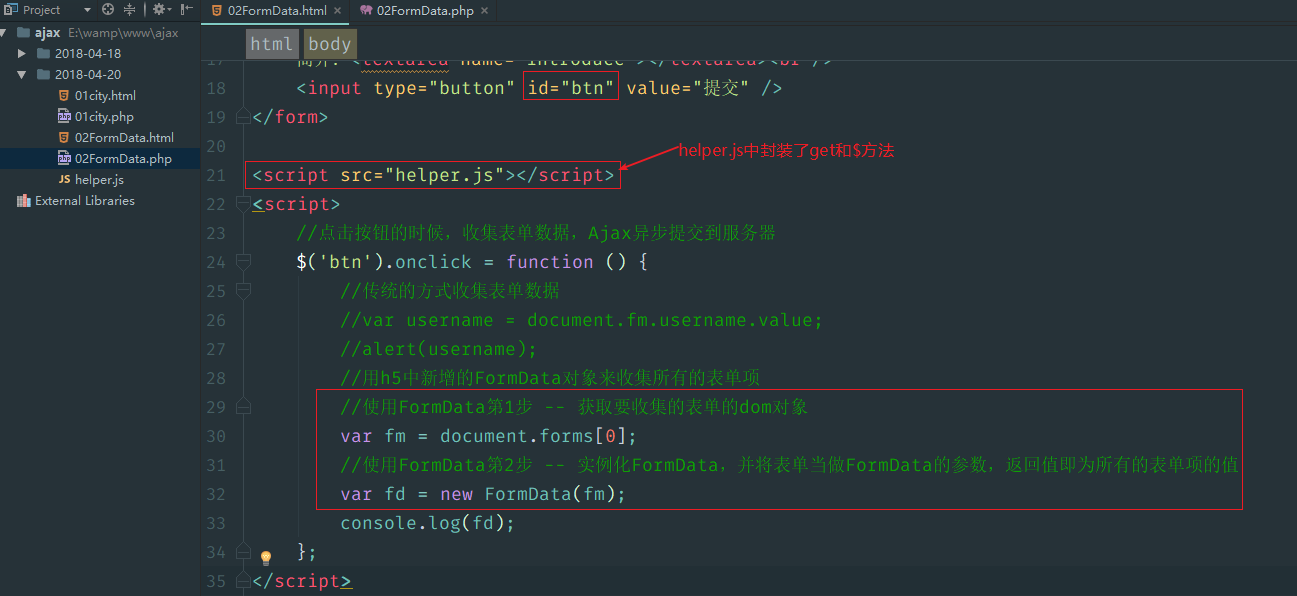
②、将收集到的数据通过ajax对象发送给服务器

FormData：字母意思就是表单数据，这是h5中新增的一个内置对象（构造器），它可以获取任何类型的表单项的值，适用于获取大量的表单项的值。常用于发送Ajax请求。

