

Assignment 24 Kafka 2

For this task we need to start zookeeper and kafka broker

Starting Zookeeper

```
$KAFKA_HOME/bin/zookeeper-server-start.sh
```

\$KAFKA_HOME/config/zookeeper.properties

```
[2018-12-02 14:22:04,776] INFO Reading configuration from: /home/cadgild/install/kafka/kafka-2.12-0.10.1/config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2018-12-02 14:22:04,789] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DataDirCleanupManager)
[2018-12-02 14:22:04,789] INFO autopurge.interval set to 0 (org.apache.zookeeper.server.DataDirCleanupManager)
[2018-12-02 14:22:04,789] INFO Purge task is scheduled for execution (org.apache.zookeeper.server.DataDirCleanupManager)
[2018-12-02 14:22:04,789] WARN Either no config or no quorum defined in config, running in standalone mode (org.apache.zookeeper.server.quorum.QuorumPeerMain)
[2018-12-02 14:22:04,838] INFO Reading configuration from: /home/cadgild/install/kafka/kafka-2.12-0.10.1/config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2018-12-02 14:22:04,839] INFO Starting server (org.apache.zookeeper.server.ZooKeeperServerMain)
[2018-12-02 14:22:04,851] INFO Server environment:java.version=2.4.0.1, built on 02/06/2016 03:10 GMT (org.apache.zookeeper.server.ZooKeeperServer)
[2018-12-02 14:22:04,854] INFO Server environment:java.vendor=Oracle Corporation (org.apache.zookeeper.server.ZooKeeperServer)
[2018-12-02 14:22:04,854] INFO Server environment:java.version=1.8_151 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-12-02 14:22:04,854] INFO Server environment:java.vendor=Oracle Corporation (org.apache.zookeeper.server.ZooKeeperServer)
[2018-12-02 14:22:04,854] INFO Server environment:java.home=/usr/java/jdk1.8_151/jre (org.apache.zookeeper.server.ZooKeeperServer)
[2018-12-02 14:22:04,855] INFO Server environment:java.class.path=/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/apollonice-repackaged-2.4.0.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/avro-1.7.7.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/caffeine-2.6.1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-beanutils-1.9.0.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-codec-1.10.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-io-2.4.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-lang3-3.4.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-math3-3.6.1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-net-3.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/commons-pool2-2.6.2.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/concurrent-jmx-jmxri-1.3.1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/connect-jmx-jmxri-1.0.1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/connect-runtime-0.10.1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/guava-18.0.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/hk2-api-2.4.0-b34.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/hk2-locator-2.4-0-b34.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/hk2-utils-2.4-0-b34.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jackson-annotations-2.6.0.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jackson-core-2.6.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jackson-databind-2.6.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jackson-jaxrs-base-2.6.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jackson-jaxrs-provider-2.6.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jackson-module-jaxb-annotations-2.6.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jaxb-annotations-2.6.3.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/javax-annotation-api-1.2.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/javax-inject-1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/javax-servlet-api-2.0.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/javax.ws.rs-api-2.0.1.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jsr305-3.0.2.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jersey-container-servlet-2.22.2.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jersey-container-servlet-core-2.22.2.jar:/home/cadgild/install/kafka/kafka-2.12-0.10.1/bin:/lib/jersey-media-jaxb-2.22.2.jar
```

