Student Name: **A Z Hasnain Kabir**
Student ID: **200042102**

CSE 4410
Database Management Systems II
Lab 7

Introduction

In this lab, we are going to learn about NoSQL using MongoDB

Scenario

- Think of a scenario of social media platform that allows users to create profiles, post, follow other users, and like posts. Generally, the user provides his basic information for example name, email, password (here you don't have to think of hashing password), phone no, date of birth, address, and profile creation date. Other than that users may add their working status (like what is he/she doing currently, and how long they are doing this), bios, hobbies, and so on. Any user can post content and anybody can like and comment on that post. Additionally, a user can follow others.

Task 1

Create necessary collections with respective attributes.

Explanation

I have created the necessary collections in mongoDB compass. The .json file will be attached with the submission.

Task 2

Insertion

(a) An entry of a user with only email, name and password.

(b) Multiple entries of users with basic info and hobbies.

(c) An entry of a user with basic info and working status.

(d) Add multiple followers for multiple users.

(e) Multiple posts with content, creation time, like and the user id who likes.

(f) Add multiple comments for multiple posts.

Explanation

In MongoDB, the operation insertOne will insert a document inside a collection. We have to write the queries in json format. Also, the operation insertMany inserts multiple documents, and for update of any document, we need two things, one is filter and another is the update. In the updateMany function, we have set filter of "post id" to 1 indicating that only posts with the id of 1 will be updated. The set part updates the document's exact key and value.

Code

```
//2A
db.user.insertOne({
    "email": "nibir@iut-dhaka.edu",
    "password" : "Nibir1234",
    "name": "Nibir"
});
```

```
//2B
db.user.insertMany([
    {
        "email": "tahlil@iut-dhaka.edu",
        "password" : "tahlil123",
        "name" : "K M Tahlil Mahfuz Faruk",
        "phone no" : "012333333",
        "date of birth" : new ISODate("2002-05-18"),
        "profile created" : new Date(),
        "hobbies" : ["being lazy", "being racist"]
    },
    {
        "email": "Mamun@iut-dhaka.edu",
        "password" : "mamun123",
        "name" : "Mamunur Rahman",
        "phone no" : "012333333",
        "date of birth" : new ISODate("2002-03-21"),
        "profile created" : new Date(),
        "hobbies" : ["Reddit", "Contemplating life"]
    }
]);

//2C
db.user.insertOne({
    "email": "muaz@iut-dhaka.edu",
    "password" : "Muaz1234",
    "name": "Muazul Islam",
    "phone no" : "012333333",
    "date of birth" : new ISODate("2002-02-21"),
    "working status" : "racism"
});
```

```
//2D
db.follower.insertMany([
    {
        "user email" : "nibir@iut-dhaka.edu",
        "follower emails" :["tahlil@iut-dhaka.edu", "muaz@iut-dhaka.edu"]
    },
    {
        "user email" : "muaz@iut-dhaka.edu",
        "follower emails" :["tahlil@iut-dhaka.edu"]
    }
]);
```

```
//2E
db.post.insertMany([
    {
        "post id" : "1",
        "title" : "Chilling with da bois",
        "time" : new ISODate("2023-03-10"),
        "likes" : ["nibir@iut-dhaka.edu", "tahlil@iut-dhaka.edu"]
    },
    {
        "post id" : "1",
        "title" : "Chilling alone",
        "time" :  new ISODate("2023-03-12"),
        "likes" : ["Mamun@iut-dhaka.edu"]
    }
]);
db.post.insertMany([
    {
        "post id" : "2",
        "title" : "I hate life",
        "time" :  new ISODate("2023-03-14"),
        "likes" : ["nibir@iut-dhaka.edu", "tahlil@iut-dhaka.edu"]
    },
    {
        "post id" : "2",
        "title" : "I hate me",
        "time" : new ISODate("2023-03-14"),
        "likes" : ["Mamun@iut-dhaka.edu"]
    }
]);
db.post.insertOne(
    {
        "post id" : "3",
        "title" : "Covfefe",
        "time" : new ISODate("2022-04-14"),
        "likes" : ["nibir@iut-dhaka.edu", "tahlil@iut-dhaka.edu"]
    }
);
```

```
//2F

db.post.updateMany(
    {"post id" : "1"},
    {$set: {"comments" : ["Why was I not invited?","Damn"]} }
);

db.post.updateMany(
    {"post id" : "2"},
    {$set: {"comments" : ["Dont hate","Damn"]} }
);

db.post.updateMany(
    {"post id" : "3"},
    {$set: {"comments" : ["say what?","Damn"]} }
);
```

Task 3

Explanation

For retrieving values, in mongoDB, I have taken help of a function called find. In 3A, after finding the values we sort it using a parameter called time. It can be noticed that the value of time is set to -1 indicating descending order. For 3B, we have used the find() function but this time we have used a filter to find exact values.

Data Retrieving

(a) Display the most recent to oldest posts along with their poster.

(b) Show all the posts that were created in the last 24 hours.

(c) Show all the users who have more than 3 followers.

(d) Show all the users who are following more than 3 users.

Code

```
//3A
db.post.find().sort({"time": -1});

//3B
db.post.find({"time": {$gt : new Date( Date.now() - 24*60*60 * 1000)}});
```

3C

```
db.follower.aggregate([
    {
      $match:{
        "follower emails.1":{$exists : true}
      }
    },
    { $project: { _id: 1, "user email": 1} }
    ]);
```