Apache Kafka - Consumer Group Example

```java
import java.util.Properties;

import java.util.Arrays;

import org.apache.kafka.clients.consumer.KafkaConsumer;

import org.apache.kafka.clients.consumer.ConsumerRecords;

import org.apache.kafka.clients.consumer.ConsumerRecord;


public class ConsumerGroup {

    public static void main(String[] args) throws Exception {

        if(args.length < 2){

            System.out.println("Usage: consumer <topic> <groupname>");

            return;

        }


        String topic = args[0].toString();

        String group = args[1].toString();

        Properties props = new Properties();

        props.put("bootstrap.servers", "localhost:9092");

        props.put("group.id", group);

        props.put("enable.auto.commit", "true");

        props.put("auto.commit.interval.ms", "1000");

        props.put("session.timeout.ms", "30000");

        props.put("key.deserializer",

            "org.apache.kafka.common.serialization.StringDeserializer");

        props.put("value.deserializer",

            "org.apache.kafka.common.serialization.StringDeserializer");

        KafkaConsumer<String, String> consumer = new KafkaConsumer<String,
String>(props);
```

```
        consumer.subscribe(Arrays.asList(topic));

        System.out.println("Subscribed to topic " + topic);

        int i = 0;


        while (true) {

            ConsumerRecords<String, String> records = consumer.poll(100);

                for (ConsumerRecord<String, String> record : records)

                    System.out.printf("offset = %d, key = %s, value = %s\n",

                    record.offset(), record.key(), record.value());

        }

    }
}
```

Compilation

```
javac -cp "/path/to/kafka/kafka_2.11-0.9.0.0/libs/*" ConsumerGroup.java
```

Execution

```
>>java -cp "/path/to/kafka/kafka_2.11-0.9.0.0/libs/*":.
ConsumerGroup <topic-name> my-group
>>java -cp "/home/bala/Workspace/kafka/kafka_2.11-0.9.0.0/libs/*":.
ConsumerGroup <topic-name> my-group
```

Here we have created a sample group name as my-group with two consumers. Similarly, you can create your group and number of consumers in the group.

Input

Open producer CLI and send some messages like −

```
Test consumer group 01
Test consumer group 02
```

Output of the First Process

```
Subscribed to topic Hello-kafka

offset = 3, key = null, value = Test consumer group 01
```

Output of the Second Process

```
Subscribed to topic Hello-kafka
offset = 3, key = null, value = Test consumer group 02
```