TIC3001 Task 5

- Name: Ke Yule
- Student Number: A0211495H E0493826
- Github: https://github.com/keyule/3001-Task5

View the markdown version for better formatting at:
**

Task 5 - Caching

Set Up

- Database: PostgreSQL
- Caching: Redis
- API: Express.js

Code for the GET endpoint:

```javascript
app.get('/', async (req, res) => {
    redisClient.get('stuff' , async (error, stuff) => {
        if (error ) console.error(error)
        if (stuff != null){
            res.send(JSON.parse(stuff))
        } else {
            try{
                const data = await pool.query("SELECT * FROM morefields")
                res.send(data.rows)
                redisClient.set('stuff', JSON.stringify(data.rows))
            } catch {
                res.status(500).send('Error Occurs')
            }
        }
    })
})
```

Demonstration

1. Running Everything

```
docker-compose up --build -d
```

2. Check containers

```
PS C:\Users\Yule Ke\Desktop\Task5> docker ps
CONTAINER ID    IMAGE           COMMAND                  CREATED
STATUS          PORTS                    NAMES
539206eadc4f    postgres        "docker-entrypoint.s…"   5 seconds ago    Up 3
seconds    5432/tcp                 task5-db-1
3d314eb5c621    task5-app       "docker-entrypoint.s…"   5 seconds ago    Up 3
seconds    0.0.0.0:3000->3000/tcp   task5-app-1
4af7dcd18c5c    redis:alpine    "docker-entrypoint.s…"   5 seconds ago    Up 3
seconds    0.0.0.0:6379->6379/tcp   task5-redis-1
```

3. First GET request: 125ms

```
Task 4 >docker-compose exec kafka-1 kafka-topics --describe --topic test-topic --bootstrap-server kafka-1:9092
Topic: test-topic        TopicId: wIzddWTHTg6DCjzb3S28uA PartitionCount: 1        ReplicationFactor: 3    Configs:
        Topic: test-topic        Partition: 0    Leader: 2        Replicas: 2,1,3 Isr: 2,1,3
Task 4 >
```

4. Subsequent Get request: 41ms

```
Task 4 >docker-compose exec kafka-1 kafka-topics --describe --topic test-topic --bootstrap-server kafka-1:9092
Topic: test-topic        TopicId: wIzddWTHTg6DCjzb3S28uA PartitionCount: 1        ReplicationFactor: 3    Configs:
        Topic: test-topic        Partition: 0    Leader: 2        Replicas: 2,1,3 Isr: 2,1,3
Task 4 >
```

5. Flush the redis cache to see if it increases back up

```
PS C:\Users\Yule Ke\Desktop\Task5> docker compose exec redis redis-cli
127.0.0.1:6379> flushall
OK
```

6. Another Get Request: 117ms

```
Task 4 >docker-compose exec kafka-1 kafka-topics --describe --topic test-topic --bootstrap-server kafka-1:9092
Topic: test-topic        TopicId: wIzddWTHTg6DCjzb3S28uA PartitionCount: 1        ReplicationFactor: 3    Configs:
        Topic: test-topic        Partition: 0    Leader: 2        Replicas: 2,1,3 Isr: 2,1,3
Task 4 >
```

Appendix

docker-compose.yml

```yaml
version: "3"
services:
  redis:
    image: 'redis:alpine'
    ports:
      - '6379:6379'
  db:
    image: postgres
    environment:
      POSTGRES_PASSWORD: password123
      POSTGRES_USER: user123
      POSTGRES_DB: db123
```

```
        volumes:
            - ./init.sql:/docker-entrypoint-initdb.d/init.sql
    app:
        build: .
        ports:
            - 3000:3000
```

        volumes:
            - ./init.sql:/docker-entrypoint-initdb.d/init.sql
    app:
        build: .
        ports:
            - 3000:3000