Download kafka from below link kafka_2.13-3.5.2

Kafka Download Link - https://kafka.apache.org/downloads

Extract the tgz file and its inner folder keep the extarcated folder in D: drive or any drive

Commands to start Zookeeper and Kafka in windows :

Open the folder having bin in terminal and execute below commands :-

Command 1 -- .\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties

Command2 - .\bin\windows\kafka-server-start.bat .\config\server.properties

Commands to start Zookeeper and Kafka in Linux:-

```
# For Linux users(ubuntu)

Command1 - bin/zookeeper-server-start.sh config/zookeeper.properties

Command2 -    bin/kafka-server-start.sh config/server.properties
```

To check any topics got created in kafka command for that is :-

For windows -

.\bin\windows\kafka-topics.bat --list --bootstrap-server localhost:9092

If Kafka is running, you should see a list of topics.

Now for demo create a springboot project with web and kafka dependency.

```
<dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-web</artifactId>
                </dependency>
                <dependency>
                        <groupId>org.springframework.kafka</groupId>
                        <artifactId>spring-kafka</artifactId>
                </dependency>
```

Keep below code in controller

```
@RestController
@RequestMapping("/rest/api")
public class FetchMessageFromClient {

       @Autowired
    KafkaProducer kafkaProducer;


    @GetMapping(value = "/producer")
```

```
    public String sendMessage(@RequestParam("message") String message)
    {
        kafkaProducer.sendMessageToTopic(message);
        return "Message sent Successfully to the your code decode topic ";
    }

}
```

And then in service layer create two classes one “KafkaProducer” and the other One “KafkaListners”

```
@Service
public class KafkaProducer {

      @Autowired
      private KafkaTemplate<String, String> kafkaTemplate;
       // first type is topic data type(topic name String) 2nd one type of data to be
passed
      public void sendMessageToTopic(String message) {
            kafkaTemplate.send("CodeDecodeTopic", message);
      }

}


@Service
public class KafkaListner {

      @KafkaListener(topics = "CodeDecodeTopic", groupId = "codedecode-group")
      public void listenToCodeDecodeKafkaTopic(String messageReceived) {
            System.out.println("Message received is " + messageReceived);
      }
}
```

Application.yml should have below configurations :-

```
spring:
  kafka:
    producer:
      bootstrap-servers: localhost:9092
      key-serializer: org.apache.kafka.common.serialization.StringSerializer
      value-serializer: org.apache.kafka.common.serialization.StringSerializer
```

same configuration for separate consumer service as well

```
spring:
  kafka:
    consumer:
      bootstrap-servers: localhost:9092
      key-serializer: org.apache.kafka.common.serialization.StringSerializer
```

```
value-serializer: org.apache.kafka.common.serialization.StringSerializer
```

How to check kafaka running is ubuntu or not command

sudo systemctl status kafka