

# 目录

- JSONP跨域

## 1.回顾ajax

1. 创建 `ajax` 对象 `xhr`
2. 调用 `open('get/post', 'ip', true)` 方法, `post` 设置请求头
3. 调用 `send()`
4. 等待数据响应

- `ajax` 请求本地服务器数据 正常请求
- `ajax` 请求别的服务器上的数据
  - 报错: **已拦截跨源请求** (原因: CORS 头缺少 'Access-Control-Allow-Origin')
    - `ajax` **只能下载同源的数据**,跨源的数据禁止下载。
      - 原因: 保证服务器的安全
    - 同源策略 (满足三者)
      1. 同协议
      2. 同域名/IP
      3. 同端口号
  - 跨源的方式:
    1. 修改 `ajax` 的同源协议(不建议)
    2. 委托 `php` 文件进行跨源
      - `php` 页面是没有跨源协议的,可以通过当前 `html` 页面请求 `php` 页面,通过 `php` 页面进行跨源
    3. JSONP 跨域

## 3.认识JSONP跨域

- `img` 的 `src` 属性可以完成跨源。
- `script` 的 `src` 也是可以跨源的
- JSONP 跨域原理
  - 通过 `script` 标签 `src` 属性
    1. 将要下载的数据放在 `demo.js` 中的函数调用

```
download('数据')
```

2. 在`index.html` 中, 先写一个 `script` ,里面写 `download(date)`

```
download(date){  
    alert(date);  
}
```

3. 引入 `deom.js` ,就可以完成数据下载

```
<script>  
    function download(date){  
        alert(date);  
    }  
</script>  
<script src="./03demo.js"></script>  
// 03demo.js内容 => download('I am string');  
// 利用函数的声明和调用
```

## 4.动态创建script标签

```
var oScript = document.createElement("script");  
oScript.src = "https://query.asilu.com/weather/baidu";  
document.body.appendChild(oScript);
```

## 5.天气查询——案例

- 在线 JSONP 接口测试: <http://cdn.asilu.com/#weather>
- 拿到 url 的数据, 如何变成 jsonp 跨域的方式, 进行函数调用
  - 原url: <https://query.asilu.com/weather/baidu>
  - 进行操作后的url: [https://query.asilu.com/weather/baidu?city="上海"&callback=download](https://query.asilu.com/weather/baidu?city=%E4%B8%AD%E6%B5%B7&callback=download)
    - 在原本的url后进行字符串拼接
      - city="上海"
      - callback=download (download是要调用函数的函数名)

## 6.百度搜索框

- JSONP 测试地址: <http://suggestion.baidu.com/su?wd=关键词&cb=download;>
  - wd 后拼接 关键词
  - cb 后拼接 函数名