Starting broker

```
$KAFKA_HOME/bin/kafka-server-start.sh $KAFKA_HOME/config/server.properties
```

```
[acadgild@localhost ~]$ $KAFKA_HOME/bin/kafka-server-start.sh $KAFKA_HOME/config/server.properties
[2018-12-02 14:18:02,011] INFO KafkaConfig values:
  advertised.host.name = null
  advertised.listeners = null
  advertised.port = null
  authorizer.class.name =
  auto.create.topics.enable = true
  auto.leader.rebalance.enable = true
  background.threads = 10
  broker.id = 0
  broker.id.generation.enable = true
  broker.rack = null
  compression.type = producer
  connections.max.idle.ms = 600000
  controlled.shutdown.enable = true
  controlled.shutdown.max.retries = 3
  controlled.shutdown.retry.backoff.ms = 5000
  controller.socket.timeout.ms = 30000
  default.replication.factor = 1
  delete.topic.enable = false
  fetch.purgatory.purge.interval.requests = 1000
  group.max.session.timeout.ms = 300000
  group.min.session.timeout.ms = 6000
  host.name =
  inter.broker.protocol.version = 0.10.1-IV2
  leader.imbalance.check.interval.seconds = 300
  leader.imbalance.per.broker.percentage = 10
  listeners = null
  log.cleaner.backoff.ms = 15000
```

```
[acadgild@localhost ~]$ jps
8392 QuorumPeerMain
6696 Kafka
8922 Jps
You have new mail in /var/spool/
[acadgild@localhost ~]$
```

Task 1:

Create a java program MyKafkaProducer.java that takes a file name and delimiter as input arguments. It should read the content of file line by line. Fields in the file are in following order

