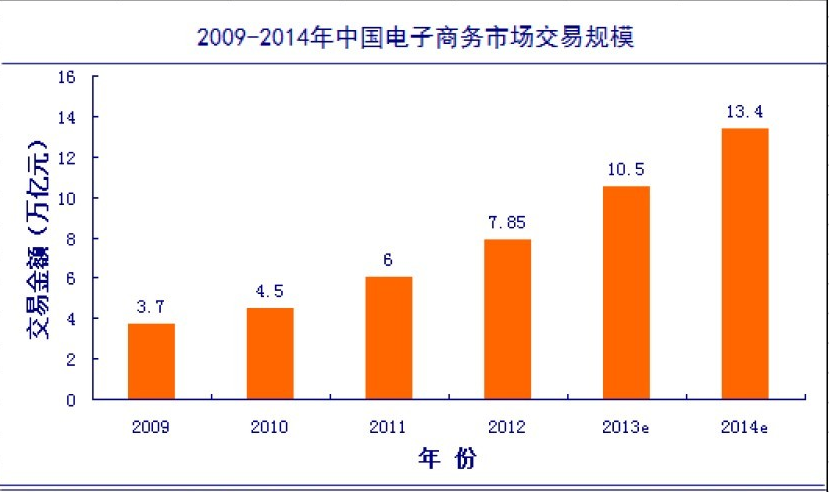
# 今日大纲

* 了解电商行业
* 了解企业开发流程
* 搭建商城后台系统
* 完成商品类目显示

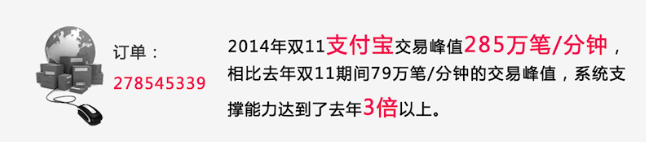
# 电商行业

## 电商行业发展



## 淘宝双十一





2015双11：



## 电商行业技术特点

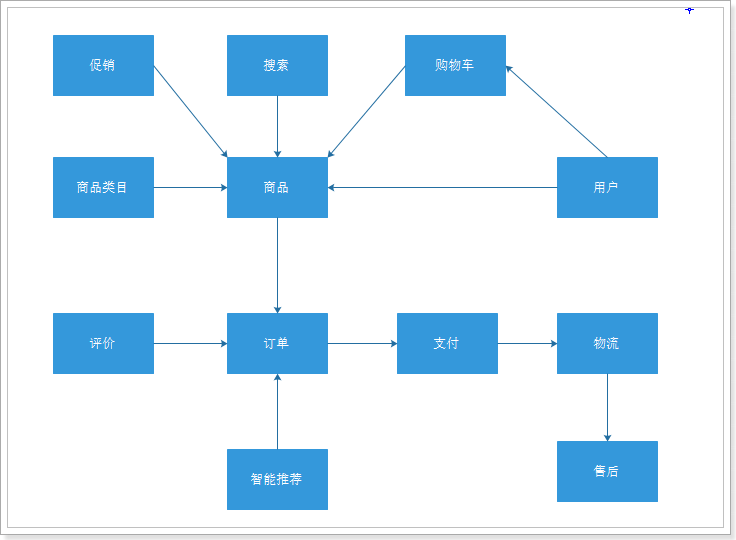
* 技术新
* 技术范围广
* 分布式
* 高并发/集群/负载均衡/高可用
* 海量数据
* 业务发展
* 系统安全

# 商城简介

## 电商行业的概念

* B2C：商家对个人，如：京东、亚马逊、当当等
* C2C：个人对个人，如：淘宝网、拍拍网
* B2B：商家对商家，如：阿里巴巴、八方资源网等
* O2O：线上和线下结合，如：饿了么、美团等
* P2P：在线金融，贷款，如：网贷之家、人人聚财等
* B2C平台：天猫、京东、一号店、当当等

## 功能模块

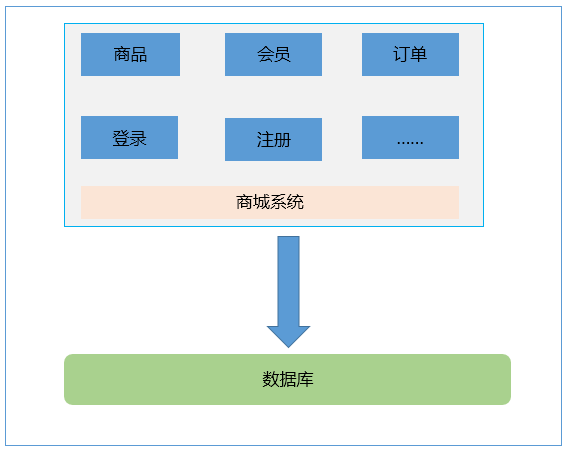


思维导图



# 系统架构

## 传统架构



存在的问题：

1. 开发、维护代码比较困难
2. 系统内部的功能模块之间的耦合度太高了
3. 无法针对每个模块的特性做优化
4. 无法做服务器的水平扩展
5. 无法存储海量数据

## 分布式系统架构



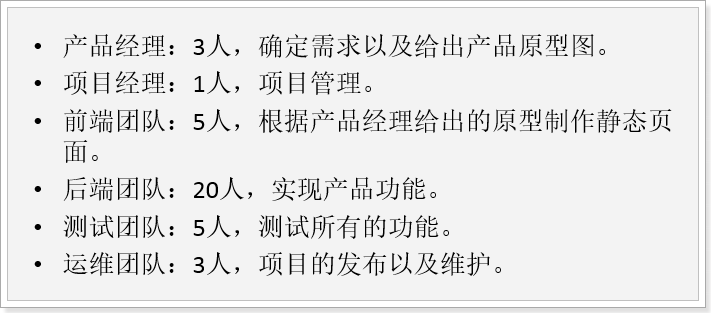
已经解决以上存在的问题，但是，带来了新问题：

1. 系统间的服务的调用变得复杂
2. 如何解决错综复杂的服务调用关系：服务注册中心。

## 系统说明

* 后台管理系统：商品管理、订单管理、商品分类、商品规格属性、用户管理、网站内容发布等。
* 前台系统（商城）：首页、用户注册、登录、浏览商品、下单、收藏商品。
* 会员系统：维护个人信息、收获地址管理、查询已下订单、查看收藏商品等。
* 订单系统：下单、查询订单、修改订单、处理订单等
* 单点登录系统：为多个系统之间提供统一的登录入口
* 搜索系统：提供商品的搜索功能

## 人员配置



## 开发流程



## 开发技术

* Spring、SpringMVC、MyBatis
* JSP、JSTL 、jQuery、KindEditor、CSS、DIV
* Redis（缓存服务器）
* ~~Lucene、Solr(搜索)~~
* HttpClient(调用系统服务)
* MYSQL
* Nginx/Tomcat(WEB服务器)
* Quartz(定时任务)
* RabbitMQ(消息队列)

# 系统开发

## 开发工具和环境

* Eclipse/Idea
* Maven
* Tomcat
* JDK1.7
* MYSQL5.6
* Nginx1.5.1
* Redis2.8.9
* WIN7
* SVN

