

说明文档

SSM整合

搭建Mybatis环境

1. 引入相关坐标
 1. mybatis: mybatis框架包
 2. mysql-connector-java: mysql包
 3. junit: 单元测试包
2. 在domain包下创建对象的实体类（com.zichen.domain）。属性名可以使用驼峰命名，但需要在mybaits的配置文件进行设置。
 1. Department
 2. Employee: 注意需要加上department这个变量，因为department和employee是一对多的关系，department属于外键，在查询employee信息时，要通过和进行多表查询。
3. 创建Dao接口：定义增删该查的方法（com.zichen.dao）
4. 在resources包下创建jdbc.properties文件，用来配置数据库的基本信息
5. 在resources包下创建mybatisConfig.xml文件，用来配置Mybatis的环境
 1. 引入jdbc.properties文件
 2. 开启驼峰映射
 3. 设置别名
 4. 配置Mybatis环境
 5. 加载映射文件
6. 在resources包下创建接口的映射文件（com.zichen.dao）
 1. 在创建employeeDao的映射文件时，要通过和进行多表查询。

```
...  
  
    <resultMap id="employeeMap" type="Employee">  
        <id property="empId" column="emp_id"></id>  
        <result property="empName" column="emp_name"></result>  
        <result property="empDepartId" column="emp_depart_id"></result>  
        <result property="empPosition" column="emp_position"></result>  
        <result property="empStartDate" column="emp_start_date"></result>  
        <result property="empTel" column="emp_tel"></result>  
  
        <association property="department" javaType="Department">  
            <id property="departId" column="emp_depart_id"></id>  
            <result property="departName" column="depart_name"></result>
```

```
        <result property="departManager" column="depart_manager">
    </result>
        <result property="managerTel" column="manager_tel"></result>
        <result property="managerEmail" column="manager_email">
    </result>
    </association>
</resultMap>
...

```

7. 编写测试类进行测试

1. 创建sqlSessionFactory对象
2. 获取sqlSession
3. 获取employeeDao和departmentDao的代理对象
4. 调用方法，进行测试

