

**BOS物流项目**

# 教学计划

1、演示权限demo

2、权限概述

* + 认证
  + 授权

3、常见的权限控制方式

* + url拦截权限控制
  + 方法注解权限控制

4、创建权限数据模型

* + 权限表
  + 角色表
  + 用户表
  + 角色权限关系表
  + 用户角色关系表

5、apache shiro框架简介

* + 官网和下载
  + 框架核心功能
  + 框架调用流程

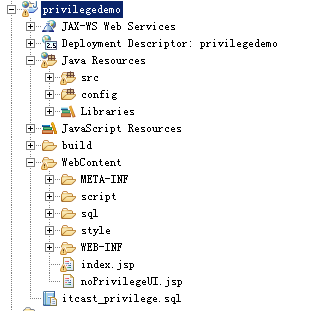
6、基于shiro框架进行认证操作

# 演示权限demo



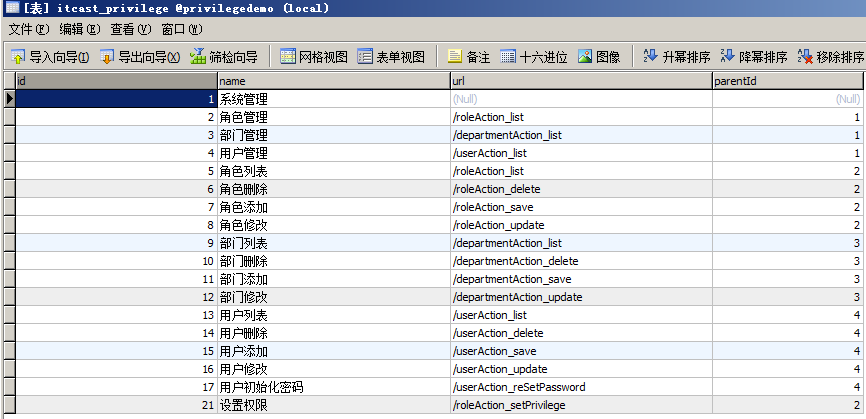
如何运行项目：

第一步：在eclipse中引入上面的项目

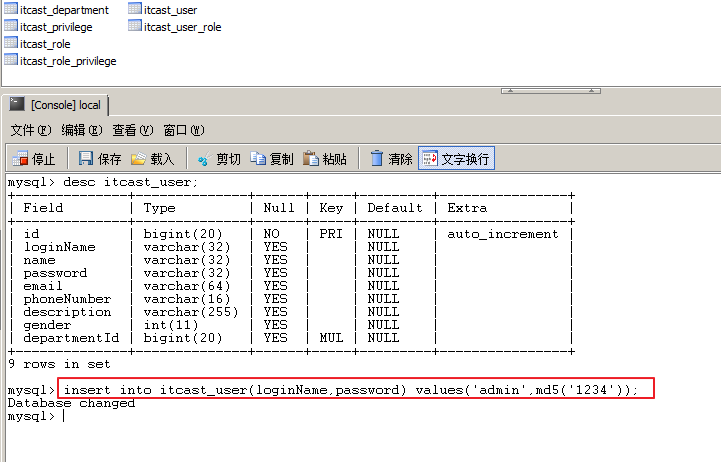


第二步：创建一个数据库privilegedemo，并执行项目中的sql脚本





第三步：启动项目，可以完成自动建表，为itcast\_user表插入一条初始化数据



# 权限概述

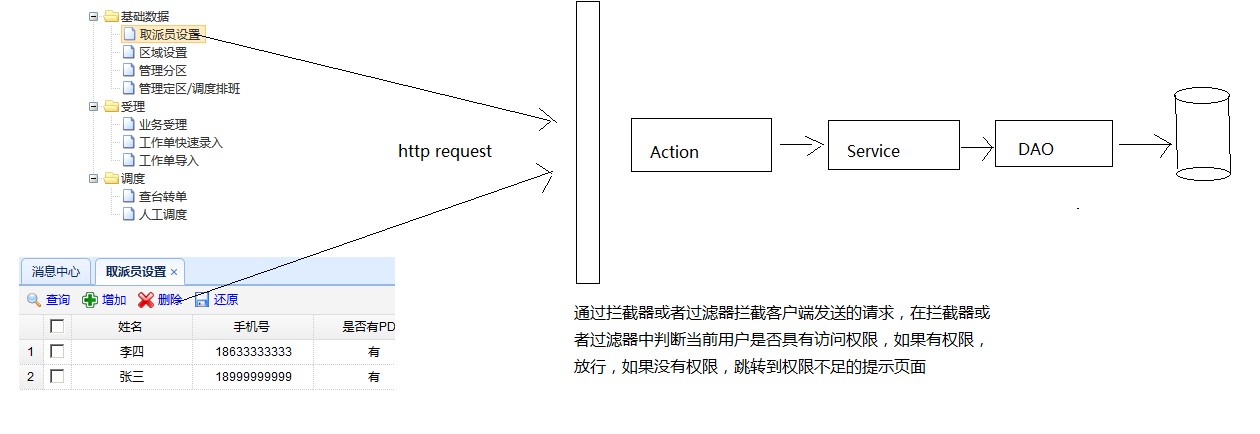
认证：系统提供的用于识别用户身份的功能，通常登录功能就是认证功能-----让系统知道你是谁？？