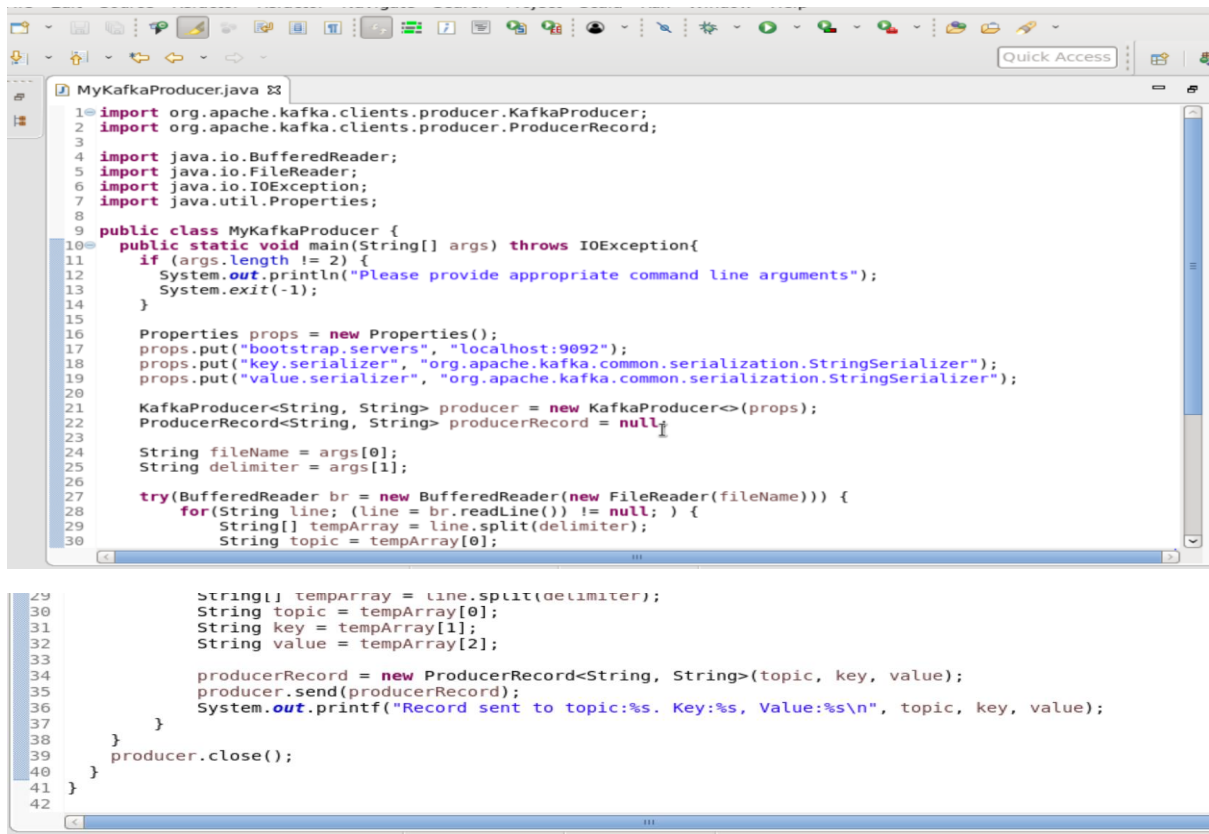
1. Kafka Topic Name
2. Key
3. value

For every line, insert the key and value to the respective Kafka broker in a fire and forget mode. After record is sent, it should print appropriate message on screen. Pass dataset_producer.txt as the input file and - as delimiter

Below is the data file

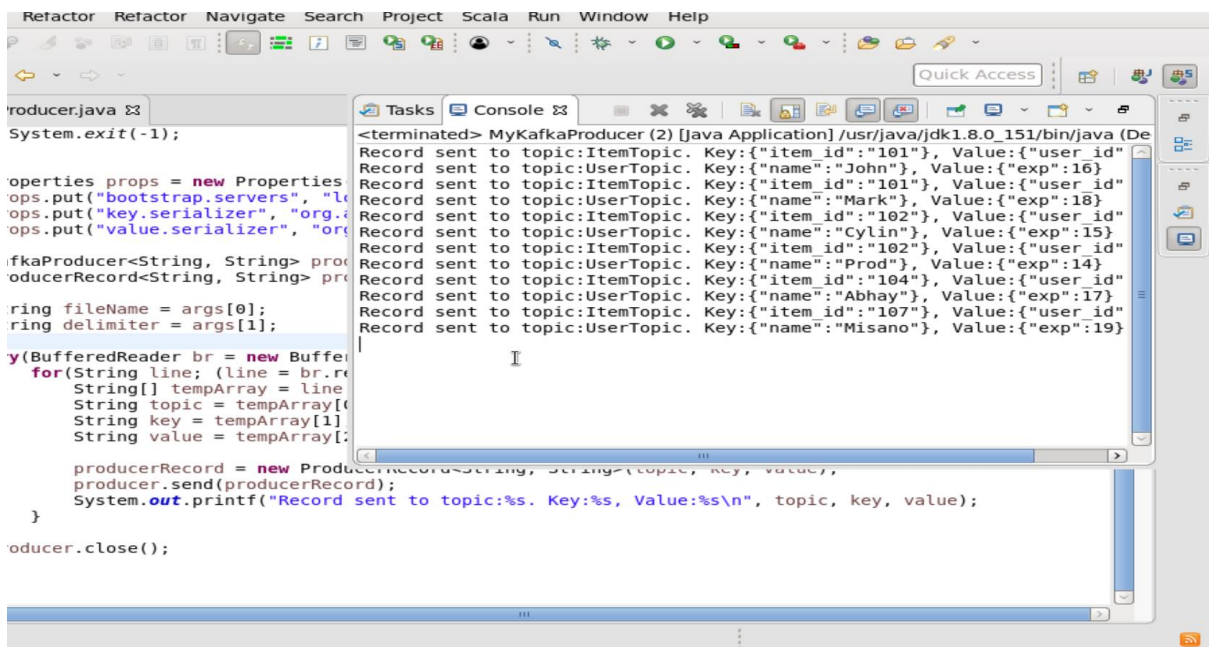
```
[acadgild@localhost ~]$ ls dat*
dataset_producer.txt
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ cat dataset_producer.txt
ItemTopic-{"item_id":"101"}-{"user_id":"U101"}
UserTopic-{"name":"John"}-{"exp":16}
ItemTopic-{"item_id":"101"}-{"user_id":"U106"}
UserTopic-{"name":"Mark"}-{"exp":18}
ItemTopic-{"item_id":"102"}-{"user_id":"U110"}
UserTopic-{"name":"Cylin"}-{"exp":15}
ItemTopic-{"item_id":"102"}-{"user_id":"U101"}
UserTopic-{"name":"Prod"}-{"exp":14}
ItemTopic-{"item_id":"104"}-{"user_id":"U102"}
UserTopic-{"name":"Abhay"}-{"exp":17}
ItemTopic-{"item_id":"107"}-{"user_id":"U104"}
UserTopic-{"name":"Misano"}-{"exp":19}
[acadgild@localhost ~]$
```