## 创建itao-parent

### pom.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**modelVersion**>4.0.0</**modelVersion**>  <**groupId**>com.zhiyou100</**groupId**>  <**artifactId**>itao-parent</**artifactId**>  <**version**>1.0-SNAPSHOT</**version**>  <**packaging**>pom</**packaging**>   <**dependencies**>  *<!-- junit start-->* <**dependency**>  <**groupId**>junit</**groupId**>  <**artifactId**>junit</**artifactId**>  <**version**>4.12</**version**>  <**scope**>test</**scope**>  </**dependency**>  *<!-- junit end-->  <!--spring start-->* <**dependency**>  <**groupId**>org.springframework</**groupId**>  <**artifactId**>spring-context</**artifactId**>  <**version**>4.3.4.RELEASE</**version**>  </**dependency**>  <**dependency**>  <**groupId**>org.springframework</**groupId**>  <**artifactId**>spring-aspects</**artifactId**>  <**version**>4.3.4.RELEASE</**version**>  </**dependency**>  <**dependency**>  <**groupId**>org.springframework</**groupId**>  <**artifactId**>spring-jdbc</**artifactId**>  <**version**>4.3.4.RELEASE</**version**>  </**dependency**>  <**dependency**>  <**groupId**>org.springframework</**groupId**>  <**artifactId**>spring-test</**artifactId**>  <**version**>4.3.4.RELEASE</**version**>  <**scope**>test</**scope**>  </**dependency**>   <**dependency**>  <**groupId**>org.springframework</**groupId**>  <**artifactId**>spring-web</**artifactId**>  <**version**>4.3.4.RELEASE</**version**>  </**dependency**>   <**dependency**>  <**groupId**>org.springframework</**groupId**>  <**artifactId**>spring-webmvc</**artifactId**>  <**version**>4.3.4.RELEASE</**version**>  </**dependency**>   *<!--spring-json依赖-->* <**dependency**>  <**groupId**>com.fasterxml.jackson.core</**groupId**>  <**artifactId**>jackson-core</**artifactId**>  <**version**>2.8.1</**version**>  </**dependency**>   <**dependency**>  <**groupId**>com.fasterxml.jackson.core</**groupId**>  <**artifactId**>jackson-databind</**artifactId**>  <**version**>2.8.1</**version**>  </**dependency**>   <**dependency**>  <**groupId**>com.fasterxml.jackson.core</**groupId**>  <**artifactId**>jackson-annotations</**artifactId**>  <**version**>2.8.1</**version**>  </**dependency**>    *<!-- Servlet -->* <**dependency**>  <**groupId**>javax.servlet</**groupId**>  <**artifactId**>javax.servlet-api</**artifactId**>  <**version**>3.0.1</**version**>  <**scope**>provided</**scope**>  </**dependency**>  *<!-- JSP -->* <**dependency**>  <**groupId**>javax.servlet.jsp</**groupId**>  <**artifactId**>jsp-api</**artifactId**>  <**version**>2.2</**version**>  <**scope**>provided</**scope**>  </**dependency**>  *<!-- JSTL -->* <**dependency**>  <**groupId**>javax.servlet</**groupId**>  <**artifactId**>jstl</**artifactId**>  <**version**>1.2</**version**>  <**scope**>runtime</**scope**>  </**dependency**>   <**dependency**>  <**groupId**>joda-time</**groupId**>  <**artifactId**>joda-time</**artifactId**>  <**version**>2.9.4</**version**>  </**dependency**>   *<!--spring end-->  <!-- database begin -->* <**dependency**>  <**groupId**>mysql</**groupId**>  <**artifactId**>mysql-connector-java</**artifactId**>  <**version**>5.1.40</**version**>  <**scope**>runtime</**scope**>  </**dependency**>  <**dependency**>  <**groupId**>com.alibaba</**groupId**>  <**artifactId**>druid</**artifactId**>  <**version**>1.0.26</**version**>  </**dependency**>  <**dependency**>  <**groupId**>org.mybatis</**groupId**>  <**artifactId**>mybatis</**artifactId**>  <**version**>3.3.0</**version**>  </**dependency**>  <**dependency**>  <**groupId**>org.mybatis</**groupId**>  <**artifactId**>mybatis-spring</**artifactId**>  <**version**>1.2.3</**version**>  </**dependency**>  *<!-- database end -->  <!-- loging begin-->* <**dependency**>  <**groupId**>org.slf4j</**groupId**>  <**artifactId**>slf4j-log4j12</**artifactId**>  <**version**>1.7.7</**version**>  </**dependency**>  *<!-- loging end -->  <!-- Mybatis Generator -->* <**dependency**>  <**groupId**>org.mybatis.generator</**groupId**>  <**artifactId**>mybatis-generator-core</**artifactId**>  <**version**>1.3.5</**version**>  <**scope**>compile</**scope**>  <**optional**>true</**optional**>  </**dependency**>   <**dependency**>  <**groupId**>tk.mybatis</**groupId**>  <**artifactId**>mapper</**artifactId**>  <**version**>3.3.9</**version**>  </**dependency**>  *<!-- 分页插件 -->* <**dependency**>  <**groupId**>com.github.pagehelper</**groupId**>  <**artifactId**>pagehelper</**artifactId**>  <**version**>4.1.6</**version**>  </**dependency**>  </**dependencies**>   <**build**>  <**resources**>  <**resource**>  <**directory**>${basedir}/src/main/java</**directory**>  <**includes**>  <**include**>\*\*/\*.xml</**include**>  </**includes**>  </**resource**>  <**resource**>  <**directory**>${basedir}/src/main/resources</**directory**>  </**resource**>  </**resources**>  <**plugins**>  <**plugin**>  <**artifactId**>maven-compiler-plugin</**artifactId**>  <**configuration**>  <**source**>1.7</**source**>  <**target**>1.7</**target**>  </**configuration**>  </**plugin**>  <**plugin**>  <**groupId**>org.mybatis.generator</**groupId**>  <**artifactId**>mybatis-generator-maven-plugin</**artifactId**>  <**version**>1.3.2</**version**>  <**configuration**>  <**configurationFile**>${basedir}/src/main/resources/generator/generatorConfig.xml</**configurationFile**>  <**overwrite**>true</**overwrite**>  <**verbose**>true</**verbose**>  </**configuration**>  <**dependencies**>  <**dependency**>  <**groupId**>mysql</**groupId**>  <**artifactId**>mysql-connector-java</**artifactId**>  <**version**>5.1.40</**version**>  </**dependency**>  <**dependency**>  <**groupId**>tk.mybatis</**groupId**>  <**artifactId**>mapper</**artifactId**>  <**version**>3.3.9</**version**>  </**dependency**>  </**dependencies**>  </**plugin**>  </**plugins**>  </**build**> </**project**> |

## 创建itao-common

### pom.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**parent**>  <**artifactId**>itao-parent</**artifactId**>  <**groupId**>com.zhiyou100</**groupId**>  <**version**>1.0-SNAPSHOT</**version**>  </**parent**>  <**modelVersion**>4.0.0</**modelVersion**>  <**artifactId**>itao-common</**artifactId**> </**project**> |

## 创建itao-manage

### 分析

com.zhiyou100.manage.model

com.zhiyou100.manage.mapper

com.zhiyou100.manage.service

com.zhiyou100.manage.controller

### itao-manage

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**parent**>  <**artifactId**>itao-parent</**artifactId**>  <**groupId**>com.zhiyou100</**groupId**>  <**version**>1.0-SNAPSHOT</**version**>  </**parent**>  <**modelVersion**>4.0.0</**modelVersion**>   <**artifactId**>itao-manage</**artifactId**>  <**packaging**>pom</**packaging**>  <**modules**>  <**module**>itao-manage-model</**module**>  <**module**>itao-manage-mapper</**module**>  <**module**>itao-manage-service</**module**>  <**module**>itao-manage-web</**module**>  </**modules**>   <**dependencies**>  <**dependency**>  <**groupId**>com.zhiyou100</**groupId**>  <**artifactId**>itao-common</**artifactId**>  <**version**>1.0-SNAPSHOT</**version**>  </**dependency**>  </**dependencies**> </**project**> |