授权：系统授予用户可以访问哪些功能的许可（证书）----让系统知道你能做什么？？

# 常见的权限控制方式

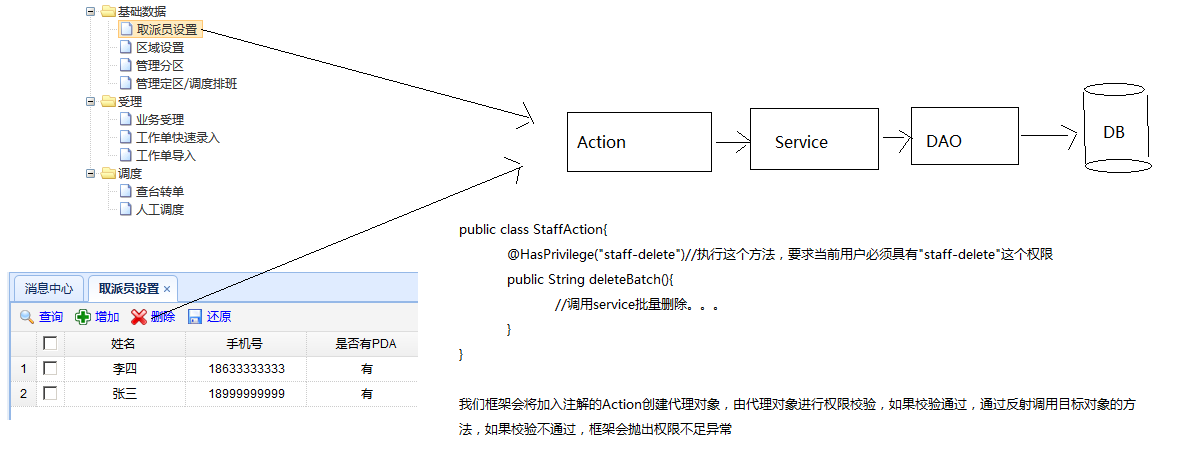
## URL拦截权限控制

底层基于拦截器或者过滤器实现



## 方法注解权限控制

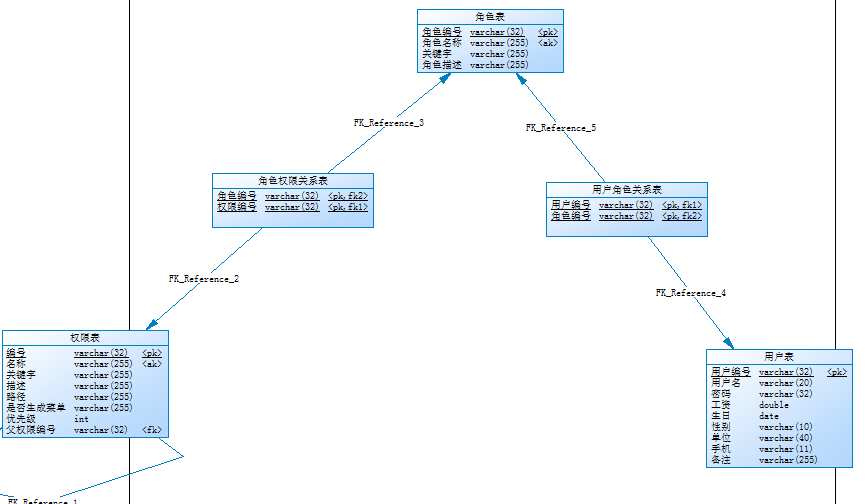
底层基于代理技术实现，为Action创建代理对象，由代理对象进行权限校验



# 创建权限数据模型

* 权限表
* 角色表
* 用户表
* 角色权限关系表
* 用户角色关系表

角色就是权限的集合，引入角色表，是为了方便授权



# apache shiro框架简介

官网：shiro.apache.org



* 下载文件：



* shiro框架的核心功能：

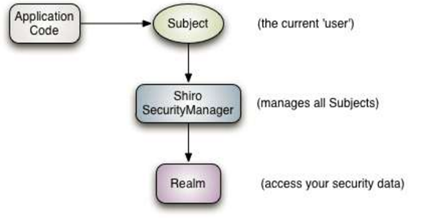
认证

授权

会话管理

加密

* shiro框架认证流程

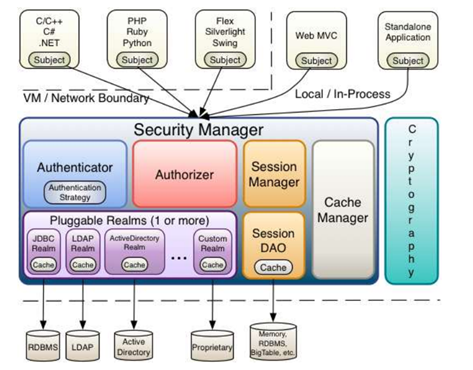


Application Code：应用程序代码，由开发人员负责开发的

Subject：框架提供的接口，代表当前用户对象

SecurityManager：框架提供的接口，代表安全管理器对象

Realm：可以开发人员编写，框架也提供一些，类似于ＤＡＯ，用于访问权限数据

