**品优购电商系统开发**

**第2章**

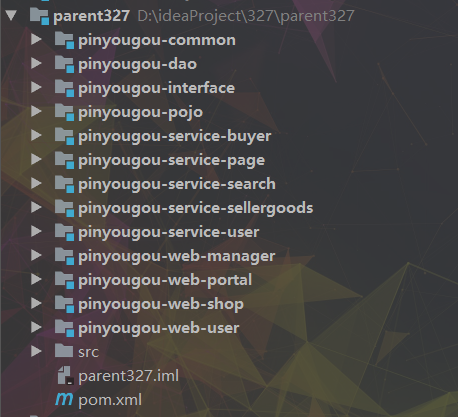
**四大框架整合SSMD**

传智播客.黑马程序员

# 1.品优购-框架搭建

## 1.1工程结构分析与设计

最终完整的工程结构如下：



工程说明：

parent 聚合工程

pinyougou-pojo 通用实体类层

pinyougou-dao 通用数据访问层

pinyougou-interface 服务层接口

pinyougou -service-xxxxx 某服务层实现

pinyougou -web-xxxxx 某web工程

## 1.2创建数据库表

执行资源文件夹中pinyougou-db.sql

## 1.3搭建框架

注意:所有项目配置文件在品优购项目\03配置文件\第一天搭建/目录中

### 1.3.1父工程

创建Maven工程parent （POM） ，groupId 为cn.itcast, artifactId 为parent , 在pom.xml中添加锁定版本信息dependencyManagement

