**项目实训报告**

**题 目： Kafka搭建与简单编程**

**专业班级： \*\*\*\*\*\*\***

**指导教师： \*\*\***

目录

[1.项目名称 1](#_Toc109429007)

[2.项目简介 1](#_Toc109429008)

[3.项目分工 2](#_Toc109429009)

[4.工作任务分配 2](#_Toc109429010)

[5.开发过程 3](#_Toc109429011)

[5.1 shop-search, shop-search-web模块实现[\*\*\*] 3](#_Toc109429012)

[shop-manager, shop-manager-web模块实现[\*\*\*] 8](#_Toc109429013)

[5.2 shop-search, shop-manager模块修改[\*\*\*] 11](#_Toc109429014)

[5.3 bookshop项目所有模块总测试[\*\*\*] 17](#_Toc109429015)

[6.展示图片 18](#_Toc109429016)

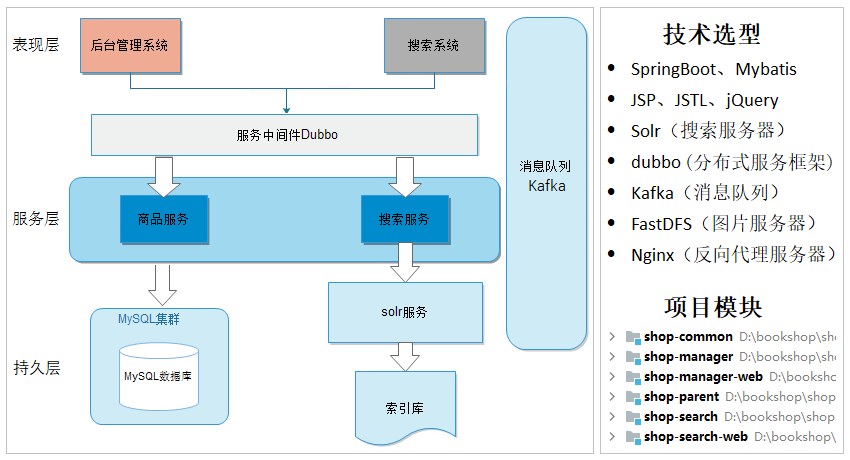
# 1.项目名称

Kafka搭建与简单编程

# 2.项目简介

本项目的目标是在Linux操作系统下搭建Zookeeper和Kafka集群并对其实现简单编程，利用Zookeeper对Kafka集群进行调度。以分布式架构编写图书后台管理系统和搜索系统，使用Kafka作为消息中间件进行系统间的通知，管理员在后台管理系统对图书进行操作时，（新增、删除、上下架等），Kafka给予搜索服务发送通知并通知SOLR服务更新索引库。

**系统架构图及技术选型**

****

**项目各模块分析**

使用单独的Maven工程构建系统的基本框架：

shop-parent Maven依赖的统一管理

shop-common 存放公用的类或功能模块

使用聚合工程搭建后台系统服务层：

shop-manager 聚合工程的父工程，负责把所有子工程聚合在一起

shop-manager-bean 聚合工程的子工程，JavaBean

shop-manager-mapper 聚合工程的子工程，DAO层

shop-manager-interface 聚合工程的子工程，业务处理层的接口

shop-manager-service 聚合工程的子工程，业务处理层

使用聚合工程搭建搜索系统服务层：

shop-search 聚合工程的父工程，负责把所有子工程聚合在一起

shop-search-interface 聚合工程的子工程，业务处理层的接口

shop-search-service 聚合工程的子工程，业务处理层

使用单独的Maven工程搭建后台系统和搜索系统表现层：

shop-manager-web 单独的Maven工程，页面控制层

shop-search-web 单独的Maven工程，页面控制层

# 3.项目分工

表 1列出项目小组成员及分工。

表 1 项目分工情况表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 学号 | 姓名 | 班级 | 项目分工 | 备注 |
| 1 | 1901010221 | \*\*\* | \*\*\*\*\* | 搭建Kafka环境  构建系统主框架  搜索系统的开发 |  |
| 2 | 1901010222 | \*\*\* | \*\*\*\*\* | 项目模块的测试 |  |
| 3 | 1901010231 | \*\*\* | \*\*\*\*\* | 前端页面的开发  后台系统的开发 |  |
| 4 | 1901010730 | \*\*\* | \*\*\*\*\* | 公共模块的开发 |  |

# 4.工作任务分配

表 2列出本项目组完成的主要模块及开发者。

表 2 项目模块及开发者明细

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 开发者 | 模块名[[1]](#footnote-1) | 开发语言[[2]](#footnote-2) | 开发形式 | 备注 |
| 1 | \*\*\* | shop-search  shop-search-web | Java、H5 | 新增 修改 测试 |  |
| 2 | \*\*\* | shop-manager  shop-search-web | Java、H5 | 新增 修改 □测试 |  |
| 3 | \*\*\* | shop-common | Java | 新增 □修改 □测试 |  |
| 4 | \*\*\* | 所有项目模块 | Java | □新增 □修改 测试 |  |

# 5.开发过程

## 5.1 shop-search, shop-search-web模块实现[\*\*\*]

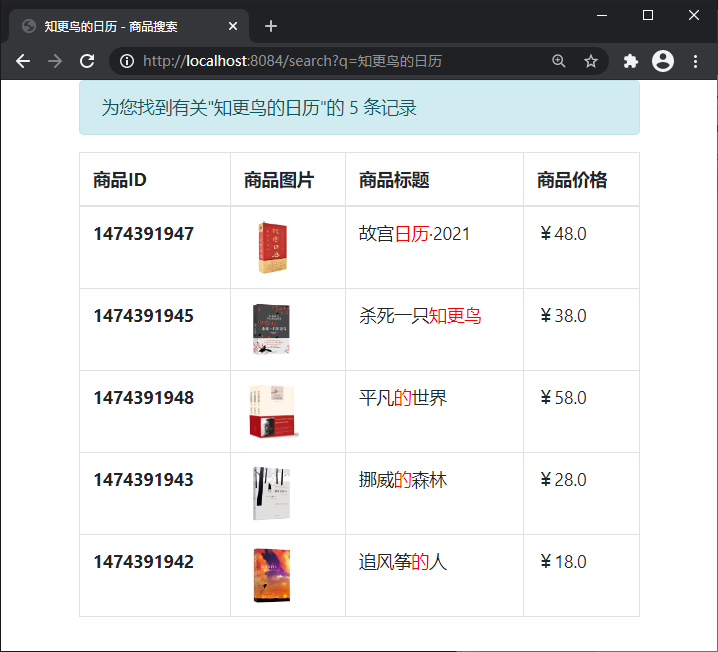
功能描述：本模块实现了图书条目的搜索功能，引入Apache SOLR由其提供搜索服务。用户在前端键入字符串时，SOLR服务会自动将字符串拆分成多个关键字，并在搜索结果页的标题（书名）关键字给予红色高亮显示。值得注意的是，搜索的结果是来源于SOLR服务所构建好的索引库，而不是直接从MySQL数据库中进行查询。

