



链滴

# RabbitMQ 消息队列搭建与发送简单消息

作者: [NiuGeH](#)

原文链接: <https://ld246.com/article/1566548338010>

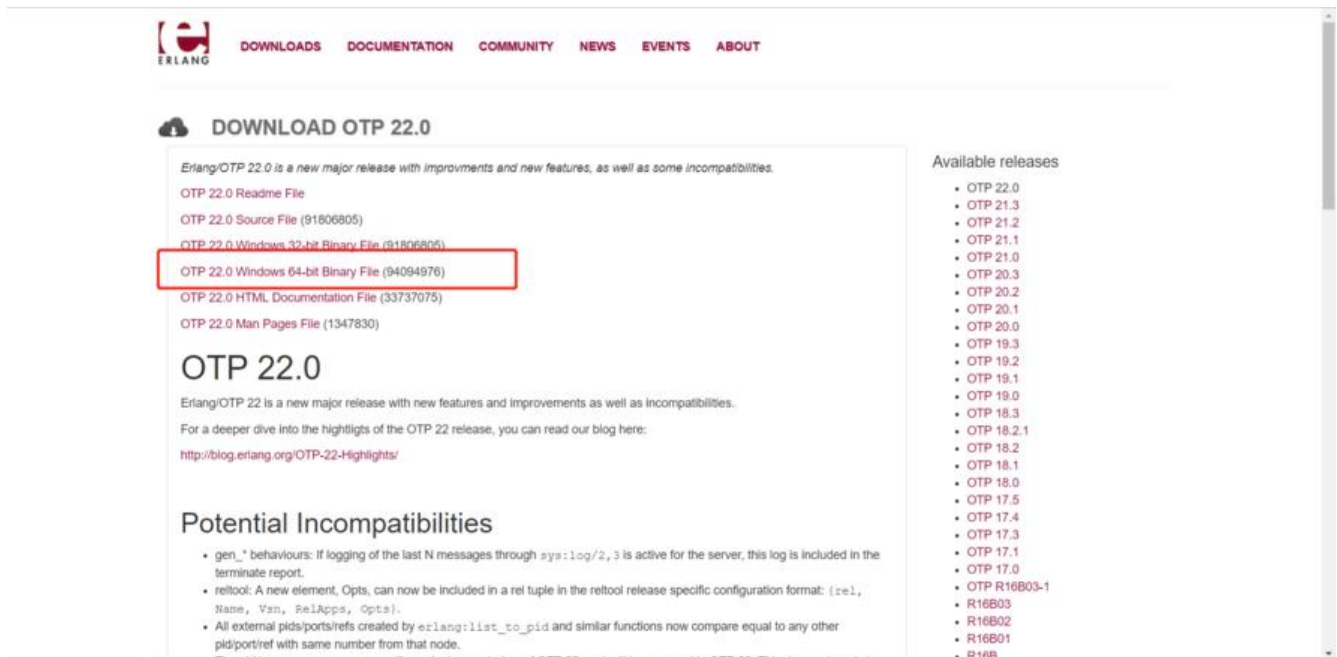
来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## RabbitMQ下载官网进行下载



下载后还必须下载并搭建Erlang环境 因为RabbitMQ 是基于Erlang 开发的



## 下载完成之后

名称	修改日期	类型	大小
otp_win64_22.0.exe	2019/8/23 12:36	应用程序	91,890 KB
rabbitmq-server-3.7.17.exe	2019/8/23 11:58	应用程序	9,876 KB

## 先安装并配置Erlang环境

一直点击下一步即可 都是默认

打开环境变量 在path后添加你的安装目录 + /bin (可以看到消息队列环境搭不上图了)

安装RabbitMQ 同Erlang一样 都是只用点击下一步即可

## 重点

安装完成后 win+R 进入cmd 再进入你安装RabbitMQ 下的目录 下的/sbin

```
选择C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.17134.950]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\NiuGeH>cd C:\Program Files\RabbitMQ Server\rabbitmq_server-3.7.17\sbin
C:\Program Files\RabbitMQ Server\rabbitmq_server-3.7.17\sbin>_
```

接着输入 `rabbitmq-plugins enable rabbitmq_management` 进行 配置

完成后如下

```
Microsoft Windows [版本 10.0.17134.950]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\NiuGeH>cd C:\Program Files\RabbitMQ Server\rabbitmq_server-3.7.17\sbin
C:\Program Files\RabbitMQ Server\rabbitmq_server-3.7.17\sbin>rabbitmq-plugins enable rabbitmq_management
Enabling plugins on node rabbit@LAPTOP-6PIQBCJ0:
rabbitmq_management
The following plugins have been configured:
  rabbitmq_management
  rabbitmq_management_agent
  rabbitmq_web_dispatch
Applying plugin configuration to rabbit@LAPTOP-6PIQBCJ0...
Plugin configuration unchanged.

C:\Program Files\RabbitMQ Server\rabbitmq_server-3.7.17\sbin>_
```