Java program MyKafkaProducer.java screenshot (original code attached)



```
1 import org.apache.kafka.clients.producer.KafkaProducer;
2 import org.apache.kafka.clients.producer.ProducerRecord;
3
4 import java.io.BufferedReader;
5 import java.io.FileReader;
6 import java.io.IOException;
7 import java.util.Properties;
8
9 public class MyKafkaProducer {
10     public static void main(String[] args) throws IOException{
11         if (args.length != 2) {
12             System.out.println("Please provide appropriate command line arguments");
13             System.exit(-1);
14         }
15
16         Properties props = new Properties();
17         props.put("bootstrap.servers", "localhost:9092");
18         props.put("key.serializer", "org.apache.kafka.common.serialization.StringSerializer");
19         props.put("value.serializer", "org.apache.kafka.common.serialization.StringSerializer");
20
21         KafkaProducer<String, String> producer = new KafkaProducer<>(props);
22         ProducerRecord<String, String> producerRecord = null;
23
24         String fileName = args[0];
25         String delimiter = args[1];
26
27         try(BufferedReader br = new BufferedReader(new FileReader(fileName))) {
28             for(String line; (line = br.readLine()) != null; ) {
29                 String[] tempArray = line.split(delimiter);
30                 String topic = tempArray[0];
31                 String key = tempArray[1];
32                 String value = tempArray[2];
33
34                 producerRecord = new ProducerRecord<String, String>(topic, key, value);
35                 producer.send(producerRecord);
36                 System.out.printf("Record sent to topic:%s. Key:%s, Value:%s\n", topic, key, value);
37             }
38         }
39         producer.close();
40     }
41 }
42 }
```

Output



```
<terminated> MyKafkaProducer (2) [Java Application] /usr/java/jdk1.8.0_151/bin/java (De
Record sent to topic:ItemTopic. Key:{\"item_id\":\"101\"}, Value:{\"user_id\"
Record sent to topic:UserTopic. Key:{\"name\":\"John\"}, Value:{\"exp\":16}
Record sent to topic:ItemTopic. Key:{\"item_id\":\"101\"}, Value:{\"user_id\"
Record sent to topic:UserTopic. Key:{\"name\":\"Mark\"}, Value:{\"exp\":18}
Record sent to topic:ItemTopic. Key:{\"item_id\":\"102\"}, Value:{\"user_id\"
Record sent to topic:UserTopic. Key:{\"name\":\"Cylin\"}, Value:{\"exp\":15}
Record sent to topic:ItemTopic. Key:{\"item_id\":\"102\"}, Value:{\"user_id\"
Record sent to topic:UserTopic. Key:{\"name\":\"Prod\"}, Value:{\"exp\":14}
Record sent to topic:ItemTopic. Key:{\"item_id\":\"104\"}, Value:{\"user_id\"
Record sent to topic:UserTopic. Key:{\"name\":\"Abhay\"}, Value:{\"exp\":17}
Record sent to topic:ItemTopic. Key:{\"item_id\":\"107\"}, Value:{\"user_id\"
Record sent to topic:UserTopic. Key:{\"name\":\"Misano\"}, Value:{\"exp\":19}
```

List the topics you can find ItemTopic and UserTopic

`$KAFKA_HOME/bin/kafka-topics.sh --list --zookeeper localhost:2181`

```
[acadgild@localhost ~]$ $KAFKA_HOME/bin/kafka-topics.sh --list --zookeeper localhost:2181
ItemTopic
KeyLessTopic
KeyedTopic
TestTopic
UserTopic
__consumer_offsets
my-replicated-topic
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$
```

