//Apache Kafka advanced concepts to read from a file based on delimiters.

```
//Reading Kafka topic Key, value from a file based upon certain
delimiters.
//Demo of reading line by line from file & insert respective key-val pair
to the broker in fire & forget mode.
//Demo of interaction with the broker based on some acknowledgement
received from broker.
//Specifying 3 read re-tries by application before throwing an exception.

package producer;
//Diifferent imports
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;


//Declaring a public class MyKafka_producer
public class MyKafka_producer {


//Main function. The start of execution of the program
public static void main(String[] args) throws IOException{

//Initialization object of properties class.
Properties props = new Properties();

//setting the properties parameters
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");

//setting producer properties
KafkaProducer<String, String> producer = new KafkaProducer<>(props);
```

```
//Setting producerRecord String type to NULL
ProducerRecord<String, String> producerRecord = null;

//setting file path placed on local
String fileName =
"/home/acadgild/Desktop/TestHadoop/kafka/dataset_producer.txt";

//Setting delimeter value to hyphen as the fields are separated using this
delimiter
String delimiter = "-";

//Initialization of BufferReader object with arguments as FileReader
object
try(BufferedReader br = new BufferedReader(new FileReader(fileName))) {

//Reading each non-null line from the input file.
for(String line; (line = br.readLine()) != null; ) {

//Splitting value of each line based upon a delimiter & storing into Array
String[] tempArray = line.split(delimiter);

//Fetching topic at 0th index
String topic = tempArray[0];

//Fetching key from first index
String key = tempArray[1];

//Fetching value @ 2nd index of the array
String value = tempArray[2];

//Initializing mew ProducerRecord with the topic, key value fetched from
above
//the key and value are sent to the repsective Kafka broker in a fire and
forget mode.
producerRecord = new ProducerRecord<String, String>(topic, key, value);

//After record is sent printing appropriate message on screen.
producer.send(producerRecord);

//Printing the value at the output console.
System.out.printf("Record sent to topic:%s. Key:%s, Value:%s\n", topic,
key, value);
}
}

//Terminating the producer connection.
```

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