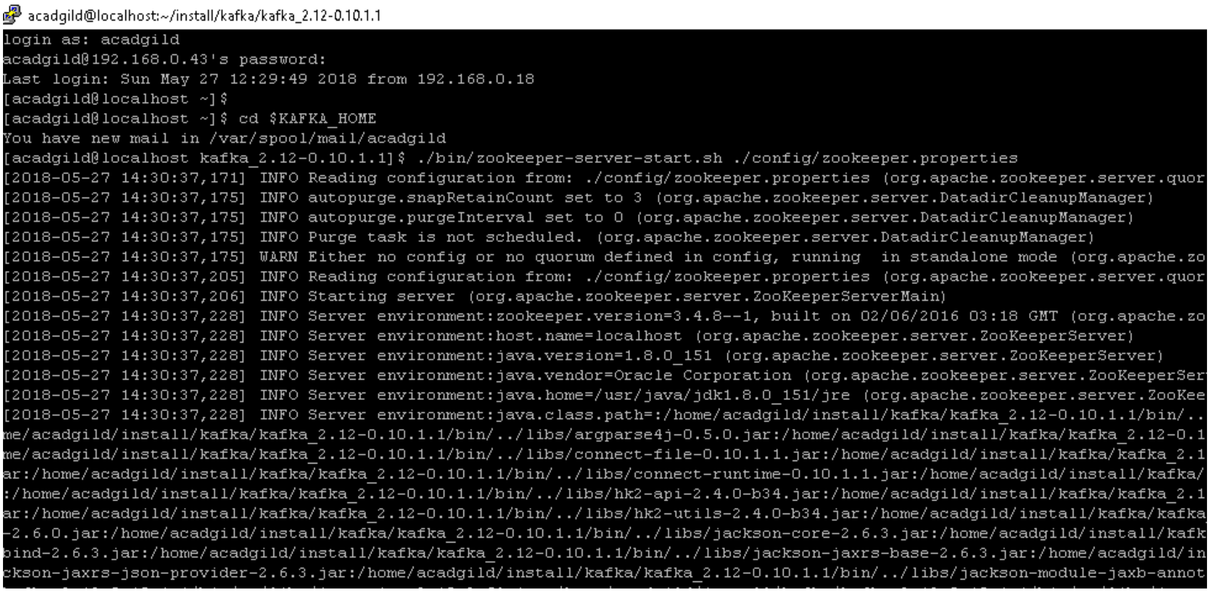
ASSIGNMENT 24.1

To solve all the tasks we have created two java programs MyKafkaProducer.java and MyKafkaProducerWithAck.java .The explanation of these codes is present in their respective java files.

We need to start the **zookeeper** and the **Kafka** broker as show below in the screen shots.

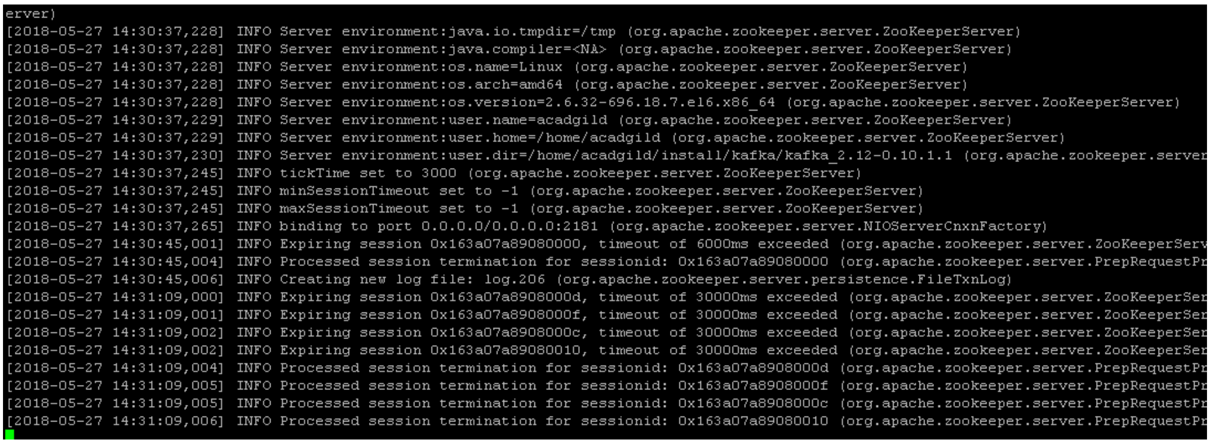
First, we will go to KAFKA directory by changing the directory path and run the command to start zookeeper server by using following commands.

