import java.sql.\*;

public class lab7 {

static final String DB\_URL = "jdbc:mysql://localhost/";

static final String USER = "root";

static final String PASSWORD = "root";

static final String DATABASE\_NAME = "ESM";

public static void main(String[] args) {

try {

// Load the JDBC driver

Class.forName("com.mysql.cj.jdbc.Driver");

// Open a connection

Connection connection = DriverManager.getConnection(DB\_URL, USER, PASSWORD);

// Create a statement

Statement statement = connection.createStatement();

// Create a new database

createDatabase(statement);

// Switch to the new database

statement.executeUpdate("USE " + DATABASE\_NAME);

// Initialize the database with some employee records

initializeDatabase(statement);

// Perform CRUD operations

// (You can replace these with your specific operations)

// Read (Retrieve) operation

readData(statement);

// Create operation

createData(statement, "John Doe", 60000.0, "IT");

readData(statement);

// Update operation

updateData(statement, 1, 65000.0);

readData(statement);

// Delete operation

deleteData(statement, 3);

readData(statement);

// Close connections

statement.close();

connection.close();

} catch (Exception e) {

e.printStackTrace();

}

}

private static void createDatabase(Statement statement) throws SQLException {

statement.executeUpdate("CREATE DATABASE IF NOT EXISTS " + DATABASE\_NAME);

}

private static void initializeDatabase(Statement statement) throws SQLException {

statement.executeUpdate("CREATE TABLE IF NOT EXISTS employees (" +

"id INT AUTO\_INCREMENT PRIMARY KEY," +

"name VARCHAR(100) ," +

"salary DOUBLE ," +

"department VARCHAR(50) )");

// Insert some initial employee records

statement.executeUpdate("INSERT INTO employees (name, salary, department) VALUES " +

"('Alice Smith', 70000.0, 'HR'), " +

"('Bob Johnson', 80000.0, 'Finance'), " +

"('Charlie Brown', 60000.0, 'IT'), " +

"('David Wilson', 75000.0, 'Marketing'), " +

"('Eva Davis', 90000.0, 'Engineering'), " +

"('Frank Miller', 65000.0, 'Sales'), " +

"('Grace Taylor', 72000.0, 'Finance'), " +

"('Henry Harris', 68000.0, 'IT'), " +

"('Ivy Jones', 95000.0, 'Engineering'), " +

"('Jack Anderson', 85000.0, 'HR')");

}

private static void readData(Statement statement) throws SQLException {

ResultSet resultSet = statement.executeQuery("SELECT \* FROM employees");

while (resultSet.next()) {

System.out.println("ID: " + resultSet.getInt("id") +

", Name: " + resultSet.getString("name") +

", Salary: " + resultSet.getDouble("salary") +

", Department: " + resultSet.getString("department"));

}

resultSet.close();

}

private static void createData(Statement statement, String name, double salary, String department) throws SQLException {

statement.executeUpdate("INSERT INTO employees (name, salary, department) VALUES " +

"('" + name + "', " + salary + ", '" + department + "')");

System.out.println("\n\nEmployee record created successfully.");

}

private static void updateData(Statement statement, int id, double newSalary) throws SQLException {

statement.executeUpdate("UPDATE employees SET salary = " + newSalary + " WHERE id = " + id);

System.out.println("\n\nEmployee record updated successfully.");

}

private static void deleteData(Statement statement, int id) throws SQLException {

statement.executeUpdate("DELETE FROM employees WHERE id = " + id);

System.out.println("\n\nEmployee record deleted successfully.");

}

}