运行与测试：

①打开<http://localhost:8084/>输入需要搜索的图书关键字，可以是多个关键字。



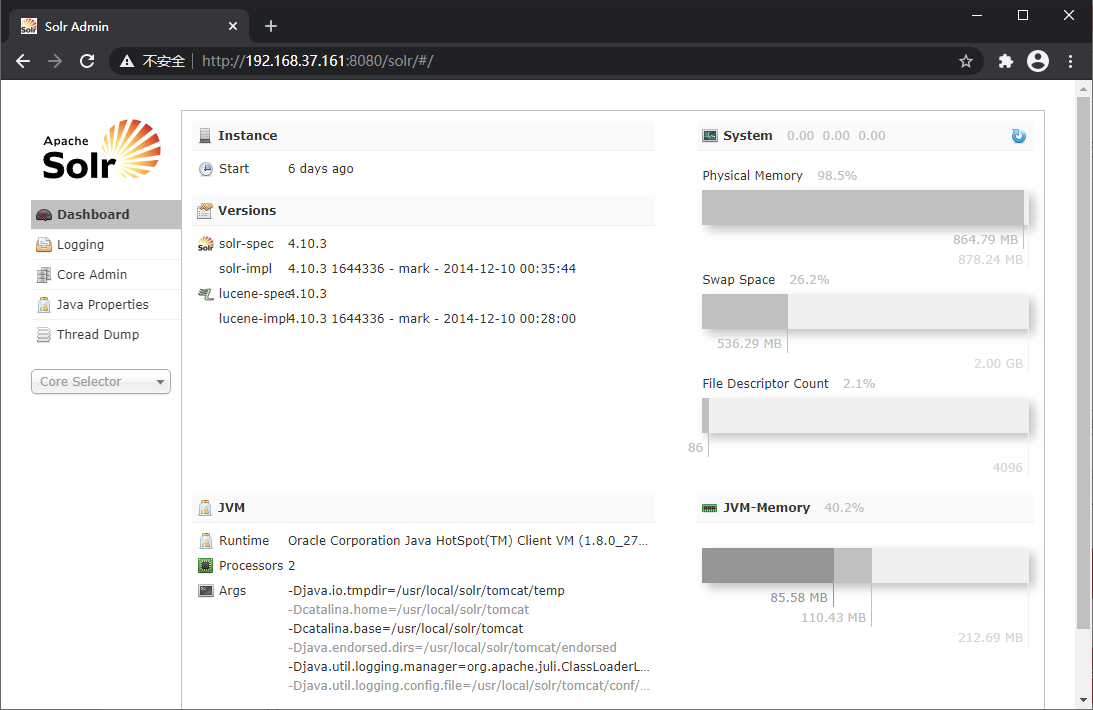
②通过观察查询结果页发现：之前输入的字符串被自动拆分成多个关键字去查询图书，被拆分的关键字注有红色高亮字体。