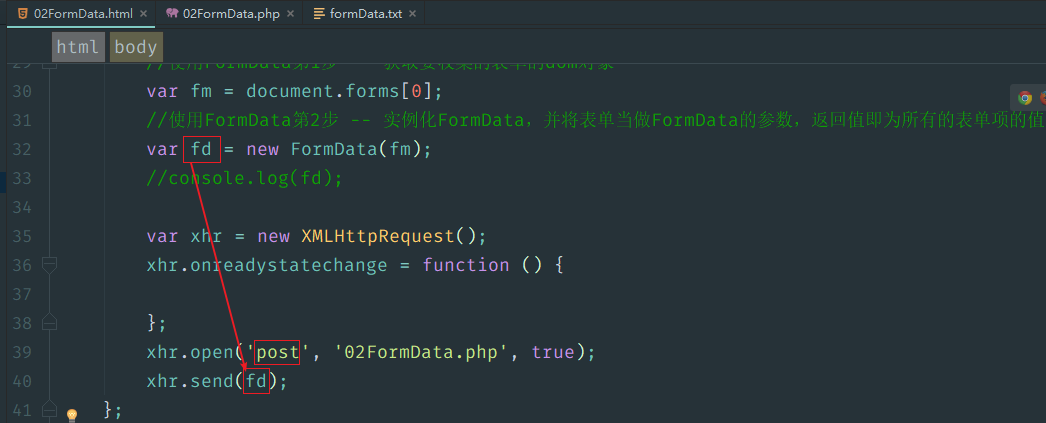
界面：



使用FormData收集表单项的数据：



通过Ajax发送post请求，将FormData获取的数据提交到服务器：



服务器保存提交过来的数据，并保存：



测试，发现成功了。

**总结**：

1、使用FormData可以一次性收集到表单的所有内容

2、使用FormData收集的数据，通过Ajax的post请求发送的时候，不用指定header。

3、print\_r($\_POST, true); true表示将打印的结果当做返回值返回

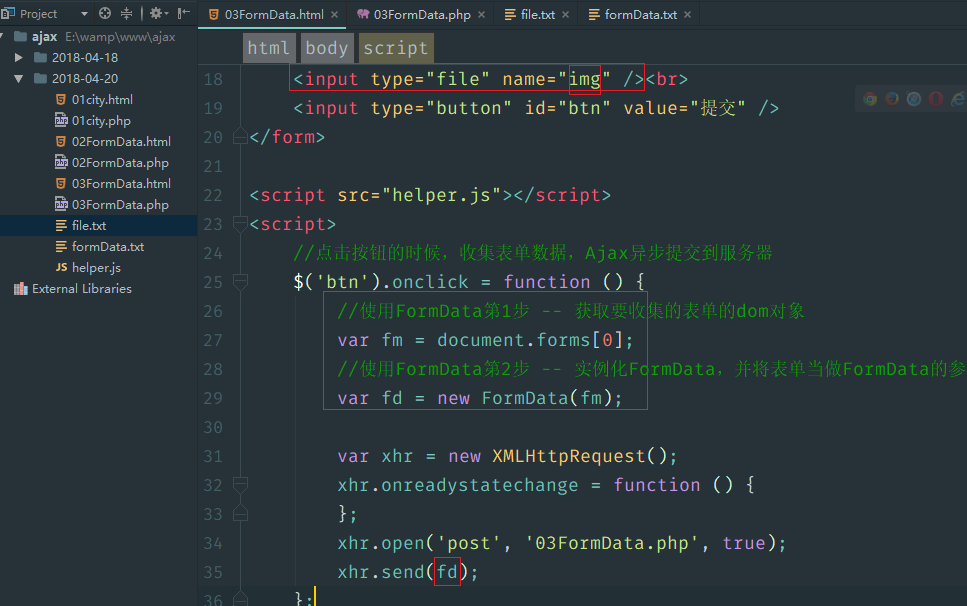
4、h5是IE9才支持部分，IE10才完全支持h5，所以FormData在IE9+版本的浏览器中才能够使用。

