# Spring-boot使用RabbitMQ

1. 什么是消息队列（MQ）

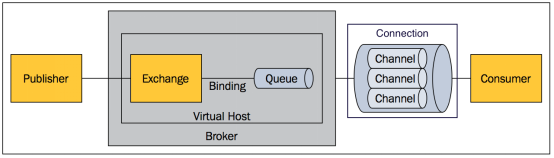
它是一种应用程序的通讯方式，可以用于分布式系统中，消息的生产者将信息不断写入消息队列中，消费者可以不断从队列中读取生产者发送的信息。而RabbitMQ就是其中的一种。

队列是RabbitMQ的内部对象，用于存储消息。生产者生产消息并投递到队列中，消费者可以从队列中获取消息并消费

多个消费者可以订阅同一个队列，这时队列中的消息会被平均分摊给多个消费者进行处理，而不是每个消费者都收到所有的消息并处理。

在RabbitMQ实际应用中，生产者将消息发送到Exchange，再通过Binding将Exchange与Queue关联起来，再建立网络连接（如TCP）,在网络连接中建立多条可复用的双向虚拟信道，不管是发布消息、订阅队列还是接收消息，这些动作都是通过信道完成（如果每次发送消息都通过建立tcp连接会消耗很多资源）。

当然队列和交换器都是存在于虚拟主机上的，虚拟主机是共享相同的身份认证和加密环境的独立服务器域，每个 vhost 本质上就是一个 mini 版的 RabbitMQ 服务器，拥有自己的队列、交换器、绑定和权限机制。

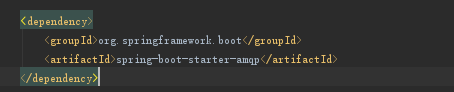


1. 为什么要用消息队列

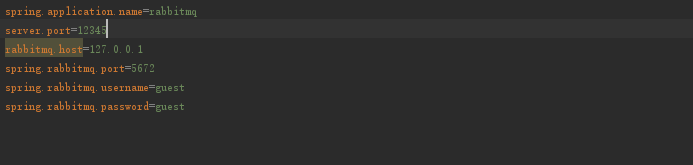
消息队列是一种应用间异步协调机制，假如在一个分布式系统中有一个订单模块，当用户点击下单操作之后，可能会与库存模块，短信通知模块，支付模块进行数据交互，这时就可以将不用立即生效的的操作分析出来进行异步处理，在点击下单的主流程（比如扣减库存，生成相应单据）完成后，发送一条消息到MQ让主流程快速结束给用户返回结果，而另外生成单独线程拉取MQ的消息让给用户发送短信并完成支付。

1. 如何在springboot中使用rabbitmq

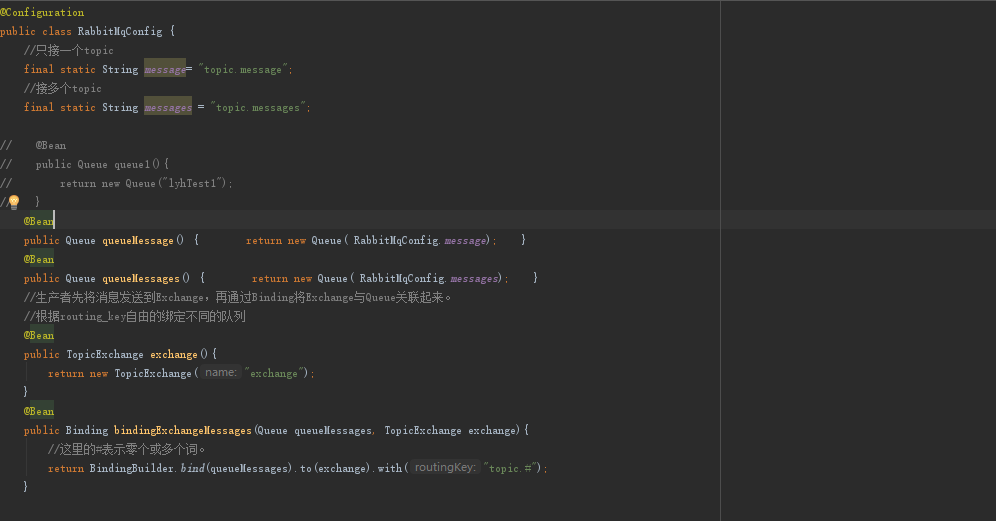
1，新建springboot项目，在maven中添加如下依赖



1. 在application.properties添加rabbitmq的相关信息，并安装rabbitmq（搜索安装教程）



1. 添加配置类

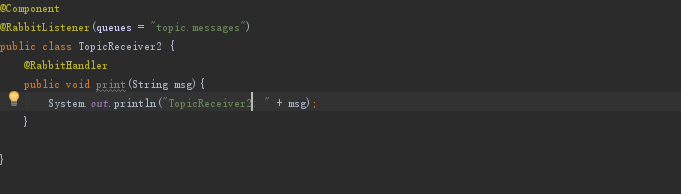


1. 写一个测试的api用于生产消息

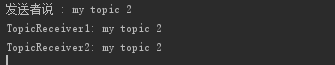


1. 写2个测试的receive消费消息





6，浏览器访问http://localhost:123456/topicSend2，终端输出：



浏览器访问http://localhost:123456/topicSend1，终端输出：

