

2-10 【基础额外补充-图文】RabbitMQ高级特性

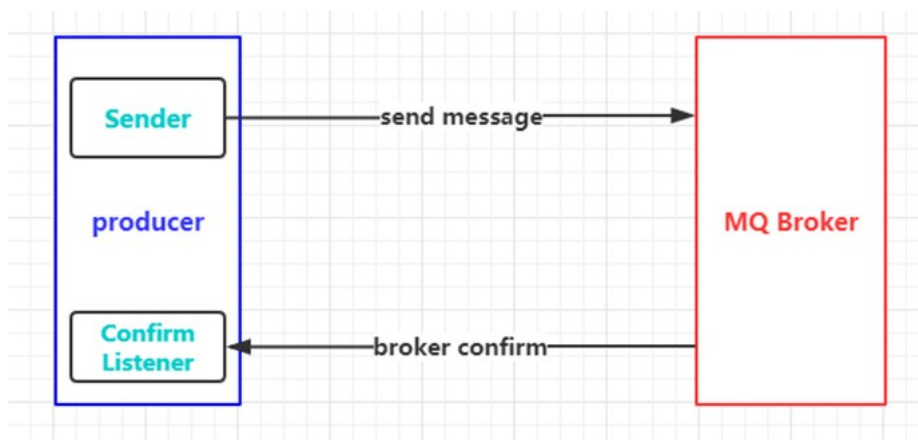
课前说明

考虑到一些同学对RabbitMQ高级特性已经遗忘的差不多了，也收到一些没有接触过RabbitMQ的同学的反馈，所以额外为大家准备了快入回顾或入门RabbitMQ图文教程【部分内容也会补充视频讲解】希望能够帮助到这部分同学。

Confirm确认消息

理解Confirm消息确认机制，是学习RabbitMQ的知识点中非常非常非常（重要的事情说三遍）重要的！

- 消息的确认，是指生产者投递消息后，如果Broker收到消息，则会给我们生产者一个应答。
- 生产者进行接收应答，用来确定这条消息是否正常的发送到Broker，这种方式也是消息的可靠性投递的核心保障
- 接下来我们通过一幅图说明Confirm消息，大致上就是producer发送消息给broker，broker把消息进行持久化，之后由broker主动发送ack响应给消息的生产者确认这次响应，注意这里的响应过程是异步的。



- 代码级别如何实现Confirm确认消息呢，我们一起来梳理一下，简单两步骤即可：
 - 第一步：在channel上开启确认模式：channel.confirmSelect()
 - 第二步：在channel上添加监听：addConfirmListener，监听成功和失败的返回结果，根据具体的结果对消息进行重新发送、或继续处理！
- 相关代码示例：
 - Producer

```
package com.bfxy.rabbitmq.api.confirm;

import com.rabbitmq.client.Channel;
import com.rabbitmq.client.Connection;
import com.rabbitmq.client.ConnectionFactory;
import com.rabbitmq.client.QueueingConsumer;
import com.rabbitmq.client.QueueingConsumer.Delivery;

public class Consumer {

    public static void main(String[] args) throws Exception {
        //1 创建ConnectionFactory
        ConnectionFactory connectionFactory = new ConnectionFactory();
        connectionFactory.setHost("192.168.11.76");
        connectionFactory.setPort(5672);
        connectionFactory.setVirtualHost("/");
        //2 获取Connection
        Connection connection = connectionFactory.newConnection();
        //3 通过Connection创建一个新的Channel
```

检测到您还没有关注慕课网服务号，无法接收课程更新通知。请扫描二维码即可绑定



下一节

2-8 附：RabbitMQ急速入门

播放下一节

重新观看