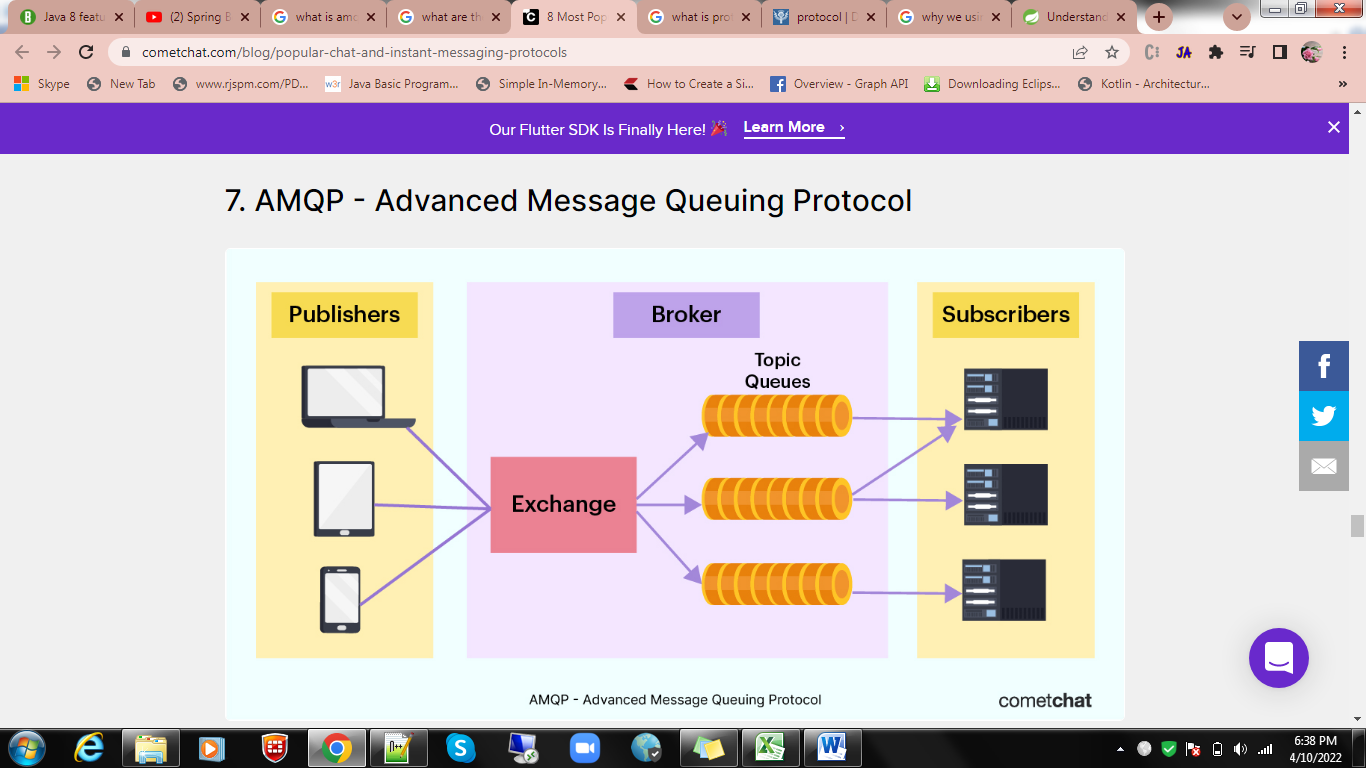
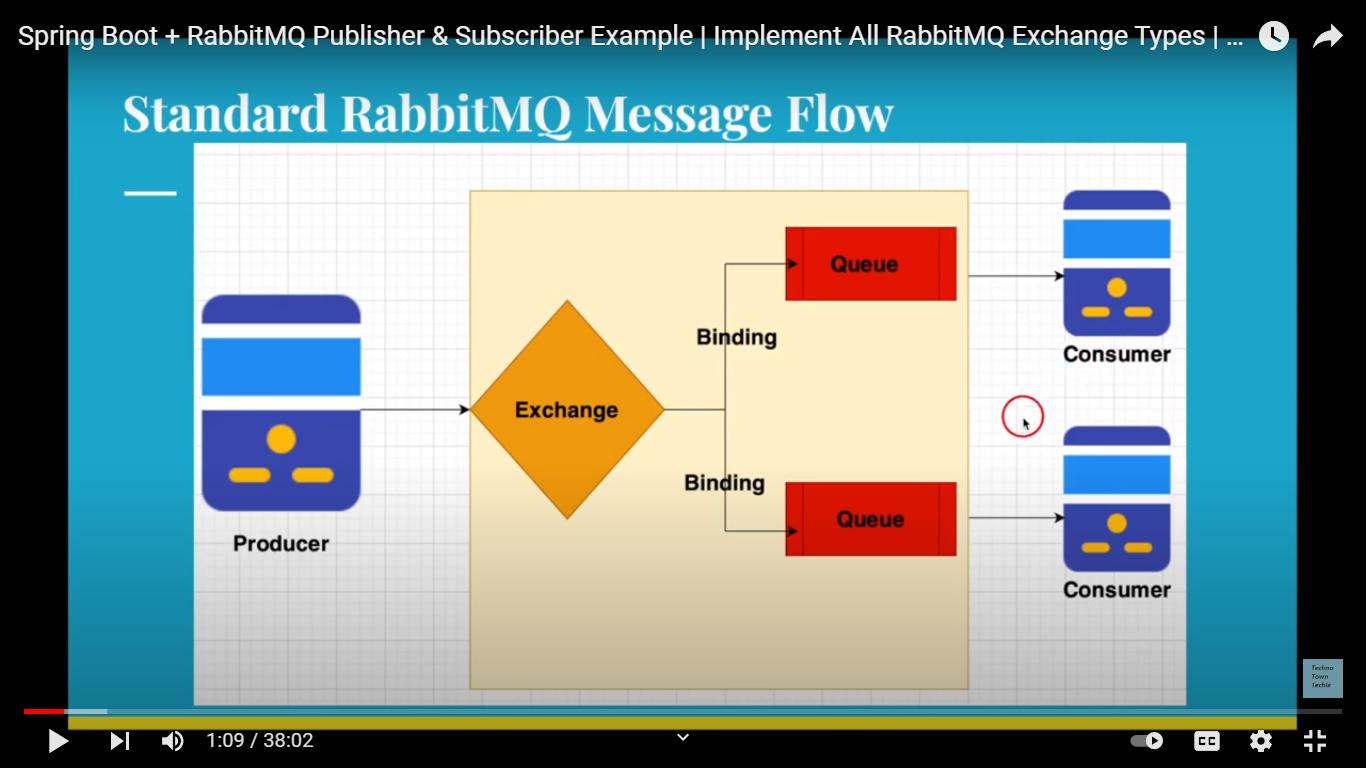
**RABBIT MQ:**

* Message broker that accepts and forward the messages.
* Uses AMQP(Advanced Message Queuing Protocol)
* AMQP, [Advanced Message Queuing Protocol](https://www.amqp.org/about/what), is an open-source, standardized protocol for asynchronous messaging. AMQP shines in multi-client environments due to its use of a streamed binary messaging system which allows for the interoperability of clients from different vendors.



