

Jquery跨域获得Json的简单实例_jquery_脚本之家

下面小编就为大家带来一篇Jquery跨域获得Json的简单实例。小编觉得挺不错的，现在分享给大家，也给大家做个参考。一起跟随小编过来看看吧

这两天用 Jquery 跨域取数据的时候，经常碰到 invalid label 这个错误，十分的郁闷，老是取不到服务器端发送回来的 json 值，

一般跨域用到的两个方法为：\$.ajax 和\$.getJSON

最后，仔细安静下来，细读 json 官方文档后发现这么一段：

JSON数据是一种能很方便通过JavaScript解析的结构化数据。如果获取的数据文件存放在远程服务器上（域名不同，也就是跨域获取数据），则需要使用jsonp类型。使用这种类型的话，会创建一个查询字符串参数 callback=?，这个参数会加在请求的URL后面。服务器端应当在JSON数据前加上回调函数名，以便完成一个有效的JSONP请求。如果要指定回调函数的参数名来取代默认的callback，可以通过设置\$.ajax()的jsonp参数。

其实jquery跨域的原理是通过外链 <script> 来实现的,然后在通过回调函数加上回调函数的参数来实现真正的跨域

Jquery 在每次跨域发送请求时都会有callback这个参数，其实这个参数的值就是回调函数名称，所以，服务器端在发送json数据时，应该把这个参数放到前面，这个参数的值往往是随机生成的，如：jsonp1294734708682，同时也可以通过 \$.ajax 方法设置 callback 方法的名称。明白了原理后，服务器端应该这样发送数据：

```
string message = "jsonp1294734708682({\"userid\":0,\"username\":\"null\"})";
```

这样，json 数据 {"userid":0,"username":"null"} 就作为了 jsonp1294734708682 回调函数的一个参数

下面我们来开始实例假如正常返回的数据：

```
{ "success": [{ "id": 1, "title": "title 1" }, { "id": 2, "title": "title 2" }, { "id": 3, "title": "title 3"}] }
```

下面我们来介绍下jquery的自己个参数

```
/**
 *@dataType (String)
 *"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。"json": 返回 JSON 数据 。
 *"jsonp": JSONP 格式。使用 JSONP 形式调用函数时，如 "myurl?callback=?" jQuery 将自动替换
? 为正确的函数名，以执行回调函数。
```

```

* "text": 返回纯文本字符串
*/

/**
*@jsonp (String) 在一个jsonp请求中重写回调函数的名字。
*这个值用来替代在"callback=?"这种GET或POST请求中URL参数里的"callback"部分,
*比如{jsonp:'onJsonPLoad'}会导致将"onJsonPLoad=?"传给服务器。
*/

/**
*@jsonpCallback (String)
*为jsonp请求指定一个回调函数名。这个值将用来取代jQuery自动生成的随机函数名。
*这主要用来让jQuery生成度独特的函数名, 这样管理请求更容易, 也能方便地提供回调函数和错误处理。
*你也可以在想让浏览器缓存GET请求的时候, 指定这个回调函数名。
*/

```

1 不指定jsonp的名称、

```

$.ajax({
    url: 'http://lifeloopdev.info/get_events',
    dataType: "jsonp",
    data: "offset=0&num_items=10",
    username: 'username',
    password: 'password',
    success: function (data) {
        $.each(data.success, function (i, item) {
            $("body").append('<h1>' + item.title + '</h1>');
        });
    }
});

```

服务器需要这样返回数据示例:

```

Response.ContentType="text/html; charset=utf-8";
String callback = Request.QueryString["callback"].ToString();
Response.Write(callback + "{ \"success\": [{ \"id\": 1, \"title\": \"title 1\" }, { \"id\": 2, \"title\": \"title 2\" }, { \"id\": 3, \"title\": \"title 3\"}] }");

```

2 指定jsonp名称, 和返回函数名称的function、

```

//这里我们自己指定了jsonp的callback的名字

```

```
$.ajax({
  url: 'http://lifeloopdev.info/get_events',
  dataType: "jsonp",
  data: "offset=0&num_items=10",
  username: 'username',
  password: 'password',
  jsonp: "successCallback",
  jsonpCallback: 'successCallback'
});

function successCallback(data) {
  $.each(data.success, function (i, item) {
    $("body").append('<h1>' + item.title + '</h1>');
  });
}
```

服务器需要这样返回数据示例:

```
Response.ContentType="text/html; charset=utf-8";
String callback = Request.QueryString["successCallback"].ToString();
Response.Write(callback + "{ \"success\": [{ \"id\": 1, \"title\": \"title 1\" }, { \"id\": 2, \"title\": \"title 2\" }, { \"id\": 3, \"title\": \"title 3\"}] }");
```

3指定jsonp名称, 不指定返回函数名称的function、

```
$.ajax({
  url: 'http://lifeloopdev.info/get_events',
  dataType: "jsonp",
  data: "offset=0&num_items=10",
  username: 'username',
  password: 'password',
  jsonp: "successCallback",
  success: function (data) {
    $.each(data.success, function (i, item) {
      $("body").append('<h1>' + item.title + '</h1>');
    });
  }
});
```

服务器需要这样返回数据示例:

```
Response.ContentType="text/html; charset=utf-8";
String callback = Request.QueryString["successCallback"].ToString();
```

```
Response.Write(callback + "{ \"success\": [{ \"id\": 1, \"title\": \"title 1\" }, {  
\"id\": 2, \"title\": \"title 2\" }, { \"id\": 3, \"title\": \"title 3\"}] }");
```

4使用getJSON() 获取数据、

```
/**  
*注意:  
*这里调用的地址中jsoncallback=? 是关键的所在! 其中我们关心的是 jsoncallback=? 起什么作用  
了? 原来 jsoncallback=? 被替换后, 会把方法名称传给服务器。  
*我们在服务器端要做什么工作呢? 服务器要接受参数 jsoncallback , 然后把 jsoncallback 的值作  
为 JSON 数据方法名称返回 .  
*/  
$.getJSON("http://192.168.20.86/friend/getMyJsonData.aspx?jsoncallback=?", function  
(data) {  
    $.each(data.success, function (i, item) {  
        $("body").append('<h1>' + item.title + '</h1>');  
    });  
});
```

服务器需要这样返回数据示例:

```
Response.ContentType="text/html; charset=utf-8";  
String callback = Request.QueryString["jsoncallback"].ToString();  
Response.Write(callback + "{ \"success\": [{ \"id\": 1, \"title\": \"title 1\" }, {  
\"id\": 2, \"title\": \"title 2\" }, { \"id\": 3, \"title\": \"title 3\"}] }");
```

以上这篇Jquery跨域获得Json的简单实例就是小编分享给大家的全部内容了, 希望能给大家一个参考, 也希望大家多多支持脚本之家。

您可能感兴趣的文章:

- 1