Shiro框架的简单认证（入门）：

**public class** AuthenticationTest {  
 SimpleAccountRealm **simpleAccountRealm**=**new** SimpleAccountRealm();  
 @Before  
 **public void** addUser(){  
 **simpleAccountRealm**.addAccount(**"mark"**,**"123456"**,**"admin"**,**"role1"**);//后两个是角色

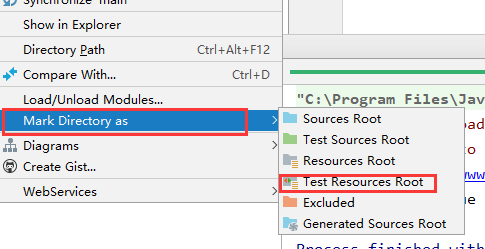
}  
 @Test  
 **public void** testAuthentication(){  
 DefaultSecurityManager defaultSecurityManager=**new** DefaultSecurityManager();  
 defaultSecurityManager.setRealm(**simpleAccountRealm**);  
  
 SecurityUtils.*setSecurityManager*(defaultSecurityManager);  
 Subject subject= SecurityUtils.*getSubject*();  
  
 UsernamePasswordToken tocken=**new** UsernamePasswordToken(**"mark"**,**"12456"**);  
 subject.login(tocken);  
 System.***out***.println(**"isAuthenticated:"**+subject.isAuthenticated());

*//授权（检查角色）*subject.checkRoles(**"admin"**,**"role21"** );

subject.logout();  
 }  
}

自定义ini 型realm

在项目中建立resources文件夹，并右击文件夹：



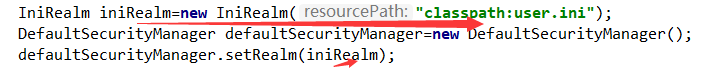
再在文件夹下新建文件user.ini：

[users]  
mark=123456,admin,role1  
[roles]  
admin=user:delete,user:update

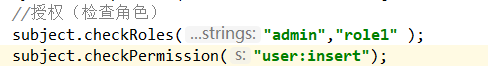
其中第二行的admin和role1是角色

第四行的user：delete等是权限

然后就可以声明并使用了：



Check授权代码：



内置一一对应的jdbcrealm：



这个是固定了表和字段名（jdbcRealm默认的）

自定义在testAuthentication方法中加入如下：

这就定义了认证表为：user1 字段名为password和userName。授权的两个同理。