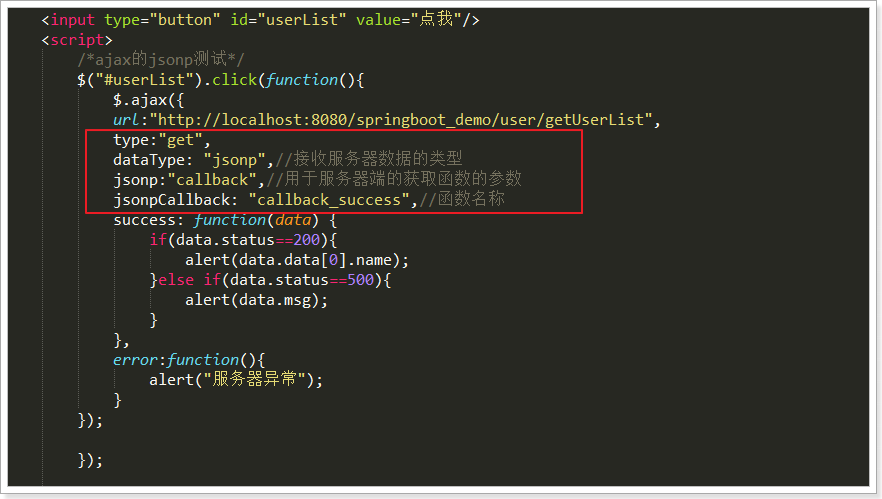
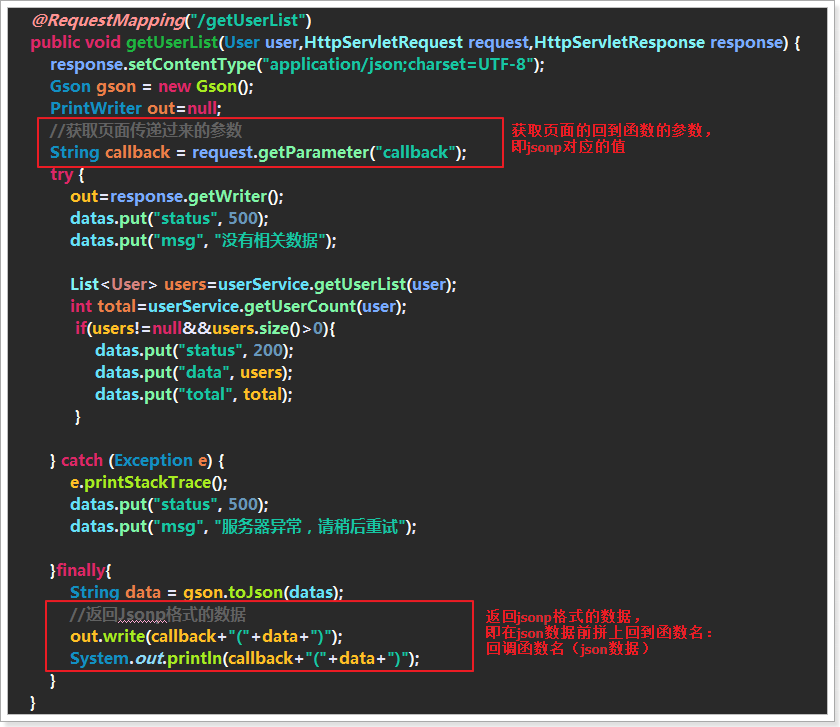
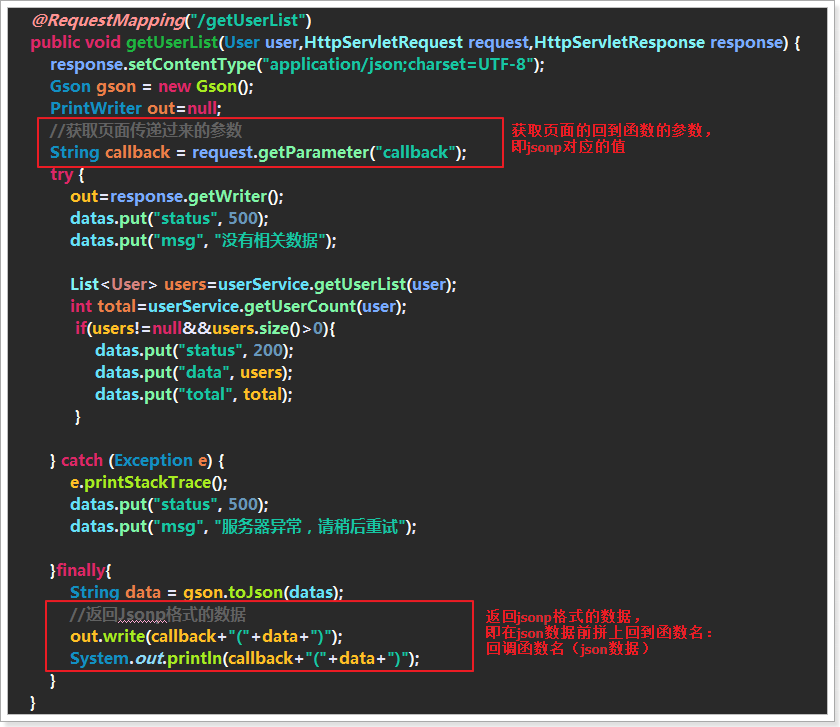
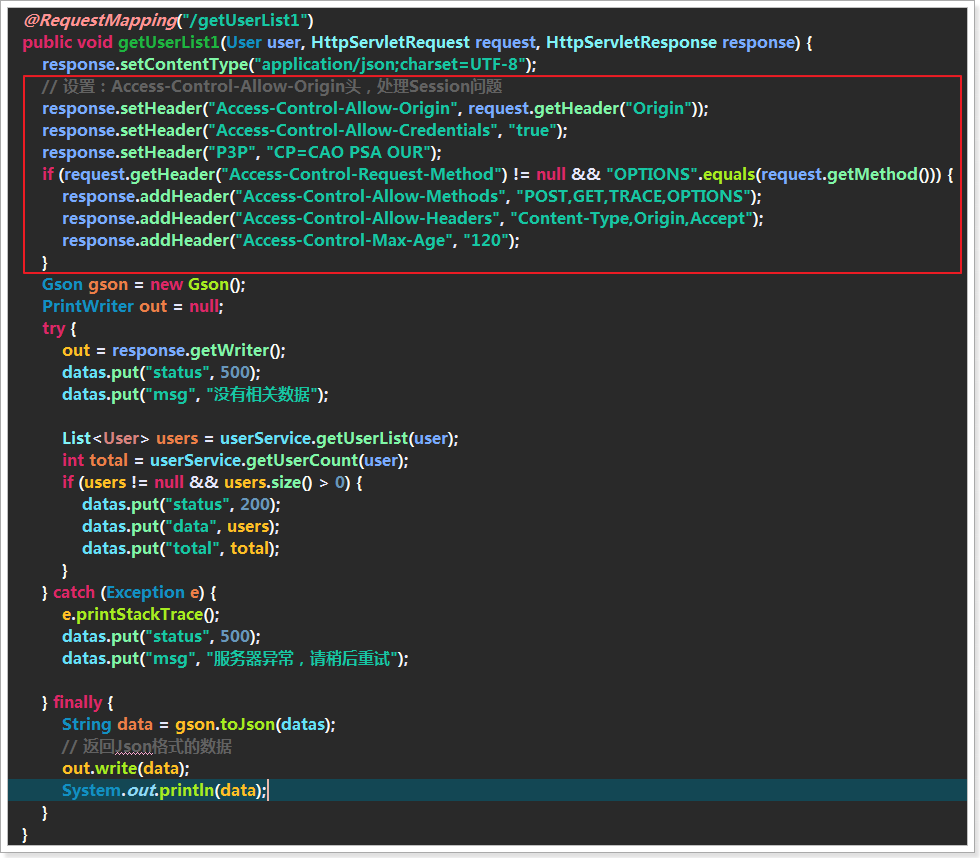