cd $KAFKA_HOME
./bin/zookeeper-server-start.sh ./config/zookeeper.properties



acadgild@localhost:~/install/kafka/kafka_2.12-0.10.1.1

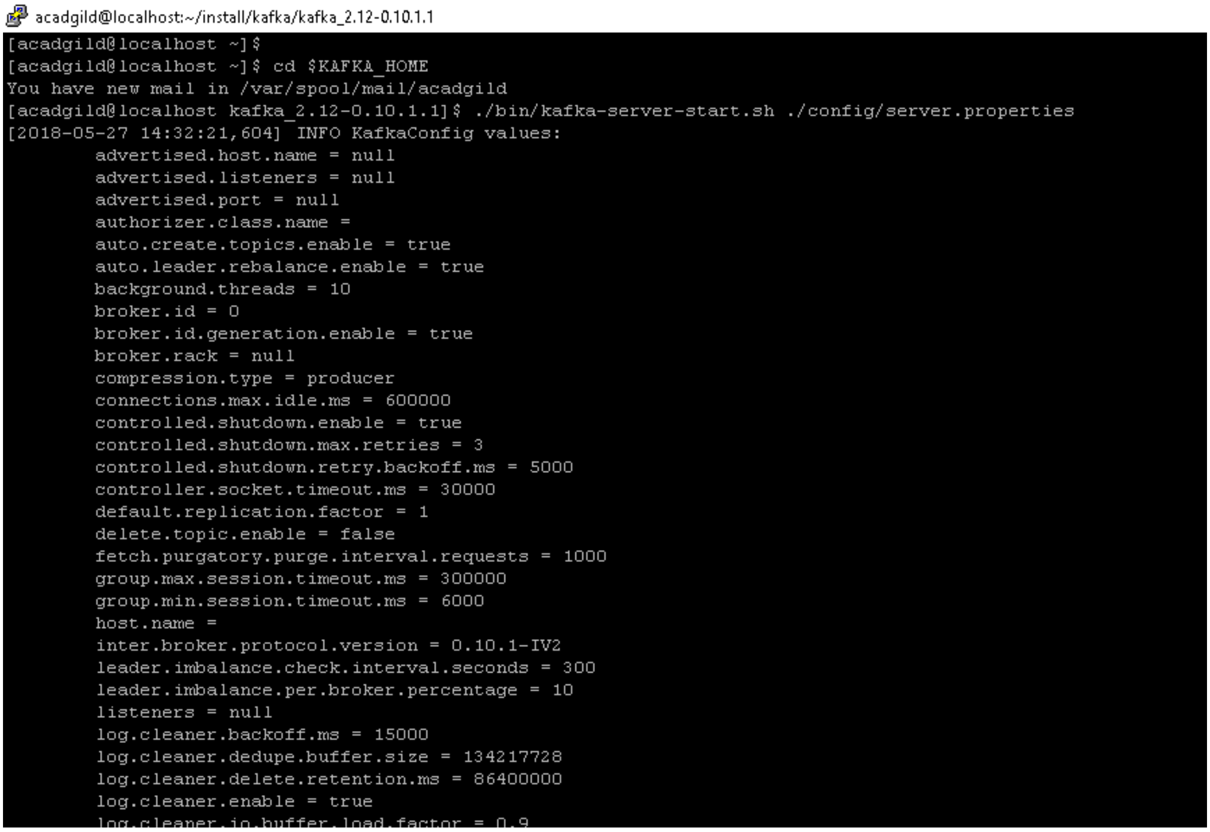
```
login as: acadgild
acadgild@192.168.0.43's password:
Last login: Sun May 27 12:29:49 2018 from 192.168.0.18
[acadgild@localhost ~]$
[acadgild@localhost ~]$ cd $KAFKA_HOME
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./bin/zookeeper-server-start.sh ./config/zookeeper.properties
[2018-05-27 14:30:37,171] INFO Reading configuration from: ./config/zookeeper.properties (org.apache.zookeeper.server.quor
[2018-05-27 14:30:37,175] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DatadirCleanupManager)
[2018-05-27 14:30:37,175] INFO autopurge.purgeInterval set to 0 (org.apache.zookeeper.server.DatadirCleanupManager)
[2018-05-27 14:30:37,175] INFO Purge task is not scheduled. (org.apache.zookeeper.server.DatadirCleanupManager)
[2018-05-27 14:30:37,175] WARN Either no config or no quorum defined in config, running  in standalone mode (org.apache.zo
[2018-05-27 14:30:37,205] INFO Reading configuration from: ./config/zookeeper.properties (org.apache.zookeeper.server.quor
[2018-05-27 14:30:37,206] INFO Starting server (org.apache.zookeeper.server.ZooKeeperServerMain)
[2018-05-27 14:30:37,228] INFO Server environment:zookeeper.version=3.4.8--1, built on 02/06/2016 03:18 GMT (org.apache.zo
[2018-05-27 14:30:37,228] INFO Server environment:host.name=localhost (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,228] INFO Server environment:java.version=1.8.0_151 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,228] INFO Server environment:java.vendor=Oracle Corporation (org.apache.zookeeper.server.ZooKeeperSer
[2018-05-27 14:30:37,228] INFO Server environment:java.home=/usr/java/jdk1.8.0_151/jre (org.apache.zookeeper.server.ZooKee
[2018-05-27 14:30:37,228] INFO Server environment:java.class.path=:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/..
me/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/argparse4j-0.5.0.jar:/home/acadgild/install/kafka/kafka_2.12-0.1
me/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/connect-file-0.10.1.1.jar:/home/acadgild/install/kafka/kafka_2.1
ar:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/connect-runtime-0.10.1.1.jar:/home/acadgild/install/kafka/
:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/hk2-api-2.4.0-b34.jar:/home/acadgild/install/kafka/kafka_2.1
ar:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/hk2-utils-2.4.0-b34.jar:/home/acadgild/install/kafka/kafka
-2.6.0.jar:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/jackson-core-2.6.3.jar:/home/acadgild/install/kafk
bind-2.6.3.jar:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/jackson-jaxrs-base-2.6.3.jar:/home/acadgild/in
ckson-jaxrs-json-provider-2.6.3.jar:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/jackson-module-jaxb-annot
```



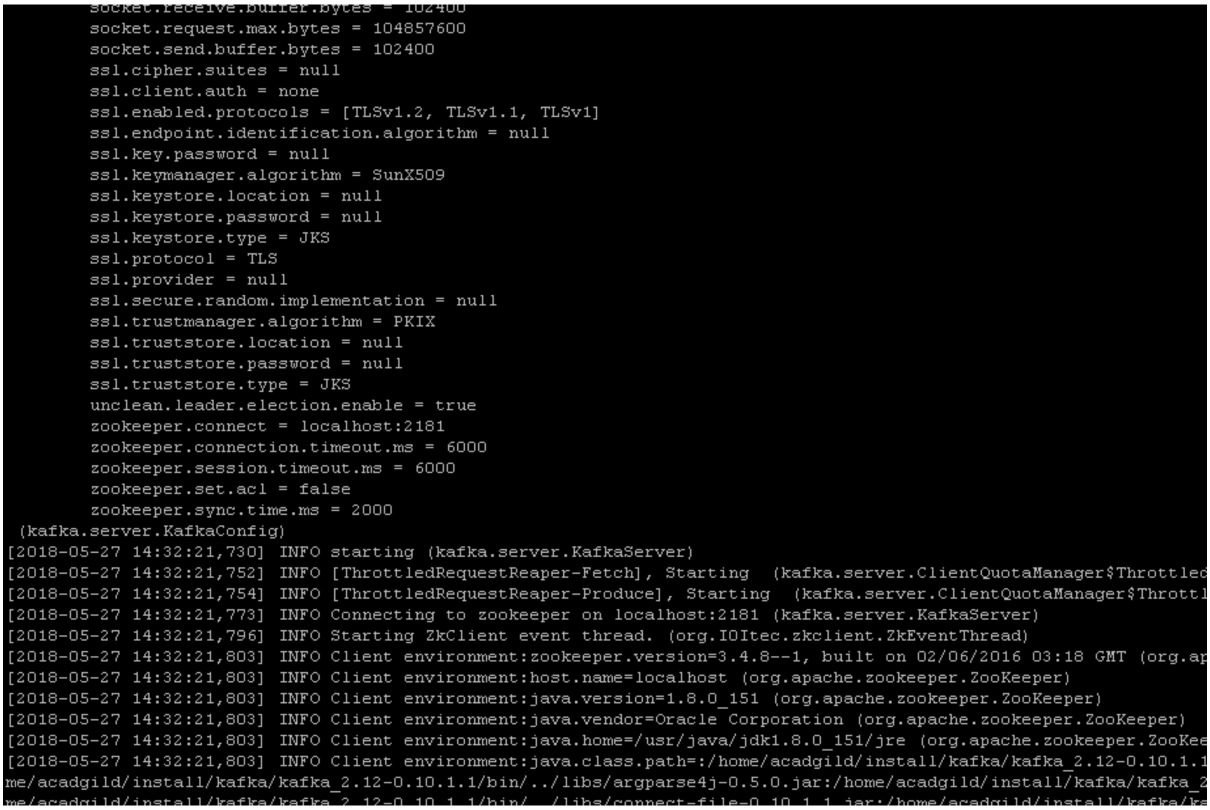
```
erver)
[2018-05-27 14:30:37,228] INFO Server environment:java.io.tmpdir=/tmp (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,228] INFO Server environment:java.compiler=<NA> (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,228] INFO Server environment:os.name=Linux (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,228] INFO Server environment:os.arch=amd64 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,228] INFO Server environment:os.version=2.6.32-696.18.7.el6.x86_64 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,229] INFO Server environment:user.name=acadgild (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,229] INFO Server environment:user.home=/home/acadgild (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,230] INFO Server environment:user.dir=/home/acadgild/install/kafka/kafka_2.12-0.10.1.1 (org.apache.zookeeper.server
[2018-05-27 14:30:37,245] INFO tickTime set to 3000 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,245] INFO minSessionTimeout set to -1 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,245] INFO maxSessionTimeout set to -1 (org.apache.zookeeper.server.ZooKeeperServer)
[2018-05-27 14:30:37,265] INFO binding to port 0.0.0.0/0.0.0.0:2181 (org.apache.zookeeper.server.NIOServerCnxnFactory)
[2018-05-27 14:30:45,001] INFO Expiring session 0x163a07a89080000, timeout of 6000ms exceeded (org.apache.zookeeper.server.ZooKeeperServ
[2018-05-27 14:30:45,004] INFO Processed session termination for sessionid: 0x163a07a89080000 (org.apache.zookeeper.server.PrepRequestPr
[2018-05-27 14:30:45,006] INFO Creating new log file: log.206 (org.apache.zookeeper.server.persistence.FileTxnLog)
[2018-05-27 14:31:09,000] INFO Expiring session 0x163a07a8908000d, timeout of 30000ms exceeded (org.apache.zookeeper.server.ZooKeeperSer
[2018-05-27 14:31:09,001] INFO Expiring session 0x163a07a8908000f, timeout of 30000ms exceeded (org.apache.zookeeper.server.ZooKeeperSer
[2018-05-27 14:31:09,002] INFO Expiring session 0x163a07a8908000c, timeout of 30000ms exceeded (org.apache.zookeeper.server.ZooKeeperSer
[2018-05-27 14:31:09,002] INFO Expiring session 0x163a07a89080010, timeout of 30000ms exceeded (org.apache.zookeeper.server.ZooKeeperSer
[2018-05-27 14:31:09,004] INFO Processed session termination for sessionid: 0x163a07a8908000d (org.apache.zookeeper.server.PrepRequestPr
[2018-05-27 14:31:09,005] INFO Processed session termination for sessionid: 0x163a07a8908000f (org.apache.zookeeper.server.PrepRequestPr
[2018-05-27 14:31:09,005] INFO Processed session termination for sessionid: 0x163a07a8908000c (org.apache.zookeeper.server.PrepRequestPr
[2018-05-27 14:31:09,006] INFO Processed session termination for sessionid: 0x163a07a89080010 (org.apache.zookeeper.server.PrepRequestPr
```

