

Web开发实训

第4讲：接口与服务集成

冯顺磊

18337149582 / fengshunlei@51xueweb.cn

2018.08

本讲主要内容

- jQuery Ajax简介
- load()方法
- \$.get()和\$.post()
- \$.getJSON()方法
- ajax()方法
- 服务集成

1. jQuery Ajax简介

- **Ajax** 是与服务器交换数据的技术，它在不刷新页面的情况下，实现数据的异步调用和页面的局部刷新。
- **jQuery**提供多个与**AJAX**有关的方法。通过 **jQuery AJAX** 方法，您能够使用 **HTTP Get** 和 **HTTP Post** 从远程服务器上请求文本、**HTML**、**XML** 或 **JSON** - 同时您能够把这些外部数据直接载入网页的被选元素中。
- 本节学习**jQuery**中常用的几个**Ajax**方法：**load()**、**\$.get()**和**\$.post()**、**\$.getJSON()**、**ajax()**。

2. load()方法

□ 用法

- load()方法从服务器加载数据，并把返回的数据放入被选元素中。

□ 语法

- \$(selector).load(URL,data,callback);

表 3-1 load() 方法参数说明↵

参数↵	必要性↵	描述↵
URL↵	必需↵	规定您希望加载的 URL，服务器端的文件或程序↵
data↵	可选↵	规定与请求一同发送的参数对象，键/值对集合↵
callback↵	可选↵	load() 方法完成后所执行的函数名称；↵ 回调函数可以设置不同的参数：↵ <u>responseTxt</u> - 包含调用成功时的结果内容↵ <u>statusTXT</u> - 包含调用的状态↵ <u>xhr</u> - 包含 <u>XMLHttpRequest</u> 对象↵

2. load()方法

□ 示例

- 示例1：load()加载内容。

```
$( '#pageContent' ).load( 'data/loadfirst.txt' );
```

2. load()方法

□ 示例

■ 示例2：load()加载内容并提示结果

```
$('#pageContent').load('data/loadfirst.txt', function (responseTxt, statusTxt, xhr) {  
    console.log(responseTxt); //响应内容  
    console.log(statusTxt); //请求结果 成功/失败  
    console.log(xhr); //对象  
    if (statusTxt == "success")  
        alert("外部内容加载成功!");  
    if (statusTxt == "error")  
        alert("Error: " + xhr.status + ": " + xhr.statusText);  
});
```

3. \$.get()和\$.post()方法

□ 用法

- jQuery `get()` 和 `post()` 方法用于通过 HTTP GET 或 POST 请求与服务器进行数据交换。**GET**-从服务器请求数据，**POST**-向服务器提交数据或从服务器下载数据。

□ 语法

- `$.get(URL,callback);`
- `$.post(URL,data,callback);`

3. \$.get()和\$.post()方法

□ 语法

表 3-2 \$.get()和\$.post()方法参数说明

参数	必要性	描述
URL	必需	规定您希望加载的 URL，服务器端的文件或程序
data	可选	规定与请求一同发送的参数对象，键/值对集合
callback	可选	请求完成后所执行的函数名称； 回调函数可以设置不同的参数： data-被请求页面的内容 status-被请求页面的状态 xhr-包含 XMLHttpRequest 对象

3. \$.get()和\$.post()方法

□ 示例

- \$.get()、\$.post()请求数据

```
$.get('data/getdata.txt', function (data, status) {  
    console.log(data);  
    console.log(status);  
    $('#pageContent').html(data);  
});
```

4. \$.getJSON()方法

□ 用法

- `getJSON()` 方法使用 **AJAX** 的 **HTTP GET** 请求获取 **JSON** 数据, 会将获取到的**json**字符串自动转换为**json**对象。

□ 语法

- `$.getJSON(url,data,success(data,status,xhr))`

4. \$.getJSON()方法

□ 语法

表 3-3 \$.getJSON() 方法参数说明

参数	必要性	描述
URL	必需	规定您希望加载的 URL，服务器端的文件或程序
data	可选	规定与请求一同发送的参数对象，键/值对集合
callback	可选	请求完成后所执行的函数名称； 回调函数可以设置不同的参数： data-被请求页面的内容 status-被请求页面的内容 xhr-包含 XMLHttpRequest 对象

4. \$.getJSON()方法

□ 示例

■ getJSON() 获取数据

```
$.getJSON('data/getdata.txt', function (data, status, xhr) {  
    console.log(data);  
    console.log(status);  
    console.log(xhr);  
    $('#pageContent').html(JSON.stringify(data));  
});
```

5. \$.ajax()方法

□ 用法

- `ajax()` 方法用于执行 **AJAX**（异步 **HTTP**）请求。以上讲到的 **jQuery AJAX** 方法都使用 `ajax()` 方法。该方法通常用于其他方法不能完成的请求。