|  |
| --- |
| <!-- 集中定义依赖版本号 -->  <properties>  <junit.version>4.12</junit.version>  <spring.version>4.2.4.RELEASE</spring.version>  <pagehelper.version>4.0.0</pagehelper.version>  <servlet-api.version>2.5</servlet-api.version>  <dubbo.version>2.8.4</dubbo.version>  <zookeeper.version>3.4.7</zookeeper.version>  <zkclient.version>0.1</zkclient.version>  <mybatis.version>3.2.8</mybatis.version>  <mybatis.spring.version>1.2.2</mybatis.spring.version>  <mybatis.paginator.version>1.2.15</mybatis.paginator.version>  <mysql.version>5.1.32</mysql.version>  <druid.version>1.0.9</druid.version>  <commons-fileupload.version>1.3.1</commons-fileupload.version>  <freemarker.version>2.3.23</freemarker.version>  <activemq.version>5.11.2</activemq.version>  <security.version>3.2.3.RELEASE</security.version>  <solrj.version>4.10.3</solrj.version>  <ik.version>2012\_u6</ik.version>  </properties>  <dependencyManagement>  <dependencies>  <!-- Spring -->  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-beans</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-webmvc</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jdbc</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-aspects</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jms</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context-support</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-test</artifactId>  <version>${spring.version}</version>  </dependency>  <!-- dubbo相关 -->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>dubbo</artifactId>  <version>${dubbo.version}</version>  </dependency>  <dependency>  <groupId>org.apache.zookeeper</groupId>  <artifactId>zookeeper</artifactId>  <version>${zookeeper.version}</version>  </dependency>  <dependency>  <groupId>com.github.sgroschupf</groupId>  <artifactId>zkclient</artifactId>  <version>${zkclient.version}</version>  </dependency>  <dependency>  <groupId>junit</groupId>  <artifactId>junit</artifactId>  <version>4.9</version>  </dependency>  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>fastjson</artifactId>  <version>1.2.28</version>  </dependency>  <dependency>  <groupId>javassist</groupId>  <artifactId>javassist</artifactId>  <version>3.11.0.GA</version>  </dependency>  <dependency>  <groupId>commons-codec</groupId>  <artifactId>commons-codec</artifactId>  <version>1.10</version>  </dependency>  <dependency>  <groupId>com.github.pagehelper</groupId>  <artifactId>pagehelper</artifactId>  <version>${pagehelper.version}</version>  </dependency>  <!-- Mybatis -->  <dependency>  <groupId>org.mybatis</groupId>  <artifactId>mybatis</artifactId>  <version>${mybatis.version}</version>  </dependency>  <dependency>  <groupId>org.mybatis</groupId>  <artifactId>mybatis-spring</artifactId>  <version>${mybatis.spring.version}</version>  </dependency>  <dependency>  <groupId>com.github.miemiedev</groupId>  <artifactId>mybatis-paginator</artifactId>  <version>${mybatis.paginator.version}</version>  </dependency>  <!-- MySql -->  <dependency>  <groupId>mysql</groupId>  <artifactId>mysql-connector-java</artifactId>  <version>${mysql.version}</version>  </dependency>  <!-- 连接池 -->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>druid</artifactId>  <version>${druid.version}</version>  </dependency>  <dependency>  <groupId>org.csource.fastdfs</groupId>  <artifactId>fastdfs</artifactId>  <version>1.2</version>  </dependency>  <!-- 文件上传组件 -->  <dependency>  <groupId>commons-fileupload</groupId>  <artifactId>commons-fileupload</artifactId>  <version>${commons-fileupload.version}</version>  </dependency>  <!-- 缓存 -->  <dependency>  <groupId>redis.clients</groupId>  <artifactId>jedis</artifactId>  <version>2.8.1</version>  </dependency>  <dependency>  <groupId>org.springframework.data</groupId>  <artifactId>spring-data-redis</artifactId>  <version>1.7.2.RELEASE</version>  </dependency>  <dependency>  <groupId>org.freemarker</groupId>  <artifactId>freemarker</artifactId>  <version>${freemarker.version}</version>  </dependency>  <dependency>  <groupId>org.apache.activemq</groupId>  <artifactId>activemq-all</artifactId>  <version>${activemq.version}</version>  </dependency>  <!-- 身份验证 -->  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-web</artifactId>  <version>4.1.0.RELEASE</version>  </dependency>  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-config</artifactId>  <version>4.1.0.RELEASE</version>  </dependency>  <dependency>  <groupId>com.github.penggle</groupId>  <artifactId>kaptcha</artifactId>  <version>2.3.2</version>  <exclusions>  <exclusion>  <groupId>javax.servlet</groupId>  <artifactId>javax.servlet-api</artifactId>  </exclusion>  </exclusions>  </dependency>  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-cas</artifactId>  <version>4.1.0.RELEASE</version>  </dependency>  <dependency>  <groupId>org.jasig.cas.client</groupId>  <artifactId>cas-client-core</artifactId>  <version>3.3.3</version>  <!-- 排除log4j包冲突 -->  <exclusions>  <exclusion>  <groupId>org.slf4j</groupId>  <artifactId>log4j-over-slf4j</artifactId>  </exclusion>  </exclusions>  </dependency>  <!-- solr客户端 -->  <dependency>  <groupId>org.apache.solr</groupId>  <artifactId>solr-solrj</artifactId>  <version>${solrj.version}</version>  </dependency>  <dependency>  <groupId>com.janeluo</groupId>  <artifactId>ikanalyzer</artifactId>  <version>${ik.version}</version>  </dependency>  <dependency>  <groupId>org.apache.httpcomponents</groupId>  <artifactId>httpcore</artifactId>  <version>4.4.4</version>  </dependency>  <dependency>  <groupId>org.apache.httpcomponents</groupId>  <artifactId>httpclient</artifactId>  <version>4.5.3</version>  </dependency>  <dependency>  <groupId>dom4j</groupId>  <artifactId>dom4j</artifactId>  <version>1.6.1</version>  </dependency>  <dependency>  <groupId>xml-apis</groupId>  <artifactId>xml-apis</artifactId>  <version>1.4.01</version>  </dependency>  <dependency>  <groupId>com.github.wxpay</groupId>  <artifactId>wxpay-sdk</artifactId>  <version>0.0.3</version>  </dependency>  </dependencies>  </dependencyManagement>  <dependencies>  <dependency>  <groupId>javax.servlet</groupId>  <artifactId>servlet-api</artifactId>  <version>2.5</version>  <scope>provided</scope>  </dependency>  </dependencies>  <build>  <plugins>  <!-- java编译插件 -->  <plugin>  <groupId>org.apache.maven.plugins</groupId>  <artifactId>maven-compiler-plugin</artifactId>  <version>3.2</version>  <configuration>  <source>1.7</source>  <target>1.7</target>  <encoding>UTF-8</encoding>  </configuration>  </plugin>  </plugins>  </build> |