**RABBITMQ MESSAGE FLOW:**



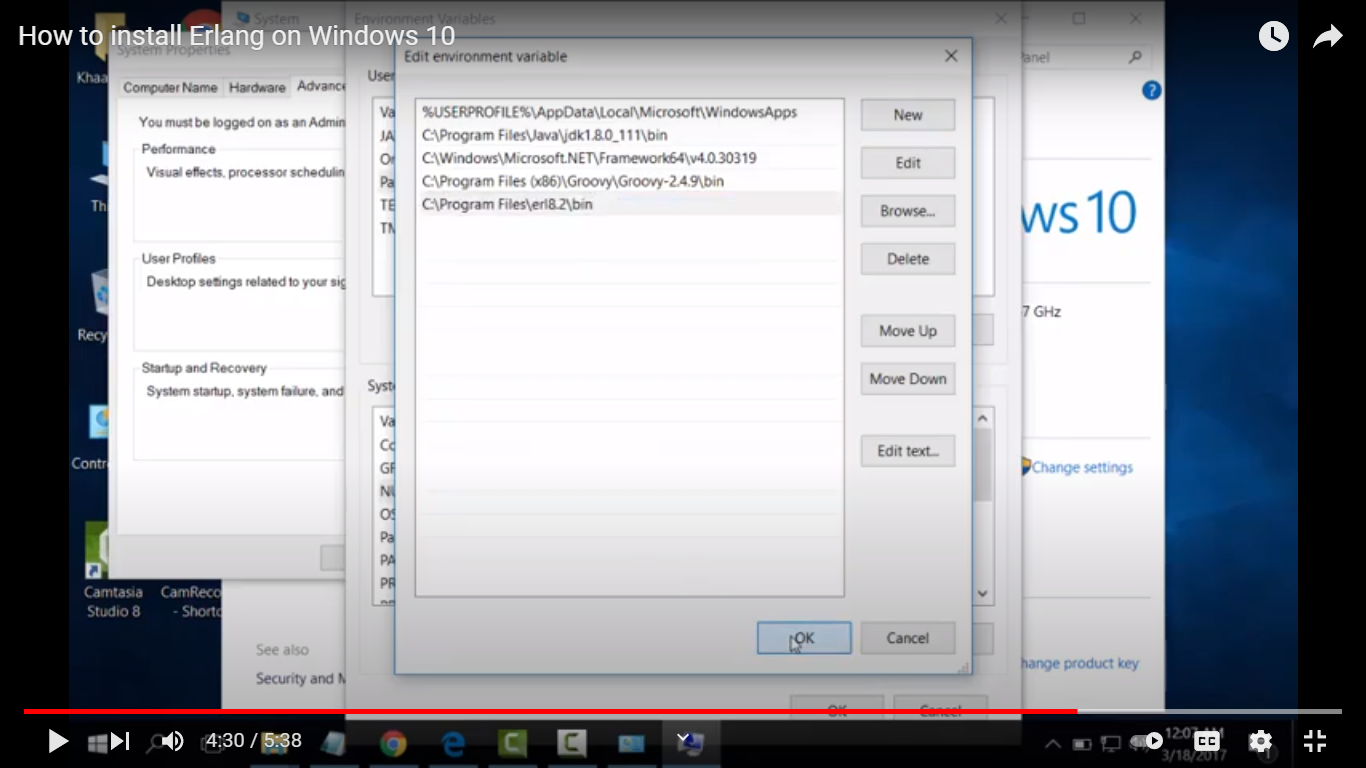
**INSTALLATION STEPS:**

Step 1: Install erlang using the below link:

<https://www.erlang.org/downloads>

Step 2: Download windows installer

Step 3: Install the software

Step 4: Set environment path: 

Step 5: To start the erlang:

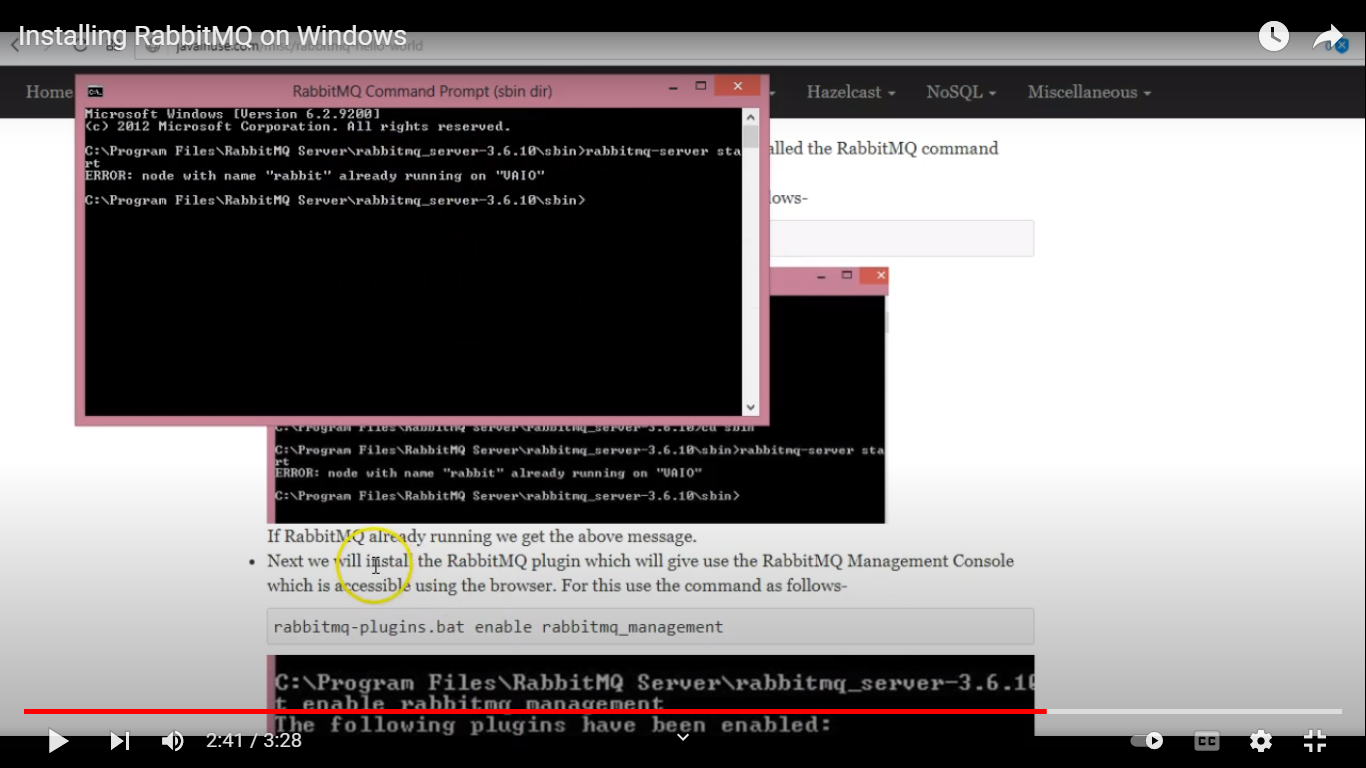
Cmd prompt🡪 type **erl 🡪** Installed erlang version will be displayed.