### itao-manage-model

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**parent**>  <**artifactId**>itao-manage</**artifactId**>  <**groupId**>com.zhiyou100</**groupId**>  <**version**>1.0-SNAPSHOT</**version**>  </**parent**>  <**modelVersion**>4.0.0</**modelVersion**>   <**artifactId**>itao-manage-model</**artifactId**>  </**project**> |

### itao-manage-mapper

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**parent**>  <**artifactId**>itao-manage</**artifactId**>  <**groupId**>com.zhiyou100</**groupId**>  <**version**>1.0-SNAPSHOT</**version**>  </**parent**>  <**modelVersion**>4.0.0</**modelVersion**>   <**artifactId**>itao-manage-mapper</**artifactId**>  <**dependencies**>  <**dependency**>  <**groupId**>com.zhiyou100</**groupId**>  <**artifactId**>itao-manage-model</**artifactId**>  <**version**>1.0-SNAPSHOT</**version**>  </**dependency**>  </**dependencies**> </**project**> |

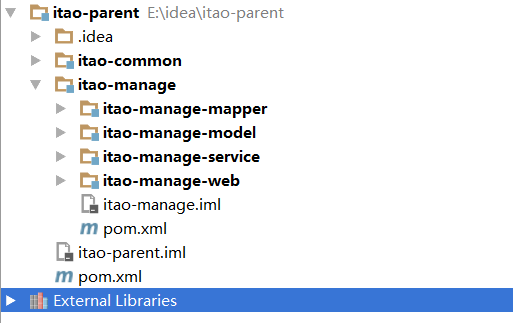
### itao-manage-service

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**parent**>  <**artifactId**>itao-manage</**artifactId**>  <**groupId**>com.zhiyou100</**groupId**>  <**version**>1.0-SNAPSHOT</**version**>  </**parent**>  <**modelVersion**>4.0.0</**modelVersion**>   <**artifactId**>itao-manage-service</**artifactId**>  <**dependencies**>  <**dependency**>  <**groupId**>com.zhiyou100</**groupId**>  <**artifactId**>itao-manage-mapper</**artifactId**>  <**version**>1.0-SNAPSHOT</**version**>  </**dependency**>  </**dependencies**> </**project**> |

### itao-manage-web

|  |
| --- |
| <**project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4\_0\_0.xsd"**>  <**parent**>  <**artifactId**>itao-manage</**artifactId**>  <**groupId**>com.zhiyou100</**groupId**>  <**version**>1.0-SNAPSHOT</**version**>  </**parent**>  <**modelVersion**>4.0.0</**modelVersion**>  <**artifactId**>itao-manage-web</**artifactId**>  <**packaging**>war</**packaging**>   <**dependencies**>  <**dependency**>  <**groupId**>com.zhiyou100</**groupId**>  <**artifactId**>itao-manage-service</**artifactId**>  <**version**>1.0-SNAPSHOT</**version**>  </**dependency**>  </**dependencies**>   <**build**>  <**finalName**>itao-manage-web</**finalName**>  </**build**> </**project**> |

### 创建完成



## 子工程之间的依赖关系

itao-manage-web 🡺 itao-manage-service 🡺 itao-manage-mapper 🡺itao-manage-model

## 导入依赖

导入依赖的原则：

1、在使用依赖的最底层导入。

2、运行时所需要的依赖在web工程中加入。

3、所有的工程都需要的依赖应该在聚合工程（itao-manage）中导入。

导入SSM依赖：

Spring和SpringMVC的依赖在itao-manage-service中导入；

## jdbc.properties

|  |
| --- |
| *#数据库连接信息* **jdbc.driverClassName**=**com.mysql.jdbc.Driver jdbc.url**=**jdbc:mysql://localhost:3306/itao jdbc.username**=**root jdbc.password**=**123456 mapper.plugin**=**tk.mybatis.mapper.generator.MapperPlugin mapper.Mapper**=**tk.mybatis.mapper.common.Mapper** |

## log4j.properties

|  |
| --- |
| **log4j.rootLogger**=**DEBUG, Console** *#Console* **log4j.appender.Console**=**org.apache.log4j.ConsoleAppender log4j.appender.Console.layout**=**org.apache.log4j.PatternLayout log4j.appender.Console.layout.ConversionPattern**=**%-5p - %m%n** *#log4j.logger.org.springframework=ERROR* **log4j.logger.org.mybatis.spring**=**ERROR log4j.logger.org.apache.ibatis.logging.LogFactory**=**ERROR log4j.logger.com.alibaba.druid.pool.DruidDataSource**=**ERROR** *#log4j.logger.com.github.pagehelper=DEBUG #log4j.logger.com.github.pagehelper.mapper = TRACE* |

## mybatis-config.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"** *?>* **<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"  "http://mybatis.org/dtd/mybatis-3-config.dtd"*>*** <**configuration**>  *<!-- mybatis默认是没有开启延迟加载的 需要手动开启 -->* <**settings**>  *<!-- 延迟加载 默认false -->* <**setting name="lazyLoadingEnabled" value="true"** />  *<!-- 积极加载 默认true -->* <**setting name="aggressiveLazyLoading" value="false"** />  *<!--开启缓存-->* <**setting name="cacheEnabled" value="true"**/>  </**settings**>  <**plugins**>  *<!-- com.github.pagehelper为PageHelper类所在包名 -->* <**plugin interceptor="com.github.pagehelper.PageHelper"**>  *<!-- 4.0.0以后版本可以不设置该参数 -->* <**property name="dialect" value="mysql"** />  *<!-- 该参数默认为false -->  <!-- 设置为true时，会将RowBounds第一个参数offset当成pageNum页码使用 -->  <!-- 和startPage中的pageNum效果一样 -->* <**property name="offsetAsPageNum" value="true"** />  *<!-- 该参数默认为false -->  <!-- 设置为true时，使用RowBounds分页会进行count查询 -->* <**property name="rowBoundsWithCount" value="false"** />  *<!-- 设置为true时，如果pageSize=0或者RowBounds.limit = 0就会查询出全部的结果 -->  <!-- （相当于没有执行分页查询，但是返回结果仍然是Page类型） -->* <**property name="pageSizeZero" value="true"** />  *<!-- 3.3.0版本可用 - 分页参数合理化，默认false禁用 -->  <!-- 启用合理化时，如果pageNum<1会查询第一页，如果pageNum>pages会查询最后一页 -->  <!-- 禁用合理化时，如果pageNum<1或pageNum>pages会返回空数据 -->* <**property name="reasonable" value="false"** />  *<!-- 3.5.0版本可用 - 为了支持startPage(Object params)方法 -->  <!-- 增加了一个`params`参数来配置参数映射，用于从Map或ServletRequest中取值 -->  <!-- 可以配置pageNum,pageSize,count,pageSizeZero,reasonable,orderBy,不配置映射的用默认值 -->  <!-- 不理解该含义的前提下，不要随便复制该配置 -->* <**property name="params"  value="pageNum=pageHelperStart;pageSize=pageHelperRows;"** />  *<!-- 支持通过Mapper接口参数来传递分页参数 -->* <**property name="supportMethodsArguments" value="false"** />  *<!-- always总是返回PageInfo类型,check检查返回类型是否为PageInfo,none返回Page -->* <**property name="returnPageInfo" value="none"** />  </**plugin**>  </**plugins**> </**configuration**> |

