[jQuery Ajax 实例 ($.ajax、$.post、$.get)](http://jun1986.iteye.com/blog/1399242)

**博客分类：**

* [jquery](http://jun1986.iteye.com/category/153794)
* [经验+注意](http://jun1986.iteye.com/category/171218)

Jquery在异步提交方面封装的很好，直接用AJAX非常麻烦，Jquery大大简化了我们的操作，不用考虑浏览器的诧异了。

推荐一篇不错的jQuery Ajax 实例文章，忘记了可以去看看，地址为：http://www.cnblogs.com/yeer/archive/2009/07/23/1529460.html 和 http://www.w3school.com.cn/jquery/

$.post、$.get是一些简单的方法，如果要处理复杂的逻辑，还是需要用到jQuery.ajax()

**一、$.ajax的一般格式**

$.ajax({

     type: 'POST',

     url: *url* ,

    data: *data* ,

    success: *success* ,

    dataType: *dataType*

});

**二、$.ajax的参数描述**

参数 描述

|  |
| --- |
|  |
| *url* | 必需。规定把请求发送到哪个 URL。 |
| *data* | 可选。映射或字符串值。规定连同请求发送到服务器的数据。 |
| *success(data, textStatus, jqXHR)* | 可选。请求成功时执行的回调函数。 |
| *dataType* | 可选。规定预期的服务器响应的数据类型。  默认执行智能判断（xml、json、script 或 html）。 |

**三、$.ajax需要注意的一些地方：**

  1.data主要方式有三种，html拼接的，json数组，form表单经serialize()序列化的；通过dataType指定，不指定智能判断。

  2.$.ajax只提交form以文本方式，如果异步提交包含<file>上传是传过不过去,需要使用jquery.form.js的$.ajaxSubmit

**四、$.ajax我的实际应用例子**

**Js代码  [收藏代码](javascript:void())**

1. //1.$.ajax带json数据的异步请求
2. **var** aj = $.ajax( {
3. url:'productManager\_reverseUpdate',// 跳转到 action
4. data:{
5. selRollBack : selRollBack,
6. selOperatorsCode : selOperatorsCode,
7. PROVINCECODE : PROVINCECODE,
8. pass2 : pass2
9. },
10. type:'post',
11. cache:**false**,
12. dataType:'json',
13. success:**function**(data) {
14. **if**(data.msg =="true" ){
15. // view("修改成功！");
16. alert("修改成功！");
17. window.location.reload();
18. }**else**{
19. view(data.msg);
20. }
21. },
22. error : **function**() {
23. // view("异常！");
24. alert("异常！");
25. }
26. });

29. //2.$.ajax序列化表格内容为字符串的异步请求
30. **function** noTips(){
31. **var** formParam = $("#form1").serialize();//序列化表格内容为字符串
32. $.ajax({
33. type:'post',
34. url:'Notice\_noTipsNotice',
35. data:formParam,
36. cache:**false**,
37. dataType:'json',
38. success:**function**(data){
39. }
40. });
41. }

44. //3.$.ajax拼接url的异步请求
45. **var** yz=$.ajax({
46. type:'post',
47. url:'validatePwd2\_checkPwd2?password2='+password2,
48. data:{},
49. cache:**false**,
50. dataType:'json',
51. success:**function**(data){
52. **if**( data.msg =="false" ) //服务器返回false，就将validatePassword2的值改为pwd2Error，这是异步，需要考虑返回时间
53. {
54. textPassword2.html("<font color='red'>业务密码不正确！</font>");
55. $("#validatePassword2").val("pwd2Error");
56. checkPassword2 = **false**;
57. **return**;
58. }
59. },
60. error:**function**(){}
61. });

64. //4.$.ajax拼接data的异步请求
65. $.ajax({
66. url:'<%=request.getContextPath()%>/kc/kc\_checkMerNameUnique.action',
67. type:'post',
68. data:'merName='+values,
69. async : **false**, //默认为true 异步
70. error:**function**(){
71. alert('error');
72. },
73. success:**function**(data){
74. $("#"+divs).html(data);
75. }
76. });

jquery中的ajax方法参数

**1.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。

**2.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。

**3.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。

**4.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。

**5.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。

**6.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。

**7.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。

**8.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  
            }  
**9.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  
          }

**10.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  
         }

**11.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  
       }

**12.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"。该默认值适合大多数应用场合。

**13.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;  
            }

**14.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;  
            }

**15.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。

**16.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。

**17.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。

**18.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。

**19.password**：  
要求为String类型的参数，用于响应HTTP访问认证请求的密码。

**20.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。

**21.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。

案例代码：

[复制代码](javascript:void(0);)

$(function(){

$('#send').click(function(){

$.ajax({

type: "GET",

url: "test.json",

data: {username:$("#username").val(), content:$("#content").val()},

dataType: "json",

success: function(data){

$('#resText').empty(); //清空resText里面的所有内容

var html = '';

$.each(data, function(commentIndex, comment){

html += '<div class="comment"><h6>' + comment['username']

+ ':</h6><p class="para"' + comment['content']

+ '</p></div>';

});

$('#resText').html(html);

}

});

});

});

[复制代码](javascript:void(0);)