Ref: <https://www.youtube.com/watch?v=uhsIu-zP7Hs>

Step 6: Download rabbitmq (Link : <https://www.rabbitmq.com/download.html>)

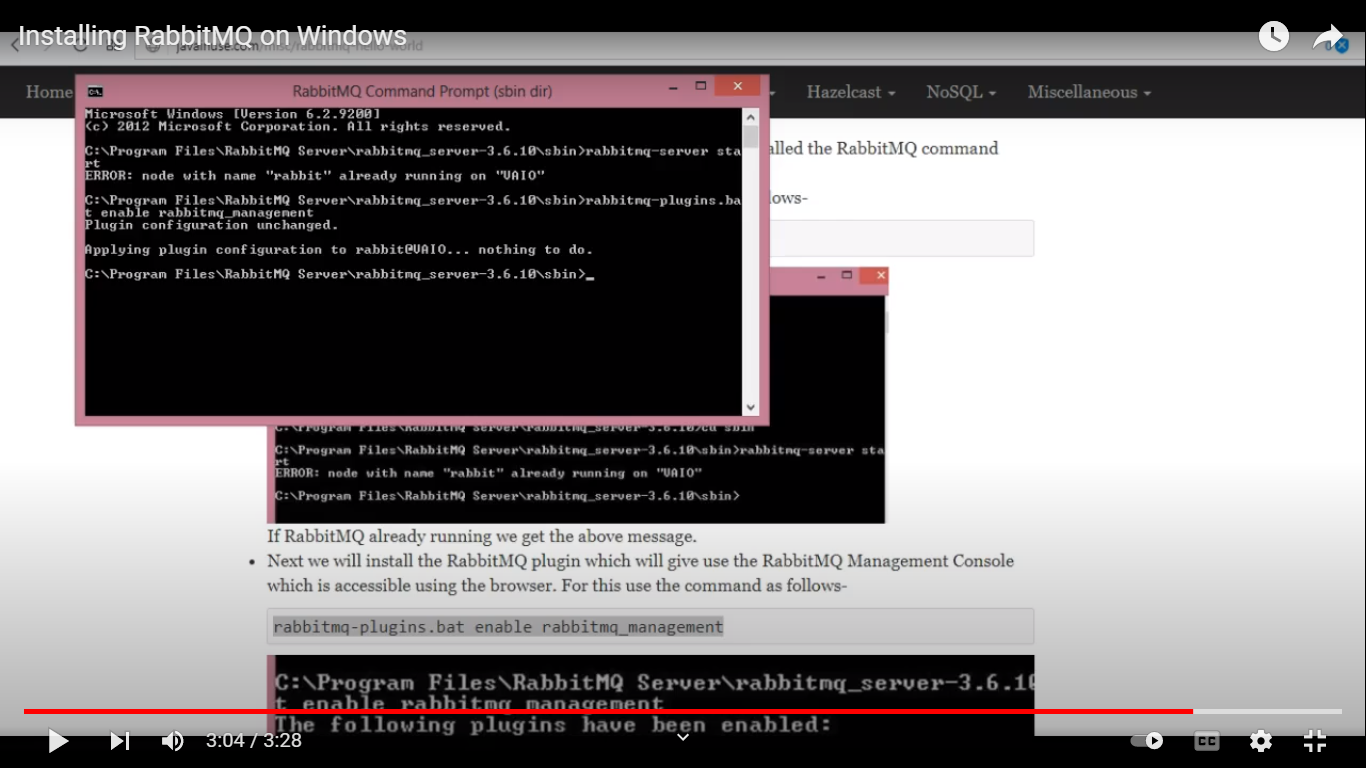
Go to the link🡪 Click windows link(Windows: [Chocolatey or Installer](https://www.rabbitmq.com/install-windows.html) ) 🡪Download the .exe file([rabbitmq-server-3.9.14.exe](https://github.com/rabbitmq/rabbitmq-server/releases/download/v3.9.14/rabbitmq-server-3.9.14.exe))

Step 7: Open rabbitmq cmd prompt(search for rabbitmq) and start it.