2、使用FormData完成文件上传

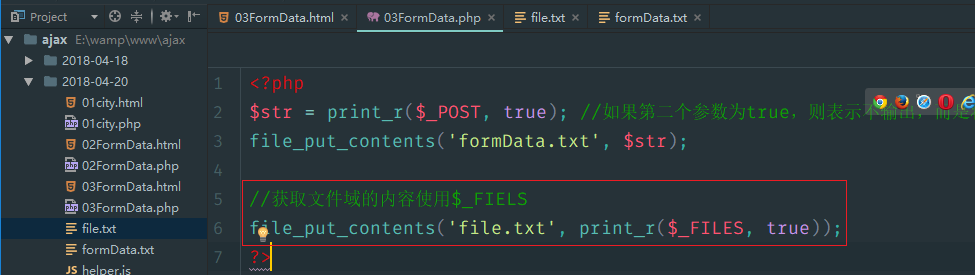
本节课我们将学习以下内容：

①、配合FormData，完成文件异步上传工作

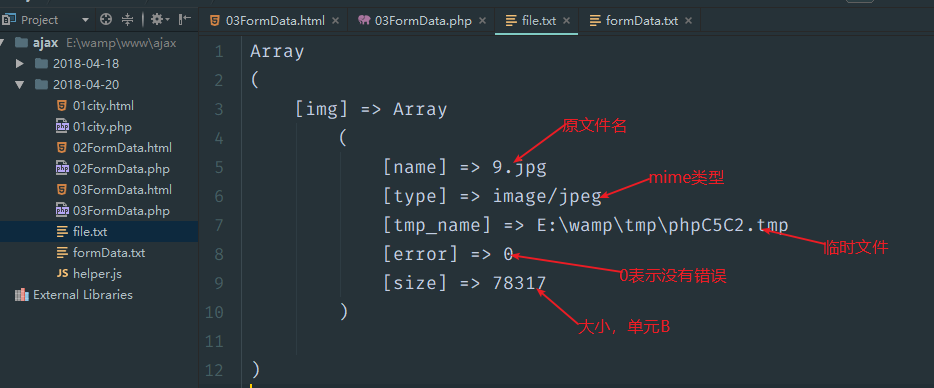
HTML代码：



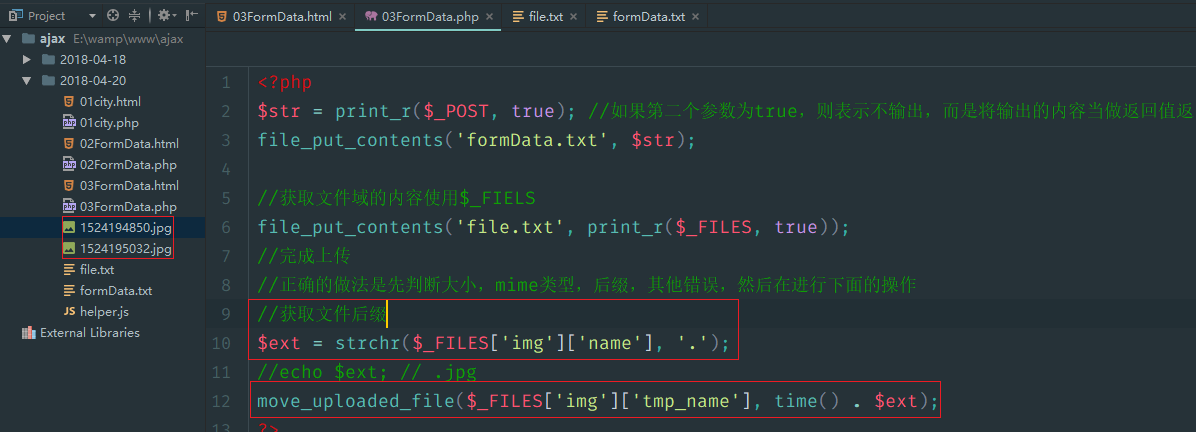
PHP代码：



检测，可以拿到上传文件的信息。



PHP代码，完成文件上传：



3、使用FormData上传大文件（上传进度条）

本节课我们将学习以下内容：

①、完成异步上传中，用进度条展示上传进度。

上传大文件，应该先配置php.ini。

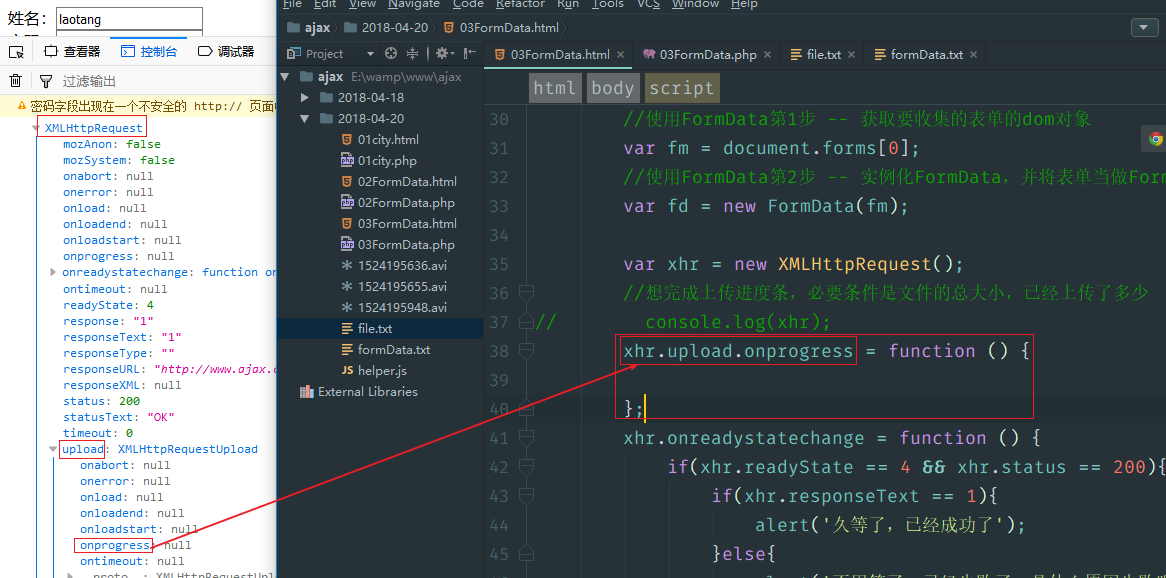
post\_max\_size：表示允许post提交的内容的最大值

upload\_max\_filesize：表示允许提交的附件的最大值。

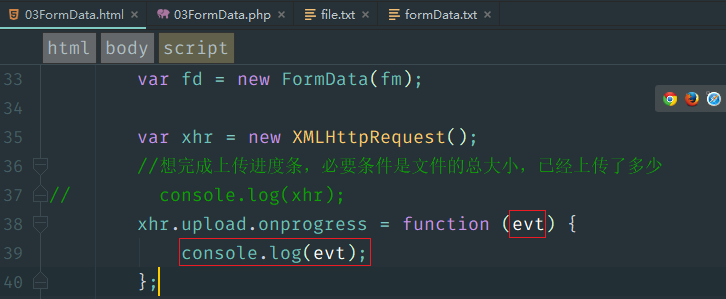
max\_execution\_time = 30：最大响应时间

**重启Apache**。

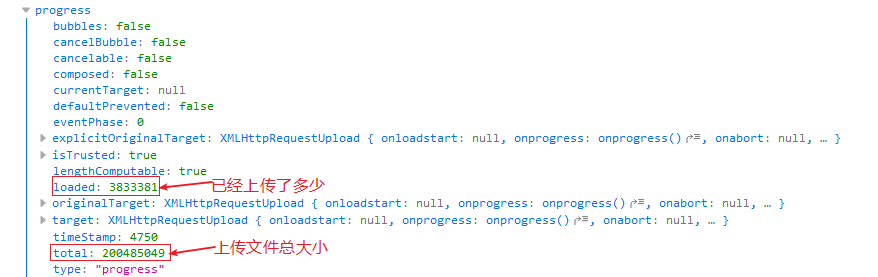
