Angular4项目分为**工程代码**和生产代码,在本地调试一般都是工程代码,这样联调接口的话,每次写一个接口都要丢到服务器上测试,严重影响效率,所以我们需要做的就是能在工程项目上联调接口,随时能看到效果,但是后端代码不是部署在本机的话就会有跨域问题,于是我们便需要去着重去解决跨域问题!这样后端代码随时改,前端也可以随时更改看到效果,实现真正的前后端分离!

对于Angular4解决跨域问题,应该是开发者已经想到这个问题,所以解决这个问题很简单!那就是反向代理!!

下面介绍反向代理的方法:

1.首先，需要在项目的根目录（与package.json一级的目录下）建立一个JSON文件,文件名”proxy.config.json”

{  
 "/api": {  
 "target": "http://192.168.1.163:8084/",   
 "pathRewrite": {"^/api" : ""}   
 }  
}

**/api 是代理的名称，一般都是请求的ip地址后面的第一个参数**

比如angular服务的IP地址为：192.168.1.112:4201；

访问 ：<http://192.168.1.112:4201/api/login>；  
实际访问就会变为 <http://192.168.1.163:8084/login>；注意“/api”消失了，因为

“pathRewrite”的值将“/api”去掉了

2. 然后配置”package.json”文件  
（ --proxy-config proxy.config.json）

"scripts": {  
 "ng": "ng",  
 "start": "ng serve",  
 "start-shen": "ng serve --host 192.168.1.112 --port 4201 --proxy-config proxy.config.json",  
 "build": "ng build --prod --env=prod --aot",  
 "test": "ng test",  
 "lint": "ng lint",  
 "e2e": "ng e2e"  
},

# Golang语言签名bug

Gologn 后端语言会将json字符串中的“&”当做url来接续，在返回的时候会将“&”转义成Unicode码“\u2006”; 如果前后端都需要进行签名验证的时候需要注意。Golang将“\u2006”作为6个字符。前端会当做一个字符。

# Angular url中是否带“#”

url是否有#号，由useHash字段确定 true为有#号，false为无#号

当后端用nginx代理的解决跨域的时候，url中必须用#号，否则nginx 找不到url对应的路径报404错误

@NgModule({  
 imports: [  
 RouterModule.*forRoot*(  
 appRoutes,  
 {  
 enableTracing: **false**, // <-- debugging purposes only  
 useHash: **true**// true：有“#”；false：无“#”}  
 )  
 ],  
 exports: [  
 RouterModule  
 ],  
 providers: []  
})  
**export class** AppRoutingModule { }