**今天学习的内容：**

1、 权限概述（认证、授权）

2、 常见的权限控制的方式（url拦截、方法注解权限控制）

3、 权限数据模型（权限表、角色表、用户表、角色权限关系表、用户角色关系表）

4、 Apache Shiro框架----权限控制框架

5、 在BOS项目中应用shiro框架进行权限控制

## 01权限概述（认证、授权）.avi

系统中提供了很多功能，并不是每个人登录系统之后，可以操作的功能是完全相同的。需要对系统的功能的访问进行控制。

**认证：**系统提供的用于识别用户身份的功能（通常就是登录功能）-----解决你是谁的问题

**授权：**系统中提供很多功能，需要为不同的用户根据其对应的权限，授予用户操作某些功能的能力------解决你能做什么的问题

## **02常见的权限控制方式（url拦截）.avi**

重点： 基于请求响应模式的web项目，可以用过滤器或拦截器拦截url请求，在过滤器或拦截器中判断当前用户是否具有访问当前功能的权限.如果有就放行，如果没有跳转到权限不足提示页面。

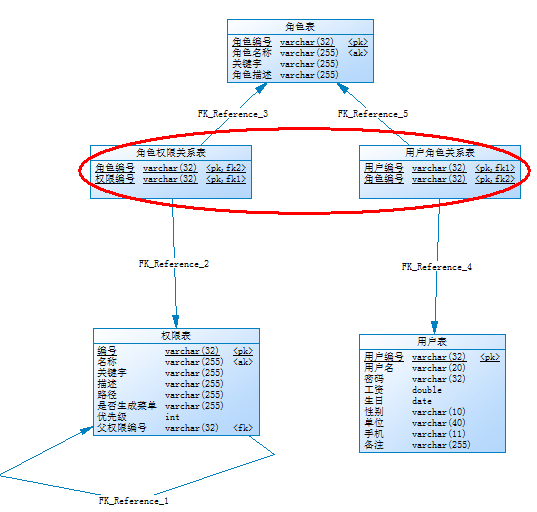
## 03**常见的权限控制方式（方法注解权限控制）**.avi

## 

重点：如果Action的方法加了注解，就会为当前Action创建代理对象，请求该方法时会判断当前用户是否具有注解上所需的权限(staff-delete)，如果有就通过反射调用该方法，如果没有抛出异常，阻止调用

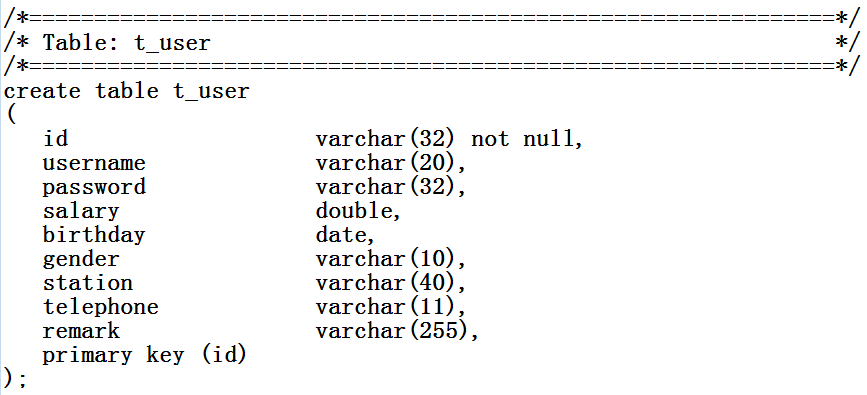
## 04权限数据模型（生成类和hbm）.avi

需要用到5张表，分别是：权限表，角色表，用户表，角色权限关系表，用户角色关系表

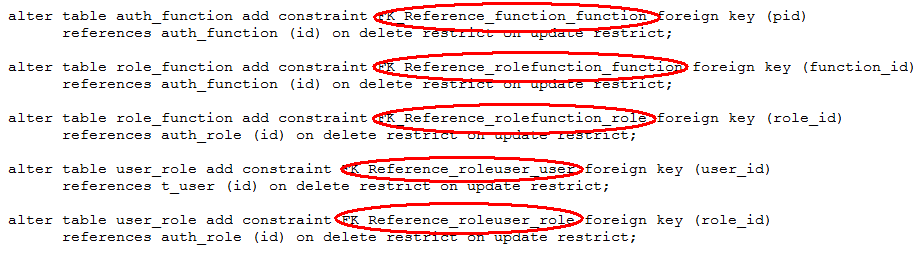


重点：

1. 角色权限关系表是用于描述角色与权限多对多关系的中间表，其中的字段既是主键又是外键，用户角色关系表也是类似的
2. 角色可以认为是一组权限的集合
3. 权限表中使用了自关联(父权限编号引用编号)
4. 由于用户表在数据库中已经存在了，pdm文件导出的sql文件需要作出修改，修改如下：
5. 删除drop user表语句：
6. 删除创建user表的语句：