以下模块均继承自此父工程

### 1.3.2通用工具类模块

创建jar模块项目common

|  |
| --- |
| <dependencies>  <dependency>  <groupId>com.github.pagehelper</groupId>  <artifactId>pagehelper</artifactId>  </dependency>  <!-- Mybatis -->  <dependency>  <groupId>org.mybatis</groupId>  <artifactId>mybatis</artifactId>  </dependency>  <dependency>  <groupId>org.mybatis</groupId>  <artifactId>mybatis-spring</artifactId>  </dependency>  <dependency>  <groupId>com.github.miemiedev</groupId>  <artifactId>mybatis-paginator</artifactId>  </dependency>  <!-- MySql -->  <dependency>  <groupId>mysql</groupId>  <artifactId>mysql-connector-java</artifactId>  </dependency>  <!-- 连接池 -->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>druid</artifactId>  </dependency>  <!-- 文件上传组件 -->  <dependency>  <groupId>org.csource.fastdfs</groupId>  <artifactId>fastdfs</artifactId>  </dependency>  <dependency>  <groupId>commons-fileupload</groupId>  <artifactId>commons-fileupload</artifactId>  </dependency>    <!-- 缓存 -->  <dependency>  <groupId>redis.clients</groupId>  <artifactId>jedis</artifactId>  </dependency>  <dependency>  <groupId>org.springframework.data</groupId>  <artifactId>spring-data-redis</artifactId>  </dependency>  <dependency>  <groupId>org.apache.httpcomponents</groupId>  <artifactId>httpclient</artifactId>  </dependency>    <!-- Spring -->  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-beans</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-webmvc</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jdbc</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-aspects</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jms</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context-support</artifactId>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-test</artifactId>  </dependency>  <!-- dubbo相关 -->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>dubbo</artifactId>  </dependency>  <dependency>  <groupId>org.apache.zookeeper</groupId>  <artifactId>zookeeper</artifactId>  </dependency>  <dependency>  <groupId>com.github.sgroschupf</groupId>  <artifactId>zkclient</artifactId>  </dependency>  <dependency>  <groupId>junit</groupId>  <artifactId>junit</artifactId>  </dependency>  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>fastjson</artifactId>  </dependency>  <dependency>  <groupId>javassist</groupId>  <artifactId>javassist</artifactId>  </dependency>  <dependency>  <groupId>commons-codec</groupId>  <artifactId>commons-codec</artifactId>  </dependency>  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-web</artifactId>  </dependency>    <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-config</artifactId>  </dependency>    <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-cas</artifactId>  </dependency>  <dependency>  <groupId>org.jasig.cas.client</groupId>  <artifactId>cas-client-core</artifactId>  <!-- <exclusions> -->  <!-- <exclusion> -->  <!-- <groupId>org.slf4j</groupId> -->  <!-- <artifactId>log4j-over-slf4j</artifactId> -->  <!-- </exclusion> -->  <!-- </exclusions> -->  </dependency>  <!-- activeMq -->  <dependency>  <groupId>org.apache.activemq</groupId>  <artifactId>activemq-all</artifactId>  </dependency>  <!-- freemarker -->  <dependency>  <groupId>org.freemarker</groupId>  <artifactId>freemarker</artifactId>  </dependency>  <!-- github -->  <dependency>  <groupId>com.github.wxpay</groupId>  <artifactId>wxpay-sdk</artifactId>  </dependency>  <!-- solr -->  <dependency>  <groupId>org.springframework.data</groupId>  <artifactId>spring-data-solr</artifactId>  <version>1.5.5.RELEASE</version>  </dependency>  </dependencies> |

### 1.3.3通用实体类模块

创建jar模块项目pojo .添加依赖于common项目

### 1.3.4通用数据访问模块

创建jar模块项目dao .添加依赖于pojo项目

### 1.3.5服务接口模块

创建jar模块项目interface , 添加依赖于pojo项目

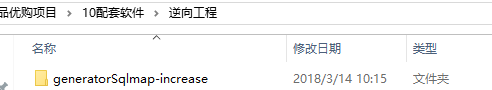
### 1.3.6 web.xml配置文件

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xmlns="http://java.sun.com/xml/ns/javaee"  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"  version="2.5">  </web-app> |

## 1.4实体类与数据访问层模块

### 1.4.1 生成代码

利用逆向工程实现实体类与数据访问层代码的自动生成



### 1.4.2 拷贝代码

将cn.itcast.core.pojo包拷贝到pojo工程

将cn.itcast.core.dao包和resources下

将cn.itcast.core.dao文件夹拷贝到dao工程

# 2.配置9个Tomcat

-Xms128m -Xmx128m -XX:PermSize=64m -XX:MaxPermSize=128m

# 3.品牌列表-后端代码

## 3.1需求分析

完成品牌管理的后端代码，在浏览器可查询品牌的数据（json格式）

## 3.2数据库表

**tb\_brand 品牌表**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 长度 | 含义 |
| Id | Bigint |  | 主键 |
| Name | Varchar | 255 | 品牌名称 |
| First\_char | Varchar | 1 | 品牌首字母 |

## 3.3后端代码

### 3.3.1 服务实现类

在service-product 工程创建BrandServiceImpl类

|  |
| --- |
| @Service  **public** **class** BrandServiceImpl **implements** BrandService {  @Autowired  **private** BrandDao brandDao;  @Override  **public** List<Brand> findAll() {  **return** brandDao.selectByExample(**null**);  } |

### 3.3.1 控制层代码

在manager-web工程创建cn.itcast.core.controller包，包下创建BrandController类

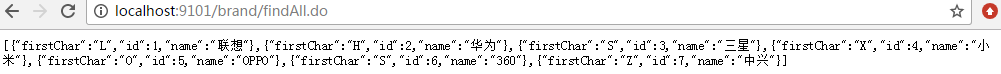
|  |
| --- |
| /\*\*  \* 品牌controller  \* **@author** Administrator  \*/  @RestController  @RequestMapping("/brand")  **public** **class** BrandController {      @Reference(url="dubbo://127.0.0.1:20880")  **private** BrandService brandService;  /\*\*  \* 返回全部列表  \* **@return**  \*/  @RequestMapping("/findAll")  **public** List<Brand> findAll(){  **return** brandService.findAll();  } |

## 3.4测试

启动product-service

启动manager-web

地址栏输入http://localhost:8080/brand/findAll.do



可以看到浏览器输出了json数据。