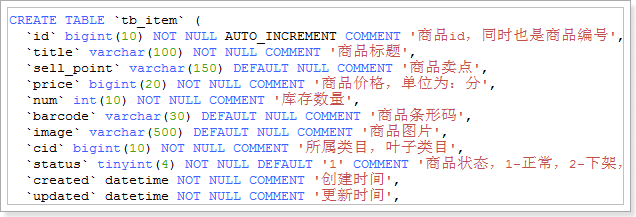
技术分析：

(1)SOLR服务的配置简介

①实现搜索系统的全局功能，首先要在Linux服务器上搭建好SOLR。



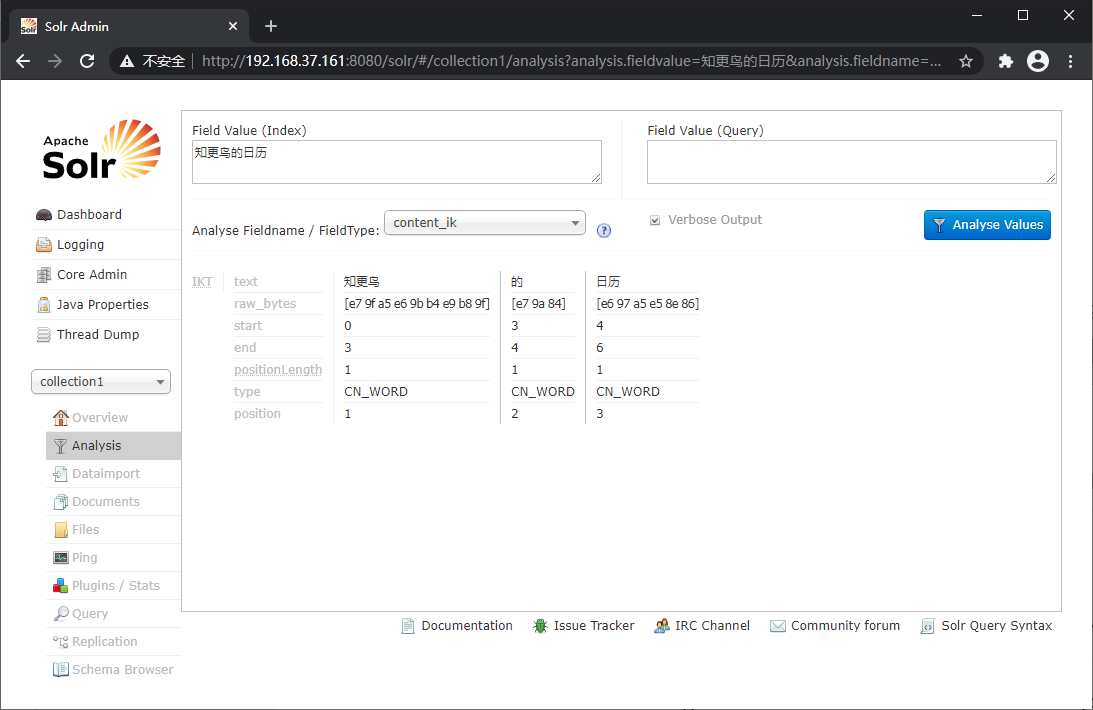
②要实现图书搜索的功能，需要配置商品的业务域，图书的数据库表：



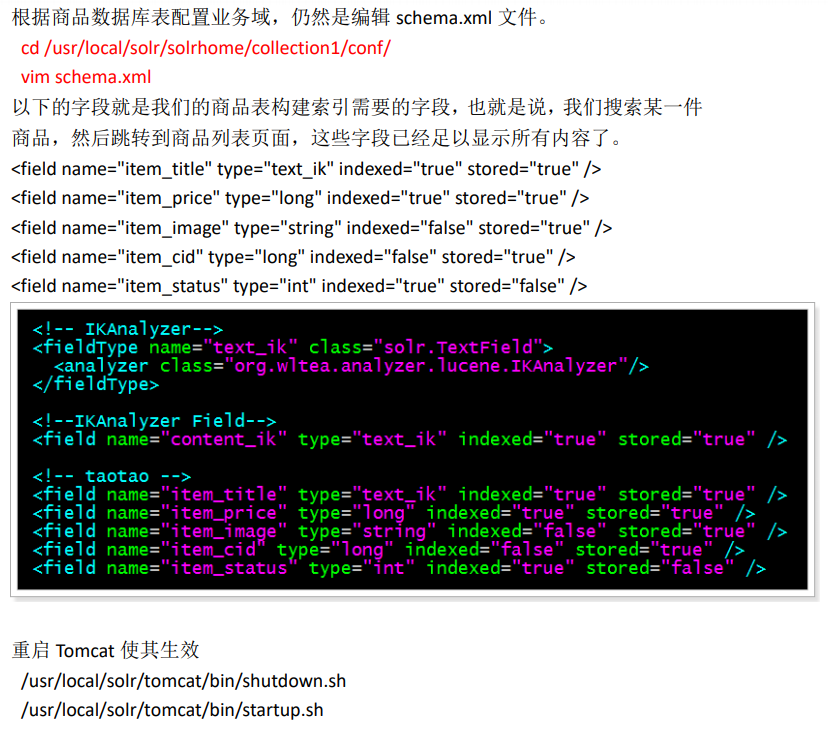
③搜索图书标题需要中文分词，需要配置中文分词器IK。



测试中文分词功能：输入“知更鸟的日历”，分词效果如下：



④根据商品数据库表配置业务域，编辑schema.xml文件。



(2)实现图书的搜索功能，编写Service层

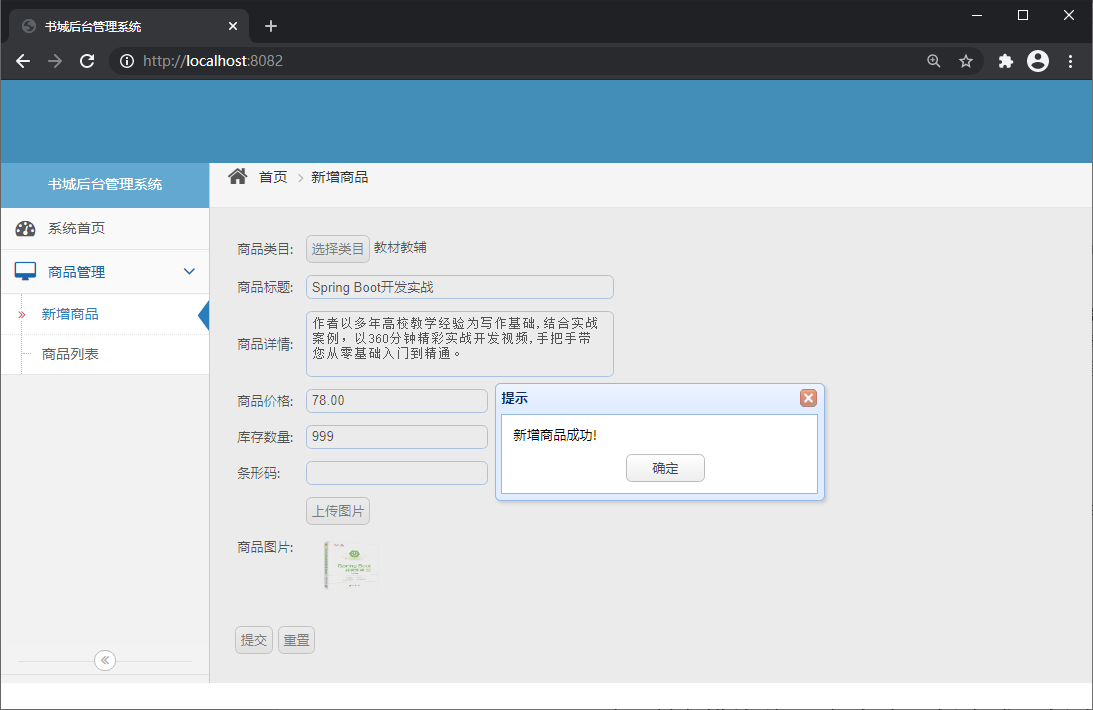


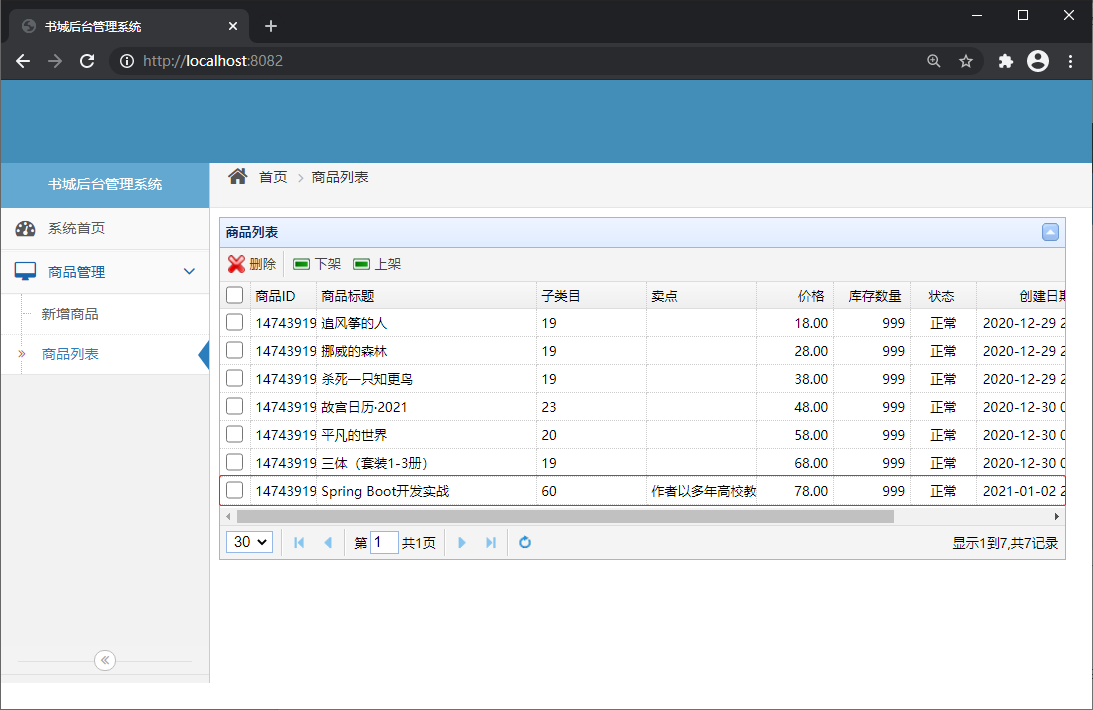
## shop-manager, shop-manager-web模块实现[\*\*\*]

功能描述：本模块实现了对图书的新增、删除以及查询功能。新增功能是指把图书信息提交到数据库中，删除功能是指在数据库删除此条记录信息，而查询功能（商品列表）则是在商品列表的页面中显示数据库所保存的全部图书信息。