Port MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Running the jar file

`java -cp $KAFKA_HOME/share/java/kafka/*:/home/acadgild/Desktop/my.jar
producer.MyKafkaProducer /home/acadgild/dataset_producer.txt -`

```
[acadgild@localhost Desktop]$ java -cp $KAFKA_HOME/share/java/kafka/*:/home/acadgild/Desktop/my.jar MyKafkaProducer /home/acadgild/dataset_producer.txt -
Record sent to topic:ItemTopic. Key:{"item_id":"101"}, Value:{"user_id":"U101"}
Record sent to topic:UserTopic. Key:{"name":"John"}, Value:{"exp":16}
Record sent to topic:ItemTopic. Key:{"item_id":"101"}, Value:{"user_id":"U106"}
Record sent to topic:UserTopic. Key:{"name":"Mark"}, Value:{"exp":18}
Record sent to topic:ItemTopic. Key:{"item_id":"102"}, Value:{"user_id":"U110"}
Record sent to topic:UserTopic. Key:{"name":"Cylin"}, Value:{"exp":15}
Record sent to topic:ItemTopic. Key:{"item_id":"102"}, Value:{"user_id":"U101"}
Record sent to topic:UserTopic. Key:{"name":"Prod"}, Value:{"exp":14}
Record sent to topic:ItemTopic. Key:{"item_id":"104"}, Value:{"user_id":"U102"}
Record sent to topic:UserTopic. Key:{"name":"Abhay"}, Value:{"exp":17}
Record sent to topic:ItemTopic. Key:{"item_id":"107"}, Value:{"user_id":"U104"}
Record sent to topic:UserTopic. Key:{"name":"Misano"}, Value:{"exp":19}
[acadgild@localhost Desktop]$
```

Item topic consumer

`$KAFKA_HOME/bin/kafka-console-consumer.sh --topic ItemTopic --from-beginning --
zookeeper localhost:2181 --property print.key=true`

```
^[[AYou have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ $KAFKA_HOME/bin/kafka-console-consumer.sh --topic ItemTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
Using the ConsoleConsumer with old consumer is deprecated and will be removed in a future major release. Consider using the new consumer by passing [bootstrap-
server] instead of [zookeeper].
{"item_id":"101"} {"user_id":"U101"}
{"item_id":"101"} {"user_id":"U106"}
{"item_id":"102"} {"user_id":"U110"}
{"item_id":"102"} {"user_id":"U101"}
{"item_id":"104"} {"user_id":"U102"}
{"item_id":"107"} {"user_id":"U104"}
{"item_id":"101"} {"user_id":"U101"}
{"item_id":"101"} {"user_id":"U106"}
{"item_id":"102"} {"user_id":"U110"}
{"item_id":"102"} {"user_id":"U101"}
{"item_id":"104"} {"user_id":"U102"}
{"item_id":"107"} {"user_id":"U104"}
{"item_id":"107"} {"user_id":"U104"}
```