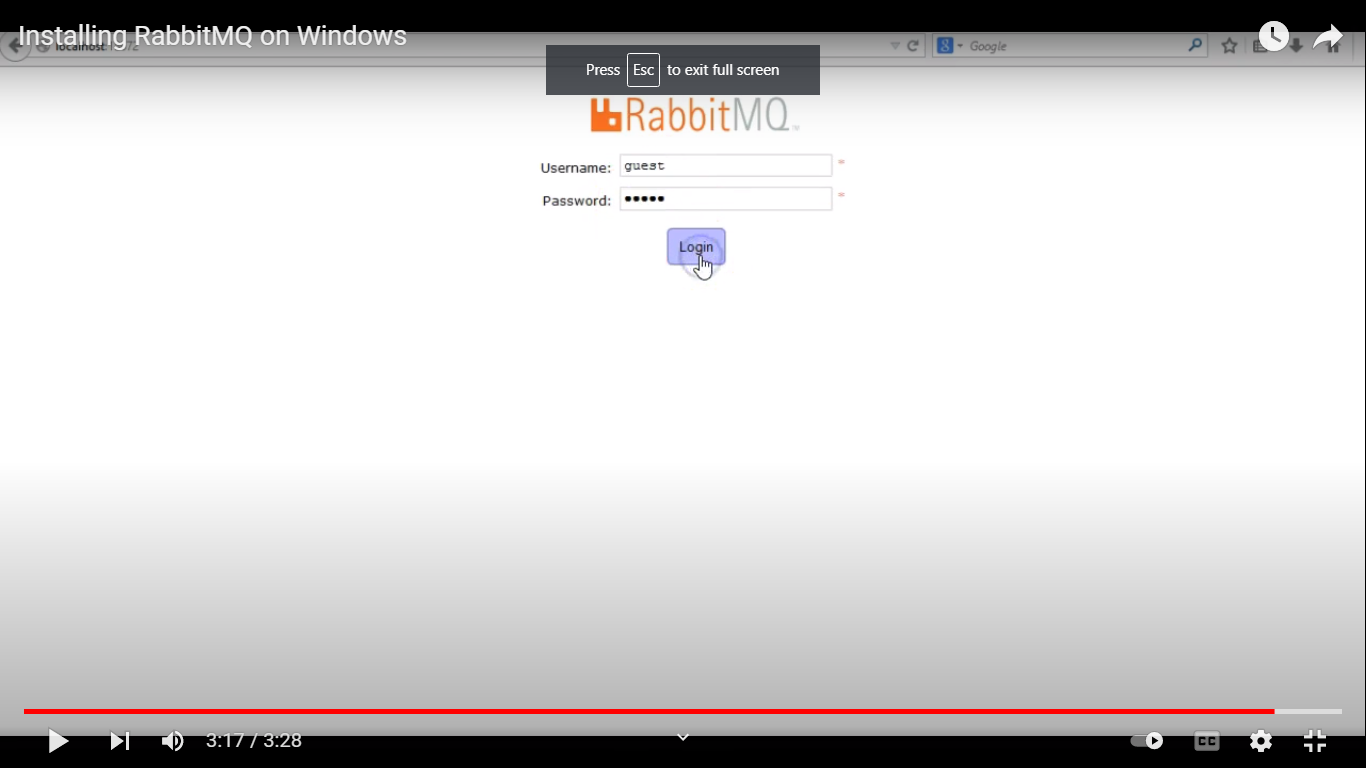


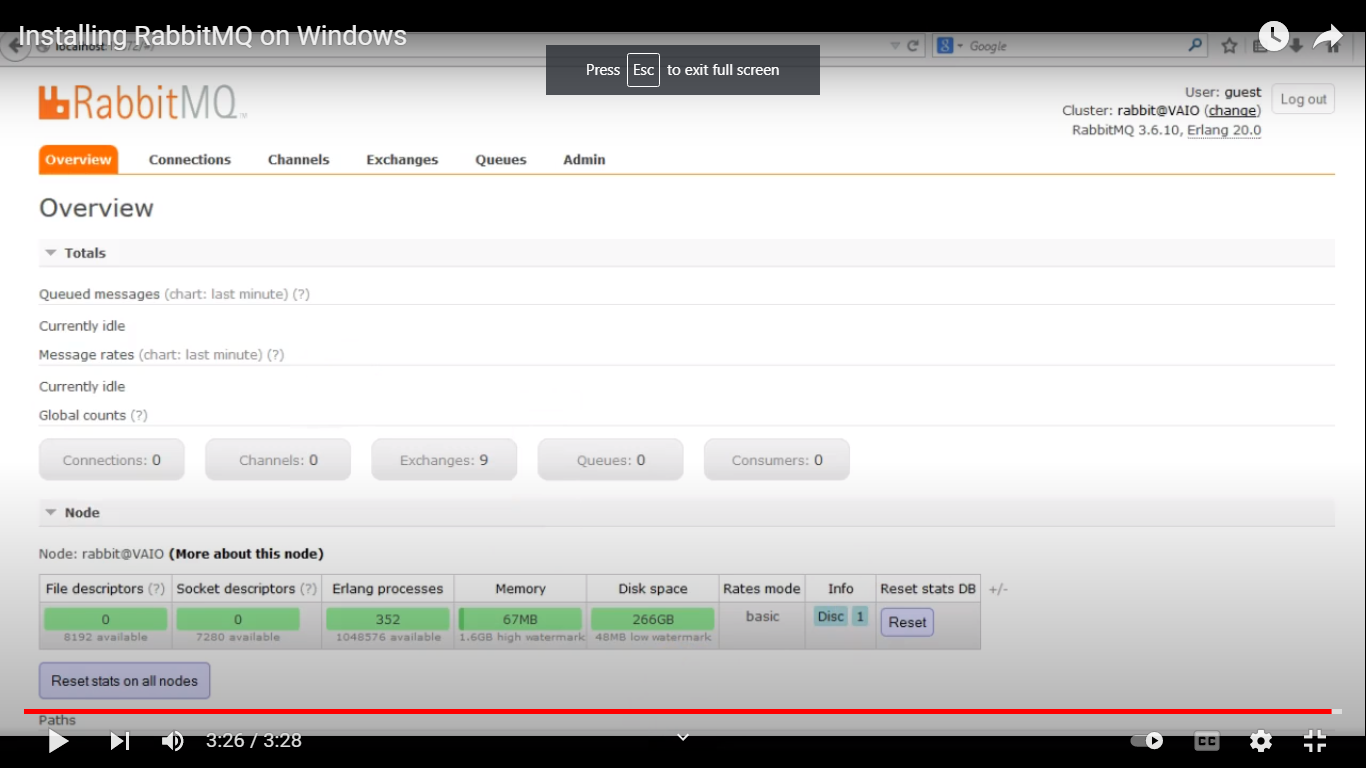
Step 8: We need to install rabbitMQ plugin for console. It will be done using the below cmd.

rabbitmq-plugins.bat enable rabbitmq\_management



Step 9: check the console by localhost:15672 🡪 password “guest”





(OR) Javatpoint ref: <https://www.javatpoint.com/installing-rabbitmq-server>

**SAMPLE SOURCE CODE:**

<https://github.com/ekim197711/springboot-rabbitmq>

**DIFFERENT TYPES OF EXCEHANGE:**



1. **Direct Exchange:**



* It check for the routing key with the queue binding key, if both are equal the message send to the particular queue alone.