## applicationContext.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**beans xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xmlns:tx="http://www.springframework.org/schema/tx"  xmlns:aop="http://www.springframework.org/schema/aop"  xmlns:context="http://www.springframework.org/schema/context"  xmlns="http://www.springframework.org/schema/beans"  xsi:schemaLocation="http://www.springframework.org/schema/beans  http://www.springframework.org/schema/beans/spring-beans.xsd  http://www.springframework.org/schema/tx  http://www.springframework.org/schema/tx/spring-tx.xsd  http://www.springframework.org/schema/aop  http://www.springframework.org/schema/aop/spring-aop.xsd  http://www.springframework.org/schema/context  http://www.springframework.org/schema/context/spring-context.xsd"**>  *<!--扫描包路径  <context:component-scan base-package="com.zhiyou100.service"/>  -->  <!-- 扫描的包路径 去掉mvc的注解 -->* <**context:component-scan base-package="com.zhiyou100"**>  <**context:exclude-filter type="annotation" expression="org.springframework.stereotype.Controller"** />  <**context:exclude-filter type="annotation" expression="org.springframework.web.bind.annotation.ControllerAdvice"** />  </**context:component-scan**>  *<!--加载配置文件-->* <**context:property-placeholder location="classpath:jdbc.properties"**/>  *<!--数据源配置-->* <**bean id="dataSource" class="com.alibaba.druid.pool.DruidDataSource" init-method="init" destroy-method="close"**>  <**property name="driverClassName" value="${jdbc.driverClassName}"**/>  <**property name="url" value="${jdbc.url}"**/>  <**property name="username" value="${jdbc.username}"**/>  <**property name="password" value="${jdbc.password}"**/>  <**property name="filters" value="stat"**/>  <**property name="maxActive" value="20"**/>  <**property name="initialSize" value="1"**/>  <**property name="maxWait" value="60000"**/>  <**property name="minIdle" value="1"**/>  <**property name="timeBetweenEvictionRunsMillis" value="60000"**/>  <**property name="minEvictableIdleTimeMillis" value="300000"**/>  <**property name="validationQuery" value="SELECT 'x'"**/>  <**property name="testWhileIdle" value="true"**/>  <**property name="testOnBorrow" value="false"**/>  <**property name="testOnReturn" value="false"**/>  </**bean**>  *<!-- 注入sqlSessionFactory -->* <**bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean"**>  *<!-- 注入数据源-->* <**property name="dataSource" ref="dataSource"**/>  *<!-- 自动扫描mapping.xml文件，\*\*表示迭代查找* <**property name="mapperLocations" value="classpath:mapper/\*"**/>  *-->*  *<!--mybatis配置文件位置-->* <**property name="configLocation" value="classpath:mybatis-config.xml"**/>  </**bean**>   *<!--扫描com.zhiyou100下的mapper接口  <bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">  <property name="basePackage" value="com.zhiyou100"/>  </bean>  -->  <!--扫描com.zhiyou100下的mapper接口-->* <**bean class="tk.mybatis.spring.mapper.MapperScannerConfigurer"**>  <**property name="basePackage" value="com.zhiyou100"**/>  *<!-- 通用Mapper通过属性注入进行配置，默认不配置时会注册Mapper<T>接口-->* <**property name="properties"**>  <**value**>  mappers=tk.mybatis.mapper.common.Mapper  </**value**>  </**property**>  </**bean**>   *<!-- 事务管理器 -->* <**bean id="transactionManager" class="org.springframework.jdbc.datasource.DataSourceTransactionManager"**>  <**property name="dataSource" ref="dataSource"**/>  </**bean**>  *<!--事务详情(事务通知/增强)， 在aop筛选基础上，确定使用什么样的事务。例如：读写、只读等  <tx:attributes> 用于配置事务详情（属性属性）  <tx:method name=""/> 详情具体配置  propagation 传播行为 ， REQUIRED：必须；REQUIRES\_NEW:必须是新的 read-only 是否是只读事物  isolation 隔离级别 read-only 只读事物  -->* <**tx:advice id="txAdvice" transaction-manager="transactionManager"**>  <**tx:attributes**>  <**tx:method name="select\*" read-only="true"**/>  <**tx:method name="find\*" read-only="true"**/>  <**tx:method name="get\*" read-only="true"**/>  <**tx:method name="\*"**/>  </**tx:attributes**>  </**tx:advice**>  *<!--支持基于注解的aspectj-->* <**aop:aspectj-autoproxy**/>  *<!--AOP编程，切入点表达式 确定增强的连接器，从而获得切入点-->* <**aop:config**>  <**aop:advisor advice-ref="txAdvice" pointcut="execution(\* com.zhiyou100..service..\*.\*(..)))"**/>  </**aop:config**> </**beans**> |

## applicationContext-servlet.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**beans xmlns="http://www.springframework.org/schema/beans"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xmlns:context="http://www.springframework.org/schema/context"  xmlns:mvc="http://www.springframework.org/schema/mvc"   xsi:schemaLocation="http://www.springframework.org/schema/beans  http://www.springframework.org/schema/beans/spring-beans.xsd  http://www.springframework.org/schema/mvc  http://www.springframework.org/schema/mvc/spring-mvc.xsd  http://www.springframework.org/schema/context  http://www.springframework.org/schema/context/spring-context.xsd"**>   *<!-- 注解扫描!!!  <context:component-scan base-package="com.zhiyou100.controller" />  -->  <!-- 自动扫描且只扫描@Controller -->* <**context:component-scan base-package="com.zhiyou100" use-default-filters="false"**>  <**context:include-filter type="annotation" expression="org.springframework.stereotype.Controller"**/>  <**context:include-filter type="annotation"  expression="org.springframework.web.bind.annotation.ControllerAdvice"**/>  </**context:component-scan**>   *<!-- 处理静态资源 让静态资源可以直接访问 <mvc:resources mapping="/static/\*\*" location="/static/"/> -->  <!-- 当在web.xml 中 DispatcherServlet使用 <url-pattern>/</url-pattern> 映射时，能映射静态资源 -->* <**mvc:default-servlet-handler**/>   *<!-- 自定义日期转换器 -->* <**bean id="conversionService"  class="org.springframework.format.support.FormattingConversionServiceFactoryBean"**>  <**property name="converters"**>  <**set**>  *<!--日期转换器 -->* <**bean class="com.zhiyou100.common.util.DateConverter"**/>  </**set**>  </**property**>  </**bean**>  *<!-- 可用在springmvc.xml配置文件中使用<mvc:annotation-driven>替代注解处理器和适配器的配置。 -->* <**mvc:annotation-driven conversion-service="conversionService"**>  <**mvc:message-converters**>  <**bean class="org.springframework.http.converter.json.MappingJackson2HttpMessageConverter"**>  *<!-- 将Jackson2HttpMessageConverter的默认格式化输出设为true -->* <**property name="prettyPrint" value="true"**/>  </**bean**>  </**mvc:message-converters**>  </**mvc:annotation-driven**>   *<!-- 配置视图解析器 -->* <**bean class="org.springframework.web.servlet.view.InternalResourceViewResolver"**>  <**property name="prefix" value="/WEB-INF/view/"**/>  <**property name="suffix" value=".jsp"**/>  </**bean**> </**beans**> |