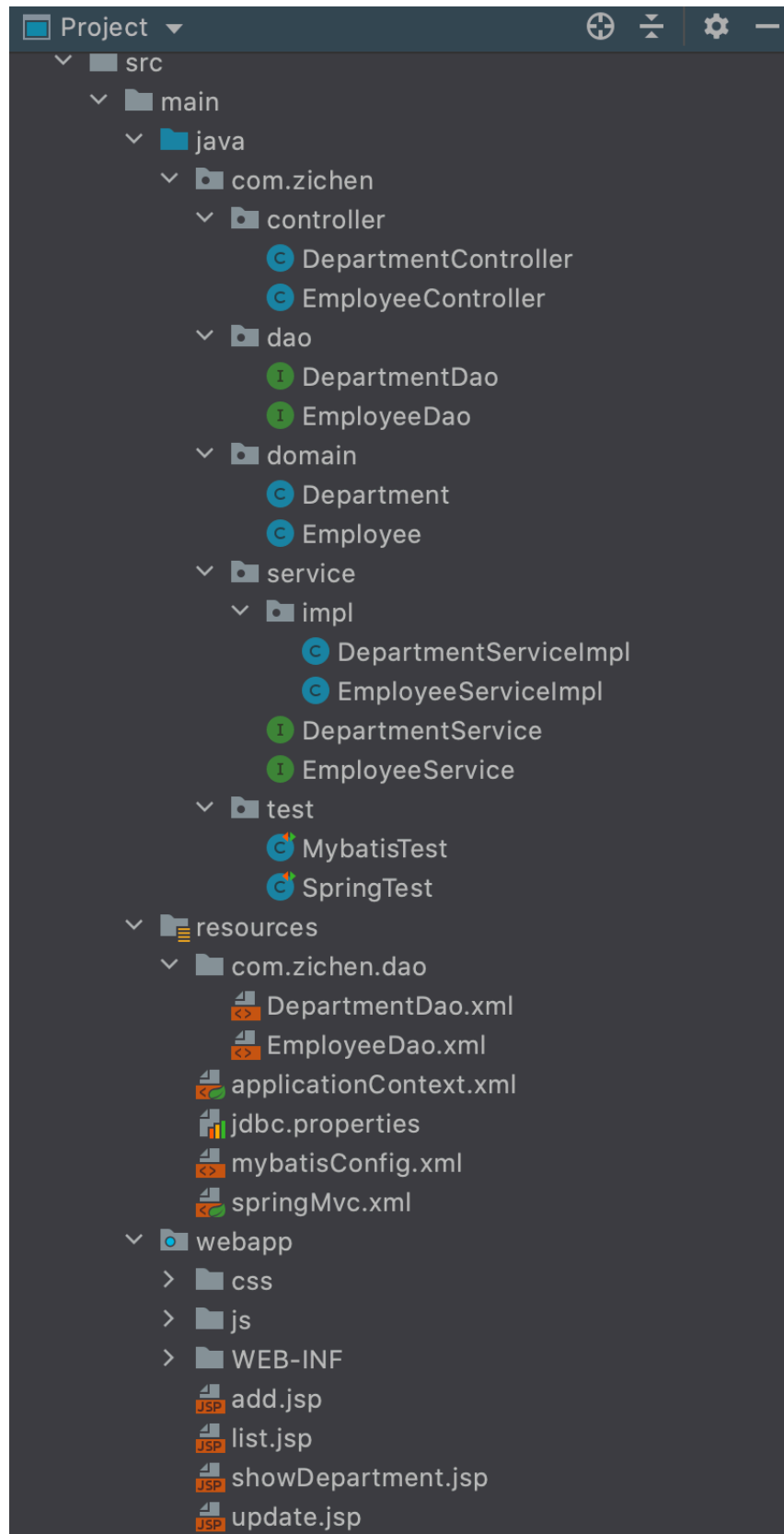
搭建Spring环境

1. 引入相关坐标
 1. spring-context: spring开发环境
 2. spring-jdbc: spring使用jdbcTemplate对jdbc的API进行封装
 3. aspectjweaver: 切面编程
 4. spring-test: spring测试
 5. spring-tx: 事务控制
 6. druid: 数据源
 7. mybatis-spring: 整合spring和mybatis
2. 在service包下面创建接口和实现类（com.zichen.service）
3. 创建applicationContext.xml文件，用于配置Spring
 1. 开启注解扫描
 2. 整合spring和mybatis：把创建代理对象的权利交给spring
 1. 引入jdbc.properties文件
 2. 设置数据源
 3. 获取代理对象
 1. 创建sqlSessionFactory对象，并引入数据源和mybatisConfig.xml文件
 2. 开启mapper映射扫描：对接口进行扫描，并产生代理对象，存到IOC容器中
4. 补全service实现类中的方法
 1. 注入employeeDao和departmentDao对象
 2. 分别调用employeeDao和departmentDao中的方法
5. 编写测试类进行测试

搭建springMVC环境

1. 引入相关坐标
 1. spring-mvc: mvc开发环境
 2. javax.servlet-api: servlet
 3. jsp-api: jsp
 4. jstl: el表达式
2. 配置springMvc.xml文件
 1. 组件扫描
 2. mvc注解增强
 3. 视图解析器
 4. 放行静态资源
3. 配置web.xml文件
 1. 配置前端控制器，并初始化加载springMvc.xml配置文件
 2. 中文乱码问题
 3. 整合spring和web容器，并加载spring配置文件applicationContext.xml
4. 在controller包下实现url映射

目录展示



结果展示

1. 员工展示

localhost:8080/cmsProject/employee/findAll

员工信息表

<input type="checkbox"/>	编号	姓名	职位	入职时间	联系方式	所属部门编号	操作
<input type="checkbox"/>	1	王自凡	客户经理	2015-08-08	18792750497	1	查询部门信息 修改员工信息
<input type="checkbox"/>	2	王子建	销售经理	2015-09-09	18792750498	2	查询部门信息 修改员工信息
<input type="checkbox"/>	3	王建国	客户经理	2015-08-08	18792750499	2	查询部门信息 修改员工信息
<input type="checkbox"/>	13	王健林	经理	2020-01-22	18792750101	2	查询部门信息 修改员工信息
删除选中		添加员工					

2. 添加员工

localhost:8080/cmsProject/add.jsp

添加员工

姓名: 王力宏
职位: 经理
入职时间: 2020-01-22
联系方式: 18792750100
所属部门编号: 2

[提交](#)

localhost:8080/cmsProject/employee/findAll

员工信息表

<input type="checkbox"/>	编号	姓名	职位	入职时间	联系方式	所属部门编号	操作
<input type="checkbox"/>	1	王自凡	客户经理	2015-08-08	18792750497	1	查询部门信息 修改员工信息
<input type="checkbox"/>	2	王子建	销售经理	2015-09-09	18792750498	2	查询部门信息 修改员工信息
<input type="checkbox"/>	3	王建国	客户经理	2015-08-08	18792750499	2	查询部门信息 修改员工信息
<input type="checkbox"/>	13	王健林	经理	2020-01-22	18792750101	2	查询部门信息 修改员工信息
<input type="checkbox"/>	14	王力宏	经理	2020-01-22	18792750100	2	查询部门信息 修改员工信息
删除选中		添加员工					