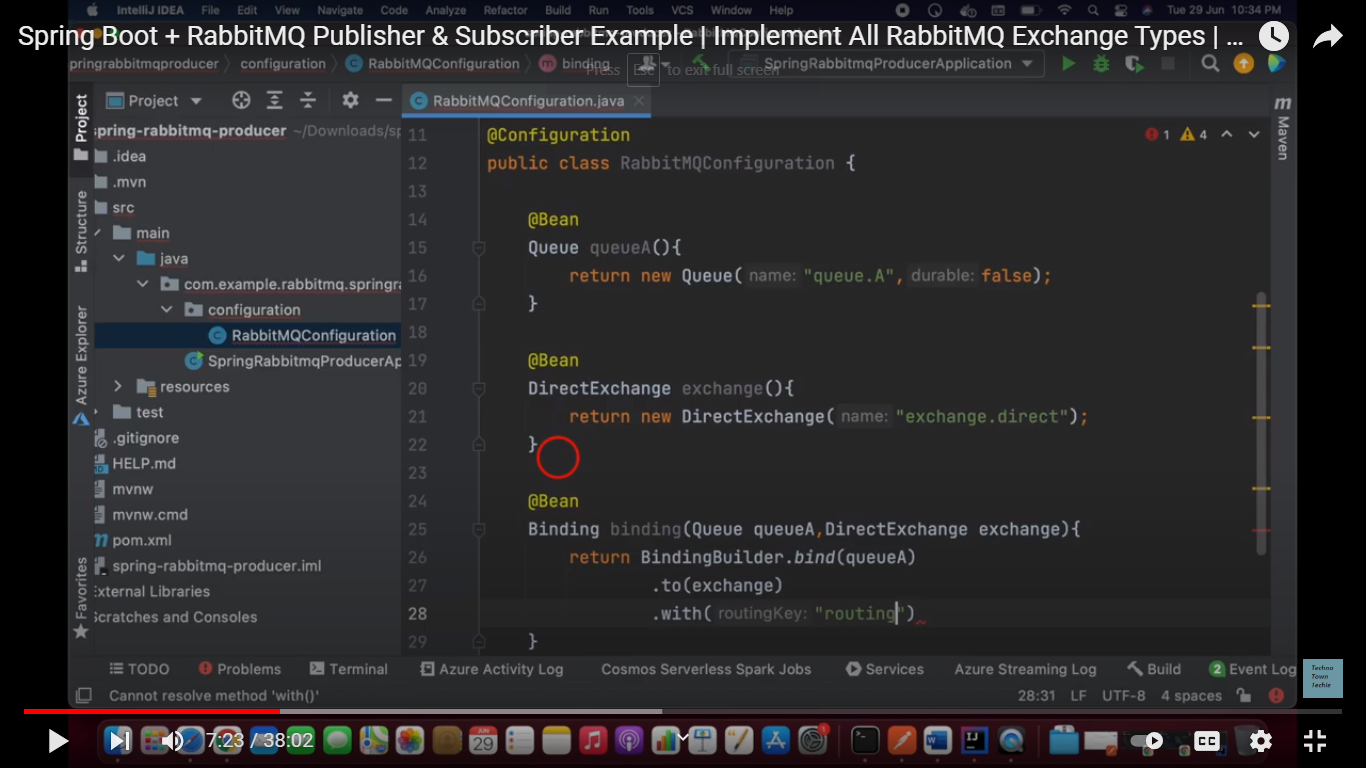
**Producer:**

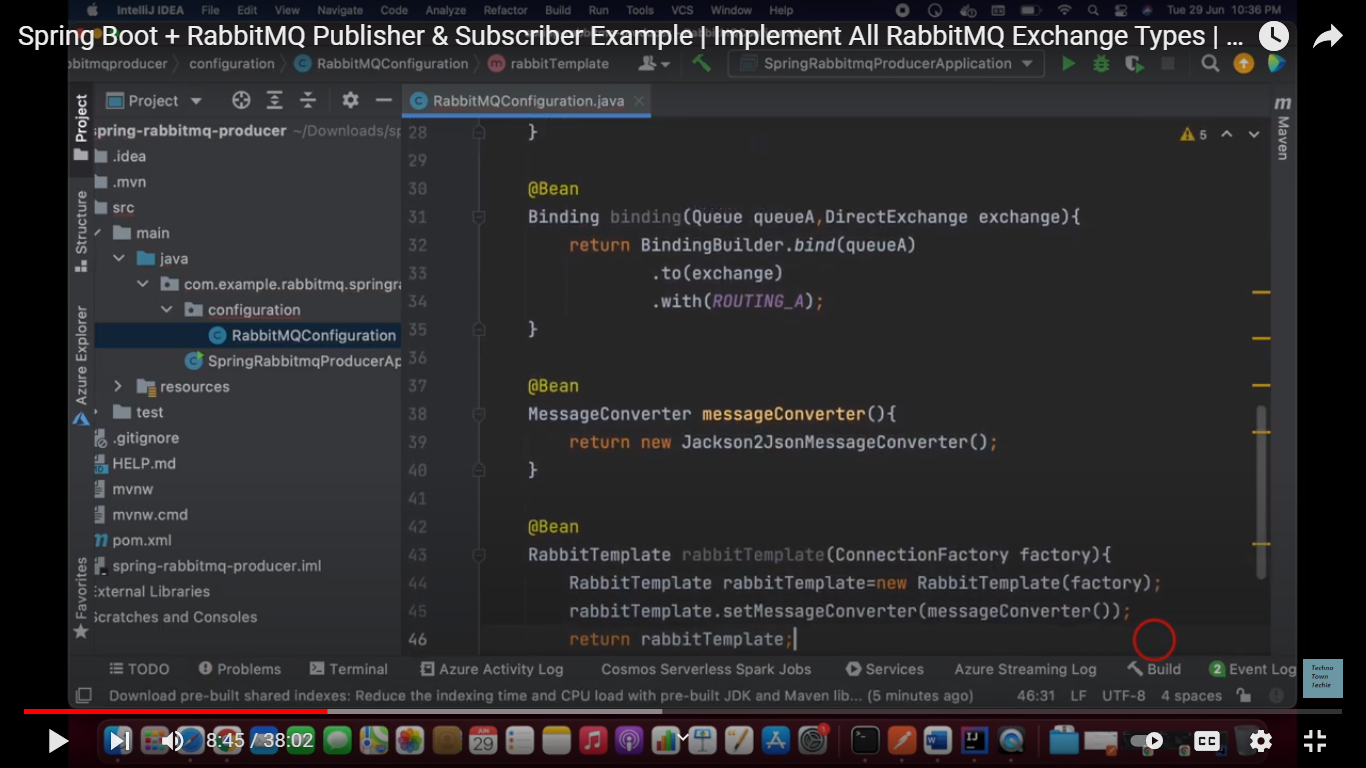
Step1: Create producer, Dependencies: web, rabbitMQ and import the project