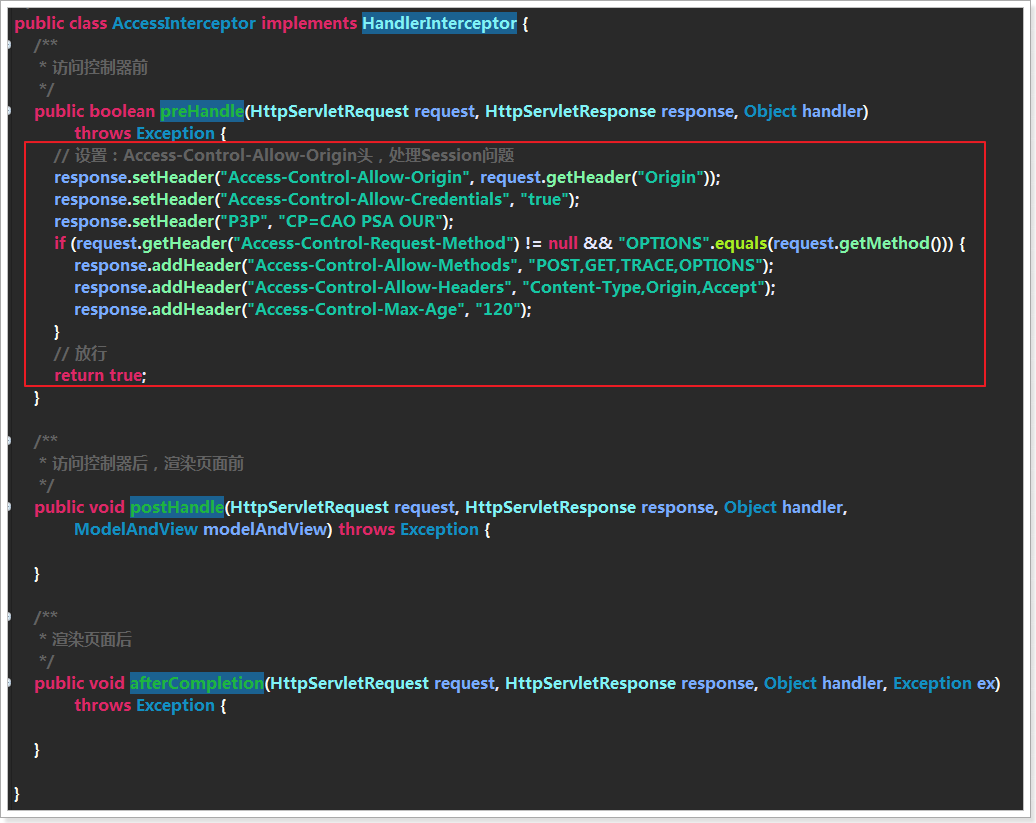
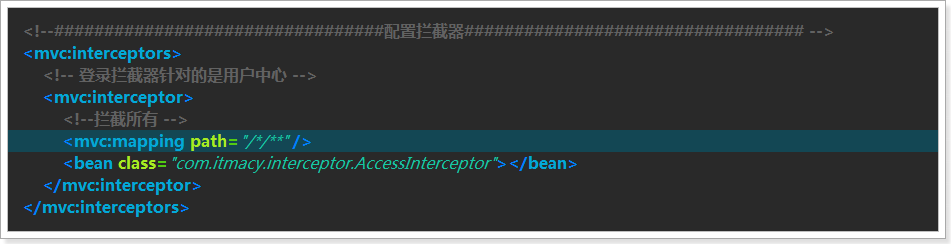
自：<https://www.cnblogs.com/itmacy/p/6958181.html>