User topic consumer

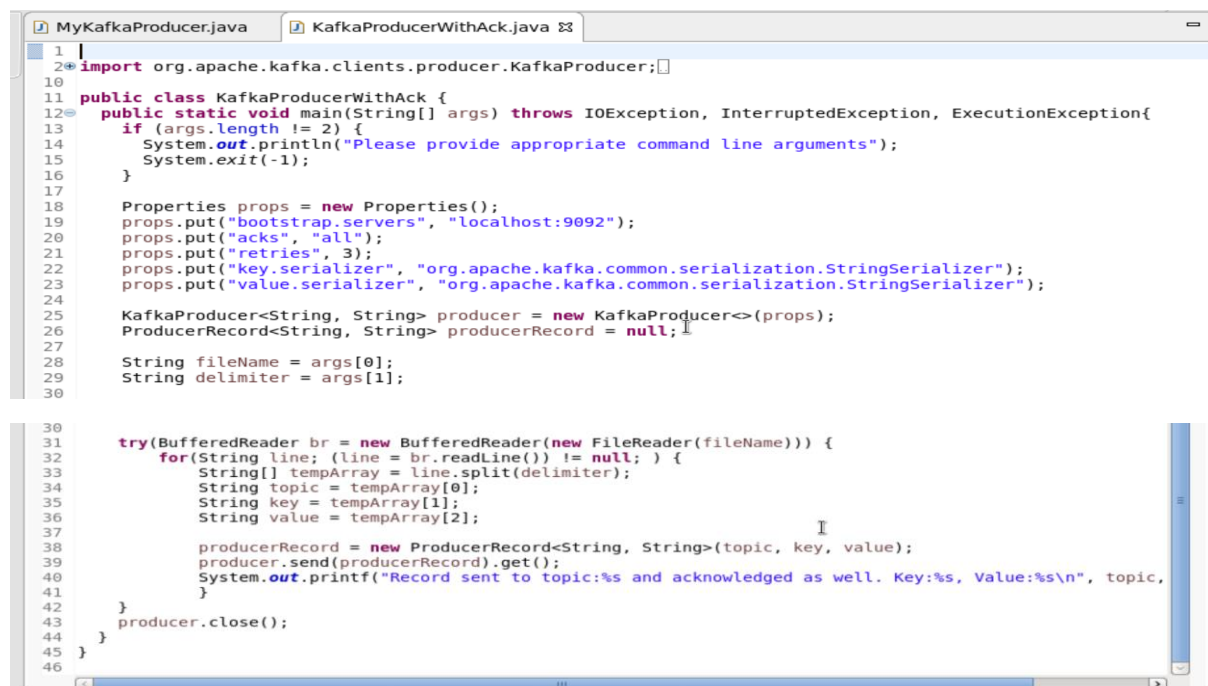
```
$KAFKA_HOME/bin/kafka-console-consumer.sh --topic UserTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
```

```
[acadgild@localhost ~]$ $KAFKA_HOME/bin/kafka-console-consumer.sh --topic UserTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
Using the ConsoleConsumer with old consumer is deprecated and will be removed in a future major release. Consider using the new consumer by passing [bootstrap.servers] instead of [zookeeper].
{"name":"John"} {"exp":16}
{"name":"Mark"} {"exp":18}
{"name":"Cylin"} {"exp":15}
{"name":"Prod"} {"exp":14}
{"name":"Abhay"} {"exp":17}
{"name":"Misano"} {"exp":19}
{"name":"John"} {"exp":16}
{"name":"Mark"} {"exp":18}
{"name":"Cylin"} {"exp":15}
{"name":"Prod"} {"exp":14}
{"name":"Abhay"} {"exp":17}
{"name":"Misano"} {"exp":19}
```

Task 2:

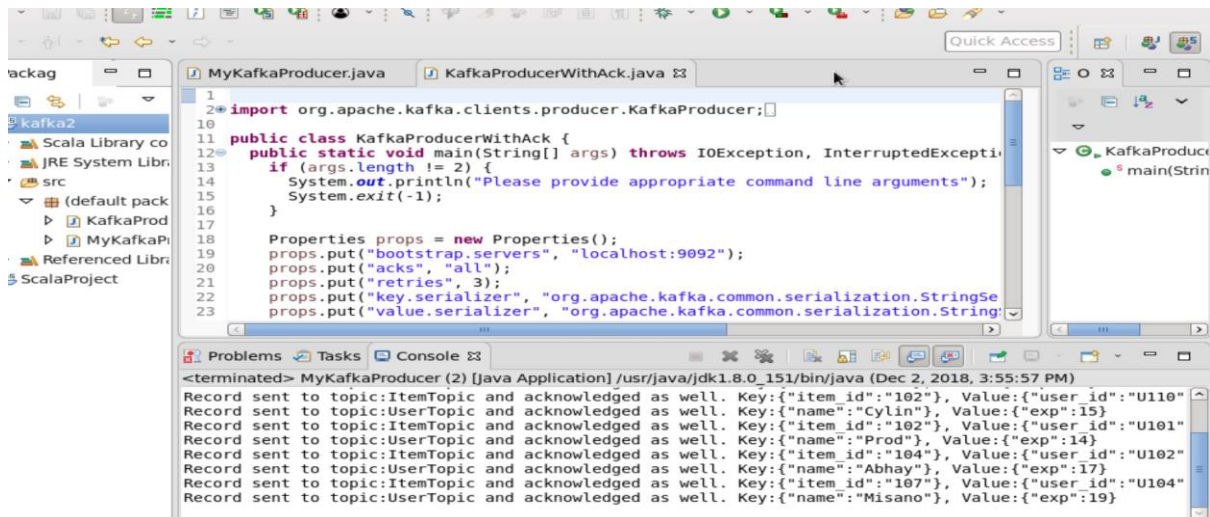
Modify the previous program MyKafkaProducer.java and create a new Java program KafkaProducerWithAck.java. This should perform the same task as of KafkaProducer.java with some modification. When passing any data to a topic, it should wait for acknowledgement. After acknowledgement is received from the broker, it should print the key and value, which has been written to a specified topic. The application should attempt for three retries before giving any exception. Pass dataset_producer.txt as the input file and - as delimiter.