Step2: Create configuration file

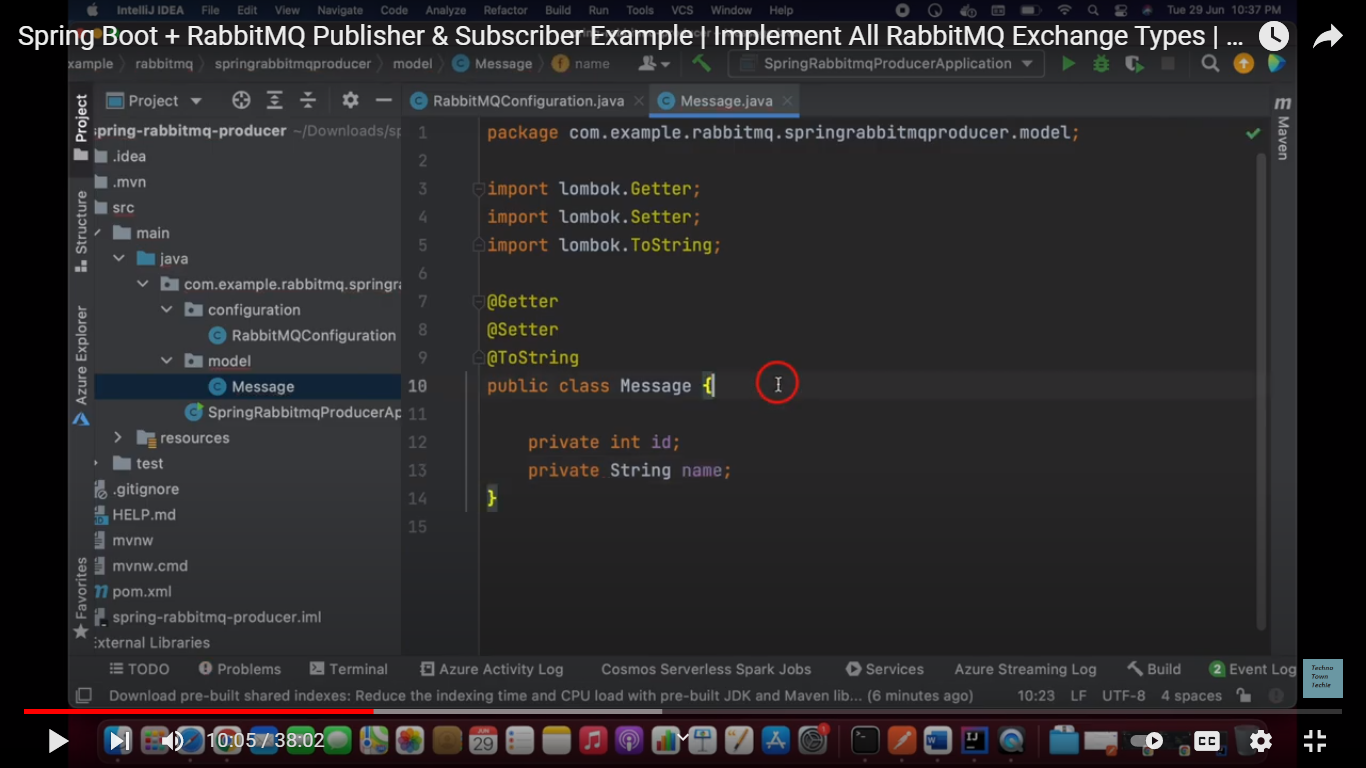
Step3: Annotate the class with @Configuration

Step4: Create queue, exchange, binding, MessageConverter and rabbitTemplate beans

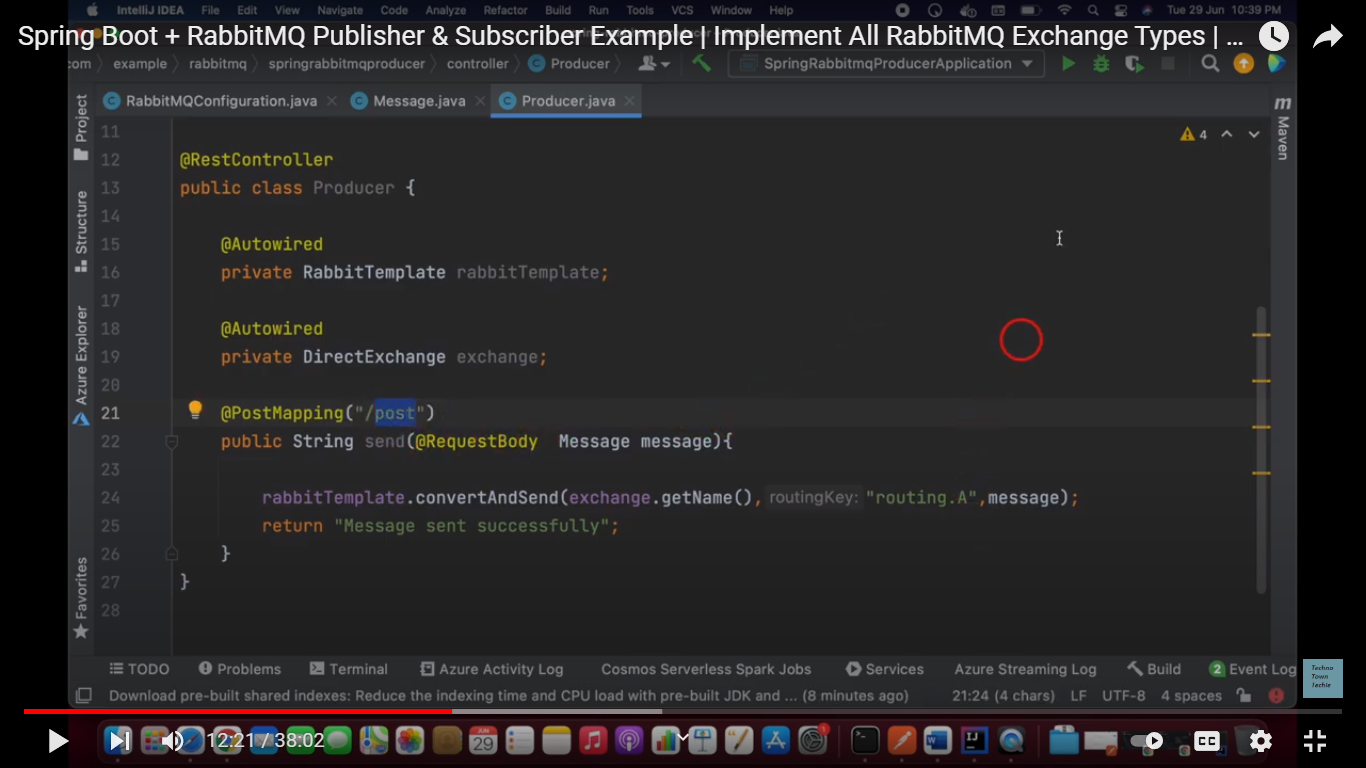




Step5: Create one model class for message



Step6: Create rest controller class with, rabbitTemplate, exchange and send API method



