TIC3001 Task 4

- Name: Ke Yule
- Student Number: A0211495H E0493826
- Github:

View the markdown version for better formatting at:
**

Task 4 - Pub-Sub Messaging

Set up the cluster

1. Start the kafka cluster using docker compoes

```
# docker-compose.yml provided in appendix cause length
docker-compose up -d
```

2. Verify its running

```
docker-compose ps
```

3. Create a new topic

```
docker-compose exec kafka-1 kafka-topics --create --topic test-topic --
partitions 1 --replication-factor 3 --bootstrap-server kafka-1:9092
```

4. Start a producer to send msgs

```
docker-compose exec kafka-1 kafka-console-producer --topic test-topic --
bootstrap-server kafka-1:9092
```

5. start a consumer

```
docker-compose exec kafka-1 kafka-console-consumer --topic test-topic --
from-beginning --bootstrap-server kafka-1:9092
```

6. Check the leader

```
docker-compose exec kafka-1 kafka-topics --describe --topic test-topic --
bootstrap-server kafka-1:9092
```

>  Show Leader

7. Kill the leader

```
docker-compose stop kafka-2
```

8. Show that the leader changed

```
docker-compose exec kafka-1 kafka-topics --describe --topic test-topic --
bootstrap-server kafka-1:9092
```

>  Show Leader Change

9. Check if topic still exists and we still can receive msgs

```
docker-compose exec kafka-3 kafka-topics --list --bootstrap-server kafka-
3:9094

docker-compose exec kafka-3 kafka-console-consumer --topic test-topic --
from-beginning --bootstrap-server kafka-3:9094
```

>  Show still can get msgs

Appendix

docker-compose.yml

```
version: '3'

services:
  zookeeper-1:
    image: zookeeper
    restart: always
    hostname: zookeeper-1
    ports:
      - '2181:2181'
    environment:
      ZOO_MY_ID: 1
      ZOO_SERVERS: server.1=zookeeper-1:2888:3888;2181 server.2=zookeeper-
2:2888:3888;2181 server.3=zookeeper-3:2888:3888;2181
    networks:
```

```
      - kafka-network

  zookeeper-2:
    image: zookeeper
    restart: always
    hostname: zookeeper-2
    environment:
      ZOO_MY_ID: 2
      ZOO_SERVERS: server.1=zookeeper-1:2888:3888;2181 server.2=zookeeper-
2:2888:3888;2181 server.3=zookeeper-3:2888:3888;2181
    networks:
      - kafka-network

  zookeeper-3:
    image: zookeeper
    restart: always
    hostname: zookeeper-3
    environment:
      ZOO_MY_ID: 3
      ZOO_SERVERS: server.1=zookeeper-1:2888:3888;2181 server.2=zookeeper-
2:2888:3888;2181 server.3=zookeeper-3:2888:3888;2181
    networks:
      - kafka-network

  kafka-1:
    image: confluentinc/cp-kafka:latest
    hostname: kafka-1
    ports:
      - '9092:9092'
    environment:
      KAFKA_BROKER_ID: 1
      KAFKA_ZOOKEEPER_CONNECT: zookeeper-1:2181,zookeeper-2:2181,zookeeper-3:2181
      KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://kafka-1:9092
      KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 3
    depends_on:
      - zookeeper-1
      - zookeeper-2
      - zookeeper-3
    networks:
      - kafka-network

  kafka-2:
    image: confluentinc/cp-kafka:latest
    hostname: kafka-2
    ports:
      - '9093:9093'
    environment:
      KAFKA_BROKER_ID: 2
      KAFKA_ZOOKEEPER_CONNECT: zookeeper-1:2181,zookeeper-2:2181,zookeeper-3:2181
      KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://kafka-2:9093
      KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 3
    depends_on:
      - zookeeper-1
      - zookeeper-2
```

```
      - zookeeper-3
    networks:
      - kafka-network

  kafka-3:
    image: confluentinc/cp-kafka:latest
    hostname: kafka-3
    ports:
      - '9094:9094'
    environment:
      KAFKA_BROKER_ID: 3
      KAFKA_ZOOKEEPER_CONNECT: zookeeper-1:2181,zookeeper-2:2181,zookeeper-3:2181
      KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://kafka-3:9094
      KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 3
    depends_on:
      - zookeeper-1
      - zookeeper-2
      - zookeeper-3
    networks:
      - kafka-network

networks:
  kafka-network:
    driver: bridge
```