

一、什么是跨域请求

1. 同域:

协议相同, ip相同, 端口相同

<http://192.168.2.117:9999/get.do>

<http://192.168.2.117:9999/user.do>

<http://192.168.2.117:9999/user/aaa/getget.do>

URL	说明	是否允许通信
http://192.168.2.117:9999/		
http://www.a.com/a.js		
http://www.a.com/b.js	同一域名下	允许
http://www.a.com/lab/a.js		
http://www.a.com/script/b.js	同一域名下不同文件夹	允许
http://www.a.com:8000/a.js		
http://www.a.com/b.js	同一域名, 不同端口	不允许
http://www.a.com/a.js		
https://www.a.com/b.js	同一域名, 不同协议	不允许
http://www.a.com/a.js		
http://70.32.92.74/b.js	域名和域名对应ip	不允许
http://www.a.com/a.js		
http://script.a.com/b.js	主域相同, 子域不同	不允许
http://www.a.com/a.js		
http://a.com/b.js	同一域名, 不同二级域名 (同上)	不允许 (cookie 这种情况下也不允许访问)
http://www.cnblogs.com/a.js		
http://www.a.com/b.js	不同域名	不允许

2. 跨域:

不是同域

3. 跨域请求

3.1 浏览器对于javascript的同源策略的限制 一般不许不同域之间相互请求

3.2 不过我们在调用css文件的时候是不受跨域影响 比如引入bootstrap框架

3.3 凡是标签拥有src这个属性的标签都可以跨域例如<script> <iframe>

二、跨域请求常用只有两种方式

1. jsonp

1.1 底层原理是script标签是不受同源策略影响的, 它可以引入来自任何地方的js文件。

1.2 jsonp只能实现get请求的跨域

示例: 服务器

```
app.get("/getStr.do", function(req, res) {  
    res.jsonp("this is ajax jsonp data");  
});
```

客户端

```
$.ajax({  
    url: "http://localhost:9999/getStr.do",  
    type: "get",  
    data: {},  
    dataType: "jsonp",  
    success: function(result) {  
        console.log(result);  
        $("div").html(result);  
    },  
    error: function(error) {  
        console.log("ajax访问出错")  
    }  
});
```

客户端如果是同源的, 即没有跨域的, 则url中可以只写"getStr.do"即可

2. cors

CORS原理只需要向响应头header中注入Access-Control-Allow-Origin,

CORS是一个W3C标准，全称是“跨域资源共享”（Cross-origin resource sharing）。

CORS对GET和POST都支持

示例:服务器

```
app.post("/getStr.do", function(req, res) {  
    res.setHeader("Access-Control-Allow-Origin", "*");  
    res.json("this is ajax data");  
});
```

说明:

(1)res.setHeader("Access-Control-Allow-Origin", "*");中的"*"表示所有其它的域都可

以跨域访问当前请求。

(2)res.setHeader("Access-Control-Allow-Origin", "http://localhost:3333");中的
"http://localhost:3333"表示只有当前电脑中域为http://localhost:3333的才能访问
当前请求，

值得注意，不要在端口号后面加"/"

客户端

```
$.ajax({  
    url: "http://localhost:9999/getStr.do",  
    type: "post",  
    data: {},  
    dataType: "json",  
    success: function(result) {  
        console.log(result);  
        $("div").html(result);  
    },  
    error: function(error) {  
        console.log("ajax访问出错")  
    }  
});
```

三、cors跨域请求示例

```
app.get("/getStr.do", function(req, res) {  
    res.setHeader("Access-Control-Allow-Origin", "*");  
    res.json("this is ajax data");  
});
```

```
$.ajax({  
    url: "http://localhost:9999/getStr.do",  
    type: "get",  
    data: {},  
    dataType: "json",  
    success: function(result) {  
        console.log(result);  
        $("div").html(result);  
    },  
    error: function(error) {  
        console.log("ajax访问出错")  
    }  
});
```

```
});
```

```
const app = express();

// cors:跨域资源共享
app.use(function(req, res, next){
  res.setHeader('Access-Control-Allow-Origin', '*');
  next();
});

app.get('/user', (req, res)=>{
  res.json({username:'蒋应龙'});
});

app.listen(81, () => {
  console.log(`Server started on 81`);
});
```