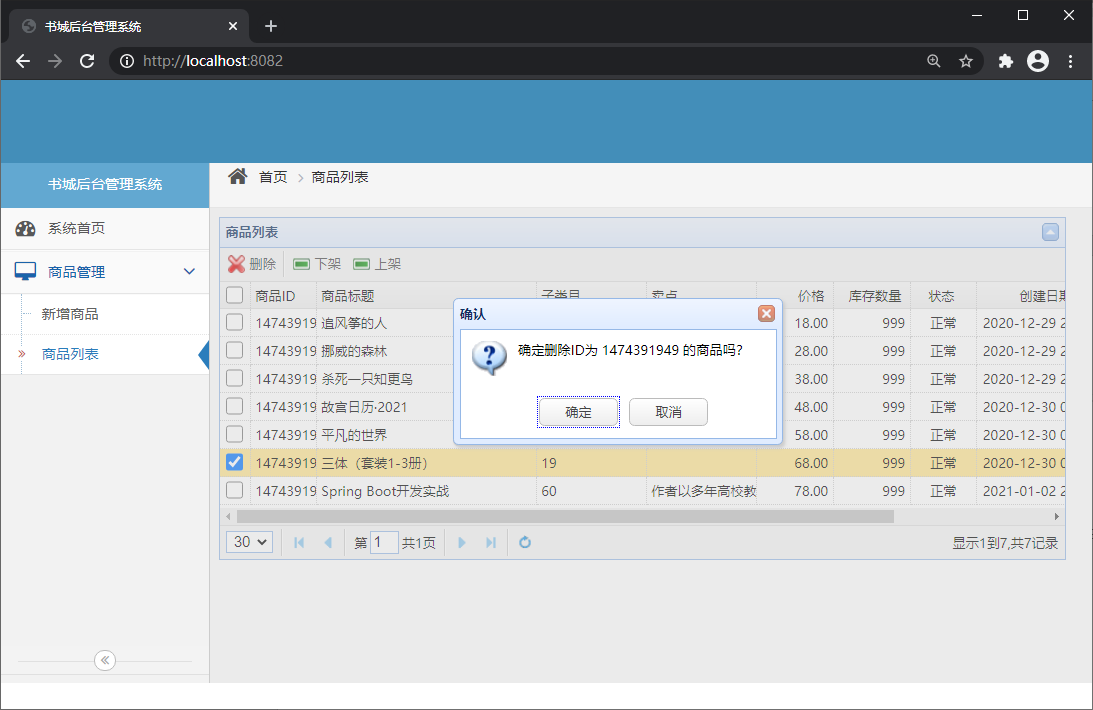
运行与测试：打开后台管理系统地址<http://localhost:8082/>