## web.xml

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**web-app xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee  http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"  version="3.0"**>   *<!--Spring入口文件的配置-->  <!-- 确定配置文件位置 -->* <**context-param**>  <**param-name**>contextConfigLocation</**param-name**>  <**param-value**>classpath:applicationContext.xml</**param-value**>  </**context-param**>  *<!-- 配置spring 监听器，加载xml配置文件 -->* <**listener**>  <**listener-class**>org.springframework.web.context.ContextLoaderListener</**listener-class**>  </**listener**>   *<!-- 解决PUT请求无法提交表单数据的问题 -->* <**filter**>  <**filter-name**>HttpMethodFilter</**filter-name**>  <**filter-class**>org.springframework.web.filter.HttpPutFormContentFilter</**filter-class**>  </**filter**>  <**filter-mapping**>  <**filter-name**>HttpMethodFilter</**filter-name**>  <**url-pattern**>/\*</**url-pattern**>  </**filter-mapping**>   *<!-- 将POST请求转化为DELETE或者是PUT 要用\_method指定真正的请求参数 -->* <**filter**>  <**filter-name**>HiddenHttpMethodFilter</**filter-name**>  <**filter-class**>org.springframework.web.filter.HiddenHttpMethodFilter</**filter-class**>  </**filter**>  <**filter-mapping**>  <**filter-name**>HiddenHttpMethodFilter</**filter-name**>  <**url-pattern**>/\*</**url-pattern**>  </**filter-mapping**>  *<!-- 解决POST请求的中文乱码 -->* <**filter**>  <**filter-name**>CharacterEncodingFilter</**filter-name**>  <**filter-class**>org.springframework.web.filter.CharacterEncodingFilter</**filter-class**>  <**init-param**>  <**param-name**>encoding</**param-name**>  <**param-value**>utf-8</**param-value**>  </**init-param**>  </**filter**>  <**filter-mapping**>  <**filter-name**>CharacterEncodingFilter</**filter-name**>  <**url-pattern**>/\*</**url-pattern**>  </**filter-mapping**>   *<!--SpringMVC入口文件配置 -->  <!-- DispatcherServlet：前端控制器 配置前端控制器servlet -->* <**servlet**>  <**servlet-name**>springmvc</**servlet-name**>  <**servlet-class**>org.springframework.web.servlet.DispatcherServlet</**servlet-class**>  *<!--加载前端控制器配置文件 上下文配置位置 -->* <**init-param**>  <**param-name**>contextConfigLocation</**param-name**>  <**param-value**>classpath:applicationContext-servlet.xml</**param-value**>  </**init-param**>  *<!-- 表示随WEB服务器启动 -->* <**load-on-startup**>1</**load-on-startup**>  </**servlet**>  <**servlet-mapping**>  <**servlet-name**>springmvc</**servlet-name**>  <**url-pattern**>/rest/\*</**url-pattern**>  </**servlet-mapping**> </**web-app**> |

## 导入静态页面

## 通用页面跳转

|  |
| --- |
| @RequestMapping(**"/page"**) @Controller **public class** PageController {  @RequestMapping(value = **"/{pageName}"**, method = RequestMethod.***GET***)  **public** String toPage(@PathVariable(**"pageName"**) String pageName) {  **return** pageName;  } } |

