UNIT 4.2 GRADED ASSIGNMENT

Group members

Ifra Saleem (2303.khi.deg.003)
Umaima Siddiqui (2023.KHI.DEG.033)

UNIT 4.2 GRADED ASSIGNMENT

Task:

Start Kafka using docker-compose and:
1. Create a topic.
2. List Kafka topics.
3. Inspect one of them to see the number of partitions.

Solution:

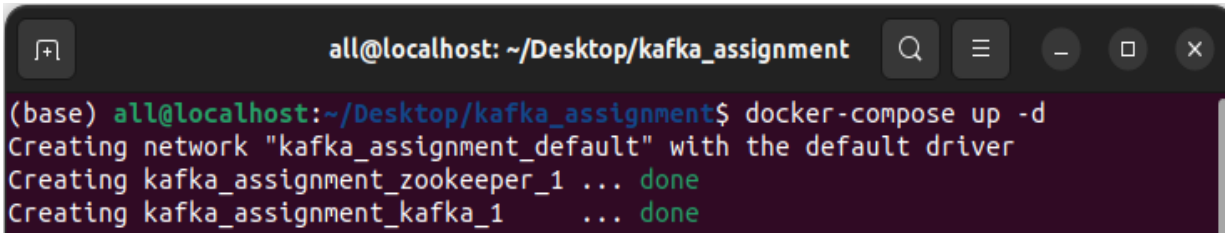
1. First, I created a docker-compose.yaml file in which I have two services, one is zookeper and another one is kafka. Both are using Confluent Platform images. This docker-compose.yaml file is used for running Apache kafka and Zookeeper. In the zookeeper service we have two environment variables ZOOKEEPER_CLIENT_PORT and ZOOKEEPER_TICK_TIME. And we are mapping container port to the host port. In the kafka service we have four environment variables.

```yaml
version: '3'
services:
  zookeeper:
    image: 'confluentinc/cp-zookeeper:latest'
    environment:
      ZOOKEEPER_CLIENT_PORT: 2181
      ZOOKEEPER_TICK_TIME: 2000
    ports:
      - '2181:2181'

  kafka:
    image: 'confluentinc/cp-kafka:latest'
    depends_on:
      - zookeeper
    environment:
      KAFKA_BROKER_ID: 1
      KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181
      KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://localhost:9092
      KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 1
    ports:
      - '9093:9093'

```

2. After creating docker-compose.yaml, we need to start the containers. We can use docker-compose up -d to start the containers in the detached mode.



3. After starting the containers, I created a create_kafka_topic.sh file in which I save following commands to create the kafka topics. I created two topics by running the command **./create_kafka_topic.sh**. It will check create_kafka_topic.sh and will execute the commands saved in that file. Replica factor is 1 which means that each partition will have one replica.

```
docker-compose exec kafka kafka-topics --create --topic topic1 --partitions 3 --replication-factor 1 --if-not-exists --bootstrap-server localhost:9092
docker-compose exec kafka kafka-topics --create --topic topic2 --partitions 4 --replication-factor 1 --if-not-exists --bootstrap-server localhost:9092

```

4. After creating two topics I used **docker-compose exec kafka kafka-topics --list --bootstrap-server localhost:9092** command to get a list of topics which I created before.

5. After this I used **docker-compose exec kafka kafka-topics --describe --topic topic2 --bootstrap-server localhost:9092** command to see the number of partitions in topic2.

Overall working: