**新巴巴运动网 项目第七天**

**传智.上官云**

# 课前回顾

1. Solr服务器搭建 1）复制lib/ext/\*.jar 到Tomcat/lib 2)复制solr.war tomcat/webapp 3)解压solr.war unzip solr.war –d solr 4)删除 solr.war 5)修改web.xml中solr/home 注解打开
2. 配置IK分词器 1）复制ik.jar到tomcat/webapp/solr/…/lib 2)配置文件.cfg.xml stopword.dic复制到tomcat/…/classes 3)修改schema.xml （自定义数据类型支持IK）text\_ik

4)自定义字段 name\_ik type=text\_ik

3、 扩展 1）修改ik.cfg.xml 打开解释 ext.dic 2)创建ext.dic 瑜伽服 换行 传智播客

4、 停止词典 1）stopword.dic 服 换行 其它

5、 Java接口SolrJ使用 SolrServer solrServer = new HttpSolrServer(); CloudSolrServer()

6、 保存商品信息到Solr服务器上（加入项目中）

7、 前台页面搜索（keyword=2016）Solr服务器 带分页 排序价格低到高

高亮

# 今天内容

主题：ActiveMQ 消息队列

1. 查询品牌（从Redis中查） Redis集群速度快 Redis中没有
2. 过滤条件查询（样式控制）
3. 去商品详情页面（单品页）
4. Sku联动（颜色变、尺码变、价格、）
5. 优化技术之一 ： 学习静态化技术（Java工程）40分钟
6. 容入项目中
7. 单点登陆系统

# 查询品牌（从Redis中查）

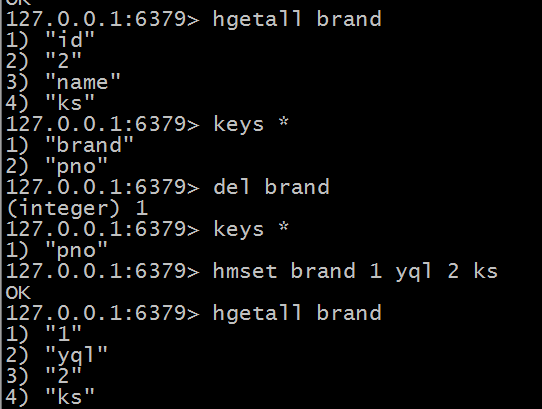
保存品牌到Redis中 （保存或修改）时保存Mysql 同时保存一份到Redis中

查询前台（系统过滤条件查询）Redis中

Redis保存 （保存字符串）

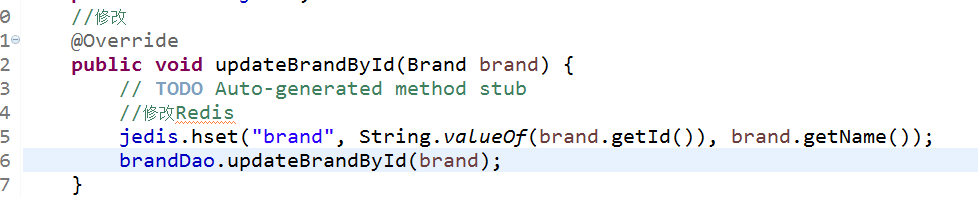
Brand对象（手动Java程序序列化成JSON字符串） K id ： V （JSON字符串）

## 命令行



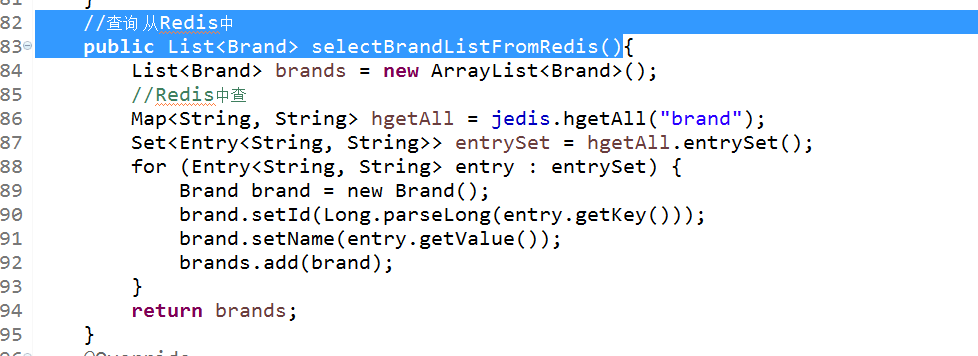
## Java 接口Jedis 重现上面命令

在BrandServiceImpl

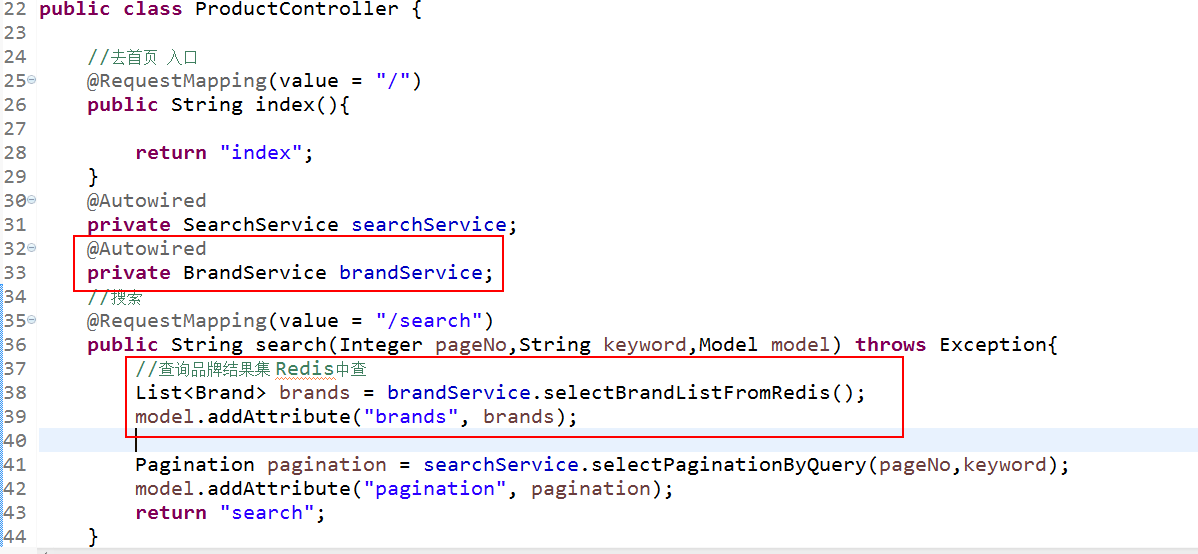


## 前台检索页面取出品牌结果集

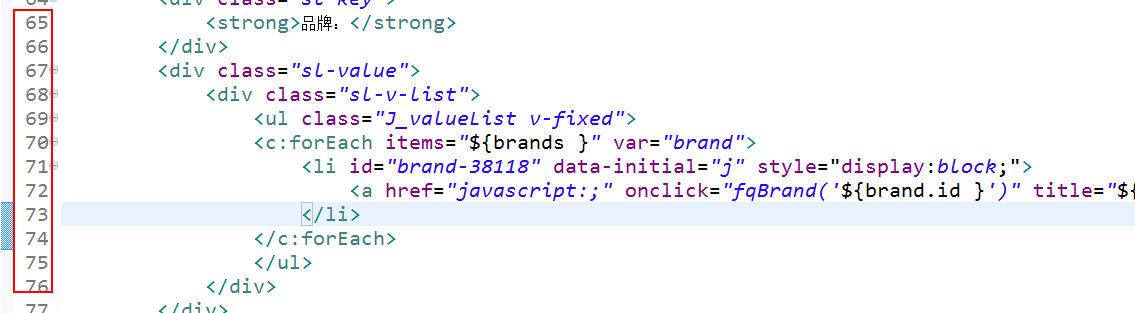
BrandService



Controller



页面

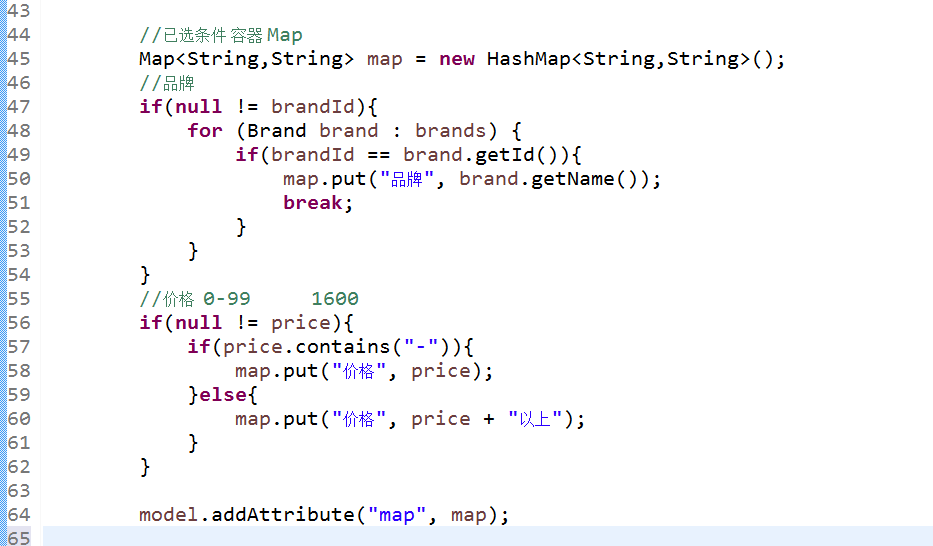


# 过滤条件

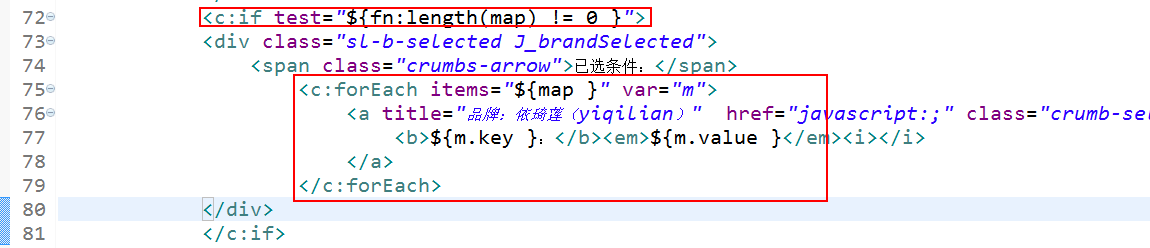
页面原型有

# 已选条件

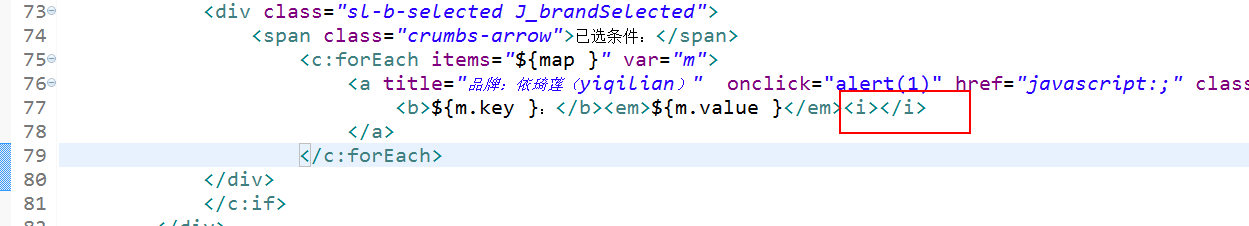
Controller



页面

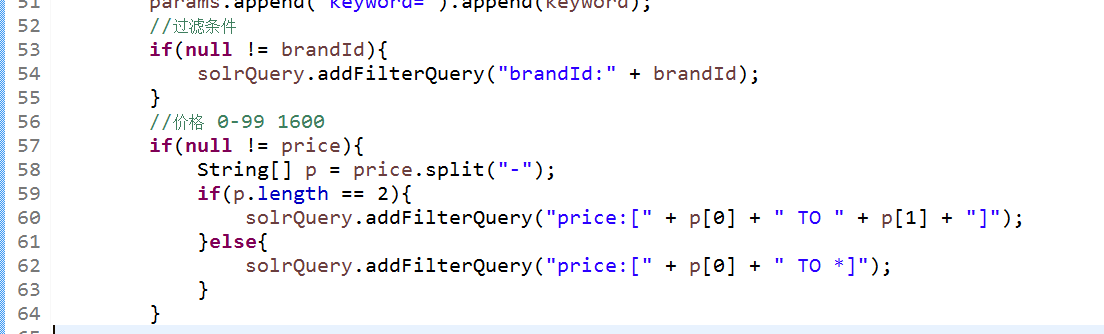


X



页面原型中有

# 过滤条件 （后台数据变化）



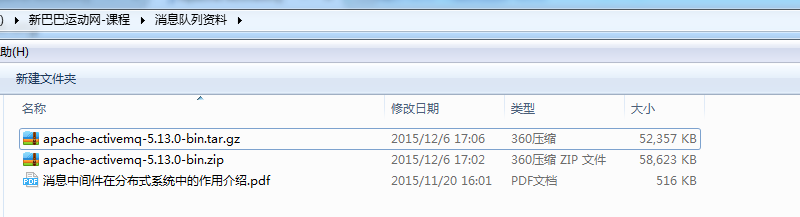
# ActiveMq

什么是ActiveMQ？

官方网站：

<http://activemq.apache.org/>





## 应用场景

财务系统

运营系统

进销存系统

订单系统

商品系统

供货商系统

。。。。。

1:银行走钱

2:成本计算

3:财务通知（确认钱） 确认

4:订单（仓库。已支付状态）

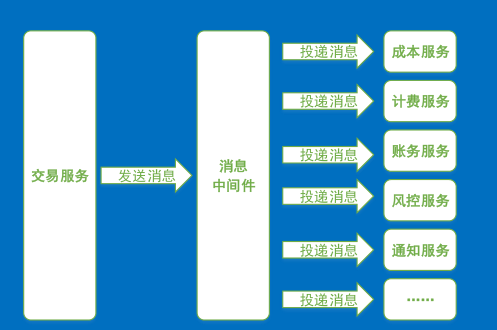
5:商家通知。。

..

…

…

支付成功!



异步处理

## 当前应用场景

Service-product 后台管理

上架 1：商品状态更改 Java接口

上架

接收消息

ActiveMQ

发送

发送

Service-cms 内容管理系统 广告

3：静态化

Service-solr 搜索

2：保存商品信到Solr服务

## ActiveMQ原理

MQ ActiveMQ

接收

四

发送

QQ

三

名称1 1,2,3

名称2

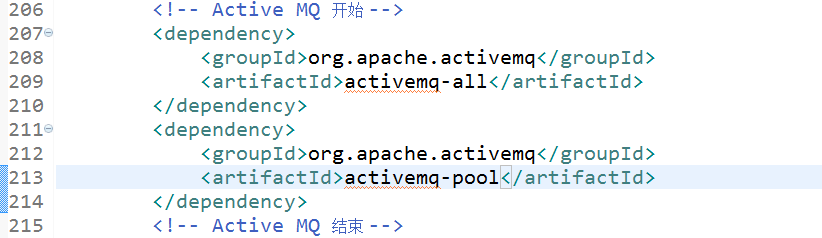
五

MQ：

第一种：队列 一对一 QQ 私聊

第二种：主题 一对多 QQ群聊

## 应用ActiveMQ



## 队列

Service-product 消息生产方

1：工厂（原厂商） Apache 提供

2：工厂连接池（原厂商）

3：Spring管理工厂连接池

4：JmsTemplate 连接ActiveMQ jdbcTemplate连接Mysql

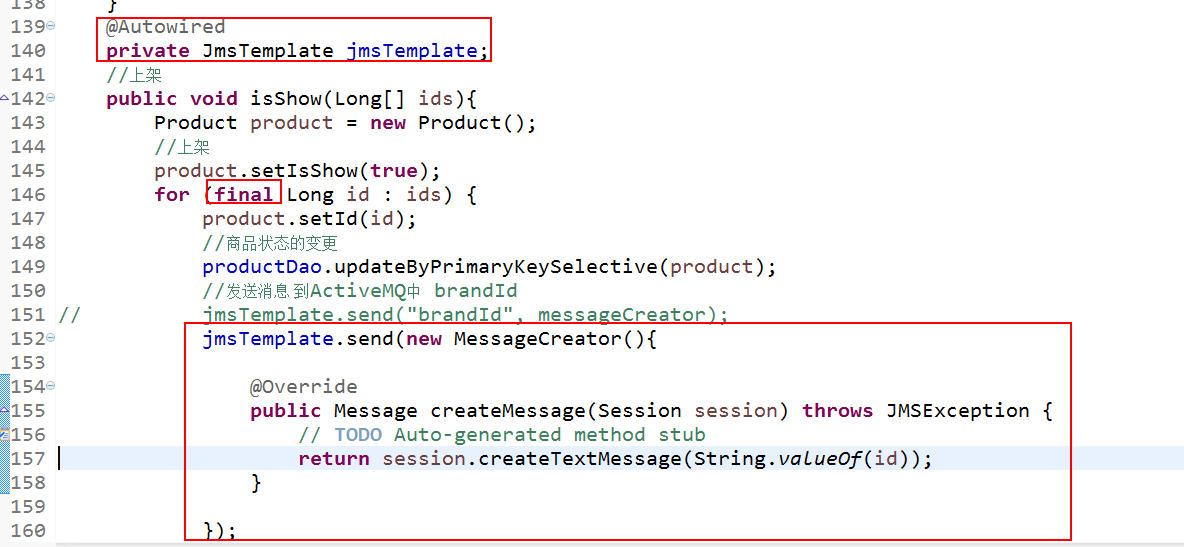
5：@AUtoward

Private JmsTemplate jmsTemplate

上架 发送（方法

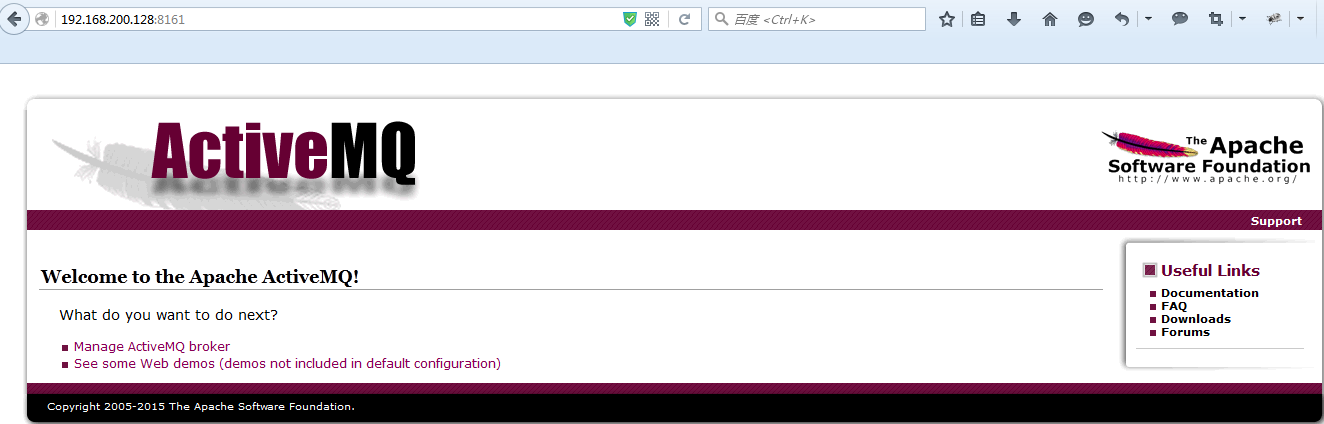
发送消息 给中间件

|  |
| --- |
| <!-- AcitveMQ 配置-->  <!-- 工厂 原厂商提供 -->  <bean id="activeMQConnectionFactory" class="org.apache.activemq.ActiveMQConnectionFactory">  <!-- 设置连接URL -->  <property name="brokerURL" value="tcp://192.168.200.128:61616"/>  <!-- 用户名、 -->  <property name="userName" value="admin"/>  <!-- 密码、 -->  <property name="password" value="admin"/>  </bean>  <!-- 工厂连接池 -->  <bean id="pooledConnectionFactoryBean" class="org.apache.activemq.pool.PooledConnectionFactoryBean">  <property name="connectionFactory" ref="activeMQConnectionFactory"/>  <!-- 最大 连接数 -->  <property name="maxConnections" value="2"/>  </bean>    <!-- Spring管理起上面的工厂 -->  <bean id="connectionFactory" class="org.springframework.jms.connection.SingleConnectionFactory">  <!-- 注入上面的工厂连接池 -->  <property name="targetConnectionFactory" ref="pooledConnectionFactoryBean"/>  </bean>    <!-- JmsTemplate由Spring提供 -->  <bean id="jmsTemplate" class="org.springframework.jms.core.JmsTemplate">  <property name="connectionFactory" ref="connectionFactory"/>  <!-- 默认目标 -->  <property name="defaultDestinationName" value="productId"/>  </bean> |

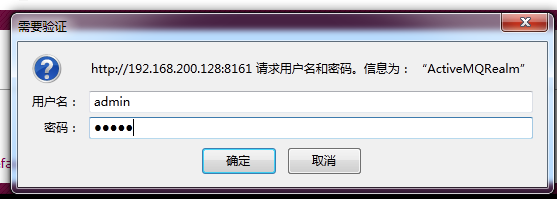


### 消息中间件

访问中间件



用户名/密码 ： admin/admin



Service-solr 消息消费方

1：工厂（原厂商） Apache 提供

2：工厂连接池（原厂商）

3：Spring管理工厂连接池

4：监听器 ActiveMQ 是否有消息 如果有接收进来

5：

处理



### 处理类