□ 语法

- `$.ajax({name:value, name:value, ... })`

5. \$.ajax()方法

□ 语法

表 3-4 \$.ajax() 方法参数说明

名称	值/描述
<u>Async</u>	布尔值，表示请求是否异步处理。默认是 <code>true</code> 。 取 <code>false</code> 表示同步请求。
<u>beforeSend(xhr)</u>	发送请求前运行的函数。
Cache	布尔值，表示浏览器是否缓存被请求页面。默认是 <code>true</code> 。
<u>complete(xhr, status)</u>	请求完成时运行的函数（在请求成功或失败之后均调用，即在 <code>success</code> 和 <code>error</code> 函数之后）。
<u>contentType</u>	发送数据到服务器时所使用的内容类型。默认是： <code>"application/x-www-form-urlencoded"</code> 。
Context	为所有 AJAX 相关的回调函数规定 "this" 值。
Data	规定要发送到服务器的数据。
<u>dataFilter(data, type)</u>	用于处理 <code>XMLHttpRequest</code> 原始响应数据的函数。
<u>dataType</u>	预期的服务器响应的数据类型。
<u>error(xhr, status, error)</u>	如果请求失败要运行的函数。

5. \$.ajax()方法

□ 语法

<code>global</code>	布尔值，规定是否为请求触发全局 AJAX 事件处理程序。默认是 <code>true</code> 。
<code>ifModified</code>	布尔值，规定是否仅在最后一次请求以来响应发生改变时才请求成功。默认是 <code>false</code> 。
<code>jsonp</code>	在一个 <code>jsonp</code> 中重写回调函数的字符串。
<code>jsonpCallback</code>	在一个 <code>jsonp</code> 中规定回调函数的名称。
<code>password</code>	规定在 HTTP 访问认证请求中使用的密码。
<code>processData</code>	布尔值，规定通过请求发送的数据是否转换为查询字符串。默认是 <code>true</code> 。
<code>scriptCharset</code>	规定请求的字符集。
<code>success(result, status, xhr)</code>	当请求成功时运行的函数。
<code>timeout</code>	设置本地的请求超时时间（以毫秒计）。
<code>traditional</code>	布尔值，规定是否使用参数序列化的传统样式。
<code>Type</code>	规定请求的类型（GET 或 POST）。
<code>url</code>	规定发送请求的 URL。默认是当前页面。
<code>username</code>	规定在 HTTP 访问认证请求中使用的用户名。
<code>xhr</code>	用于创建 <code>XMLHttpRequest</code> 对象的函数。

5. \$.ajax()方法

- ❑ 示例1：ajax异步与同步请求
- ❑ 同步和异步指ajax请求与外部程序的执行方式。
- ❑ 示例如下：

```
var i = 0;+  
var j = 0;+  
while (i < 10) {+  
    $.ajax({+  
        url: "data/getdata.txt",+  
        async: false, //是否异步执行+  
        success: function (data) {+  
            var time = new Date().getTime();+  
            console.log(i + '_' + j + '_' + time);+  
            j++;+  
        }+  
    }+  
    i++;+  
}+
```


5. \$.ajax()方法


- 示例2: jsonp接口调用
- 示例如下:

```
$.ajax({  
  url: "https://api.asilu.com/phone",  
  async: true, //是否异步执行  
  type: 'GET',  
  dataType: 'jsonp',  
  jsonpCallback: 'callback',  
  data: {  
    'phone': 15290856174  
  },  
  success: function (data) {  
    console.log(data);  
  }})
```

6.服务集成

- 网址: <http://tianqi.2345.com/plugin>
- 网站: <http://www.tianqi.com/plugin>

北京 [切换]

空气质量: 良



今天(周二)

29~38°C

雷阵雨



明天(周三)

27~36°C

晴



后天(周四)

