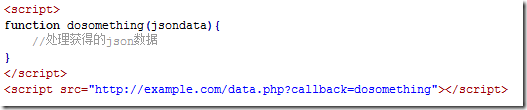
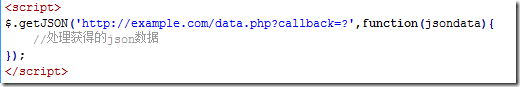
1. 内部开发说明，不外传！
   1. AJAX的跨域调用
      1. 通过jsonp跨域

方法从跨域读取一个包括一个callback调用的script文件，把数据放到function的参数中，实现跨域。



jquery封装了jsonp操作



我们不需要手动的插入script标签以及定义回掉函数。jquery会自动生成一个全局函数来替换callback=?中的问号，之后获取到数据后又会自动销毁，实际上就是起一个临时代理函数的作用。$.getJSON方法会自动判断是否跨域，不跨域的话，就调用普通的ajax方法；跨域的话，则会以异步加载js文件的形式来调用jsonp的回调函数。

服务端返回的jsonp格式：

callback({

"message":"获取成功",

"state":"1",

"result":{"name":"工作组1","id":1,"description":"11"}

})

但实际测试中，发现jquery并没自动生成和注销callback函数，需要自己定义。

function callback(data){

console.log("callback!");

console.log(data);

console.log(data.uri);

}

var url="http://"+ywzdomain+path+"?account="+account+"&sec="+makeSec()+"&tm="+tm+"&nickname=vodtest&callback=?";

console.log(url);

$.getJSON(url,function (json) {

//调用后直接运行callback()函数，永远不运行到这里

console.log(json);

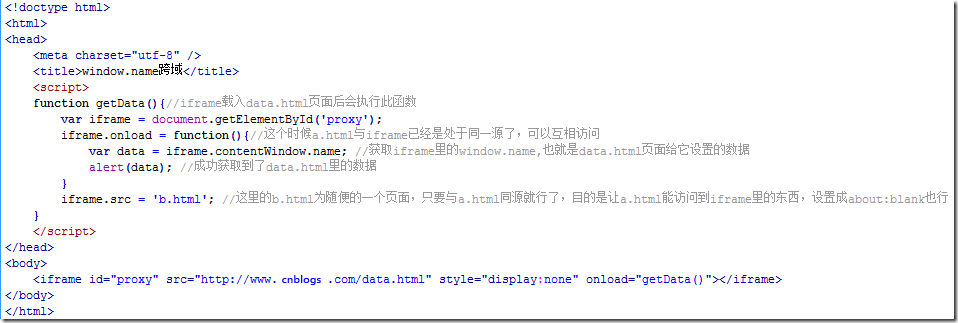
},'jsonp');

服务端：

echo "callback(".json\_encode($retAttr).")";

* + 1. 使用window.name来进行跨域

window对象有个name属性，该属性有个特征：即在一个窗口(window)的生命周期内,窗口载入的所有的页面都是共享一个window.name的，每个页面对window.name都有读写的权限，window.name是持久存在一个窗口载入过的所有页面中的，并不会因新页面的载入而进行重置。



* + 1. 使用HTML5中新引进的window.postMessage方法来跨域传送数据

window.postMessage(message,targetOrigin) 方法是html5新引进的特性，可以使用它来向其它的window对象发送消息，无论这个window对象是属于同源或不同源

* 1. 服务端统一用SIAction处理
* module:SI，action:根据不同的功能进行调用
* 在构造函数中进行协议层校验，校验不通过，处理终止，无任何返回。
* 在平台上必须配置account具有action的操作权限
* 在account的userExtAttr按约定填写commKey及SIdomain属性
  1. 数据表

在user表的扩展属性中增加两个属性：

|  |  |  |
| --- | --- | --- |
| commKey | userExtAttr | SI接口的约定通信秘钥 |
| SIdomain | userExtAttr | SI调用的有效域名 |