3. 删除员工

员工信息表

<input type="checkbox"/>	编号	姓名	职位	入职时间	联系方式	所属部门编号	操作
<input type="checkbox"/>	1	王自凡	客户经理	2015-08-08	18792750497	1	查询部门信息 修改员工信息
<input type="checkbox"/>	2	王子建	销售经理	2015-09-09	18792750498	2	查询部门信息 修改员工信息
<input type="checkbox"/>	3	王建国	客户经理	2015-08-08	18792750499	2	查询部门信息 修改员工信息
<input checked="" type="checkbox"/>	13	王健林	经理	2020-01-22	18792750101	2	查询部门信息 修改员工信息
<input checked="" type="checkbox"/>	14	王力宏	经理	2020-01-22	18792750100	2	查询部门信息 修改员工信息
删除选中		添加员工					

localhost:8080/cmsProject/employee/findAll

localhost:8080 says
您确定要删除吗?

☐

编号

姓名

职位

入职时间

联系方式

所属部门编号

操作

☐

1

王自凡

客户经理

2015-08-08

18792750497

1

查询部门信息 修改员工信息

☐

2

王子建

销售经理

2015-09-09

18792750498

2

查询部门信息 修改员工信息

☐

3

王建国

客户经理

2015-08-08

18792750499

2

查询部门信息 修改员工信息

☒

13

王健林

经理

2020-01-22

18792750101

2

查询部门信息 修改员工信息

☒

14

王力宏

经理

2020-01-22

18792750100

2

查询部门信息 修改员工信息

删除选中

添加员工

员工信息表

<input type="checkbox"/>	编号	姓名	职位	入职时间	联系方式	所属部门编号	操作
<input type="checkbox"/>	1	王自凡	客户经理	2015-08-08	18792750497	1	查询部门信息 修改员工信息
<input type="checkbox"/>	2	王子建	销售经理	2015-09-09	18792750498	2	查询部门信息 修改员工信息
<input type="checkbox"/>	3	王建国	客户经理	2015-08-08	18792750499	2	查询部门信息 修改员工信息
删除选中		添加员工					

4. 查询部门信息

员工信息表

<input type="checkbox"/>	编号	姓名	职位	入职时间	联系方式	所属部门编号	操作
<input type="checkbox"/>	1	王自凡	客户经理	2015-08-08	18792750497	1	查询部门信息 修改员工信息
<input type="checkbox"/>	2	王子建	销售经理	2015-09-09	18792750498	2	查询部门信息 修改员工信息
<input type="checkbox"/>	3	王建国	客户经理	2015-08-08	18792750499	2	查询部门信息 修改员工信息
删除选中		添加员工					

localhost:8080/cmsProject/department/findByld?departId=2

部门信息

编号	名称	部门经理	联系方式	邮箱
2	销售部	林旖曦	18792750496	yixilin@163.com

5. 修改员工信息

localhost:8080/cmsProject/employee/findByld?empld=3

更新员工信息

姓名: 王建国

职位: 客户经理

入职时间: 2015-08-08

联系方式: 18792750499

所属部门编号: 2

提交

姓名:	<input type="text" value="王建国"/>
职位:	<input type="text" value="脱口秀"/>
入职时间:	<input type="text" value="2015-08-08"/>
联系方式:	<input type="text" value="18792750499"/>
所属部门编号:	<input type="text" value="2"/>
<input type="button" value="提交"/>	

<input type="checkbox"/>	编号	姓名	职位	入职时间	联系方式	所属部门编号	操作
<input type="checkbox"/>	1	王自凡	客户经理	2015-08-08	18792750497	1	查询部门信息 修改员工信息
<input type="checkbox"/>	2	王子建	销售经理	2015-09-09	18792750498	2	查询部门信息 修改员工信息
<input type="checkbox"/>	3	王建国	脱口秀	2015-08-08	18792750499	2	查询部门信息 修改员工信息
删除选中	添加员工						