## 注意

此时百分之99的人进不去如下这个页面



你还需要 显示隐藏的文件夹 再进入你的 `C:\Users\你的用户名\AppData\Roaming\RabbitMQ\db` 将该文件夹下的所有文件删除 删除后再此执行下面程序

 otp_win64_22.0.exe	2019/8/23 12:36	应用程序	91,890 KB
 rabbitmq-server-3.7.17.exe	2019/8/23 11:58	应用程序	9,876 KB

之后进入 安装RabbitMQ 的目录下sbin

名称	修改日期	类型	大小
cuttlefish	2019/7/29 11:28	文件	458 KB
rabbitmqctl.bat	2019/7/29 11:28	Windows 批处理...	3 KB
rabbitmq-defaults.bat	2019/7/29 11:28	Windows 批处理...	2 KB
rabbitmq-diagnostics.bat	2019/7/29 11:28	Windows 批处理...	3 KB
rabbitmq-echopid.bat	2019/7/29 11:28	Windows 批处理...	2 KB
rabbitmq-env.bat	2019/7/29 11:28	Windows 批处理...	18 KB
rabbitmq-plugins.bat	2019/7/29 11:28	Windows 批处理...	3 KB
rabbitmq-server.bat	2019/7/29 11:28	Windows 批处理...	11 KB
rabbitmq-service.bat	2019/7/29 11:28	Windows 批处理...	14 KB

双击运行

等待 等待 等待

```
## ##
## ##  RabbitMQ 3.7.17. Copyright (C) 2007-2019 Pivotal Software, Inc.
##### Licensed under the MPL. See https://www.rabbitmq.com/
##### ##
##### Logs: C:/Users/NiuGeH/AppData/Roaming/RabbitMQ/log/RABBIT~1.LOG
          C:/Users/NiuGeH/AppData/Roaming/RabbitMQ/log/rabbit@LAPTOP-6PIQBCJ0_upgrade.log

Starting broker...
completed with 3 plugins.
```

- 浏览器访问 <http://localhost:15672/>

输入 guest 密码与用户名相同

## 简单的消息队列Demo

- 创建 maven 项目
- pom.xml 添加

```
<dependency>
  <groupId>com.rabbitmq</groupId>
  <artifactId>amqp-client</artifactId>
  <version>5.7.2</version>
</dependency>
```

- 创建生产者实例

```
public static void main(String[] args) {
    // 创建连接工厂
    ConnectionFactory factory = new ConnectionFactory();
    // 设置端口
```

```

factory.setHost("localhost");
factory.setVirtualHost("/");
// 用户名 密码
factory.setUsername("guest");
factory.setPassword("guest");

try {
    Connection connection = factory.newConnection();
    //Connection 来创建 Channel
    Channel channel = connection.createChannel();
    for (int i = 0 ; i < 10 ; i++){
        //生产模式中多为 json 格式
        String msg = "Rabbit消息"+i;
        channel.basicPublish("", "NiuMsg", null, msg.getBytes());
        Thread.sleep(1000);
    }
    if (channel != null){
        channel.close();
        connection.close();
    }
} catch (Exception e){
    e.printStackTrace();
}

}

```

- 创建消费者实例(接受消息)

```

public static void main(String[] args) throws Exception{
    //1. 创建连接工厂
    ConnectionFactory factory = new ConnectionFactory();
    factory.setHost("localhost");
    factory.setVirtualHost("/");
    factory.setUsername("guest");
    factory.setPassword("guest");

    //2. 创建连接
    Connection connection = factory.newConnection();

    //3. 通过 Connection 来创建 Channel
    Channel channel = connection.createChannel();

    //4. 声明一个队列
    //此处参数下面为大家介绍
    channel.queueDeclare("NiuMsg", true, false, false, null);

    //5. 创建消费者并接收消息
    Consumer consumer = new DefaultConsumer(channel) {
        @Override
        public void handleDelivery(String consumerTag, Envelope envelope,
                                   AMQP.BasicProperties properties, byte[] body)
            throws IOException {
            String message = new String(body, "UTF-8");
            System.out.println("接受到的消息 " + message);
        }
    };
}

