# 登录成功后到了欢迎登录

# 1.spring可以启用注解的方式来配置属性.

## 1.1 配置EmployeeService对象通过注解的方式来注入属性值,这是一个知识点

<!-- 配置EmployeeService对象通过注解的方式来注入属性值,这是一个知识点 -->

<bean id=*"employeeService"* class=*"com.hsp.service.imp.EmployeeService"*/>

## 1.2.到了EmployeeService中，下面的setSessionFactory是不会调用了

//当我们给某个属行 resource 时候 spring 就会通过byName 自动注入属行值

@Resource

**public** SessionFactory sessionFactory;

## 1.3.然后启用注解扫描

<!-- 启用注解扫描 -->

<context:annotation-config/>

<!-- 配置EmployeeService对象通过注解的方式来注入属性值,这是一个知识点 -->

<bean id=*"employeeService"* class=*"com.hsp.service.imp.EmployeeService"*/>

# 2.action中也使用 注解的方式来注入

**public** **class** LoginAction **extends** DispatchAction {

//当Struts 配spring接管之后就可以通过这种方式来被调用了

@Resource

EmployeeServiceInter employeeService;

# 3.如何解决懒加载问题

## 1.如果我们的雇员都属于一个部门

### 1.1新建一个部门department

**public** **class** Department {

**private** Integer id;

**private** String name;

**private** Set<Employee> emps;

### 1.2雇员中新建一个 Department department

<!-- 一个雇员属于一个部门 -->

<many-to-one name=*"department"* column=*"department\_id"* />

### 1.3配置雇员的配置文件

### 1.4配置部门的配置文件，级联

<!-- 一个部门可以有多个雇员 -->

<set name=*"emps"* cascade=*"save-update"* >

<key column=*"department\_id"* />

<one-to-many class=*"com.hlj.domain.Employee"*/>

</set>

### 1.5.然后删掉我们之前的表，重新添加用户这里就是需要将

**public** **class** DepartmentService **implements** DepartmentServiceInter {

@Resource

SessionFactory sessionFactory;

@Override

**public** **void** addDepartments(Department department) {

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

sessionFactory.getCurrentSession().save(department);

}

}

### 1.6.然后配置在spring容器中的东西

<bean id=*"departmentService"* class=*"com.hlj.service.imp.DepartmentService"*/>

## 2.解决懒加载的方法一

**public** Employee check(Employee e) {

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

String hql = "from Employee where id = ? and pwd = ?";

List<Employee> list = sessionFactory

.getCurrentSession().createQuery(hql)

.setInteger(0, e.getId()).setString(1, e.getPwd()).list();

//解决懒加载

Hibernate.*initialize*(Department.**class**);

**if**(list.size()==1){

String department = list.get(0).getDepartment().getName();

System.*out*.println(department+"#########################");

**return** list.get(0);

}**else**{

**return** **null**;

}

}

## 2.2第三种解决方案本质还是一种过滤器web中设置

<!-- spring下面中解决懒加载 -->

<filter>

<filter-name>OpenSessionInViewFilter</filter-name>

<filter-class>org.springframework.orm.hibernate3.support.OpenSessionInViewFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>OpenSessionInViewFilter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

该方法可以有效的减少对数据库的查询，缺点是和数据保持的session,时间延长了.