

2-11 创建OAuth2Realm类

140.143.132.225:8000/project-1/doc-23

`OAuth2Realm` 类是 `AuthorizingRealm` 的实现类，我们要在这个实现类中定义认证和授权的方法。因为认证与授权模块设计到用户模块和权限模块，现在我们还没有真正的开发业务模块，所以我们这里先暂时定义空的认证去授权方法，把Shiro和JWT整合起来，在后续章节我们再实现认证与授权。

在 `com.example.emos.wx.config.shiro` 中创建 `OAuth2Realm` 类。

```
1. package com.example.emos.wx.config.shiro;
2. import org.apache.shiro.authc.*;
3. import org.apache.shiro.authz.AuthorizationInfo;
4. import org.apache.shiro.authz.SimpleAuthorizationInfo;
5. import org.apache.shiro.realm.AuthorizingRealm;
6. import org.apache.shiro.subject.PrincipalCollection;
7. import org.springframework.beans.factory.annotation.Autowired;
8. import org.springframework.stereotype.Component;
9. import java.util.Set;
10. @Component
11. public class OAuth2Realm extends AuthorizingRealm {
12.     @Autowired
13.     private JwtUtil jwtUtil;
14.     @Override
15.     public boolean supports(AuthenticationToken token) {
16.         return token instanceof OAuth2Token;
17.     }
18.     /**
19.      * 授权(验证权限时调用)
20.      */
21.     @Override
22.     protected AuthorizationInfo doGetAuthorizationInfo(PrincipalCollection principals) {
23.         SimpleAuthorizationInfo info = new SimpleAuthorizationInfo();
24.         //TODO 查询用户的权限列表
25.         //TODO 把权限列表添加到info对象中
26.         return info;
27.     }
28.     /**
29.      * 认证(登录时调用)
30.      */
31.     @Override
32.     protected AuthenticationInfo doGetAuthenticationInfo(AuthenticationToken token) throws
AuthenticationException {
33.         //TODO 从令牌中获取userId，然后检测该账户是否被冻结。
34.         SimpleAuthenticationInfo info = new SimpleAuthenticationInfo();
35.         //TODO 往info对象中添加用户信息、Token字符串
36.         return info;
37.     }
38. }
```