# [跨域问题：解决跨域的三种方案](https://www.cnblogs.com/itmacy/p/6958181.html)

当前端页面与后台运行在不同的服务器时，就必定会出现跨域这一问题，本篇简单介绍解决跨域的三种方案，部分代码截图如下，仅供参考：  
**方式一：使用ajax的jsonp**  
前端代码  
  
 服务器代码  
  
 使用该方式的缺点：请求方式只能是get请求  
  
**方式二：使用jQuery的jsonp插件**  
插件下载网址：<https://github.com/jaubourg/jquery-jsonp>  
前端代码  


服务器代码  
  
 使用该方式的特点：与方式一相比，请求方式不只局限于get请求，还可以是post请求，但从服务器从获取的数据依然是jsonp格式  
  
**方式三：使用cors**  
前端代码  


服务器代码  
  
使用该方式的特点：与前两种方式相比，前端代码和未处理跨域前一样，即普通的ajax请求，但服务器代码添加了一段解决跨域的代码  
    // 设置：Access-Control-Allow-Origin头，处理Session问题  
        response.setHeader("Access-Control-Allow-Origin", request.getHeader("Origin"));  
        response.setHeader("Access-Control-Allow-Credentials", "true");  
        response.setHeader("P3P", "CP=CAO PSA OUR");  
        if (request.getHeader("Access-Control-Request-Method") != null && "OPTIONS".equals(request.getMethod())) {  
            response.addHeader("Access-Control-Allow-Methods", "POST,GET,TRACE,OPTIONS");  
            response.addHeader("Access-Control-Allow-Headers", "Content-Type,Origin,Accept");  
            response.addHeader("Access-Control-Max-Age", "120");  
        }  
  
  
**cors高级使用：在springmvc中配置拦截器**  
创建跨域拦截器实现HandlerInterceptor接口，并实现其方法，在请求处理前设置头信息，并放行  
  
在springmvc的配置文件中配置拦截器，注意拦截的是所有的文件  


[#1楼](https://www.cnblogs.com/itmacy/p/6958181.html#3709321) 2017-06-07 17:52 [我在马路边](https://www.cnblogs.com/kerbside/) 

mark

[支持(0)反对(0)](javascript:void(0);)

[#2楼](https://www.cnblogs.com/itmacy/p/6958181.html#3709343) 2017-06-07 18:20 [田想兵](https://www.cnblogs.com/tianxiangbing/) 

加一个，可以用ngix代理

[支持(0)反对(0)](javascript:void(0);)

[#3楼](https://www.cnblogs.com/itmacy/p/6958181.html#3709365)[楼主] 2017-06-07 18:57 [itmacy](https://www.cnblogs.com/itmacy/) 

[@](https://www.cnblogs.com/itmacy/p/6958181.html#3709343) 田想兵  
谢谢提醒。。。