要完成上传进度条的制作，必须给上传进度事件绑定处理函数：



在上传进度事件中，打印事件对象

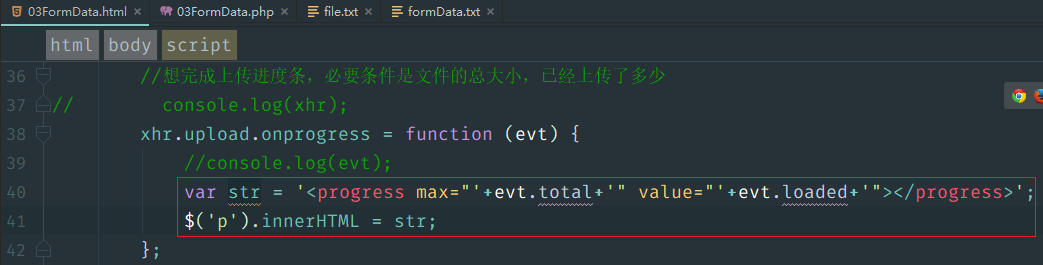


浏览器查看到的效果：



每个事件对象中的loaded表示已经上传了多少，total表示文件总大小。

下面就可以设置上传进度条。



最终效果：



# 五、解决跨域问题

Ajax是不允许跨域请求的。

1、什么是跨域请求

本节课我们将学习以下内容：

①、什么是同源政策。

②、同源政策有哪些限制

③、什么是Ajax跨域请求。

1、什么是同源政策

1995年，同源政策由 Netscape（网景） 公司引入浏览器。目前，所有浏览器都实行这个政策。

最初，它的含义是指，A 网页设置的 Cookie，B 网页不能打开，除非这两个网页“同源”。所谓“同源”指的是”三个相同“。

* + 协议相同（http https）
  + 域名相同
  + 端口相同（默认80端口）

2、同源政策有哪些限制

随着互联网的发展，“同源政策”越来越严格。目前，如果非同源，共有三种行为受到限制。

* Cookie无法读取。
* DOM 无法获得。
* AJAX 请求无效（可以发送，但浏览器会拒绝接受响应）。