(1)首先是新增图书功能，输入图书信息并点击提交后，在商品列表页查询结果。

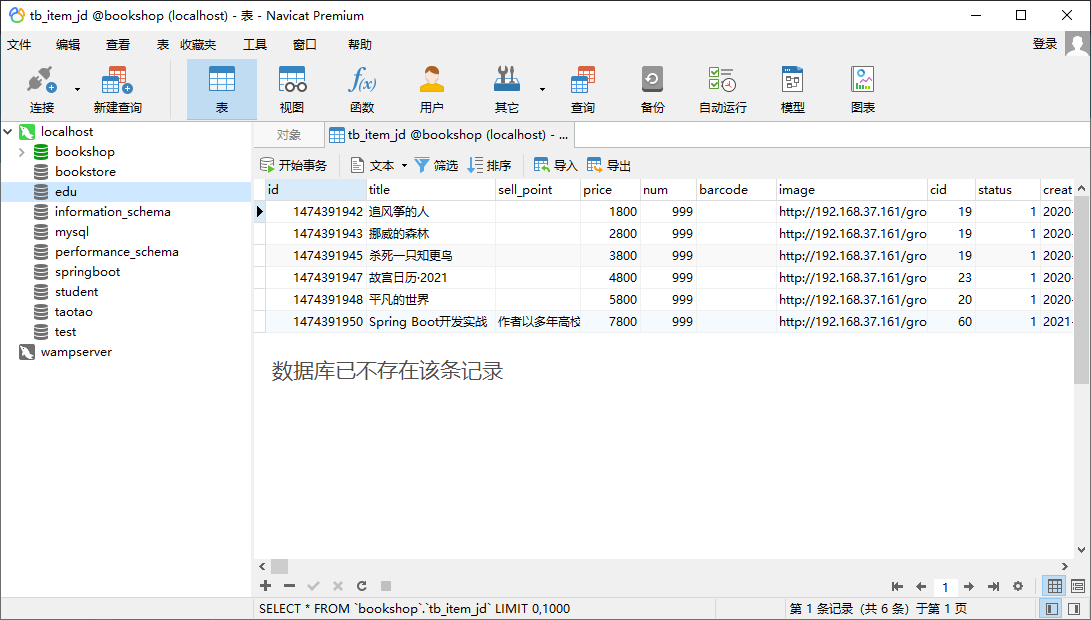




(2)然后是删除图书功能，选中图书信息并点击删除后，使用数据库工具查询结果。



(3)可见数据库中已经不存在这条记录，表明删除成功。



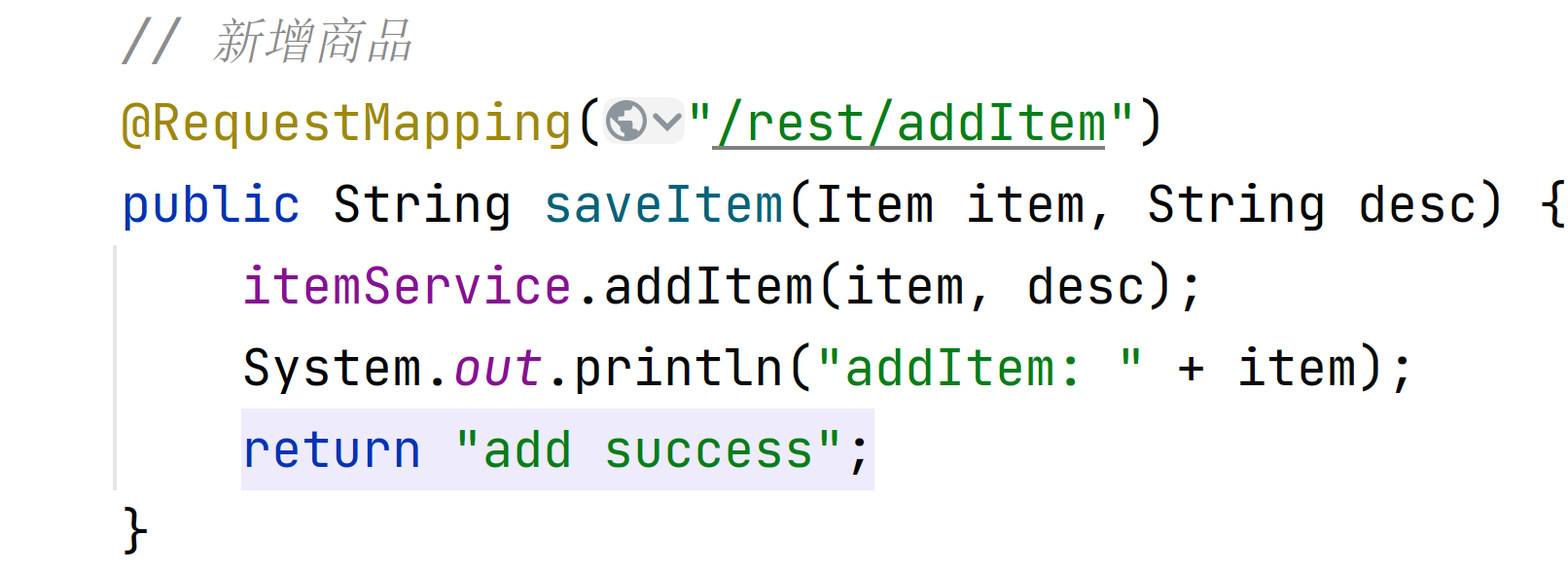
技术分析：

①图书新增功能的主要实现代码

(1)编写Service层



(2)编写Controller层



②图书删除功能的主要实现代码

(1)编写Service层



(2)编写Controller层

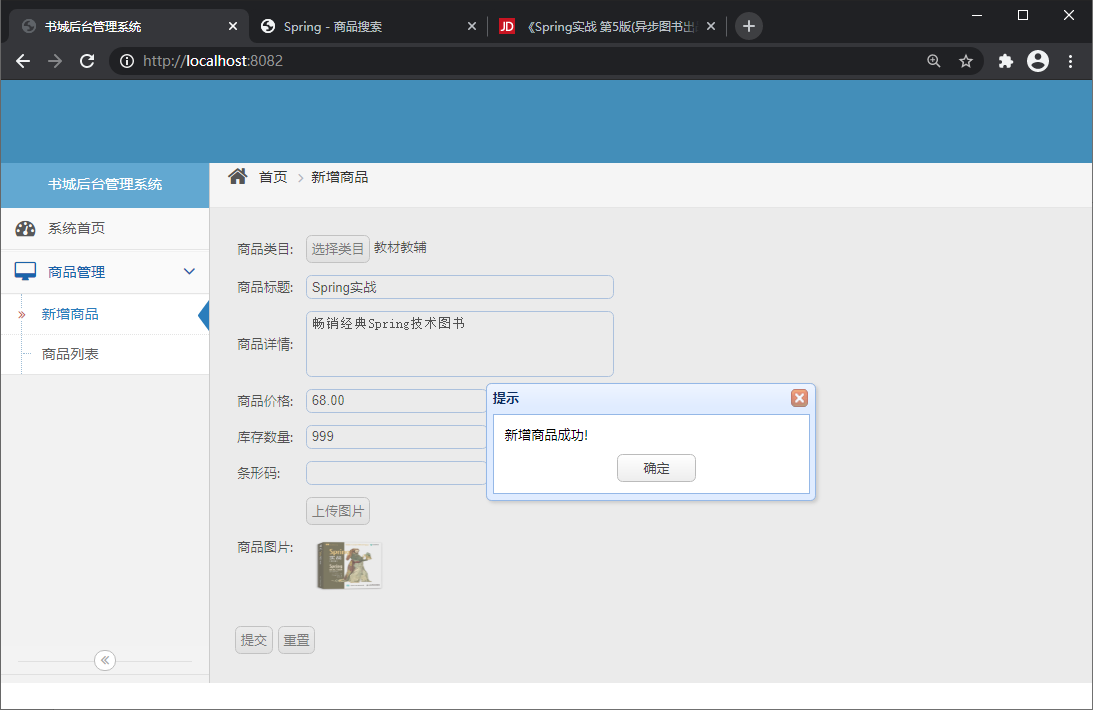


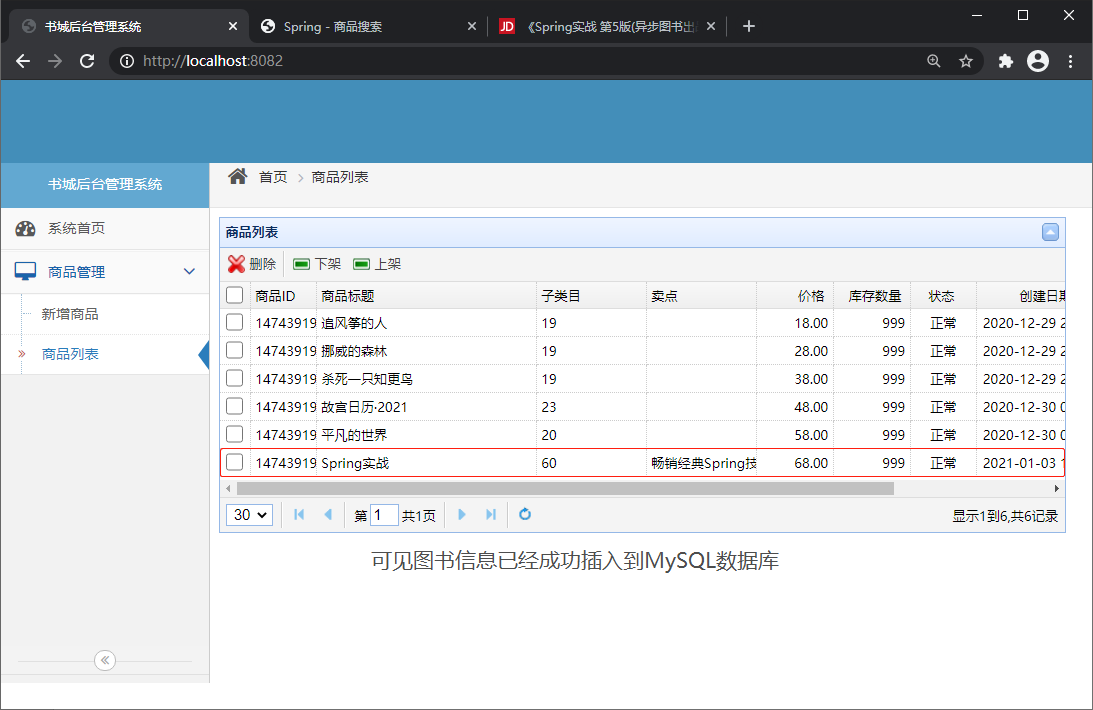
## 5.2 shop-search, shop-manager模块修改[\*\*\*]

功能描述：①本模块实现了对图书的上下架功能，引入Kafka作为消息中间件。管理员点击上架时，将选中的图书提交到SOLR索引库，从而在搜索系统能够查询到；点击下架后将图书记录从索引库删除，搜索系统此时将查询不到该图书记录。

②本模块完善了图书的新增和删除功能，引入Kafka作为消息中间件。新增图书时，将图书信息录入到数据库的基础上，同时提交到SOLR索引库；删除图书时，在数据库中删除记录的基础上，同时将该记录从索引库删除。

