

# SSM(spring+spring+Mybatis)大作业--产品管理

## 1.创建表product --- 使用Navicat

### 1.在数据库mydb中创建表sql语句如下:

```
1 DROP TABLE IF EXISTS product;
2 CREATE TABLE product (
3   product_id int(11) PRIMARY KEY AUTO_INCREMENT COMMENT '产品id',
4   product_name varchar(63) DEFAULT NULL COMMENT '产品名称',
5   product_type varchar(2) DEFAULT NULL COMMENT '产品类型: 1-通讯,2-食品,3-图书',
6   can_sale varchar(2) DEFAULT NULL COMMENT '是否上架销售: 0-否,1-是',
7   manufacturer varchar(50) DEFAULT NULL COMMENT '生产厂家'
8 );
```

## 2.创建maven工程

## 3.配置pom.xml和web.xml。直到运行服务器初始界面显示HelloWorld

## 4.创建java包结构

```
1 在src/main/java下, 创建包:
2 com.tkn.ssm.product (存放会员管理模块的所有代码)
3 com.tkn.ssm.product.model (存放模型类)
4 com.tkn.ssm.product.dao (存放内部实现的mapper接口)
5 com.tkn.ssm.product.mapper (存放mapper映射文件)
6 com.tkn.ssm.product.service (存放对外提供的服务接口)
7 com.tkn.ssm.product.service.impl (存放服务接口的实现)
8 com.tkn.ssm.product.controller (存放mvc模型中的控制器)
```

### 4.1.创建相关java业务类

#### 4.1.1.在com.tkn.ssm.product.model 创建模型类 - Product

#### 4.1.2.在com.tkn.ssm.product.dao 创建接口 - ProductMapper

#### 4.1.3.在com.tkn.ssm.product.mapper 创建映射文件 - ProductMapper.xml

##### 1.ProductMapper.xml是实现ProductMapper接口中的方法

```
1 <mapper namespace="com.tkn.ssm.product.dao.ProductMapper">
2   /*映射*/
3   <!--type的属性要在mybatis.xml中起别名或者用全路径-->
4   <resultMap id="ProductResultMapper" type="product">
5     <!--column是数据库中的列名 property是模型类中的属性值-->
6     <id column="product_id" property="productId"
7       jdbcType="INTEGER"/>
8     <result column="product_name" property="productName">
```

```

9         jdbcType="VARCHAR" />
10        <result column="product_type" property="productType"
11            jdbcType="VARCHAR" />
12        <result column="can_sale" property="canSale"
13            jdbcType="VARCHAR" />
14        <result column="manufacturer" property="manufacturer"
15            jdbcType="VARCHAR"/>
16    </resultMap>
17    /*实现接口中的方法 增删改查*/
18    /*code*/
19 </mapper>

```

#### 4.1.4.在com.tkn.ssm.product.service 创建接口 - ProductService

#### 4.1.5.在com.tkn.ssm.product.service.impl 创建实现接口类 - ProductServiceImpl

##### 1.ProductServiceImpl实现接口ProductService

2.使用@Autowired注释可以对类成员变量、方法、构造函数进行标注，完成自动装配功能

```

1  @Autowired
2  private ProductMapper productMapper;

```

#### 4.1.6.在com.tkn.ssm.product.controller 创建控制器 - ProductController

## 5.创建jsp和static目录

```

1  在src/main/webapp目录下，创建static目录，在static下创建：
2  css （存放样式文件）
3  js （存放js文件）
4  img （存放图片资源文件）

```

```

1  在src/main/webapp目录下，创建目录jsp目录，在jsp下创建：
2  product （存放jsp页面）

```

### 5.1.jsp - product目录

#### 5.1.1.product\_list.jsp

产品列表					
注册					
序号	产品名称	产品类型	生产厂家	是否上架销售	操作
1	jvm虚拟机教程	图书	人民邮电大学	是	<a href="#">查看</a> <a href="#">删除</a>
2	java微服务项目	图书	杭州电子科技大学	是	<a href="#">查看</a> <a href="#">删除</a>
3	三只松鼠	食品	雀巢公司	是	<a href="#">查看</a> <a href="#">删除</a>

### 5.1.2.priduct\_add.jsp

产品上架

产品名称

产品类型

请选择产品类型

是否上架销售

☐

是

☐

否

生产厂家

注册

重置

取消

### 5.1.3.product\_edit.jsp

会员信息修改

产品名称

jvm虚拟机教程

产品类型

图书

是否上架

☒

是

☐

否

生产厂家

人民邮电大学

保存

重置

取消

## **5.2.static目录**

### **5.2.1product\_list.css**

1.对product\_list.jsp页面进行样式修改

### **5.2.2product\_add.css**

1.对product\_add.jsp页面进行样式修改

### **5.2.2product\_edit.css**

1.对product\_edit.jsp页面进行样式修改