```



```

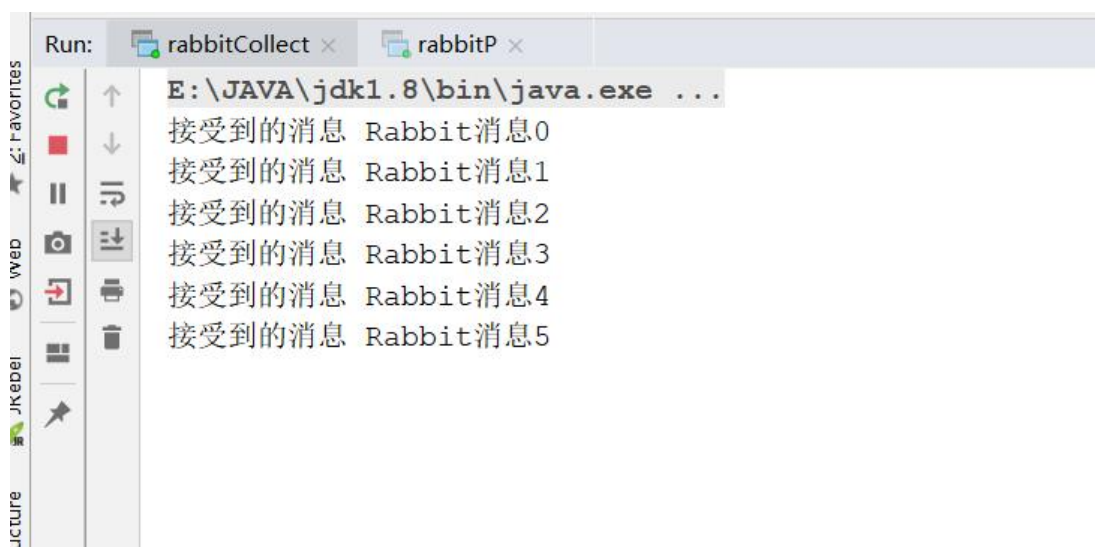
    }
};

//6. 为消费者绑定队列
channel.basicConsume("NiuMsg", true, consumer);
}

```

## 总结运行结果

- 队列声明queueDeclare的参数：第一个参数表示队列名称、第二个参数为是否持久化（true表示，队列将在服务器重启时生存）、第三个参数为是否是独占队列（创建者可以使用的私有队列，断开自动删除）、第四个参数为当所有消费者客户端连接断开时是否自动删除队列、第五个参数为队列的其他参数。
- 先运行消费者实例 然后再接着运行生产者实例
- 消费者就会接收到生产者发送来的消息



- 进入Queues NiuMsg就会进入running 状态 表示正在发送消息

Overview
Connections
Channels
Exchanges
Queues
Admin

## Queues

▼ All queues (2)

Pagination

Page 1 ▼ of 1 - Filter:  ☐ Regex ?

Overview				Messages			Message rates			+/-
Virtual host	Name	Features	State	Ready	Unacked	Total	incoming	deliver / get	ack	
/	ITEM_QUEUE	D	idle	0	0	0	0.00/s	0.00/s	0.00/s	
/	NiuMsg	D	running	0	0	0	1.2/s	1.2/s	0.00/s	

► Add a new queue

HTTP API
Server Docs
Tutorials
Community Support
Community Slack
Commercial Support
Plugins
GitHub
C

- 当生产者发送完消息 Queues 中的NiuMsg 的状态就会变为空闲状态

Overview   Connections   Channels   Exchanges   **Queues**   Admin

## Queues

▼ All queues (2)

Pagination

Page 1 ▼ of 1 - Filter:  ☐ Regex ?

Overview				Messages			Message rates				+/-
Virtual host	Name	Features	State	Ready	Unacked	Total	incoming	deliver / get	ack		
/	ITEM_QUEUE	D	idle	0	0	0	0.00/s	0.00/s	0.00/s		
/	NiuMsg	D	idle	0	0	0	0.00/s	0.00/s	0.00/s		

► Add a new queue

- 而在Connection 中只要消费者一直处于接受消息状态 连接就一直会存在

RabbitMQ 3.7.17 Erlang 22.0

Overview   **Connections**   Channels   Exchanges   Queues   Admin

## Connections

▼ All connections (1)

Pagination

Page 1 ▼ of 1 - Filter:  ☐ Regex ?

Overview			Details			Network		+/-
Virtual host	Name	User name	State	SSL / TLS	Protocol	Channels	From client	To client
/	:::1:16240	guest	running	o	AMQP 0-9-1	1	0iB/s	0iB/s

HTTP API   Server Docs   Tutorials   Community Support   Community Slack   Commercial Support   Plugins   GitHub   Changelog