3、什么是Ajax跨域请求

在Ajax请求中，只要违反了同源政策的请求，都属于跨域请求。



2、解决跨域问题

本节课我们将学习以下内容：

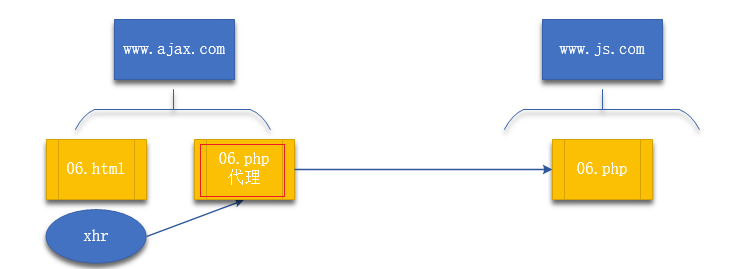
①、如何使用代理方式解决跨域问题

②、如何使用jsonp技术实现跨域请求

③、如何使用CORS技术实现跨域

1、使用代理的方式

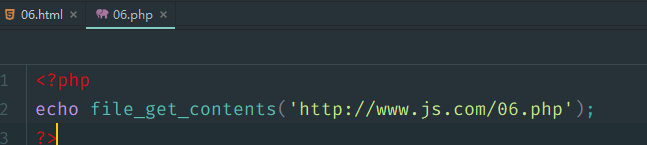
直接发送Ajax请求到其他域不允许，但是可以先发送Ajax请求到本网站中的一个php文件，然后让这个php文件去访问其他域的内容，那么这个php文件就可以认为是一个代理文件。



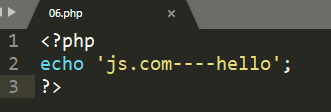
www.ajax.com下面的html文件：



www.ajax.com下的PHP代理文件：



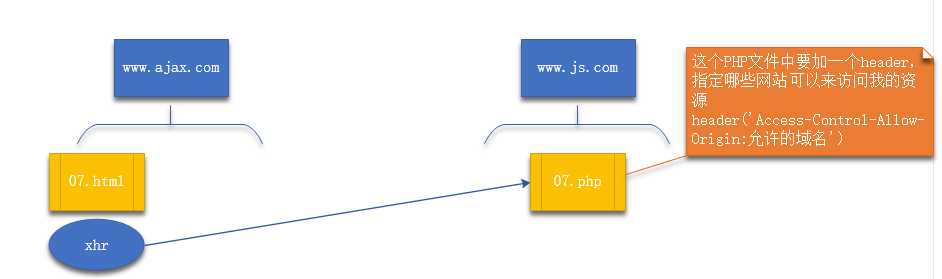
www.js.com下的PHP文件：



2、使用CORS方式

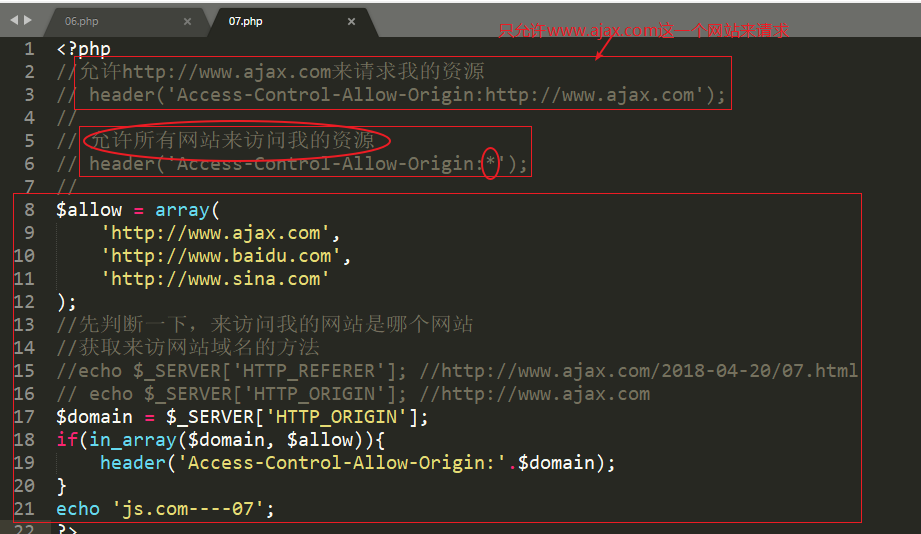
在我们访问的网站中加入一个header，该header可以控制哪些网站来请求资源。