26~34°C

多云

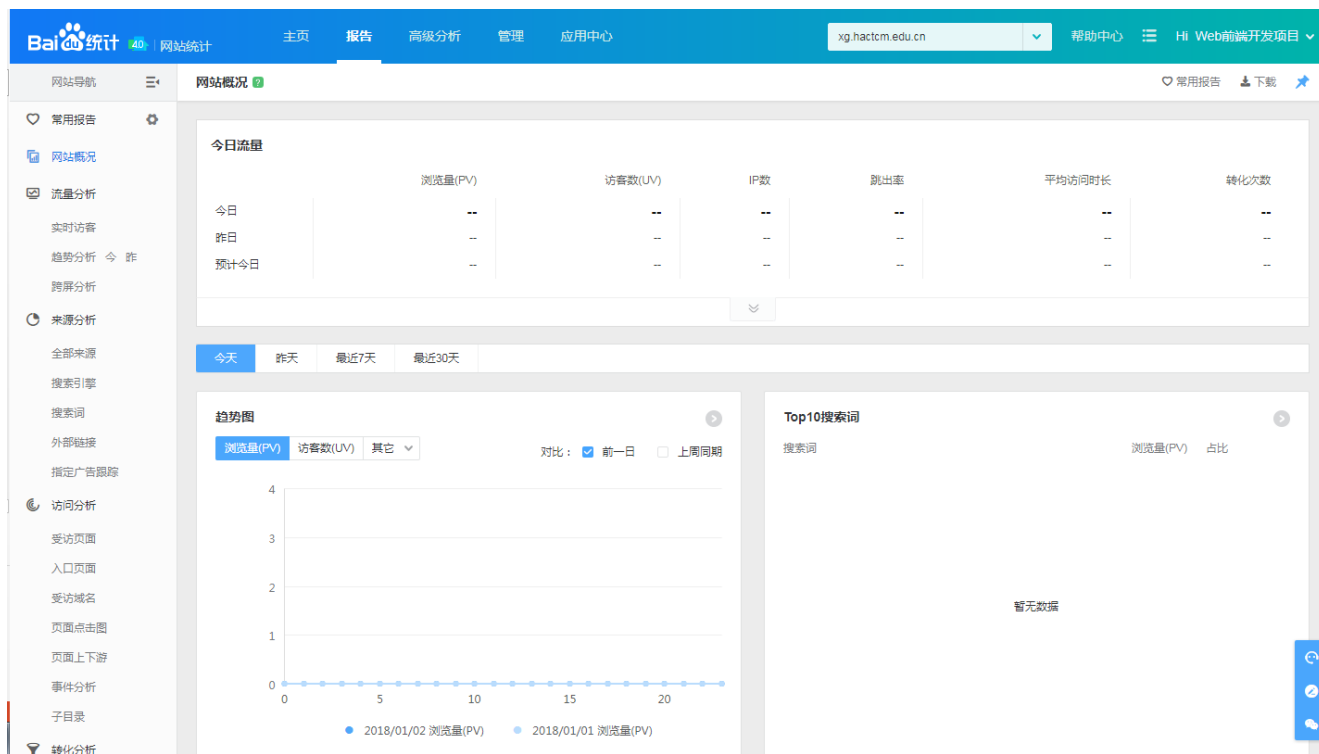
6.服务集成

- 网址: <http://share.baidu.com>



6.服务集成

□ 网址: <https://tongji.baidu.com>



6.服务集成

- 网址: <http://lbsyun.baidu.com>



6.服务集成

□ 网址: <https://connect.qq.com/intro/wpa>

> 通讯组件

· QQ分享组件

· Qzone分享组件

· 划词分享组件

· 赞组件

· 关注组件

· 签名档组件

· QQ提醒组件

6. 服务集成

□ 网址: <http://open.weibo.com/widgets>



赞组件

使用微博赞按钮，将生成的代码放置到你的博客、网站或其它支持HTML的位置，让你的用户轻松表达态度。当用户在您的站点上某个网页点击了赞按钮后，同时将同步一条包含网页内容的微博在其微博主页上。



分享按钮

将分享按钮嵌入到你的网站或移动应用里，你的访客点击它就能将你的内容分享到新浪微博，带来更高回流。



关注按钮

关注按钮可以部署在任何支持iframe的网页。用户在这些网页点击关注按钮，就可以成为你的微博粉丝。



微博话题墙

微博话题墙可以部署在任何支持HTML的网页中，提供基础版和PK版两个版本，实时展示话题相关微博、官方账号和转发热榜，并支持置顶一条微博。



微博直播

可以让用户在您的网站上针对某一热点话题进行实时讨论，用户在此发布的微博将会发布到新浪微博。

6. 服务集成

- 微博小工具
- 网址：<http://app.weibo.com/tool/signature>

基础设置 样式设置

设置尺寸：

宽 px 高 px

宽度:190-1024px,高度:75-800px

宽度自动适应

外观方案：

默认配色

自定义配色 [设置自定义配色](#)

图片内容：

显示为缩略图 显示为文字链

显示粉丝：

行

可设置1-7行，最多显示20个粉丝

效果预览：

微博

倩剑书生 河南 郑州

✓ 我自己

转发微博

最美中国风 ▼：一朝风雨，满地残红。湿了花香，几许悲凉，奈何世间无常。



TA 的粉丝 (94) [全部](#)

 ws 腾傑	 夜月风雨	 手机用户	 新手指南	 静夜里思
 蜡笔-儿	 回眸处听	 to_forgi	 不贰过00	 清风吹过

6.服务集成

□ 网址: <https://cli.im>

The screenshot displays the '草料二维码' (CaoLiao QR Code) website interface. At the top, there is a navigation bar with the logo and menu items: 产品, 解决方案, 价格, 案例, 帮助, 关于我们, and a 登录 / 注册 button. Below the navigation bar is a secondary menu with options: 文本, 网址 (selected), 名片, 文件, 图片, 微信, 小程序, 商品码, 记录码, 批量生码, 更多, and a 扫码 icon. The main content area features a text input field containing the URL 'http://ftpweb.xg.hactcm.edu.cn/'. Below the input field, there is a note: '如果想要一个二维码里显示多个网址, 可以使用 [网址导航活码](#) [查看案例](#)'. Two buttons are visible: '重新生成' and '转成活码'. A tooltip points to the '转成活码' button with the text: '活码可以随时修改二维码内容, 二维码图案不变, 还可以查看扫码统计分析。'. To the right of the input field is a large QR code. Below the QR code is a green '下载' button. Further down, there are links for '其它格式' and '快速打印'. At the bottom, there are tabs for '基本', '颜色', 'LOGO', and '美化器', and two buttons: '上传 LOGO' and '常用 LOGO'.

Thanks.