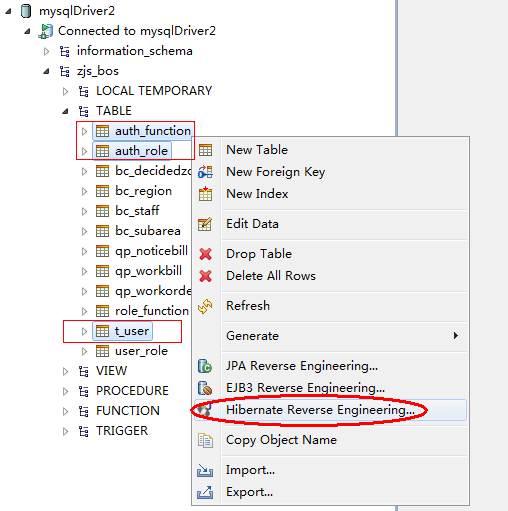
1. 由于约束名不能重复，需要修改外键约束名

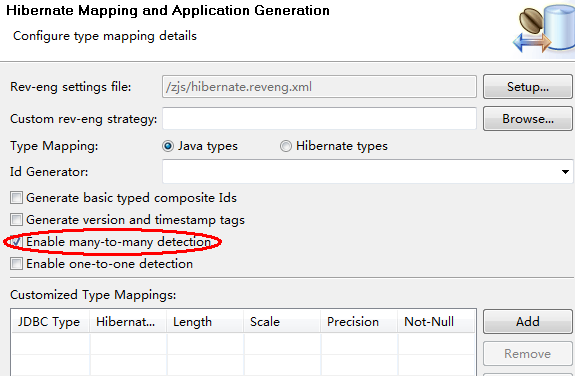


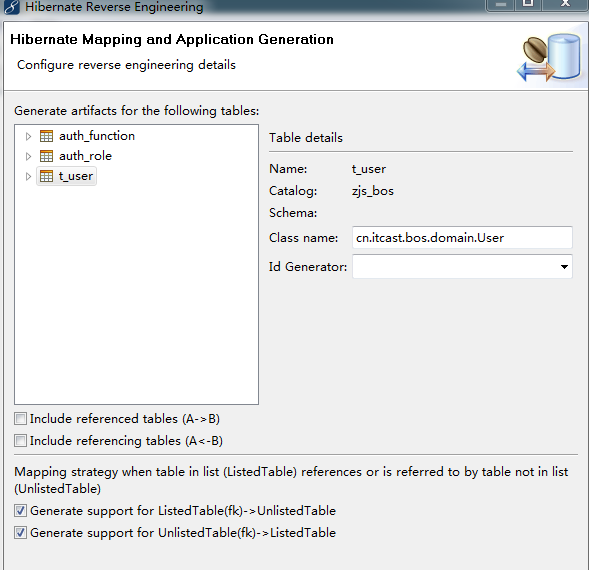
1. 使用MyEclipse反向生成实体类和映射文件时,注意以下几点：

a.备份原来的User.hbm.xml文件 (例如把里面的代码拷贝到另一个txt文件里)

b.不需要把关系表生成类，直接选中权限表，角色表和用户表进行反向生成







c.把备份的User.hbm.xml里的两个query标签拷贝到重新生成的User.hbm.xml里面

## 05 shiro框架简介.avi

Shiro框架提供的权限控制的方式：

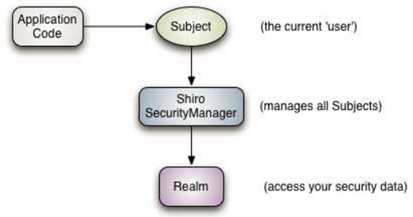
* **url拦截进行权限控制----框架提供了很多过滤器进行url拦截**
* **方法注解进行权限控制----框架提供了在方法上使用的注解，为Action创建代理对象，进行权限控制**
* **页面标签权限控制----框架提供一套标签，用于根据权限展示或者隐藏按钮或者菜单**
* **代码级别权限控制-----框架提供API，使用编程方式进行权限控制(不建议使用)**