To start **kafka server** we use the below command in KAFKA home directory

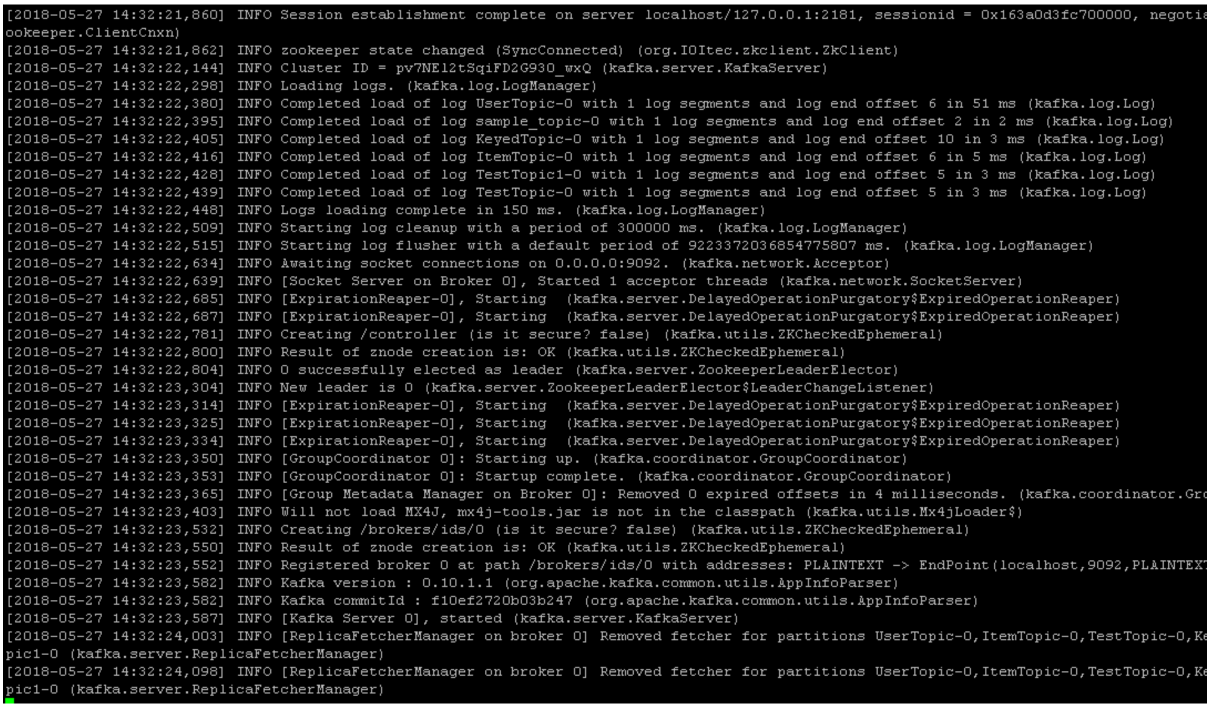
cd $KAFKA_HOME
./bin/kafka-server-start.sh ./config/server.properties



```
[acadgild@localhost ~]$
[acadgild@localhost ~]$ cd $KAFKA_HOME
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./bin/kafka-server-start.sh ./config/server.properties
[2018-05-27 14:32:21,604] 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
        log.cleaner.dedupe.buffer.size = 134217728
        log.cleaner.delete.retention.ms = 86400000
        log.cleaner.enable = true
        log.cleaner.io.buffer.load.factor = 0.9
```



```
        socket.receive.buffer.bytes = 102400
        socket.request.max.bytes = 104857600
        socket.send.buffer.bytes = 102400
        ssl.cipher.suites = null
        ssl.client.auth = none
        ssl.enabled.protocols = [TLSv1.2, TLSv1.1, TLSv1]
        ssl.endpoint.identification.algorithm = null
        ssl.key.password = null
        ssl.keymanager.algorithm = SunX509
        ssl.keystore.location = null
        ssl.keystore.password = null
        ssl.keystore.type = JKS
        ssl.protocol = TLS
        ssl.provider = null
        ssl.secure.random.implementation = null
        ssl.trustmanager.algorithm = PKIX
        ssl.truststore.location = null
        ssl.truststore.password = null
        ssl.truststore.type = JKS
        unclean.leader.election.enable = true
        zookeeper.connect = localhost:2181
        zookeeper.connection.timeout.ms = 6000
        zookeeper.session.timeout.ms = 6000
        zookeeper.set.acl = false
        zookeeper.sync.time.ms = 2000
 (kafka.server.KafkaConfig)
[2018-05-27 14:32:21,730] INFO starting (kafka.server.KafkaServer)
[2018-05-27 14:32:21,752] INFO [ThrottledRequestReaper-Fetch], Starting  (kafka.server.ClientQuotaManager$Throttled
[2018-05-27 14:32:21,754] INFO [ThrottledRequestReaper-Produce], Starting  (kafka.server.ClientQuotaManager$Throttl
[2018-05-27 14:32:21,773] INFO Connecting to zookeeper on localhost:2181 (kafka.server.KafkaServer)
[2018-05-27 14:32:21,796] INFO Starting ZkClient event thread. (org.I0Itec.zkclient.ZkEventThread)
[2018-05-27 14:32:21,803] INFO Client environment:zookeeper.version=3.4.8--1, built on 02/06/2016 03:18 GMT (org.ap
[2018-05-27 14:32:21,803] INFO Client environment:host.name=localhost (org.apache.zookeeper.ZooKeeper)
[2018-05-27 14:32:21,803] INFO Client environment:java.version=1.8.0_151 (org.apache.zookeeper.ZooKeeper)
[2018-05-27 14:32:21,803] INFO Client environment:java.vendor=Oracle Corporation (org.apache.zookeeper.ZooKeeper)
[2018-05-27 14:32:21,803] INFO Client environment:java.home=/usr/java/jdk1.8.0_151/jre (org.apache.zookeeper.ZooKee
[2018-05-27 14:32:21,803] INFO Client environment:java.class.path=:/home/acadgild/install/kafka/kafka_2.12-0.10.1.1
me/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/argparse4j-0.5.0.jar:/home/acadgild/install/kafka/kafka_2
me/acadgild/install/kafka/kafka_2.12-0.10.1.1/bin/../libs/connect-file-0.10.1.1.jar:/home/acadgild/install/kafka/ka
```



```
[2018-05-27 14:32:21,860] INFO Session establishment complete on server localhost/127.0.0.1:2181, sessionid = 0x163a0d3fc700000, negoti
ookeeper.ClientCnxn)
[2018-05-27 14:32:21,862] INFO zookeeper state changed (SyncConnected) (org.I0Itec.zkclient.ZkClient)
[2018-05-27 14:32:22,144] INFO Cluster ID = pv7NEl2tSqiFD2G930_wxQ (kafka.server.KafkaServer)
[2018-05-27 14:32:22,298] INFO Loading logs. (kafka.log.LogManager)
[2018-05-27 14:32:22,380] INFO Completed load of log UserTopic-0 with 1 log segments and log end offset 6 in 51 ms (kafka.log.Log)
[2018-05-27 14:32:22,395] INFO Completed load of log sample_topic-0 with 1 log segments and log end offset 2 in 2 ms (kafka.log.Log)
