Name:- Yash Dhumal
Roll no. :- 13373

Practical no. 9

```
pakage DBMSPracticals;

import java.util.*;

import java.sql.*;

public class StudentDatabase {

   public static void main(String[] args) {

try {

      Connection connection =

DriverManager.getConnection("jdbc:mysql://localhost:3306/practical9","root", "root");
createTable(connection);

         Scanner scanner = new Scanner(System.in);

         while (true) {

            System.out.println("Choose an operation:");

            System.out.println("1. Insert");

            System.out.println("2. Update");

            System.out.println("3. Delete");

            System.out.println("4. Display");

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

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

            int choice = scanner.nextInt();
scanner.nextLine(); // Consume newline

            switch (choice) {
case 1:

                  System.out.print("Enter roll no: ");
int rollNo = scanner.nextInt();                  scanner.nextLine();
// Consume newline                  System.out.print("Enter
student name: ");

                  String    name    =    scanner.nextLine();
System.out.print("Enter age: ");                  int age =
scanner.nextInt();
                  insertRecord(connection, rollNo, name, age);
break;                case 2:
```

```
                System.out.print("Enter roll no to update: ");
int rollNoToUpdate = scanner.nextInt();
scanner.nextLine(); // Consume newline
System.out.print("Enter new student name: ");

String newName = scanner.nextLine();
System.out.print("Enter new age: ");

 int newAge = scanner.nextInt();

                updateRecord(connection, rollNoToUpdate, newName, newAge);
break;

case 3:

                System.out.print("Enter roll no to delete: ");
int rollNoToDelete = scanner.nextInt();

                deleteRecord(connection, rollNoToDelete);
break;

 case 4:

                displayRecords(connection);
break;

case 5:

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

default:

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

            }

         }

      } catch (SQLException e) {          e.printStackTrace();

      }

   }


   private static void createTable(Connection connection) throws SQLException

String sql = "CREATE TABLE IF NOT EXISTS students (roll_no INT PRIMARY

KEY, name VARCHAR(255), age INT)";

      try (PreparedStatement stmt = connection.prepareStatement(sql)) {
stmt.executeUpdate();

      }

   }
```

```
    private static void insertRecord(Connection connection, int rollNo, String name, int age) throws
SQLException {
        String sql = "INSERT INTO students (roll_no, name, age) VALUES (?, ?, ?)";
Try
 (PreparedStatement stmt = connection.prepareStatement(sql)) {
stmt.setInt(1, rollNo);          stmt.setString(2, name);          stmt.setInt(3, age);
stmt.executeUpdate();
            System.out.println("Record inserted successfully!");
        }
    }

    private static void updateRecord(Connection connection, int rollNo, String name, int age) throws
SQLException {
        String sql = "UPDATE students SET name = ?, age = ? WHERE roll_no = ?";
        try (PreparedStatement stmt = connection.prepareStatement(sql)) {
stmt.setString(1, name);
 stmt.setInt(2, age);
stmt.setInt(3, rollNo);
            int rowsUpdated = stmt.executeUpdate();
            if (rowsUpdated > 0) {
                System.out.println("Record updated successfully!");          }
else {
                System.out.println("Record not found.");
            }
        }
    }

    private static void deleteRecord(Connection connection, int rollNo) throws SQLException {
        String sql = "DELETE FROM students WHERE roll_no = ?";
     try
(PreparedStatement stmt = connection.prepareStatement(sql)) {
stmt.setInt(1, rollNo);
            int rowsDeleted = stmt.executeUpdate();
```

```
            if (rowsDeleted > 0) {
                System.out.println("Record deleted successfully!");            }
else {
                System.out.println("Record not found.");
            }
        }
    }

    private static void displayRecords(Connection connection) throws SQLException {
String sql = "SELECT * FROM students";
        try (PreparedStatement stmt = connection.prepareStatement(sql);
             ResultSet resultSet = stmt.executeQuery()) {
System.out.println("Roll No\tName\tAge");
    While
 (resultSet.next()) {
                int rollNo = resultSet.getInt("roll_no");
String name = resultSet.getString("name");
                int age = resultSet.getInt("age");
                System.out.println(rollNo + "\t" + name + "\t" + age);
            }
        }
    }
}
```

Output :

```
Choose an operation:
1. Insert 2.
Update
3. Delete
4. Display
5. Exit
Enter your choice: 1
Enter roll no: 1
Enter student name: Anand
Enter age: 19
Record inserted successfully!
Choose an operation:
1. Insert 2.
Update
3. Delete
4. Display
5. Exit
Enter your choice: 2
Enter roll no to update: 1
Enter new student name: Bhagat
Enter new age: 20
Record updated successfully!
Choose an operation:
1. Insert 2.
Update
3. Delete
4. Display
5. Exit
Enter your choice: 4
Roll No      Name Age 1
       Bhagat 20
Choose an operation:
1. Insert 2.
Update
3. Delete
4. Display
5. Exit
Enter your choice: 3 Enter
roll no to delete: 1 Record
deleted successfully!
Choose an operation:
1. Insert 2.
Update
3. Delete
4. Display
5. Exit
Enter your choice:
```