Shiro框架下载：

Shiro.apache.org地址下载



Shiro框架执行流程：

\

几个概念：

**Application Code：**应用程序代码，由开发人员负责开发

**Subject：**当前用户

**SecurityManager：**安全管理器，管理所有的用户，认证、授权等。

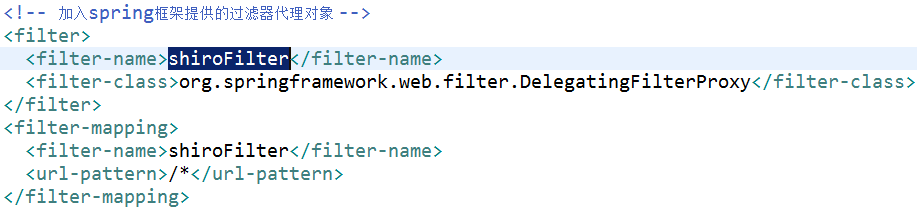
**Realm：**安全数据桥，类似于Dao，负责访问安全数据

## 06在项目中使用shiro框架\_1.avi

使用shiro框架进行项目的认证和授权操作

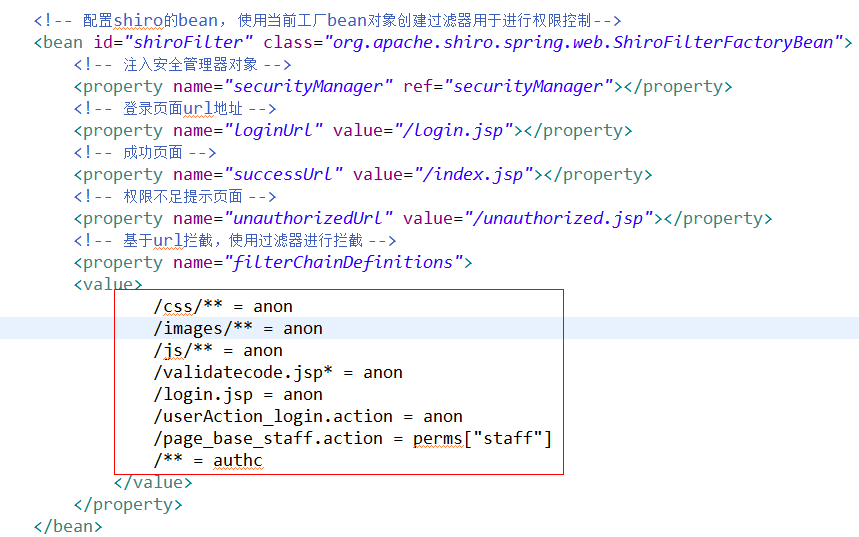
**第一步：**导入shiro的jar到项目中

**第二步：**在web.xml中配置一个过滤器代理对象，在项目启动时到spring工厂中加载一个和当前过滤器name同名的bean对象



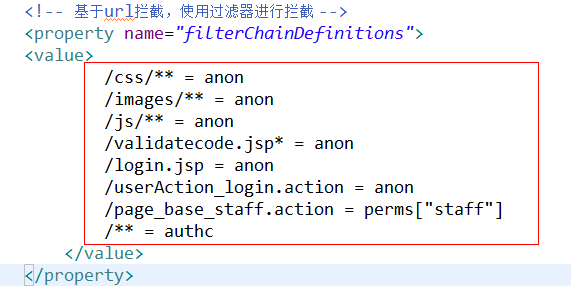
**第三步：**在spring配置文件中配置一个名称为shiroFilter的bean，这里使用过滤器进行拦截，

/css/\*\*=anon表示匿名用户可以访问css目录下的所有文件





## 07在项目中使用shiro框架\_2.avi



基于url拦截时使用的常用的过滤器及资源匹配规则语法：

能匹配访问到的资源=shiro过滤器

annno: 匿名(无须经过认证) authc:要经过认证(使用shrio的方式认证)