[2018-05-27 14:32:22,405] INFO Completed load of log KeyedTopic-0 with 1 log segments and log end offset 10 in 3 ms (kafka.log.Log)
[2018-05-27 14:32:22,416] INFO Completed load of log ItemTopic-0 with 1 log segments and log end offset 6 in 5 ms (kafka.log.Log)
[2018-05-27 14:32:22,428] INFO Completed load of log TestTopic1-0 with 1 log segments and log end offset 5 in 3 ms (kafka.log.Log)
[2018-05-27 14:32:22,439] INFO Completed load of log TestTopic-0 with 1 log segments and log end offset 5 in 3 ms (kafka.log.Log)
[2018-05-27 14:32:22,448] INFO Logs loading complete in 150 ms. (kafka.log.LogManager)
[2018-05-27 14:32:22,509] INFO Starting log cleanup with a period of 300000 ms. (kafka.log.LogManager)
[2018-05-27 14:32:22,515] INFO Starting log flusher with a default period of 9223372036854775807 ms. (kafka.log.LogManager)
[2018-05-27 14:32:22,634] INFO Awaiting socket connections on 0.0.0.0:9092. (kafka.network.Acceptor)
[2018-05-27 14:32:22,639] INFO [Socket Server on Broker 0], Started 1 acceptor threads (kafka.network.SocketServer)
[2018-05-27 14:32:22,685] INFO [ExpirationReaper-0], Starting  (kafka.server.DelayedOperationPurgatory$ExpiredOperationReaper)
[2018-05-27 14:32:22,687] INFO [ExpirationReaper-0], Starting  (kafka.server.DelayedOperationPurgatory$ExpiredOperationReaper)
[2018-05-27 14:32:22,781] INFO Creating /controller (is it secure? false) (kafka.utils.ZKCheckedEphemeral)
[2018-05-27 14:32:22,800] INFO Result of znode creation is: OK (kafka.utils.ZKCheckedEphemeral)
[2018-05-27 14:32:22,804] INFO 0 successfully elected as leader (kafka.server.ZookeeperLeaderElector)
[2018-05-27 14:32:23,304] INFO New leader is 0 (kafka.server.ZookeeperLeaderElector$LeaderChangeListener)
[2018-05-27 14:32:23,314] INFO [ExpirationReaper-0], Starting  (kafka.server.DelayedOperationPurgatory$ExpiredOperationReaper)
[2018-05-27 14:32:23,325] INFO [ExpirationReaper-0], Starting  (kafka.server.DelayedOperationPurgatory$ExpiredOperationReaper)
[2018-05-27 14:32:23,334] INFO [ExpirationReaper-0], Starting  (kafka.server.DelayedOperationPurgatory$ExpiredOperationReaper)
[2018-05-27 14:32:23,350] INFO [GroupCoordinator 0]: Starting up. (kafka.coordinator.GroupCoordinator)
[2018-05-27 14:32:23,353] INFO [GroupCoordinator 0]: Startup complete. (kafka.coordinator.GroupCoordinator)
[2018-05-27 14:32:23,365] INFO [Group Metadata Manager on Broker 0]: Removed 0 expired offsets in 4 milliseconds. (kafka.coordinator.Gro
[2018-05-27 14:32:23,403] INFO Will not load MX4J, mx4j-tools.jar is not in the classpath (kafka.utils.Mx4jLoader$)
[2018-05-27 14:32:23,532] INFO Creating /brokers/ids/0 (is it secure? false) (kafka.utils.ZKCheckedEphemeral)
[2018-05-27 14:32:23,550] INFO Result of znode creation is: OK (kafka.utils.ZKCheckedEphemeral)
[2018-05-27 14:32:23,552] INFO Registered broker 0 at path /brokers/ids/0 with addresses: PLAINTEXT -> EndPoint(localhost,9092,PLAINTEXT
[2018-05-27 14:32:23,582] INFO Kafka version : 0.10.1.1 (org.apache.kafka.common.utils.AppInfoParser)
[2018-05-27 14:32:23,582] INFO Kafka commitId : f10ef2720b03b247 (org.apache.kafka.common.utils.AppInfoParser)
[2018-05-27 14:32:23,587] INFO [Kafka Server 0], started (kafka.server.KafkaServer)
[2018-05-27 14:32:24,003] INFO [ReplicaFetcherManager on broker 0] Removed fetcher for partitions UserTopic-0,ItemTopic-0,TestTopic-0,Ke
pic1-0 (kafka.server.ReplicaFetcherManager)
[2018-05-27 14:32:24,098] INFO [ReplicaFetcherManager on broker 0] Removed fetcher for partitions UserTopic-0,ItemTopic-0,TestTopic-0,Ke
pic1-0 (kafka.server.ReplicaFetcherManager)
```

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.

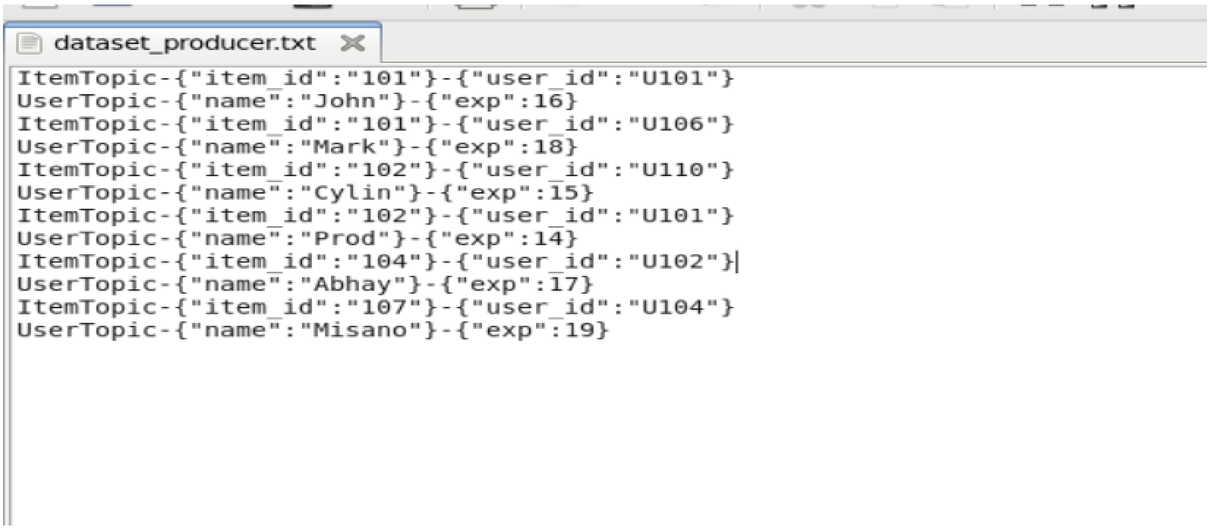
Fields in the file are in following order

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.