# 后台管理系统

## 登录页面

登录逻辑：

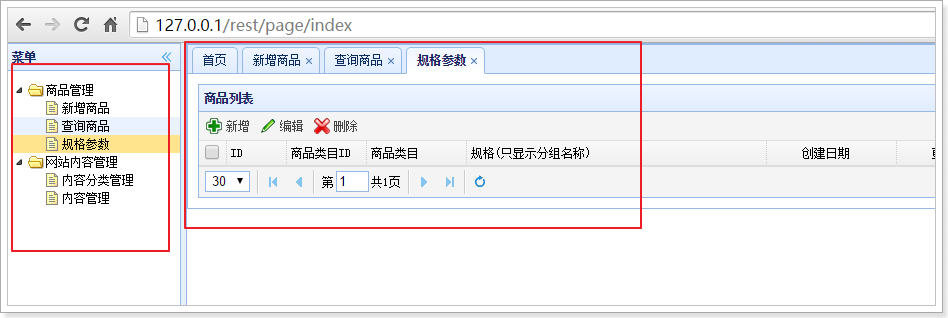


## 首页的js布局实现

### 布局

左右布局：

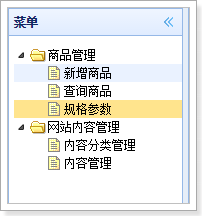




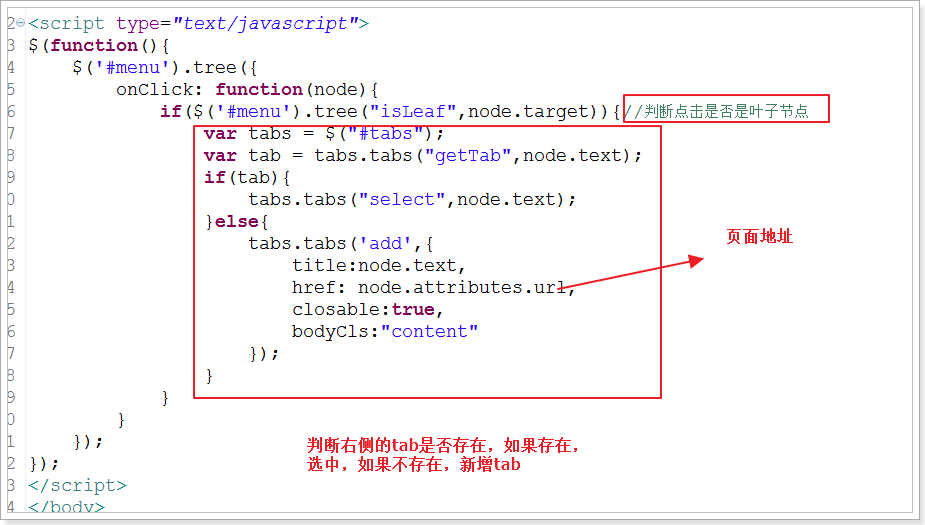
### 左侧菜单树



效果：



### 菜单的点击事件

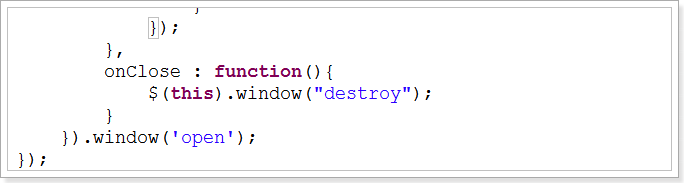


## 选择商品类目

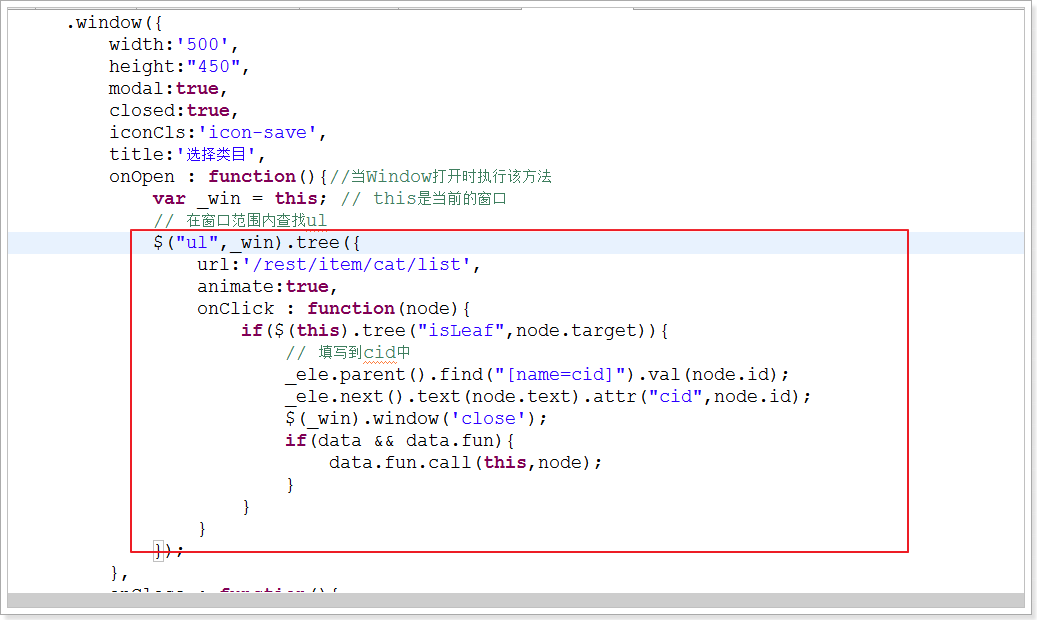
功能描述：在新增商品时点击选择类目，弹出窗口，在窗口中显示商品类目数据。

### 点击弹出框

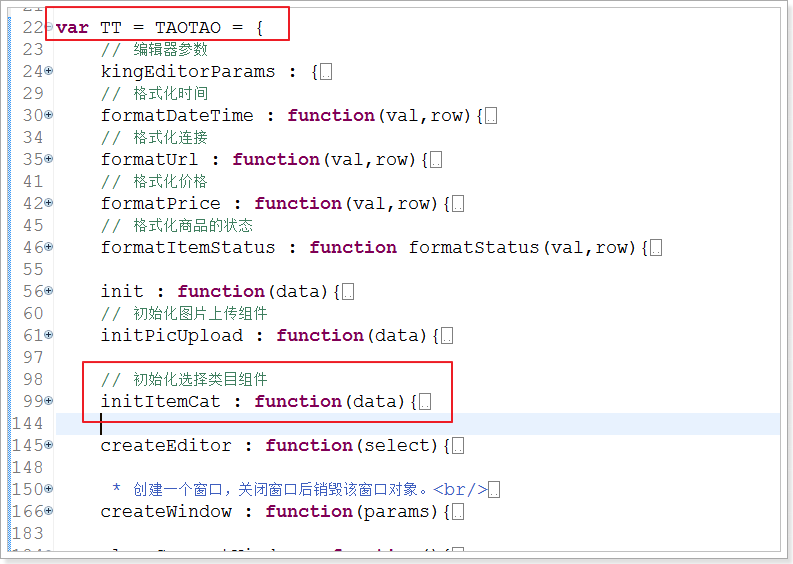




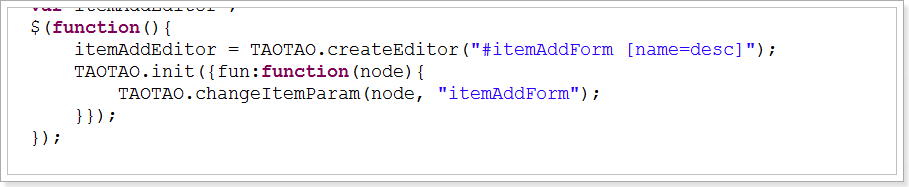
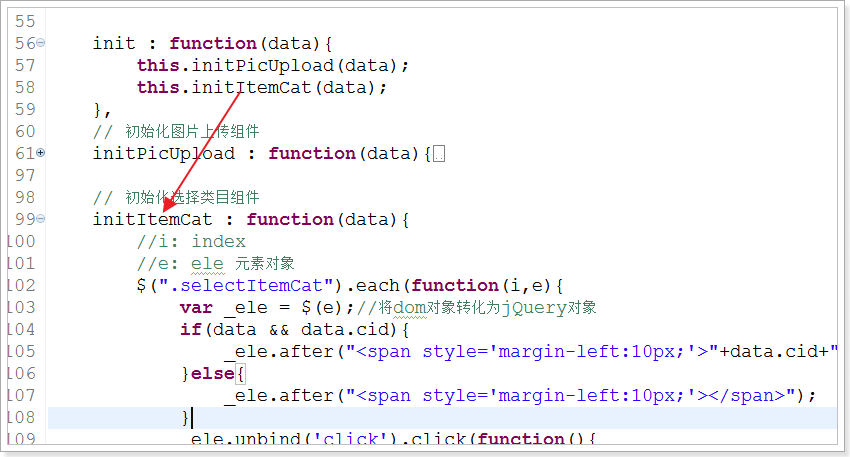
### 加载tree