Step7: The hit the API

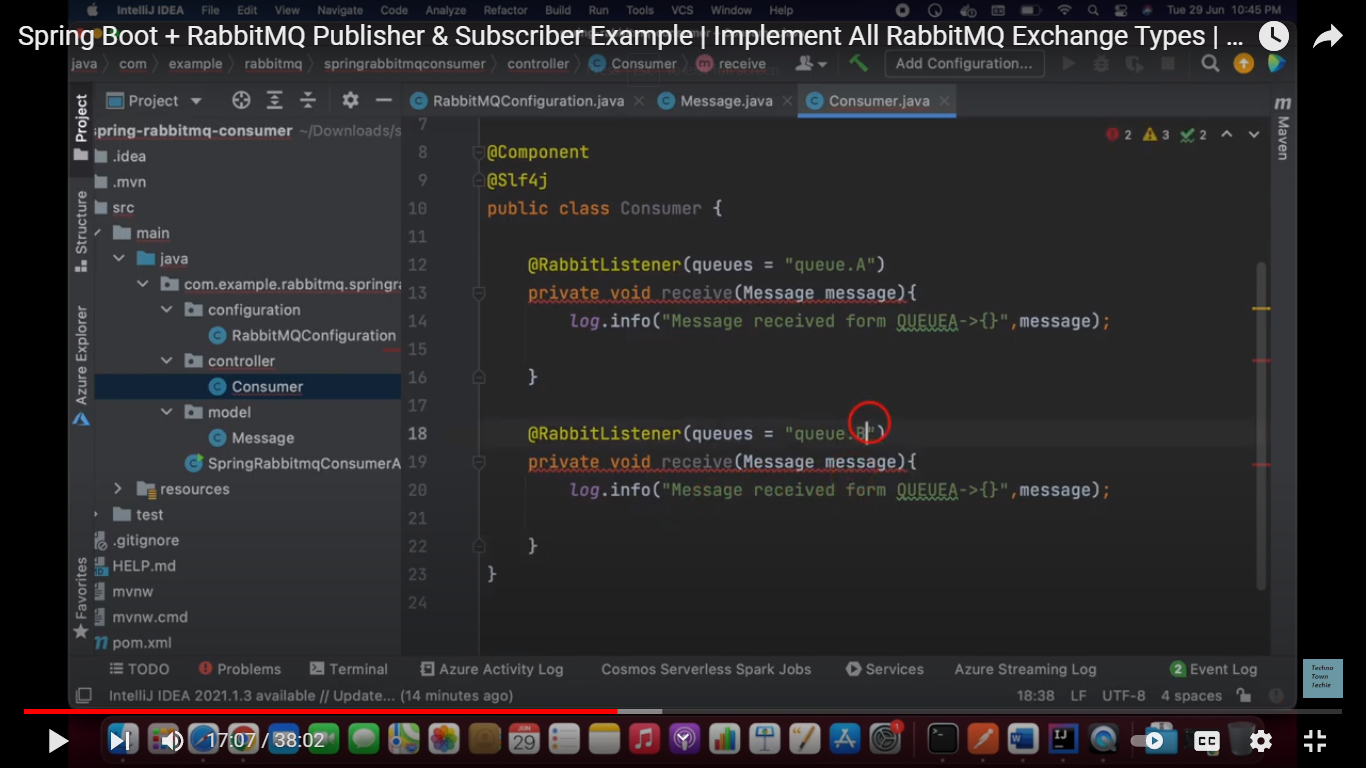
Step8: Open rabbitMQ server and check the exchange and queue, we can see the pending message there.

**Consumer:**

Step1: Create consumer 🡪 follow the same steps (Step1 to Step 5 or copy paste the configuration and model classes.

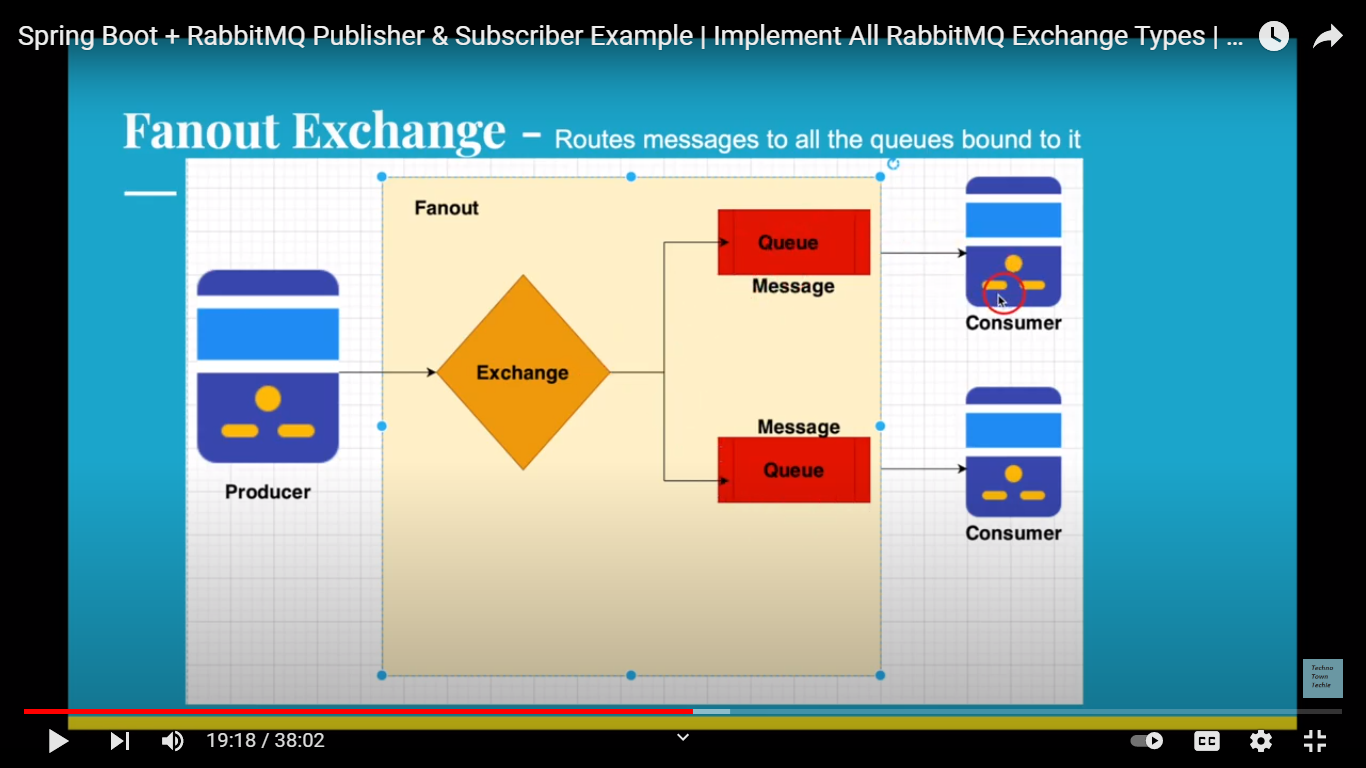
Step2: Create consumer class annotate with @Component

Step3: Create receive method



Step4: Up the consumer application, we will get the message from producer.

