package java\_mongo;

import java.util.Scanner;

import org.bson.Document;

import com.mongodb.MongoClient;

import com.mongodb.MongoClientURI;

import com.mongodb.client.FindIterable;

import com.mongodb.client.MongoCollection;

import com.mongodb.client.MongoCursor;

import com.mongodb.client.MongoDatabase;

public class DatabaseEx {

   private static MongoClient mongoClient;

   private static MongoCollection<Document> collection;

   public static void main(String[] args) {

       String uri = "mongodb://te31340:te31340@10.10.13.97/te31340\_db";

       mongoClient = new MongoClient(new MongoClientURI(uri));

       MongoDatabase db = mongoClient.getDatabase("te31340\_db");

       collection = db.getCollection("users");

       Scanner scanner = new Scanner(System.in);

       int choice;

       do {

           System.out.println("Menu:");

           System.out.println("1. Create User");

           System.out.println("2. Read User");

           System.out.println("3. Update User");

           System.out.println("4. Delete User");

           System.out.println("5. Exit");

           System.out.print("Enter your choice: ");

           choice = scanner.nextInt();

           scanner.nextLine();

           switch (choice) {

               case 1:

                   System.out.print("Enter ID: ");

                   int id = scanner.nextInt();

                   scanner.nextLine();

                   System.out.print("Enter Name: ");

                   String name = scanner.nextLine();

                   System.out.print("Enter Email: ");

                   String email = scanner.nextLine();

                   createUser(id, name, email);

                   break;

               case 2:

                   readAllUsers();

                   break;

               case 3:

                   System.out.print("Enter ID to update: ");

                   int updateId = scanner.nextInt();

                   scanner.nextLine();

                   System.out.print("Enter new Email: ");

                   String newEmail = scanner.nextLine();

                   System.out.print("Enter new Email: ");

                   String newname = scanner.nextLine();

                   updateUser(updateId, newEmail , newname);

                   break;

               case 4:

                   System.out.print("Enter ID to delete: ");

                   int deleteId = scanner.nextInt();

                   deleteUser(deleteId);

                   break;

               case 5:

                   System.out.println("Exiting...");

                   break;

               default:

                   System.out.println("Invalid choice. Please try again.");

           }

       } while (choice != 5);

       mongoClient.close();

       scanner.close();

   }

   private static void createUser(int id, String name, String email) {

       Document user = new Document("id", id).append("name", name).append("email", email);

       collection.insertOne(user);

       System.out.println("User created successfully.");

   }

   private static void readAllUsers() {

       FindIterable<Document> users = collection.find();

       boolean userFound = false;

       for (Document user : users) {

           System.out.println("User Found: " + user.toJson());

           userFound = true;

       }

       if (!userFound) {

           System.out.println("No users found.");

       }

   }

   private static void updateUser(int id, String email, String name) {

       Document updateDocument = new Document("$set", new Document("email", email).append("name", name));

       long updatedCount = collection.updateOne(new Document("id", id), updateDocument).getModifiedCount();

       if (updatedCount > 0) {

           System.out.println("User updated successfully.");

       } else {

           System.out.println("User not found.");

       }

   }

   private static void deleteUser(int id) {

       long deletedCount = collection.deleteOne(new Document("id", id)).getDeletedCount();

       if (deletedCount > 0) {

           System.out.println("User deleted successfully.");

       } else {

           System.out.println("User not found.");

       }

   }

}

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

 <modelVersion>4.0.0</modelVersion>

 <groupId>java\_mongo</groupId>

 <artifactId>java\_mongo</artifactId>

 <version>0.0.1-SNAPSHOT</version>

 <dependencies>

   <dependency>

     <groupId>org.mongodb</groupId>

     <artifactId>mongo-java-driver</artifactId>

     <version>3.12.0</version>

   </dependency>

 </dependencies>

</project>