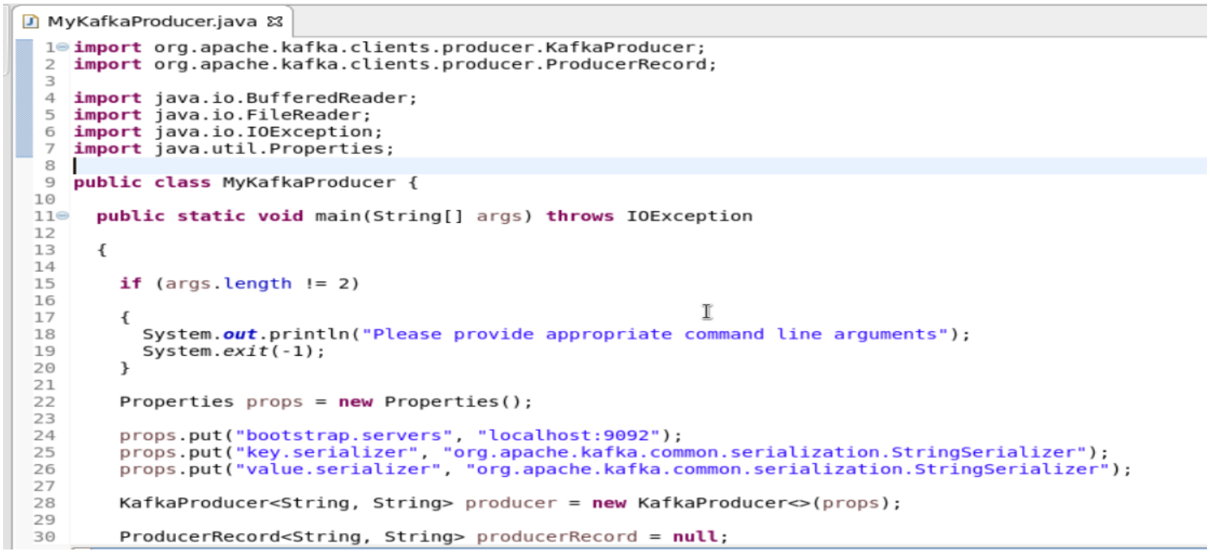
Pass dataset_producer.txt as the input file and **-** as delimiter.

Below screenshot, shows the input file **Pass dataset_producer.txt**

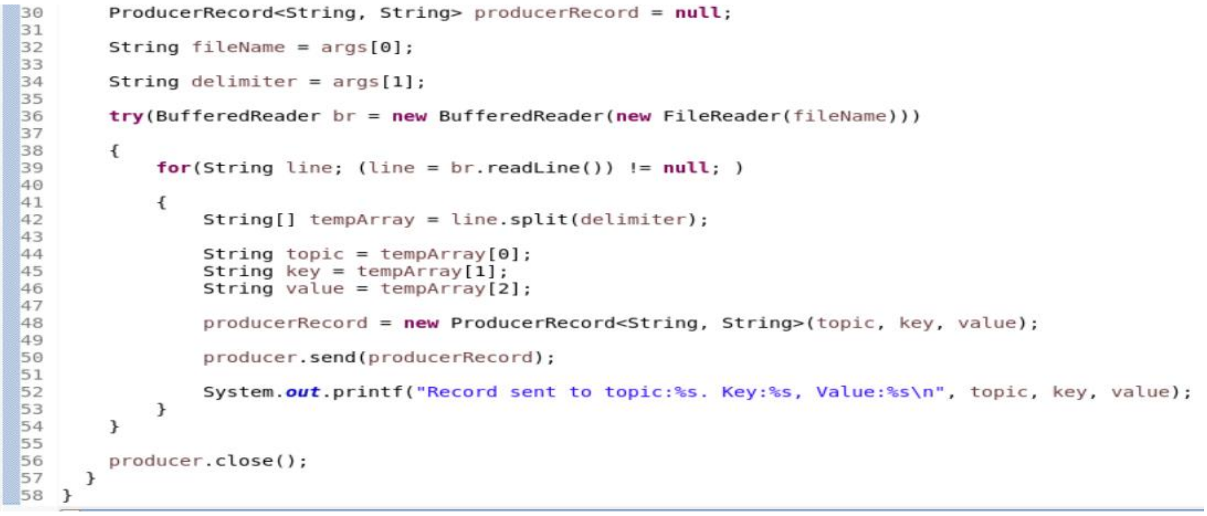


```
ItemTopic-{"item_id":"101"}-{"user_id":"U101"}
UserTopic-{"name":"John"}-{"exp":16}
ItemTopic-{"item_id":"101"}-{"user_id":"U106"}
UserTopic-{"name":"Mark"}-{"exp":18}
ItemTopic-{"item_id":"102"}-{"user_id":"U110"}
UserTopic-{"name":"Cylin"}-{"exp":15}
ItemTopic-{"item_id":"102"}-{"user_id":"U101"}
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}
```