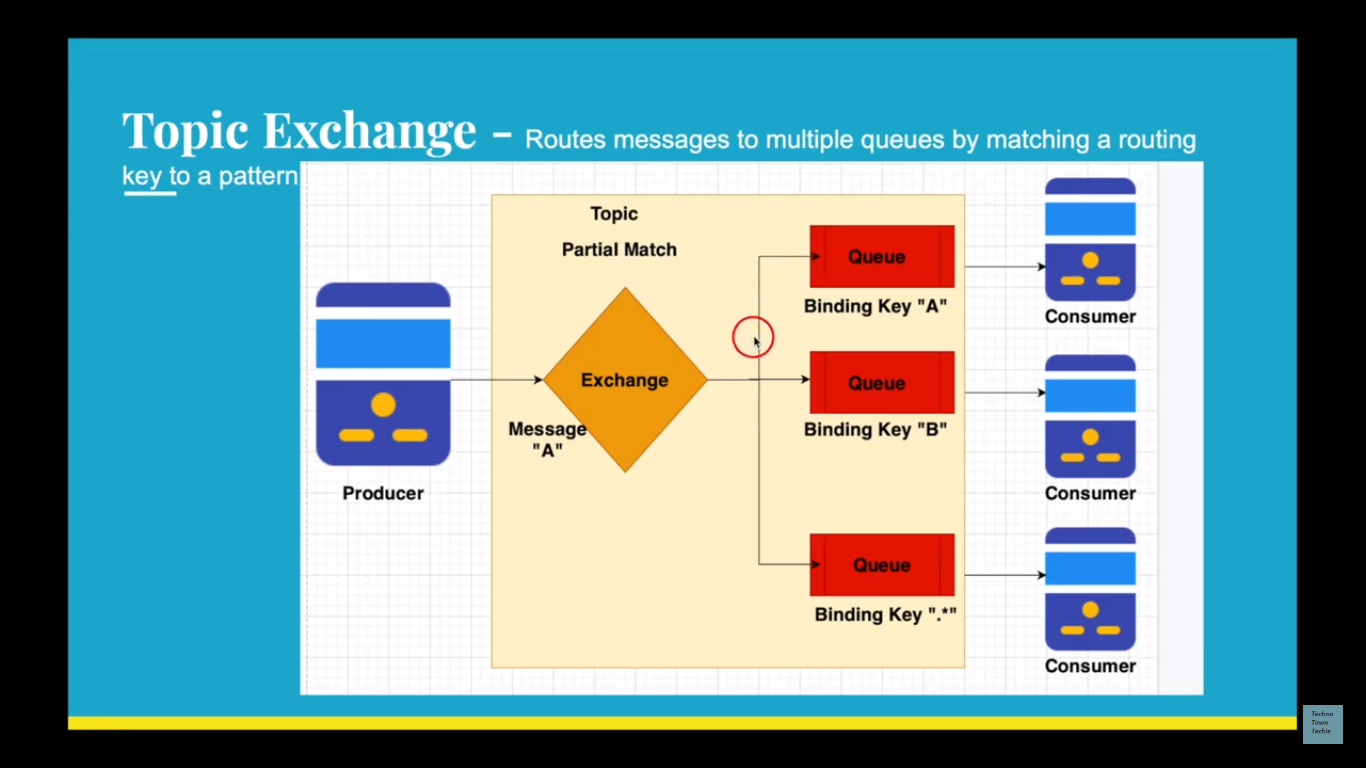
1. **Fanout Exchange:**



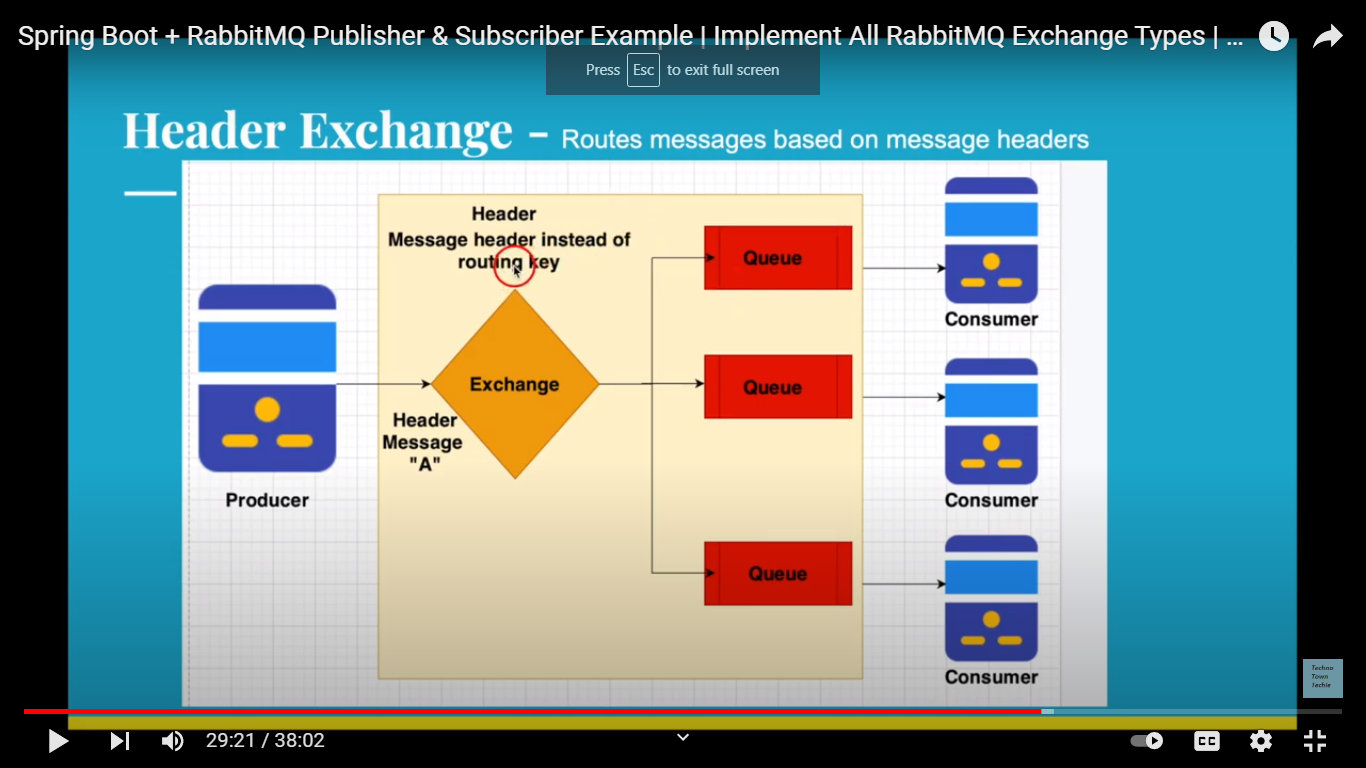
* Here message will be send to all the queue which is bind with the exchange.

Step: Change the exchange type to Fanout exchange and remove the routing keys

1. **TOPIC EXCHANGE:**



1. **HEADER EXCHANGE:**



**5.DEFAULT EXCHANGE:** 