### 定义TAOTAO对象

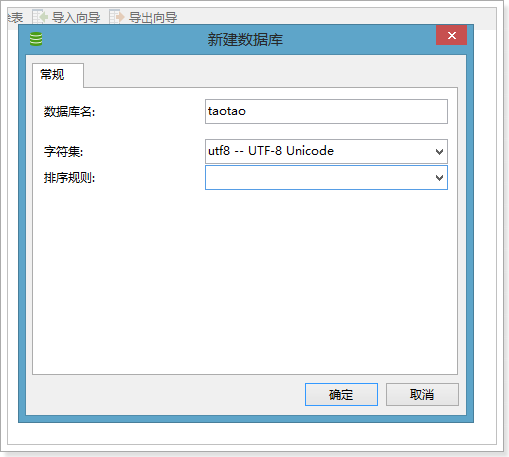


### 初始化过程

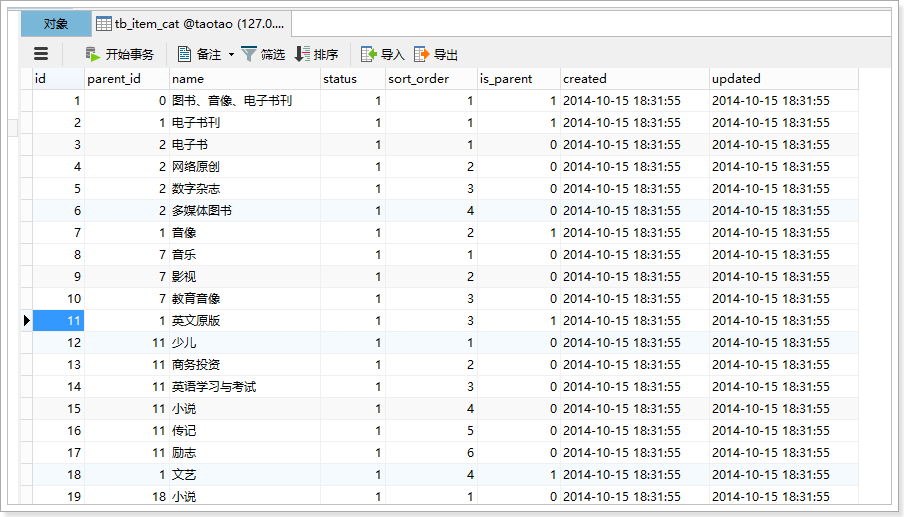
1. 通过TAOTAO.init方法完成初始化  
   
2. TT.init中调用initItemCat方法  
   
3. 给class=” selectItemCat”绑定click事件，即可实现效果

## 商品类目的数据结构

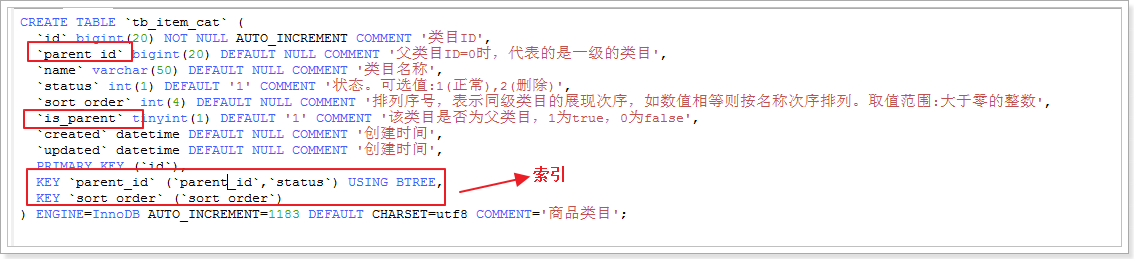
创建数据库：



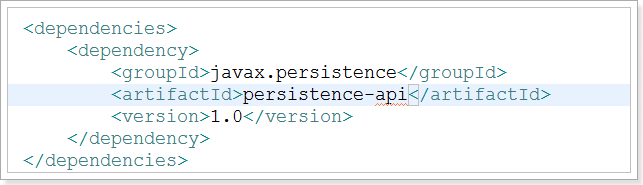
### 数据



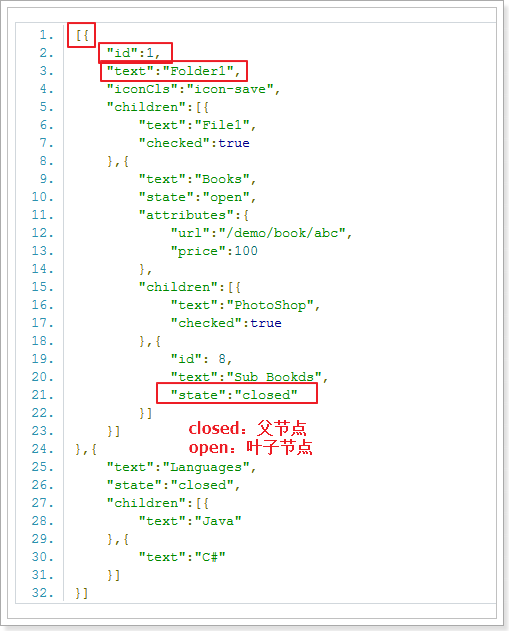
### 表结构



### 在taotao-manage-pojo中添加JPA注解依赖



### EasyUI的tree数据结构

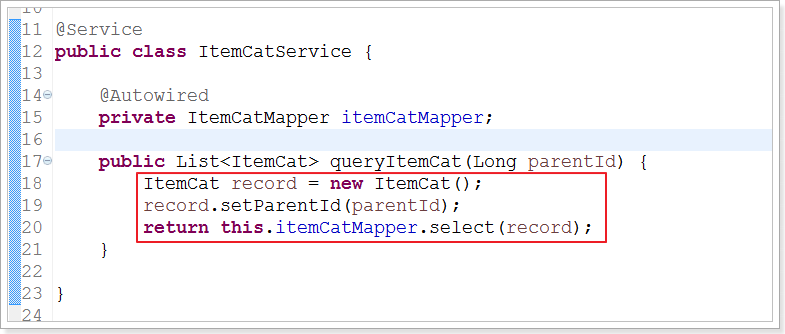


## 实现

### Controller



### Service



### ItemCat

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** javax.persistence.Table;

@Table(name = "tb\_item\_cat")

**publicclass** ItemCat **extends** BasePojo {

@Id

@GeneratedValue(strategy = GenerationType.***IDENTITY***)