Below screenshot, shows the **MyKafkaProducer** java program

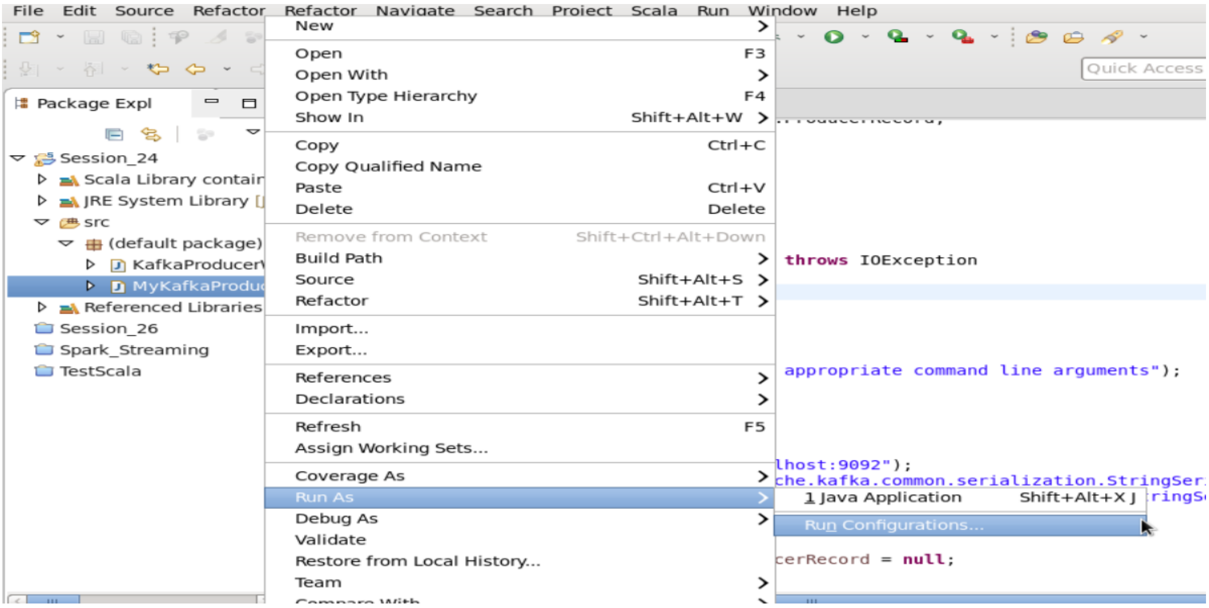


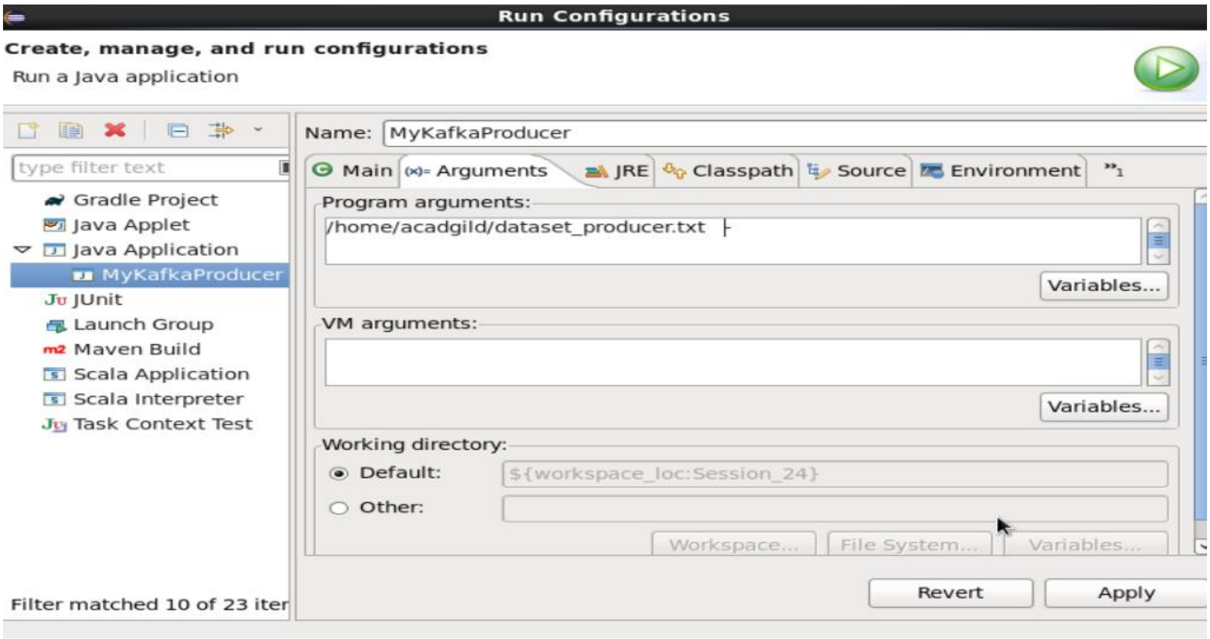
```java
import org.apache.kafka.clients.producer.KafkaProducer;
import org.apache.kafka.clients.producer.ProducerRecord;

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
import java.util.Properties;

public class MyKafkaProducer {

  public static void main(String[] args) throws IOException

  {

    if (args.length != 2)

    {
      System.out.println("Please provide appropriate command line arguments");
      System.exit(-1);
    }

    Properties props = new Properties();

    props.put("bootstrap.servers", "localhost:9092");
    props.put("key.serializer", "org.apache.kafka.common.serialization.StringSerializer");
    props.put("value.serializer", "org.apache.kafka.common.serialization.StringSerializer");

    KafkaProducer<String, String> producer = new KafkaProducer<>(props);

    ProducerRecord<String, String> producerRecord = null;
```



```
        ProducerRecord<String, String> producerRecord = null;

        String fileName = args[0];

        String delimiter = args[1];

        try(BufferedReader br = new BufferedReader(new FileReader(fileName)))

        {
            for(String line; (line = br.readLine()) != null; )

            {
                String[] tempArray = line.split(delimiter);

                String topic = tempArray[0];
                String key = tempArray[1];
                String value = tempArray[2];

                producerRecord = new ProducerRecord<String, String>(topic, key, value);

                producer.send(producerRecord);

                System.out.printf("Record sent to topic:%s. Key:%s, Value:%s\n", topic, key, value);
            }
        }

        producer.close();
    }
}
```

Let us run the java program with passing the input data file and “-“ as arguments





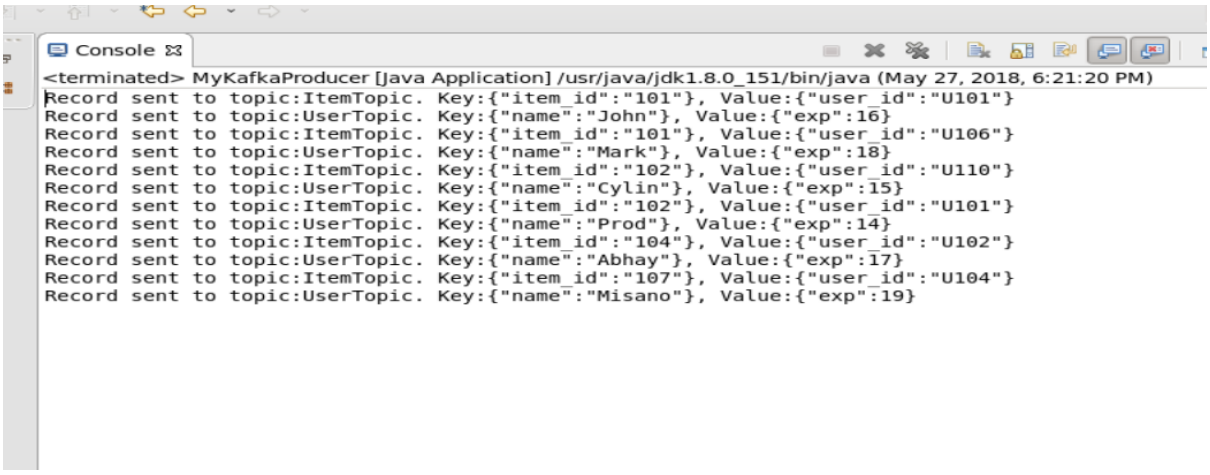
Name: MyKafkaProducer

Program arguments: /home/acadgild/dataset_producer.txt

VM arguments:

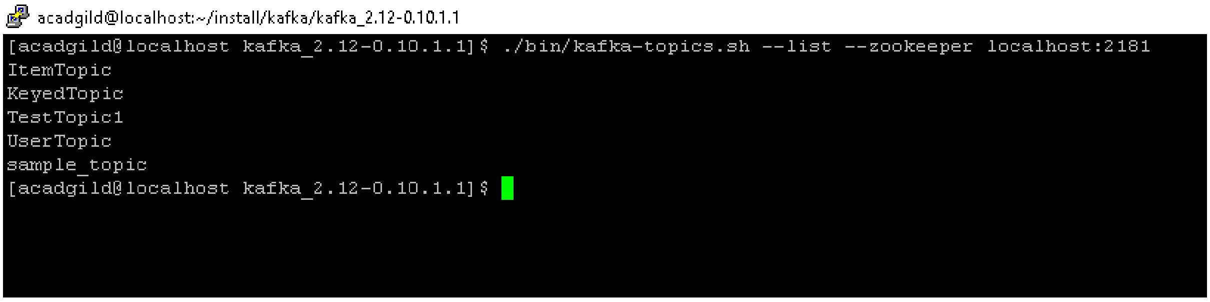
Working directory:
● Default: ${workspace_loc:Session_24}
○ Other:

Below screenshot, shows for every line, producer insert the key and value to the respective Kafka topic in broker in a fire mode or send mode



```
<terminated> MyKafkaProducer [Java Application] /usr/java/jdk1.8.0_151/bin/java (May 27, 2018, 6:21:20 PM)
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}
```

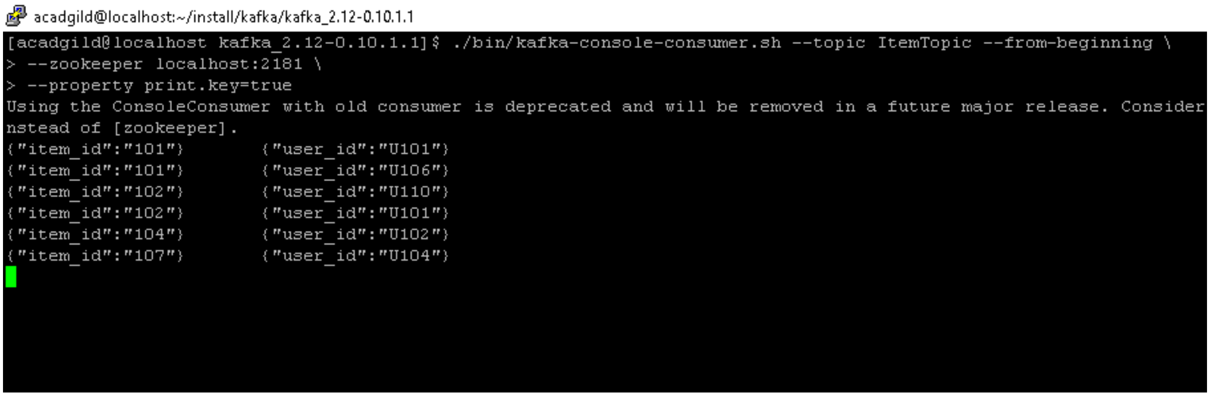
Below screenshot shows the topic created by the kafka producer program which is **ItemTopic** and **UserTopic**



acadgild@localhost:~/install/kafka/kafka_2.12-0.10.1.1

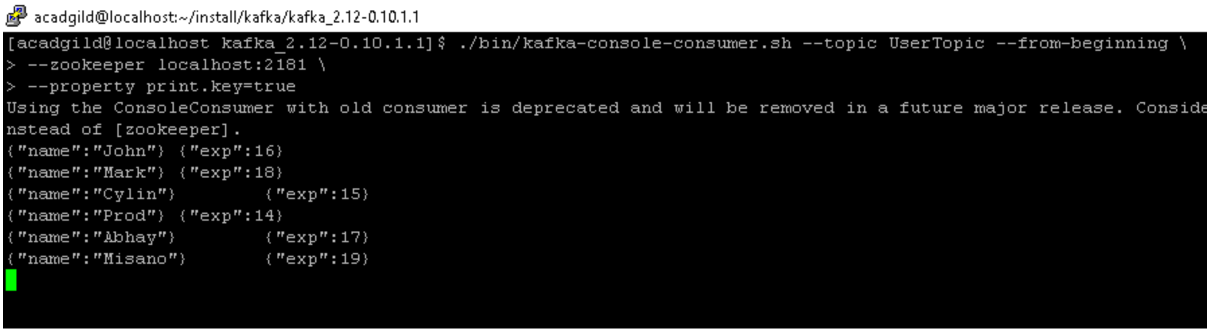
```
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./bin/kafka-topics.sh --list --zookeeper localhost:2181
ItemTopic
KeyedTopic
TestTopic1
UserTopic
sample_topic
[acadgild@localhost kafka_2.12-0.10.1.1]$
```

Running a kafka console consumer for topic **ItemTopic** to print the message in key value format as shown below



```
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./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
nstead 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"}
```

Running a kafka console consumer for topic **UserTopic** to print the message in key value format as shown below



```
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./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. Conside
nstead 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}
```

Task 2:

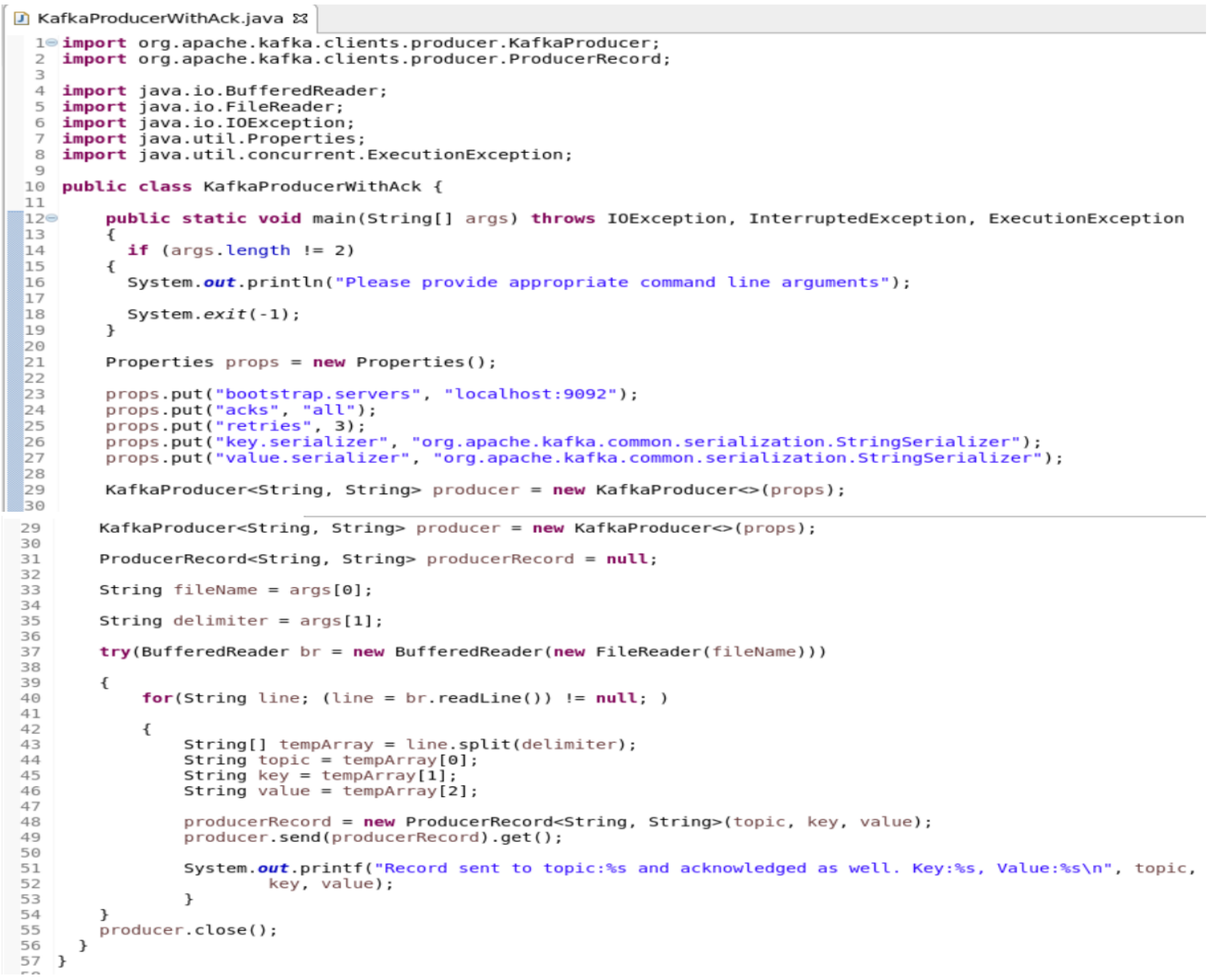
Modify the previous program MyKafkaProducer.java and create a new Java program KafkaProducerWithAck.java.