原理图如下：



需要在请求的网站中设置：Access-Control-Allow-Origin

比如，用www.ajax.com中的文件异步请求www.js.com中的07.php，需要在www.js.com/07.php中设置header头。



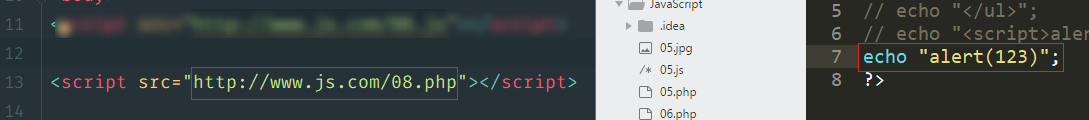
IE9+才能使用此方式。

3、使用jsonp完成跨域请求

通过html标签中的src属性可以访问另外域的内容，这就是jsonp技术。

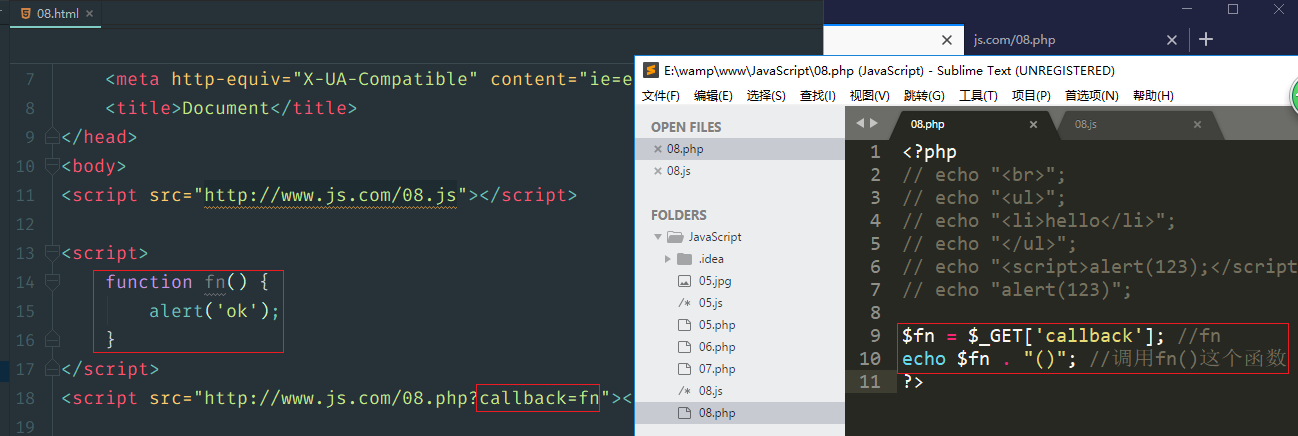
除了script标签中的src可以，其他标签的src也有相同的特点，比如img标签的src属性可以加载其他网站的图片

