

# jquery对 ajax 的支持

---

## 1.\$obj.load()

作用：载入远程的HTML文件到指定的元素中

```
$obj.load(url,data,callback)
```

`$obj`: 显示响应内容的jq元素

`url`: 请求地址

`data`: 请求参数(可省略)

方式1: 字符串传参

`"key1=value1&key2=value2"`

注：此种传参会使用 `get` 方式发送请求

方式2: 使用JS对象传参

```
{  
  key1:"value1",  
  key2:"value2"  
}
```

注：此种传参会使用 `post` 方式发送请求

`callback`: 响应成功后的回调函数(可省略)

## 2.\$.get() 和 \$.post()

作用：通过get方式异步的向远程地址发送请求

```
$.get(url,data,callback,type)
```

`url`: 请求地址

`data`: 传递到服务器端的参数

可以是字符串：`"name=sf.zh&age=18"`

也可以是js对象：

```
{  
    name: "sf.zh",  
    age: 18  
}
```

callback: 响应成功后的回调函数

```
ex: function(data){  
    console.log(data)  
}
```

type: 响应回来的数据的格式

取值如下：

- 1.html : 响应回来的文本是html文本
- 2.text : 响应回来的文本是text文本
- 3.script : 响应回来的文本是js执行脚本
- 4.json : 响应回来的文本是json格式的文

本

### 3. \$.ajax()

参数对象中的属性：

- 1.url : 字符串，表示异步请求的地址
- 2.type : 字符串，请求方式，get 或 post
- 3.data : 传递到服务器端的参数

可以是字符串：`"name=sf.zh&age=18"`

也可以是js对象：

```
{  
    name: "sf.zh",  
    age: 18  
}
```

- 4.dataType : 字符串，响应回来的数据的格式

1. 'html'
2. 'xml'
3. 'text'
4. 'script'
5. 'json'
6. 'jsonp' : 有关跨域的响应格式

5. **success**: 回调函数, 请求和响应成功时回来执行的操作

6. **error** : 回调函数, 请求或响应失败时回来执行的操作

7. **beforeSend** : 回调函数, 发送ajax请求之前执行的操作, 如果 **return false**, 则终止请求

# 跨域

## 1, 什么是跨域

跨域: 非同源的网页, 相互发送请求的过程, 就是跨域

浏览器的同源策略:

同源: 多个地址中, 相同协议, 相同域名, 相同端口被视为是"同源"

在HTTP中, 必须是同源地址才能互相发送请求, 非同源拒绝请求(<script>和<img>除外)。

http://www.tedu.cn/a.html

http://www.tedu.cn/b.html

以上地址是 "同源"

http://www.tedu.cn/a.html

https://www.tedu.cn/b.html

由于 协议不同 ， 所以不是"同源"

`http://localhost/a.html`

`http://127.0.0.1/a.html`

由于 域名不同 ， 所以不是"同源"

`http://www.tedu.cn:80/a.html`

`http://www.tedu.cn:8080/b.html`

由于端口不同 ， 所以不是"同源"

## 2, 解决方案

通过 script 标签 src 向服务器资源发送请求

由服务器资源指定前端页面的哪个方法来执行响应的数据

## 3, jquery 的跨域

jsonp - json with padding

用户传递一个callback参数给服务端，然后服务端返回数据时会将这个callback参数作为函数名来包裹住JSON数据

ex:

当前地址: <http://127.0.0.1:8000/index>

欲访问地址: <http://localhost:8000/data?callback=xx>

[X](#)

```
$.ajax({  
    url: 'xxx',  
    type: 'get',  
    dataType: 'jsonp', //指定为跨域访问  
    jsonp: 'callback', //定义了callback的参数名,  
    //以便获取callback传递过去的函数名
```

```
jsonpCallback: 'xxx' //定义jsonp的回调函数名  
});
```

//超简版本

```
$.ajax({  
    url: 'xxx',  
    type: 'get',  
    dataType: 'jsonp', //指定为跨域访问  
    success: function(data){  
        console.log(data);  
    }  
});
```