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.

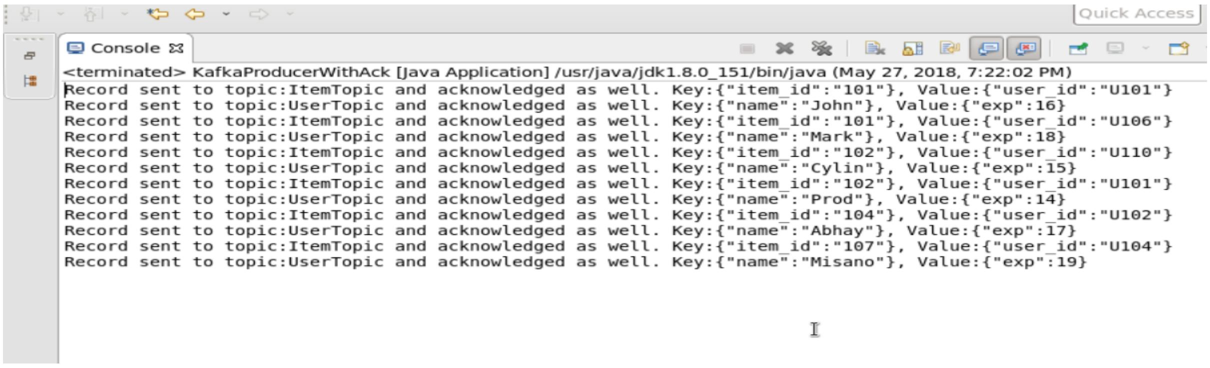
The application should attempt for three retries before giving any exception.
Pass dataset_producer.txt as the input file and - as delimiter.

Below screen, shot shows the **KafkaProducerWithAck** program



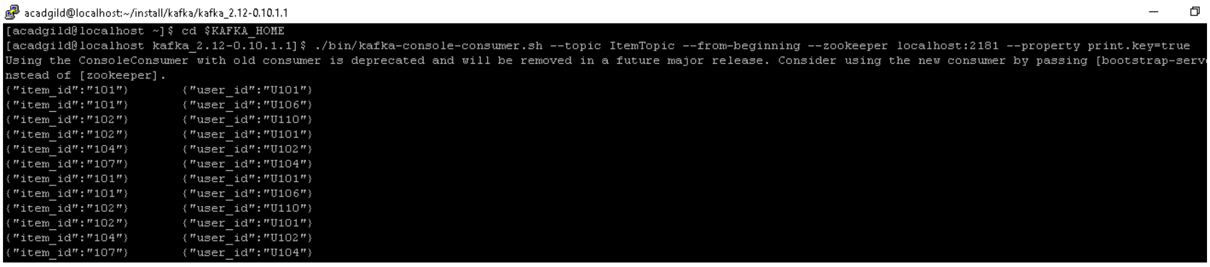
```java
import org.apache.kafka.clients.producer.KafkaProducer;
import org.apache.kafka.clients.producer.ProducerRecord;

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
import java.util.Properties;
import java.util.concurrent.ExecutionException;

public class KafkaProducerWithAck {

    public static void main(String[] args) throws IOException, InterruptedException, ExecutionException
    {
      if (args.length != 2)
    {
      System.out.println("Please provide appropriate command line arguments");

      System.exit(-1);
    }

    Properties props = new Properties();

    props.put("bootstrap.servers", "localhost:9092");
    props.put("acks", "all");
    props.put("retries", 3);
    props.put("key.serializer", "org.apache.kafka.common.serialization.StringSerializer");
    props.put("value.serializer", "org.apache.kafka.common.serialization.StringSerializer");

    KafkaProducer<String, String> producer = new KafkaProducer<>(props);

    ProducerRecord<String, String> producerRecord = null;

    String fileName = args[0];

    String delimiter = args[1];

    try(BufferedReader br = new BufferedReader(new FileReader(fileName)))

    {
        for(String line; (line = br.readLine()) != null; )

        {
            String[] tempArray = line.split(delimiter);
            String topic = tempArray[0];
            String key = tempArray[1];
            String value = tempArray[2];

            producerRecord = new ProducerRecord<String, String>(topic, key, value);
            producer.send(producerRecord).get();

            System.out.printf("Record sent to topic:%s and acknowledged as well. Key:%s, Value:%s\n", topic,
                    key, value);
            }

    }
    producer.close();
  }
}
```

Below screen shot shows the producer program acknowledge first and after receiving acknowledgment from the broker, it print the message **acknowledged as well** and key and value, which has been written to a specified topic



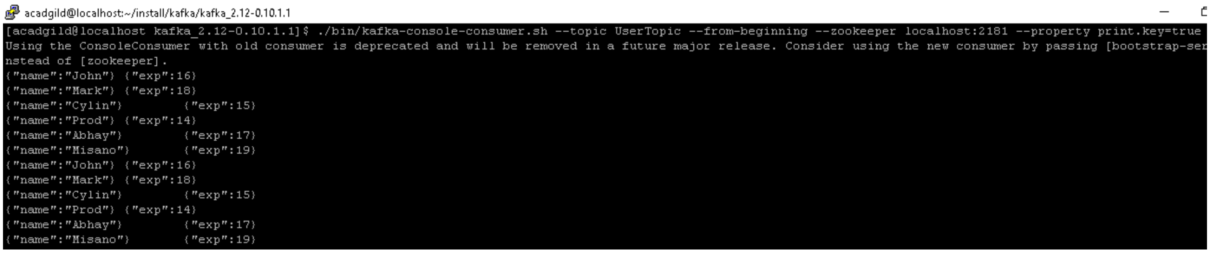
```
<terminated> KafkaProducerWithAck [Java Application] /usr/java/jdk1.8.0_151/bin/java (May 27, 2018, 7:22:02 PM)
Record sent to topic:ItemTopic and acknowledged as well. Key:{"item_id":"101"}, Value:{"user_id":"U101"}
Record sent to topic:UserTopic and acknowledged as well. Key:{"name":"John"}, Value:{"exp":16}
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}
```

Running a kafka console consumer for topic **ItemTopic** to print the message in key value format as shown below



```
[acadgild@localhost ~]$ cd $KAFKA_HOME
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./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-serv
nstead 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"}
```

Running a kafka console consumer for topic **UserTopic** to print the message in key value format as shown below



```
[acadgild@localhost kafka_2.12-0.10.1.1]$ ./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-ser
nstead 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}
```