直接请求其他的js文件，可以，并且js文件中的代码会自动执行。



页面访问ajax这个网站的08.html，可以接受到js网站中的08.php输出的内容。

加上参数的例子：

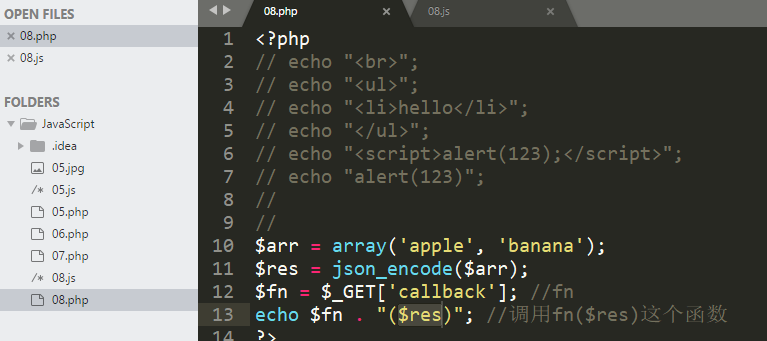


整理一下：

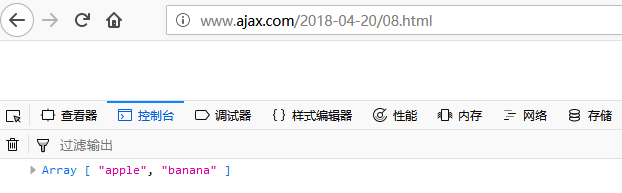
www.ajax.com中的HTML文件：



www.js.com中的08.php文件：



页面访问ajax中的html文件，可以输出js网站中返回的内容。



3、案例—获取天气信息

接口地址1：http://wthrcdn.etouch.cn/weather\_mini?citykey=101010100

接口地址2：http://www.weather.com.cn/data/sk/101010100.html

# 六、案例 – 无刷新分页

1、分页原理

比如，我的areacounty表有3125条数据。要求每页显示10条。那么共有多少页？

$count = 3125;

$pageSize = 10;  
$maxPage = ceil($count / $pageSize);

获取第1页数据的SQL：select \* from areacounty order by AID limit 0,10;

获取第2页数据的SQL：select \* from areacounty order by AID limit 10,10;

获取第3页数据的SQL：select \* from areacounty order by AID limit 20,10;

获取第$p页的数据的SQL：select \* from areacounty order by AID limit ($p-1)\*$pageSize, $pageSize;

2、传统的分页

分页代码：

<?php  
$pdo **=** new PDO('mysql:host=localhost; dbname=ajax; charset=utf8', 'root', '123');  
$pageSize **=** 10; //表示每页显示多少条  
$p **=** isset($\_GET['p']) **?** $\_GET['p'] **:** 1; //表示当前页，默认是1，  
$sql **=** "select \* from areacounty order by AID limit " **.** ($p**-**1)**\***$pageSize **.**","**.** $pageSize;  
$stmt **=** $pdo**->**prepare($sql);  
$stmt**->**execute();  
$data **=** $stmt**->**fetchAll(PDO**::FETCH\_ASSOC**);  
  
//用一个变量表示分页  
$str **=** '';  
if($p **<=** 1){  
 $str **.=** '上一页';  
}else{  
 $str **.=** '<a href="?p='**.**($p**-**1)**.**'">上一页</a>';  
}  
  
//计算最大页  
$sql2 **=** "select count(\*) from areacounty";  
$stmt2 **=** $pdo**->**prepare($sql2);  
$stmt2**->**execute();  
$count **=** $stmt2**->**fetchColumn(); //总记录数  
//echo $count;  
$maxPage **=** ceil($count **/** $pageSize);  
if($p **>=** $maxPage){  
 $str **.=** '下一页';  
}else{  
 $str **.=** '<a href="?p='**.**($p**+**1)**.**'">下一页</a>';  
}  
?>

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport"  
 content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>Document</title>  
</head>  
<body>  
<table border="1" cellpadding="2" cellspacing="0" width="500" align="center" rules="all">  
 <thead>  
 <tr>  
 <th>AID</th>  
 <th>Acode</th>  
 <th>Aname</th>  
 </tr>  
 </thead>  
 <tbody>  
 <?php  
 foreach($data as $value)**:** ?>  
 <tr>  
 <td><?php echo $value['AID'];?></td>  
 <td><?php echo $value['Acode'];?></td>  
 <td><?php echo $value['Aname'];?></td>  
 </tr>  
 <?php  
 endforeach;  
 ?>  
 </tbody>  
</table>  
<p align="center">  
 <?php echo $str;?>  
</p>  
</body>  
</html>

3、异步请求数据，完成无刷新分页

代码见11page.html和11page.php