修改原因：由于后台系统实现图书的MySQL数据管理，而搜索系统实现图书的SOLR索引库管理，就会存在同步问题：当后台系统对图书进行增删等操作时，搜索系统（索引库）却没有同步更新。如图所示：



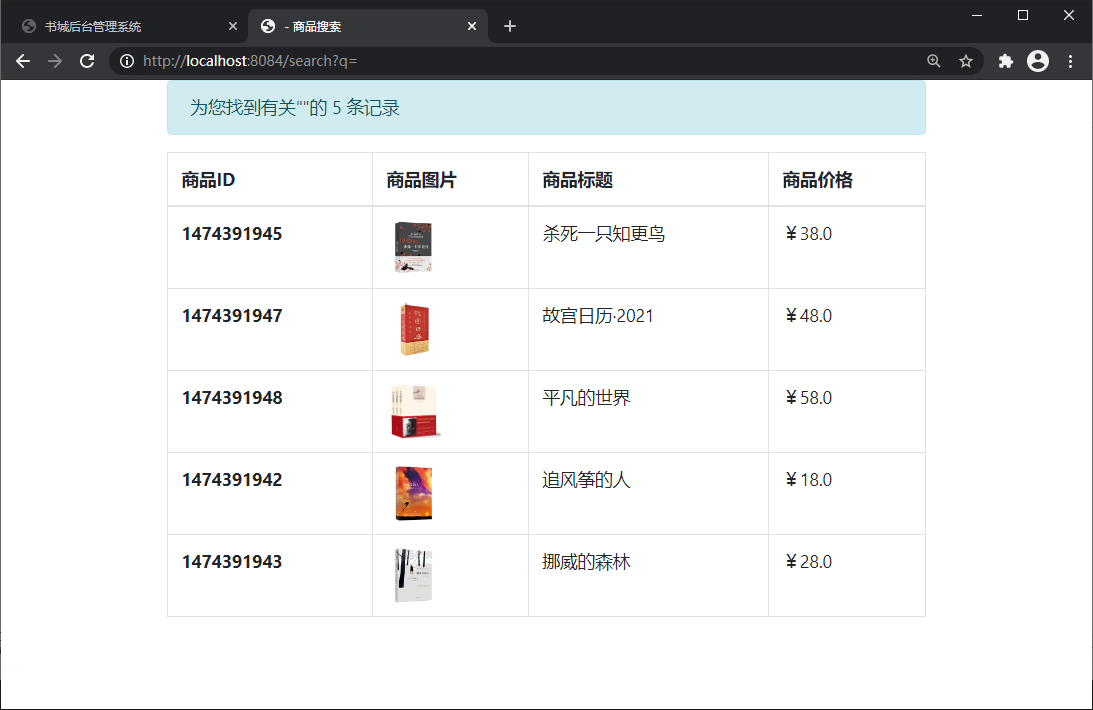




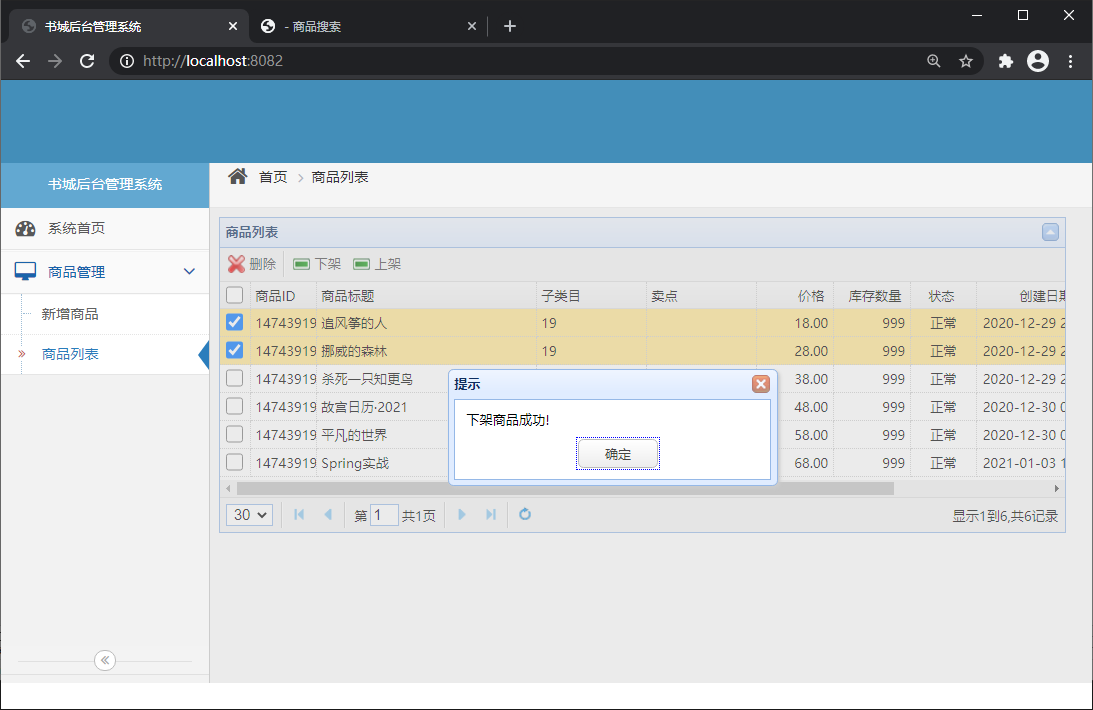
解决方案：在后台管理系统进行图书操作管理时，发送消息通知Kafka，搜索系统监听Kafka是否有消息。如果没有消息，将持续监听；如果有消息，则执行更新索引库的方法。