MyKafkaProducer.java Code screen shot (original code attached)



```
1  import org.apache.kafka.clients.producer.KafkaProducer;
2  import org.apache.kafka.clients.producer.ProducerRecord;
3
4  public class MyKafkaProducer {
5      public static void main(String[] args) throws IOException, InterruptedException, ExecutionException {
6          if (args.length != 2) {
7              System.out.println("Please provide appropriate command line arguments");
8              System.exit(-1);
9          }
10
11          Properties props = new Properties();
12          props.put("bootstrap.servers", "localhost:9092");
13          props.put("acks", "all");
14          props.put("retries", 3);
15          props.put("key.serializer", "org.apache.kafka.common.serialization.StringSerializer");
16          props.put("value.serializer", "org.apache.kafka.common.serialization.StringSerializer");
17
18          KafkaProducer<String, String> producer = new KafkaProducer<>(props);
19          ProducerRecord<String, String> producerRecord = null;
20
21          String fileName = args[0];
22          String delimiter = args[1];
23
24          try(BufferedReader br = new BufferedReader(new FileReader(fileName))) {
25              for(String line; (line = br.readLine()) != null; ) {
26                  String[] tempArray = line.split(delimiter);
27                  String topic = tempArray[0];
28                  String key = tempArray[1];
29                  String value = tempArray[2];
30
31                  producerRecord = new ProducerRecord<String, String>(topic, key, value);
32                  producer.send(producerRecord).get();
33                  System.out.printf("Record sent to topic:%s and acknowledged as well. Key:%s, Value:%s\n", topic,
34                                  key, value);
35              }
36          }
37          producer.close();
38      }
39  }
```


Out put



The screenshot shows an IDE with the file `KafkaProducerWithAck.java` open. The code defines a `KafkaProducerWithAck` class with a `main` method that takes two arguments: a topic name and a file path. It uses `KafkaProducer` to send records to the specified topic. The console output shows the program running successfully, sending records to the `ItemTopic` and `UserTopic` topics. The records are acknowledged as well. The output is as follows:

```
<terminated> MyKafkaProducer (2) [Java Application] /usr/java/jdk1.8.0_151/bin/java (Dec 2, 2018, 3:55:57 PM)
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"102"}, Value:{user_id:"U110"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Cylin"}, Value:{exp:15}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"102"}, Value:{user_id:"U101"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Prod"}, Value:{exp:14}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"104"}, Value:{user_id:"U102"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Abhay"}, Value:{exp:17}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"107"}, Value:{user_id:"U104"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Misano"}, Value:{exp:19}
```

Running the Jar

java -cp \$KAFKA_HOME/share/java/kafka/*:/home/acadgild/Desktop/my.jar
KafkaProducerWithAck /home/acadgild/dataset_producer.txt -



