**JS跨域处理**

**一、原生JS**

// 获取 XMLHttpRequest对象

xmlHttp = new XMLHttpRequest();

xmlHttp.withCredentials = true; //跨域

// 连接服务器

xmlHttp.open("post", "http://xxxx.com/api.php"); //api地址

// 设置请求头的Content-Type

xmlHttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");

// 发送数据

xmlHttp.send();

// 回调函数 success

xmlHttp.onreadystatechange = function () {

if (this.readyState == 4 && this.status == 200) {

console.log(this.responseText);

}

};

**二、Jquery**  
$.ajax({

type: 'post',

url: 'http://xxxx.com/api.php', //api地址

data: JSON.stringify(info),

contentType: "application/x-www-form-urlencoded", //设置请求头的Content-Type

dataType: "json", //如果写了json，后端就必须传JSON格式的数据，否则不执行success

xhrFields: { //跨域

withCredentials: true

},

success: function (result) {

console.log(result);

},

})

|  |  |
| --- | --- |
| **dataType** | |
| 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。(因为将使用DOM的script标签来加载) |
| 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 |

**三、后端PHP**

**头部加上如下代码：**

// 制定允许其他域名访问 必须带类似http://的协议

header("Access-Control-Allow-Origin:http://www.test.com");  
//这里只支持一个域名，所有前端测试都用统一域名，不要使用IP

header('Access-Control-Allow-Methods:POST'); // 响应类型post

header('Access-Control-Allow-Credentials:true'); //运行跨域操作

header('Access-Control-Allow-Headers:x-requested-with, content-type'); // 响应头设置

更多：https://segmentfault.com/a/1190000012469713