运行与测试：

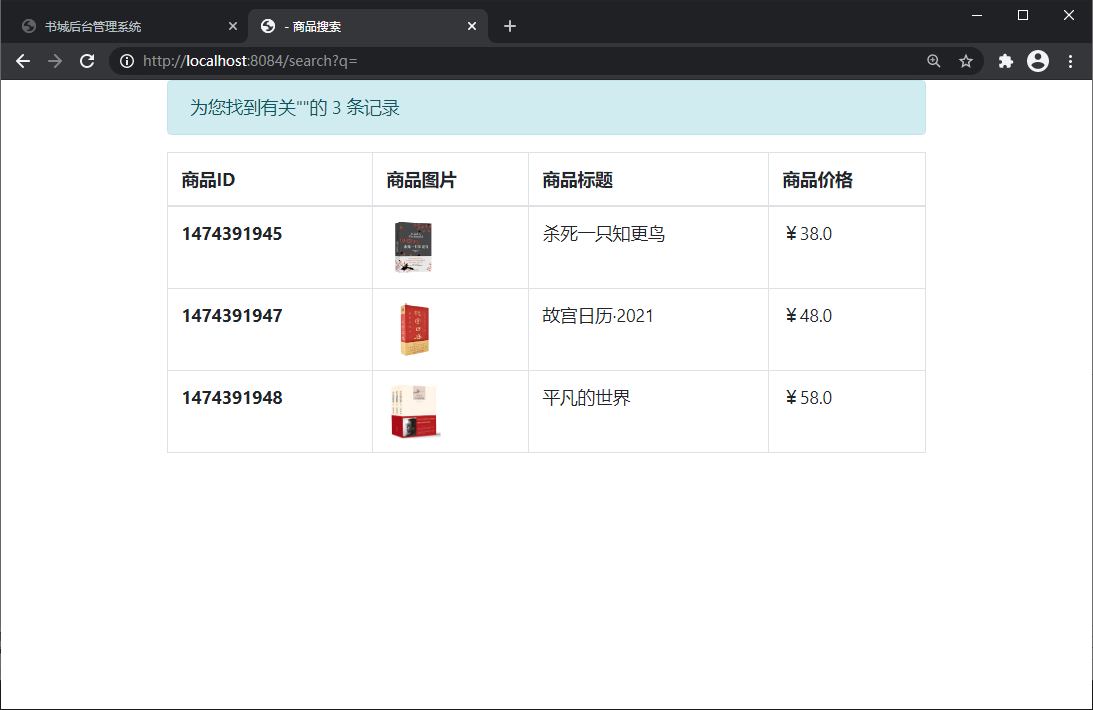
(1)首先在搜索系统查询索引库的全部记录



(2)选中其中两本图书，作下架处理（即删除索引记录）



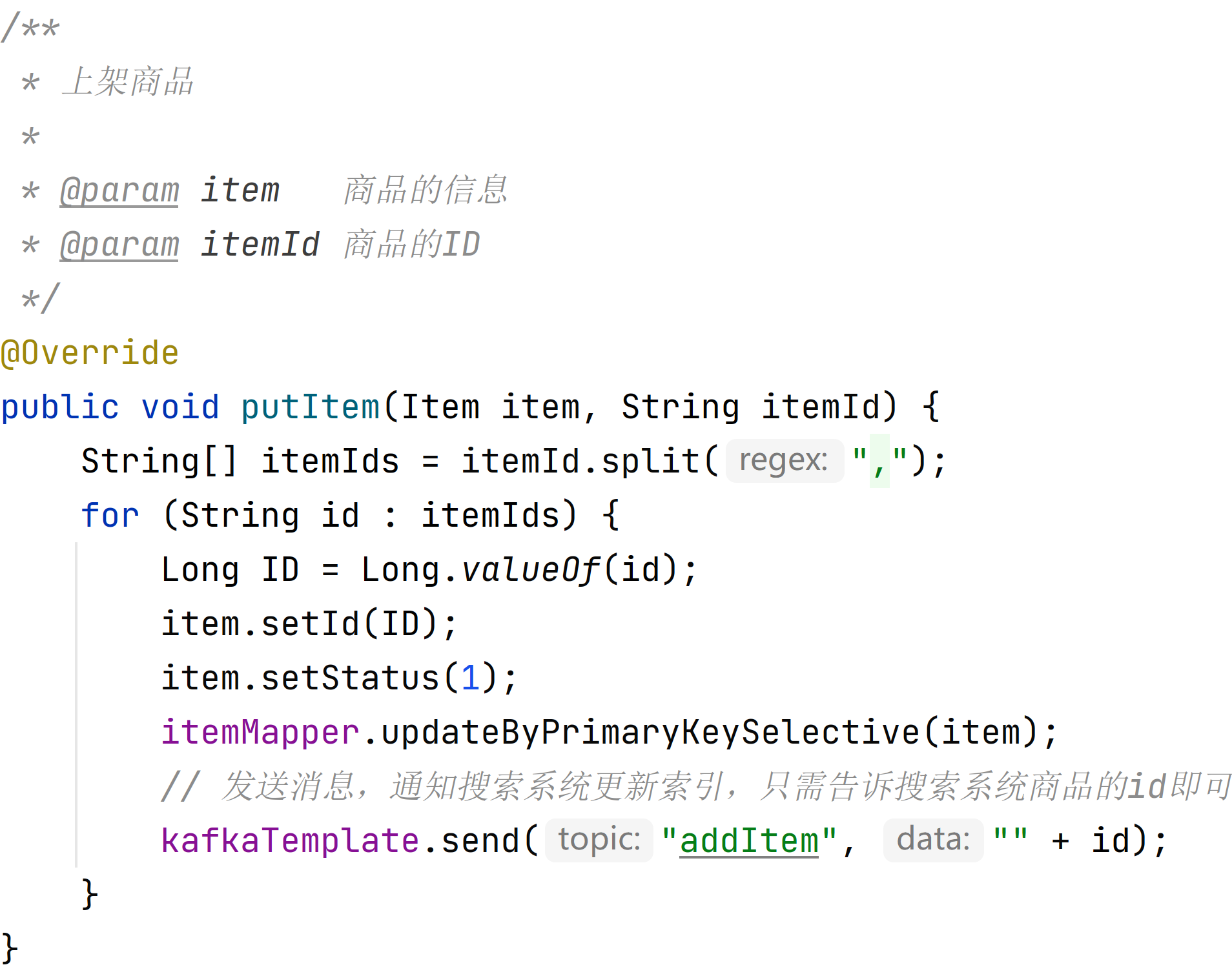
(3)再次从搜索系统查询索引库，可见选中的两本图书已经查询不到。



技术分析：

①图书上架功能的主要实现代码

(1)编写Service层



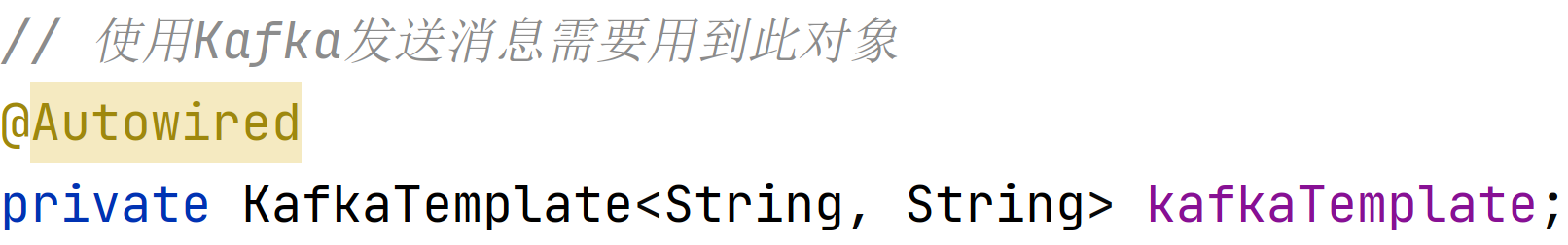


(2)编写Controller层



②使用Kafka实现消息通知的代码

(1)改造shop-manager-service的方法，加入发送消息逻辑。





(2)添加消息监听，在shop-search-service方法前面添加消息监听的注解。