The screenshot shows a terminal window with the command `java -cp $KAFKA_HOME/share/java/kafka/*:/home/acadgild/Desktop/my.jar KafkaProducerWithAck /home/acadgild/dataset_producer.txt` executed. The output shows the program sending records to the `ItemTopic` and `UserTopic` topics. The records are acknowledged as well. The output is as follows:

```
[acadgild@localhost Desktop]$ java -cp $KAFKA_HOME/share/java/kafka/*:/home/acadgild/Desktop/my.jar KafkaProducerWithAck /home/acadgild/dataset_producer.txt
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"101"}, Value:{user_id:"U101"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"John"}, Value:{exp:16}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"101"}, Value:{user_id:"U106"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Mark"}, Value:{exp:18}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"102"}, Value:{user_id:"U110"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Cylin"}, Value:{exp:15}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"102"}, Value:{user_id:"U101"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Prod"}, Value:{exp:14}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"104"}, Value:{user_id:"U102"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Abhay"}, Value:{exp:17}
Record sent to topic:ItemTopic and acknowledged as well. Key:{item_id:"107"}, Value:{user_id:"U104"}
Record sent to topic:UserTopic and acknowledged as well. Key:{name:"Misano"}, Value:{exp:19}
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost Desktop]$
```

port MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

User Topic output

```
$KAFKA_HOME/bin/kafka-console-consumer.sh --topic UserTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
```

```
[acadgild@localhost ~]$ $KAFKA_HOME/bin/kafka-console-consumer.sh --topic UserTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
Using the ConsoleConsumer with old consumer is deprecated and will be removed in a future major release. Consider using the new consumer by passing [bootstrap-server] instead of [zookeeper].
{"name":"John"} {"exp":16}
{"name":"Mark"} {"exp":18}
{"name":"Cylin"} {"exp":15}
{"name":"Prod"} {"exp":14}
{"name":"Abhay"} {"exp":17}
{"name":"Misano"} {"exp":19}
{"name":"John"} {"exp":16}
{"name":"Mark"} {"exp":18}
{"name":"Cylin"} {"exp":15}
{"name":"Prod"} {"exp":14}
{"name":"Abhay"} {"exp":17}
{"name":"Misano"} {"exp":19}
{"name":"John"} {"exp":16}
{"name":"Mark"} {"exp":18}
{"name":"Cylin"} {"exp":15}
{"name":"Prod"} {"exp":14}
{"name":"Abhay"} {"exp":17}
{"name":"Misano"} {"exp":19}
```

Item Topic output

```
$KAFKA_HOME/bin/kafka-console-consumer.sh --topic ItemTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
```

```
[acadgild@localhost ~]$ $KAFKA_HOME/bin/kafka-console-consumer.sh --topic ItemTopic --from-beginning --zookeeper localhost:2181 --property print.key=true
Using the ConsoleConsumer with old consumer is deprecated and will be removed in a future major release. Consider using the new consumer by passing [bootstrap-server] instead of [zookeeper].
{"item_id":"101"} {"user_id":"U101"}
{"item_id":"101"} {"user_id":"U106"}
{"item_id":"102"} {"user_id":"U110"}
{"item_id":"102"} {"user_id":"U101"}
{"item_id":"104"} {"user_id":"U102"}
{"item_id":"107"} {"user_id":"U104"}
{"item_id":"101"} {"user_id":"U101"}
{"item_id":"101"} {"user_id":"U106"}
{"item_id":"102"} {"user_id":"U110"}
{"item_id":"102"} {"user_id":"U101"}
{"item_id":"104"} {"user_id":"U102"}
{"item_id":"107"} {"user_id":"U104"}
{"item_id":"101"} {"user_id":"U101"}
{"item_id":"101"} {"user_id":"U106"}
{"item_id":"102"} {"user_id":"U110"}
{"item_id":"102"} {"user_id":"U101"}
{"item_id":"104"} {"user_id":"U102"}
{"item_id":"107"} {"user_id":"U104"}
```