**private** Long id;

**private** Long parentId;

**private** String name;

**private** Integer status;

**private** Integer sortOrder;

**private** Boolean isParent;

**public** Long getId() {

**return**id;

}

**publicvoid** setId(Long id) {

**this**.id = id;

}

**public** Long getParentId() {

**return**parentId;

}

**publicvoid** setParentId(Long parentId) {

**this**.parentId = parentId;

}

**public** String getName() {

**return**name;

}

**publicvoid** setName(String name) {

**this**.name = name;

}

**public** Integer getStatus() {

**return**status;

}

**publicvoid** setStatus(Integer status) {

**this**.status = status;

}

**public** Integer getSortOrder() {

**return**sortOrder;

}

**publicvoid** setSortOrder(Integer sortOrder) {

**this**.sortOrder = sortOrder;

}

**public** Boolean getIsParent() {

**return**isParent;

}

**publicvoid** setIsParent(Boolean isParent) {

**this**.isParent = isParent;

}

// 扩展字段，支持EasyUItree的显示

**public** String getText() {

**returnthis**.getName();

}

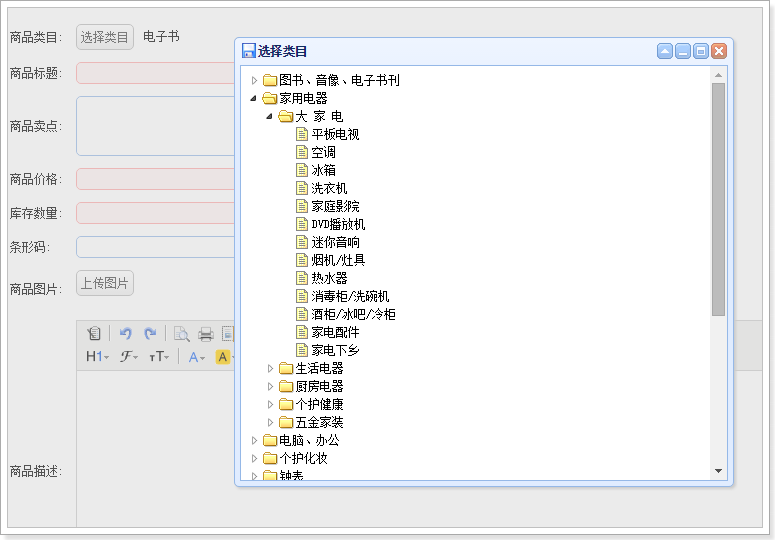
**public** String getState() {

**returnthis**.getIsParent() ? "closed" : "open";

}

}

### 效果



# 提交代码到SVN

## 账号

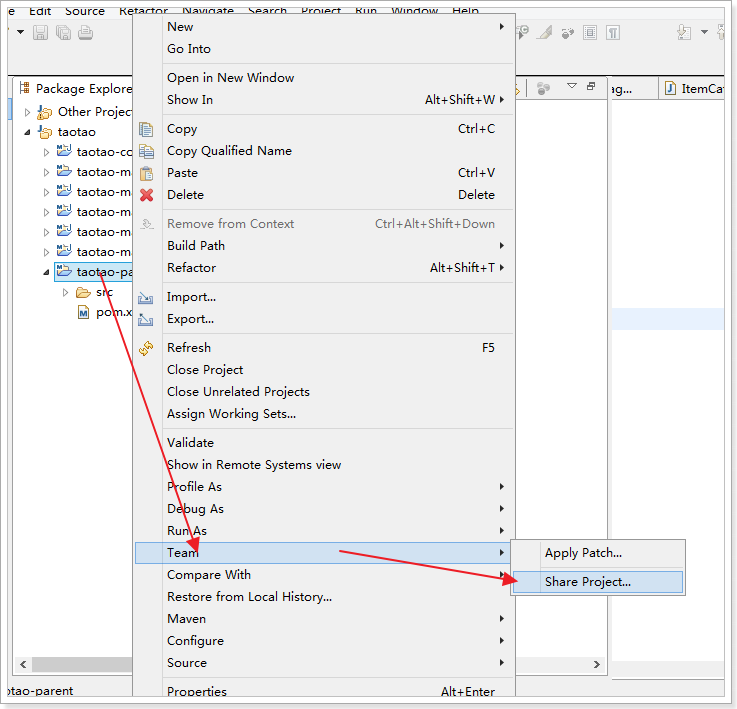
如：张三，zhangsan密码：123456

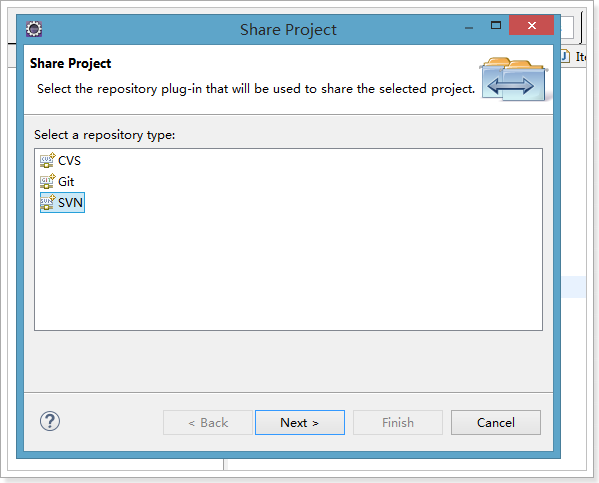
地址：svn://192.168.0.47/itcast/0114

## Maven项目提交到SVN

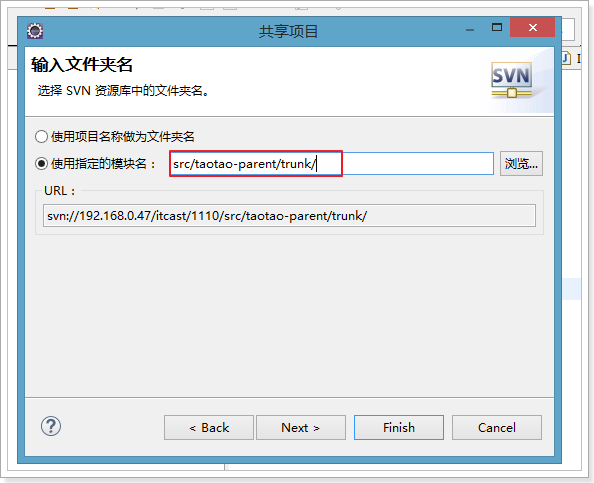
**注意：只提交src和pom.xml**

## 提交taotao-parent

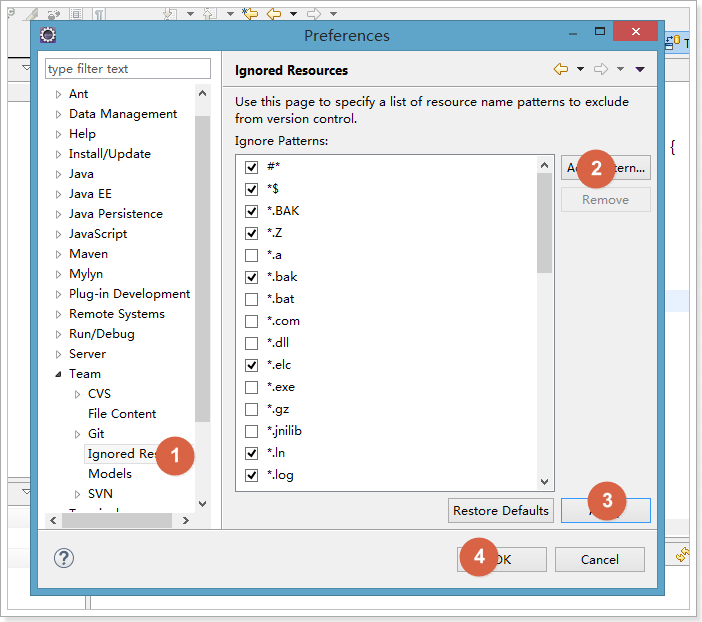




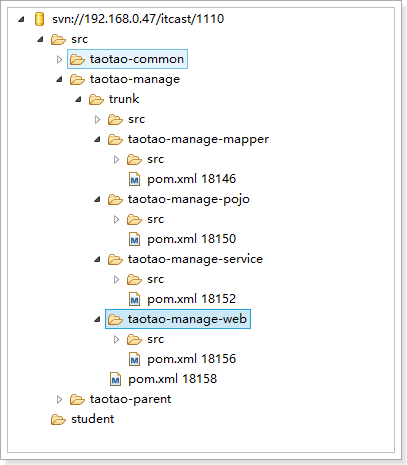




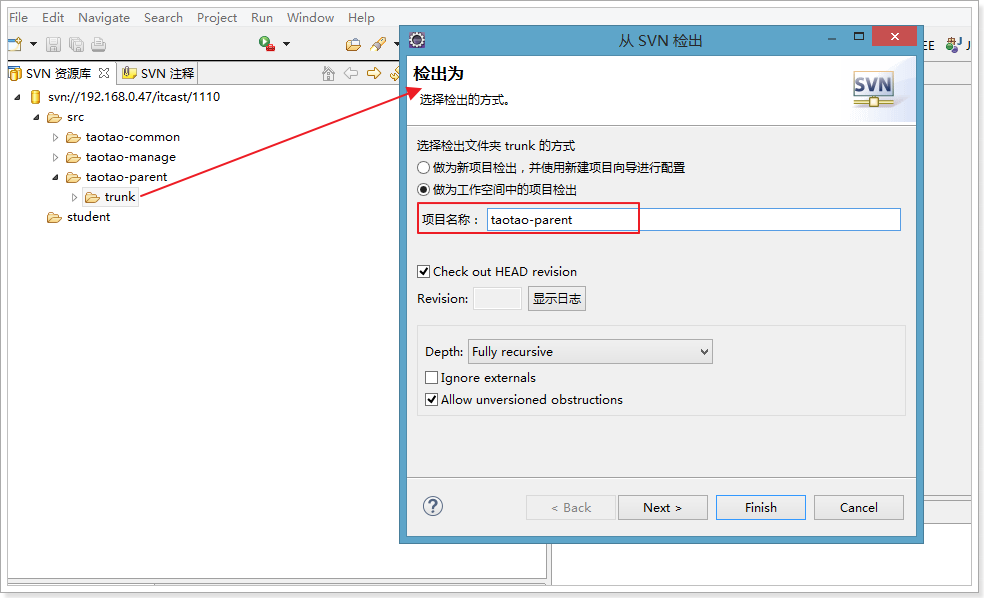
## 忽略不提交的资源



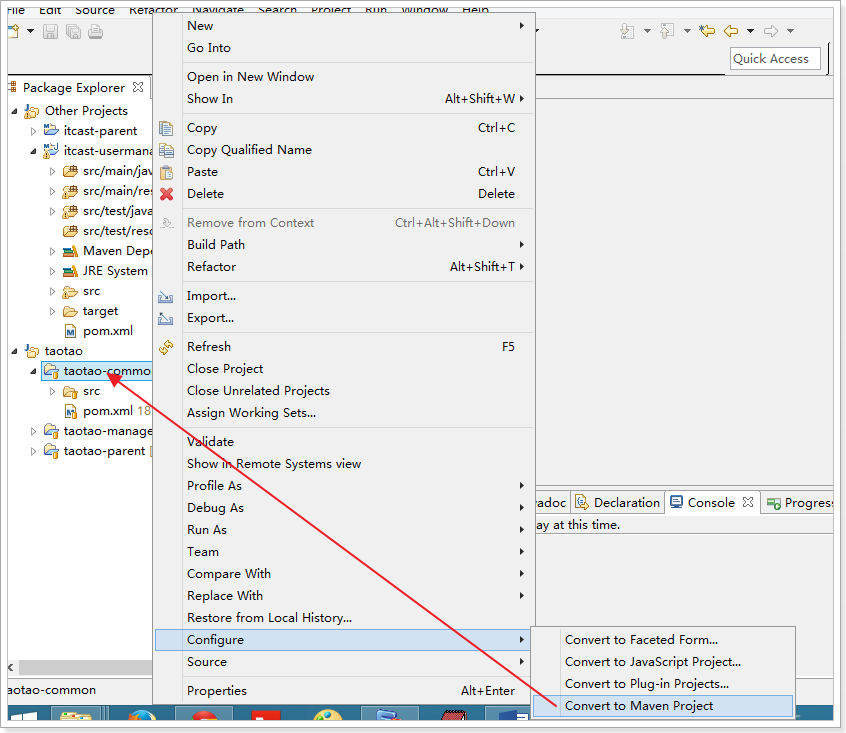
## 提交完成



# 从SVN中检出代码



转成Maven项目：



聚合工程的子工程需要导入：

