Unit 4.2 Graded Assignment

Submitted By:

1. **Ali Nasir (2303.KHI.DEG.012)**
2. **Saif ur Rehman (2303.KHI.DEG.007)**

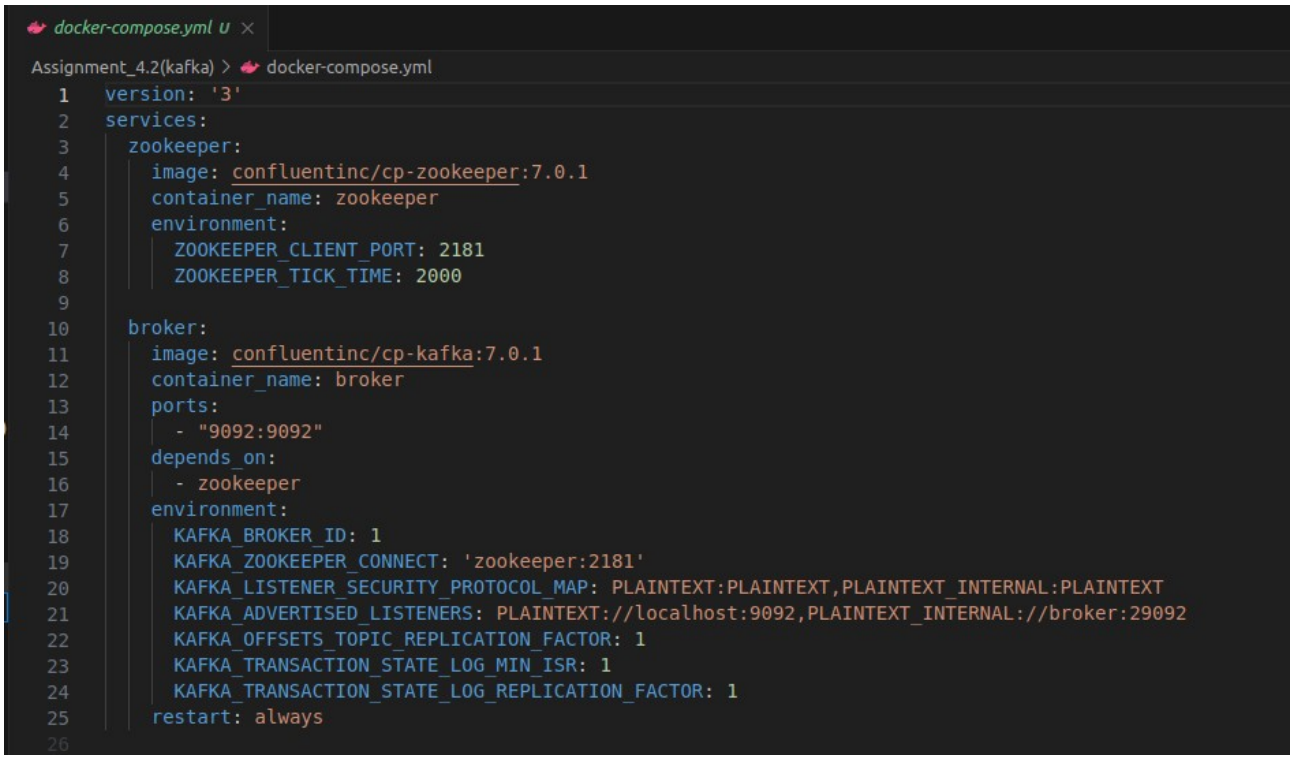
Problem Statement:

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:

→ The docker-compose.yml file contains which services we want to run the zookeeper, broker services. Zookeeper services is being run on on port 2181.



docker-compose.yml U

Assignment_4.2(kafka) > docker-compose.yml

```
version: '3'
services:
  zookeeper:
    image: confluentinc/cp-zookeeper:7.0.1
    container_name: zookeeper
    environment:
      ZOOKEEPER_CLIENT_PORT: 2181
      ZOOKEEPER_TICK_TIME: 2000

  broker:
    image: confluentinc/cp-kafka:7.0.1
    container_name: broker
    ports:
      - "9092:9092"
    depends_on:
      - zookeeper
    environment:
      KAFKA_BROKER_ID: 1
      KAFKA_ZOOKEEPER_CONNECT: 'zookeeper:2181'
      KAFKA_LISTENER_SECURITY_PROTOCOL_MAP: PLAINTEXT:PLAINTEXT,PLAINTEXT_INTERNAL:PLAINTEXT
      KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://localhost:9092,PLAINTEXT_INTERNAL://broker:29092
      KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 1
      KAFKA_TRANSACTION_STATE_LOG_MIN_ISR: 1
      KAFKA_TRANSACTION_STATE_LOG_REPLICATION_FACTOR: 1
    restart: always
```

→ In order to solve the assignment we have use the command in the code snippet we have also explained the code below.

docker exec -it broker: executes command within the docker container named broker the -it flag is used to attached to the interactive terminal.

commands.sh: commands will be read from a script file from "commands.sh."

--create --topic topica : creates a new topic in Apache Kafka with the specified topica.

--partitions 4: the number of partitions to be created for the topic 4.

--replication-factor 1: sets the replication of the topic 1.

--bootstrap-server broker:9092: This specifies the bootstrap server address for Kafka at 9092.

--list: lists the available topics in the Kafka cluster which is topica .

These commands, when executed within the "broker" container, perform operations related to topic creation, including specifying the topic name, number of partitions, and replication factor. The last command, --list, is used to retrieve a list of existing topics in the Kafka cluster.