消息队列的应用

# 1 介绍

1 网站业务日志

2 手机短信

3 邮箱邮件

4 站内信

5 订单(订单支付后调用订单系统，仓库，物流)

Success

登陆服务

登陆

其他服务

Mq中间件

日志服务

登陆日志

# 2 mq结构（在java领域，所有的mq都是基于JMS开发的,所以在写程序时大多数包都是引入的activemq依赖中的javax.jms包）

★★★★★★几个相似的名词对比，不要搞混了

Mq:消息提供者Producer -> 消息消费者consumer

SpringMvc中:返回值Produces ->参数 comsumes

①（用于webservice的rest风格中的post或get方法中的service实现类的方法上加的参数）

|  |
| --- |
| @Produces("application/json")  public Customer getCustomerById(@PathParam("id") String id) {}  @Consumes("application/x-www-form-urlencoded")// 添加是名值对形式  public String addCustomer(@BeanParam Customer customer) {} |

②@RequestMapping注解中

|  |
| --- |
| consumes = "text/plain"  produces = "text/plain" |

Dubbo:服务提供者Provider -> 服务消费者consumer

Mq分为两种：一种是队列模式，Producer 发布消息到mq服务器，然后多个consumer排队轮循去执行，只能有一个consumer执行到；

一种是订阅模式

Consumer1

Mq:queue

Producer

Consumer2

Consumer3

Mq:topic

Consumer3

Consumer2

Consumer1

Producer

# 3 日志服务功能

Mq：login

交易系统

日志系统

# 4 消息的发送和接收

Consumer1

Mq

Producer

Consumer2

过程：1Producer向mq发送消息，

2.consumer通过再mq上放监听器方式监听mq,mq一启动就被监听，（和web.xml中配置的监听spring和springmvc配置文件一样），随着mq服务器启动而启动；一旦mq有消息了，consumer就去抓消息；（不是mq向consumer推送，而是反过来consumer监听mq,从mq中抓消息）

|  |
| --- |
| 在mq的消息存储中  1 队列模式的消息，默认永久保存  2 订阅模式的消息，默认不永久保存，如果需要永久保存，需要为consumer在服务器上初始化一个永久保存的存储空间 |

# 5 任务

任务1：属性和sku搜索

任务2：完成日志系统的消息队列的使用