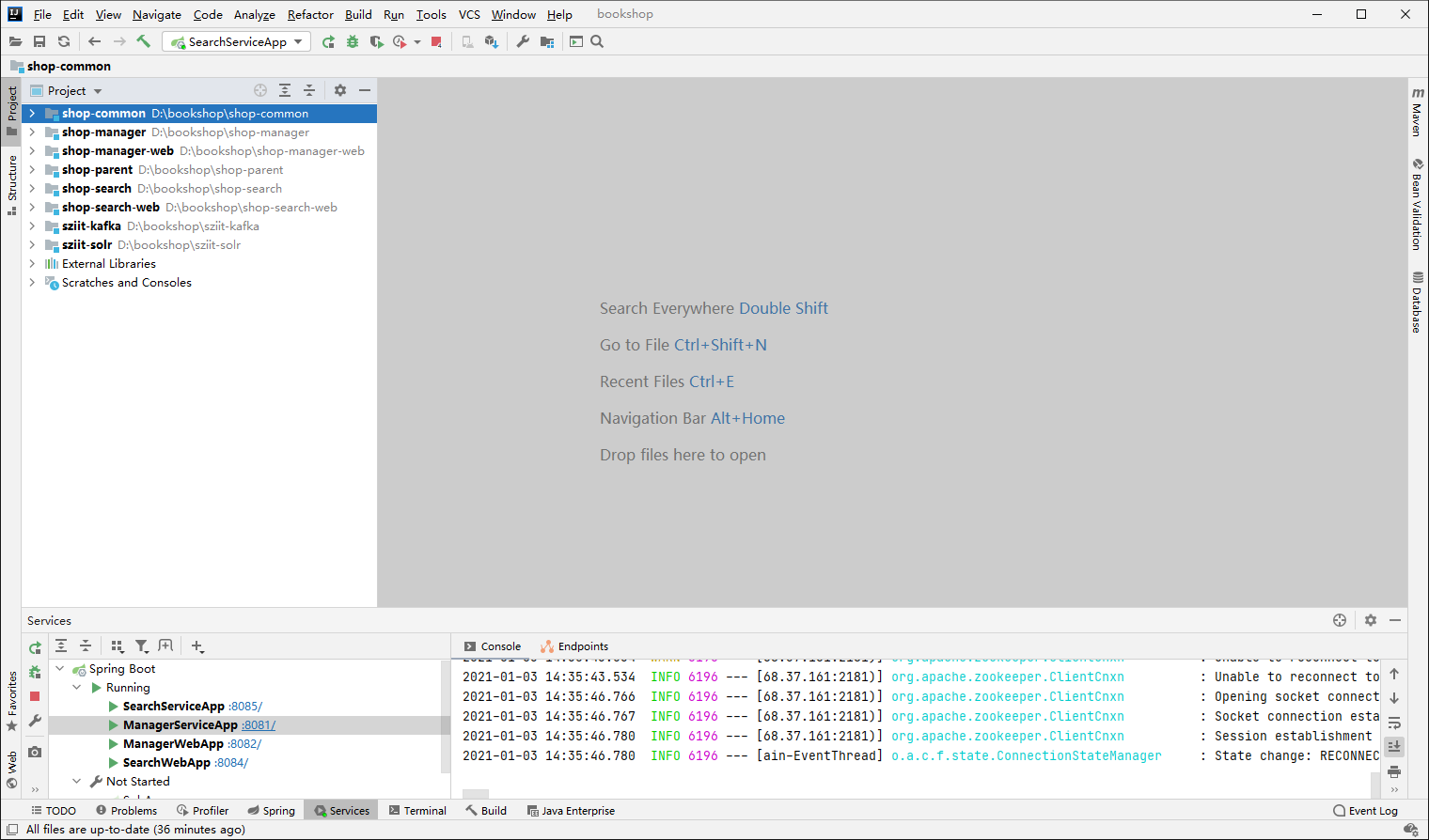


## 5.3 bookshop项目所有模块总测试[\*\*\*]

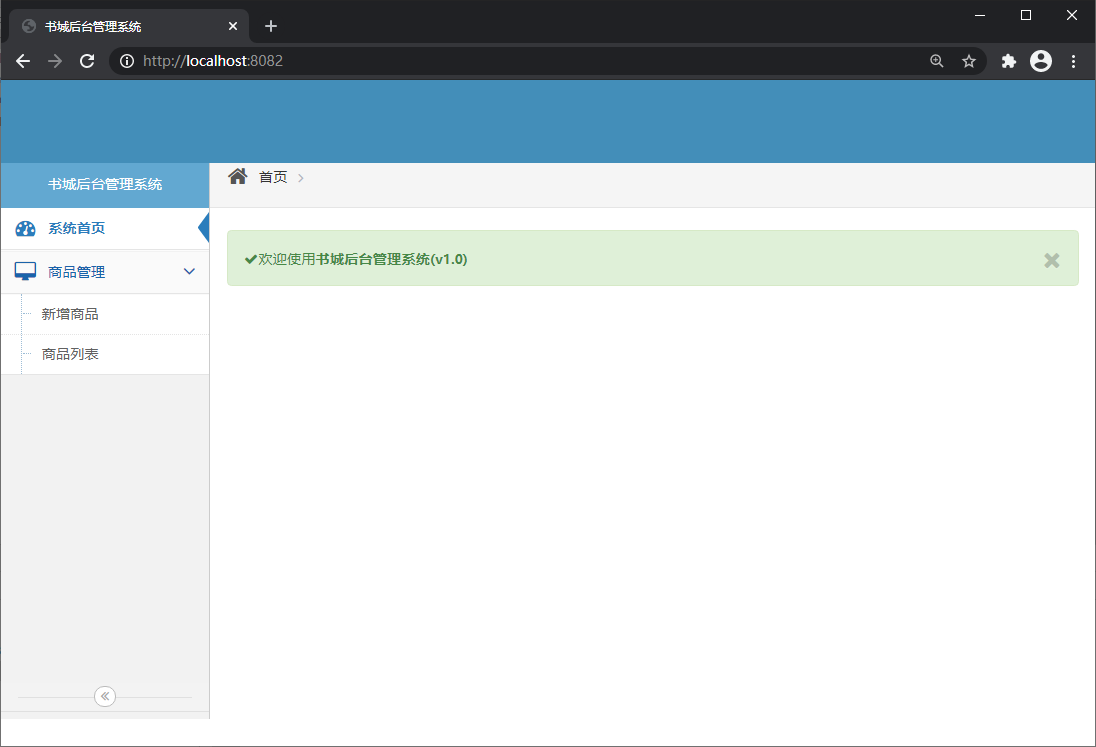
功能描述：测试本项目所有模块的功能是否正常运行，给代码补充注释。

测试用例与测试过程：使用IDEA，对逐个类逐个方法进行仔细检查。

①所有项目模块运行正常



②所有前端页面运行正常



③服务器（集群）运行正常

测试结果分析：无发现明显错误，所有模块运行正常，故测试通过。

# 6.展示图片

1. 模块指用程序代码编写的方法或H5编写的页面等。如果是程序代码编写的方法，需列举该方法的参数及类型，返回参数类型及意义。如果用Java编写，还需注明所在的包名和类名。 [↑](#footnote-ref-1)
2. 开发语言包括Java, H5, JavaScript和其它。 [↑](#footnote-ref-2)