perms[‘权限名’] :要拥有指定的权限名

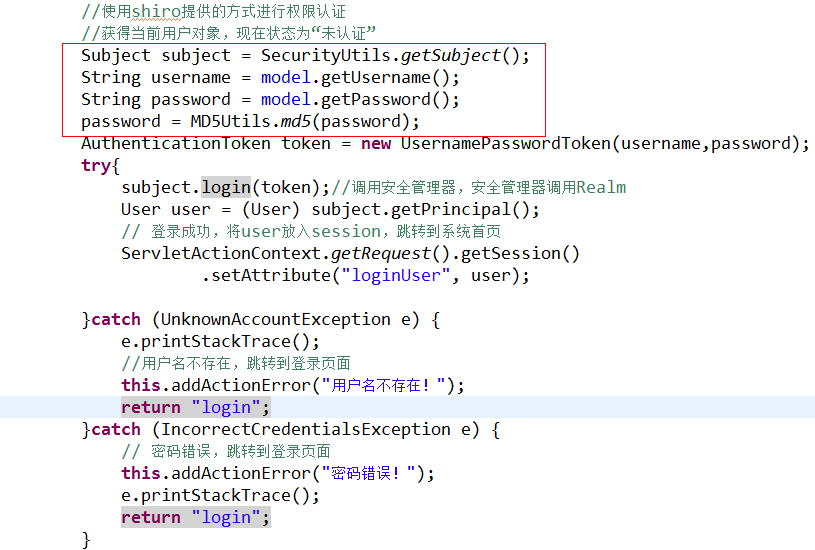
例如：/js/\*\*=anno ：任务用户都可以访问js及其子目录下的文件

/easydemo/\*\*=ahthc :认证通过的用户才可以访问easydemo及其子目录下的文件

**需要使用shrio提供的方式进行认证**，要修改UserAction的login方法，否则无法跳转到首页

## 08使用shiro方式进行认证操作.avi

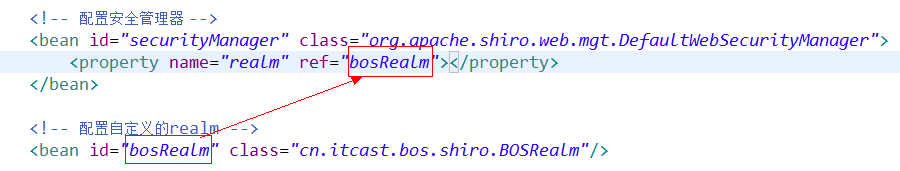
**第一步：**修改UserAction的登录方法login(),使用shiro提供的方式进行验证:



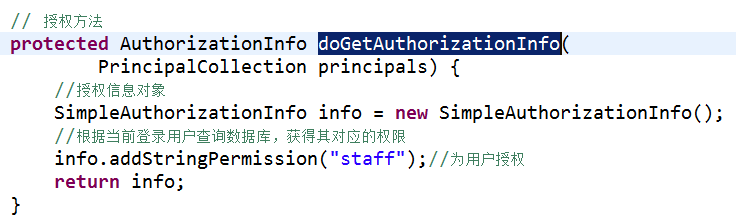
**第二步：**自定义一个Realm，进行认证和授权操作



**第三步：**注册上面自定义的Realm,并注入给安全管理器

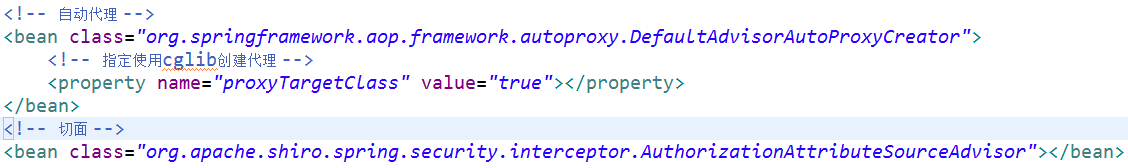


## 09 在realm中进行授权操作.avi

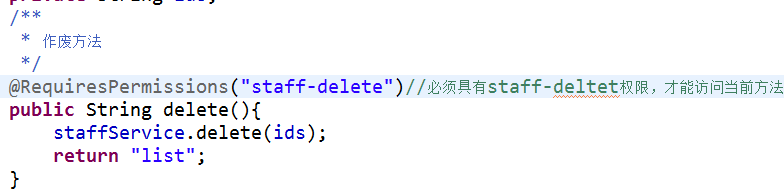


## 10 shiro方法注解进行权限控制.avi

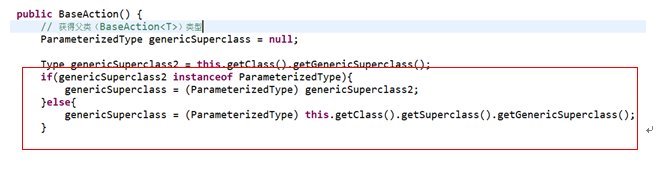
第一步：在spring配置文件中配置自动代理和切面



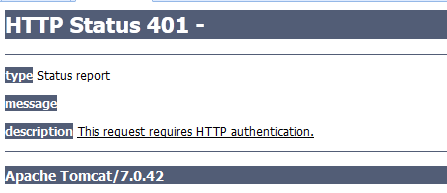
第二步：在Action方法上使用注解



第三步：修改BaseAction的构造方法



注意：在spring配置文件中指定了url拦截的规则，需要某个权限才能访问某个url，那么如果在Realm类中没有授予相应的权限**也没有指定权限不足的提示页面**，会报出状态码为401的页面；



如果在方法上使用注解指定需要某个权限，而在Realm类中没有授予相应的权限,则访问该方法时会报出权限不足的错误