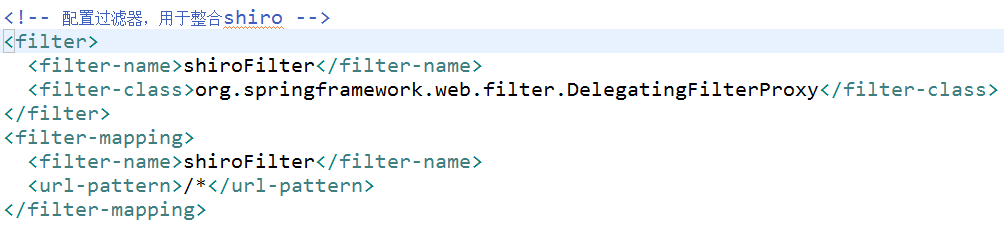


# 在BOS项目中应用shiro框架进行认证

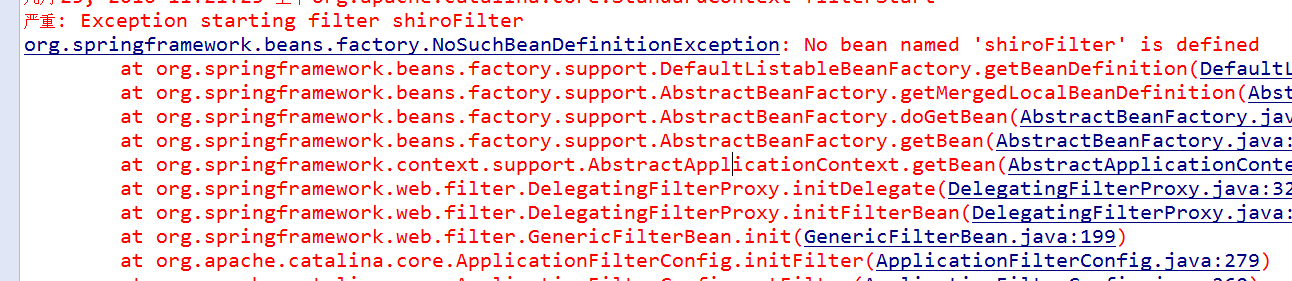
第一步：引入shiro框架相关的jar



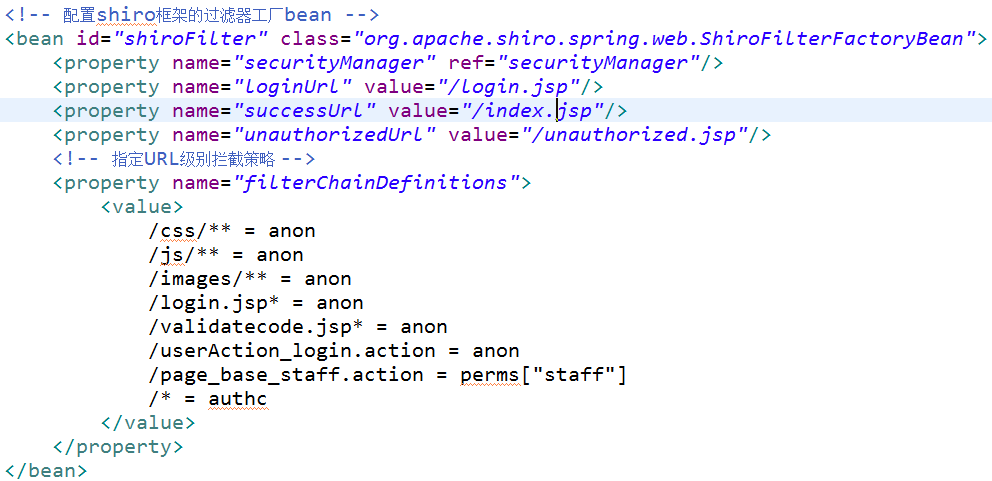
第二步：在web.xml中配置spring框架提供的用于整合shiro框架的过滤器



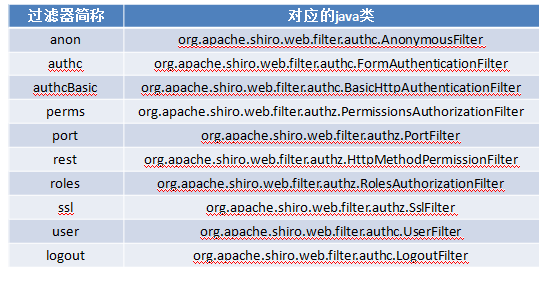
启动tomcat服务器，抛出异常：spring工厂中不存在一个名称为“shiroFilter”的bean对象



第三步：在spring配置文件中配置bean，id为shiroFilter



框架提供的过滤器：



第四步：配置安全管理器



第五步：修改UserAction中的login方法，使用shiro提供的方式进行认证操作



第六步：自定义realm，并注入给安全管理器

**public** **class** BOSRealm **extends** AuthorizingRealm{

@Autowired

**private** IUserDao userDao;

//认证方法

**protected** AuthenticationInfo doGetAuthenticationInfo(AuthenticationToken token) **throws** AuthenticationException {

System.***out***.println("realm中的认证方法执行了。。。。");

UsernamePasswordToken mytoken = (UsernamePasswordToken)token;

String username = mytoken.getUsername();

//根据用户名查询数据库中的密码

User user = userDao.findUserByUserName(username);

**if**(user == **null**){

//用户名不存在

**return** **null**;

}

//如果能查询到，再由框架比对数据库中查询到的密码和页面提交的密码是否一致

AuthenticationInfo info = **new** SimpleAuthenticationInfo(user, user.getPassword(), **this**.getName());

**return** info;

}

//授权方法

**protected** AuthorizationInfo doGetAuthorizationInfo(PrincipalCollection principals) {

// **TODO** Auto-generated method stub

**return** **null**;

}

