	SCTR's PUNE INSTITUTE OF COMPUTER TECHNOLOGY PUNE - 411043		
	Department of Electronics & Telecommunication Engineering (E&TCE)		
	ASSESMENT YEAR: 2024-2025	CLASS: TE	BATCH:- N7
	SUBJECT: Advanced JAVA Programming		
	Assignment No: 4	Roll No: 32402	Date: 05/02/2025

Programmer Name: Sahil Manoj Amrutkar

Batch: N7

Problem Statement: Create a database and perform the operation as insert and retrieve the data from the database using JDBC

INPUT :

```java
package dbConnection;

import java.sql.*;
import java.util.Scanner;

public class Exp4B {
    private Connection con;
    private Scanner scanner;

    public Exp4B() {
        scanner = new Scanner(System.in);
    }

    public static void main(String[] args) {
        Exp4B obj = new Exp4B();
        try {
            obj.getConnection();
            obj.createTable();
            obj.menu();
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            obj.closeConnection();
        }
    }

    public void getConnection() throws Exception {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            String url = "jdbc:mysql://localhost:3306/college";
            String username = "root";
            String password = "Sahil@2004";
            con = DriverManager.getConnection(url, username, password);
            System.out.println("Connection Established Successfully");
```

```java
      } catch (ClassNotFoundException | SQLException e) {
         e.printStackTrace();
      }
   }

   public void createTable() {
      String query = "CREATE TABLE IF NOT EXISTS students ("
            + "rollNo VARCHAR(10) PRIMARY KEY, "
            + "name VARCHAR(50), "
            + "subject VARCHAR(50), "
            + "marks INT)";
      try (Statement stmt = con.createStatement()) {
         stmt.executeUpdate(query);
         System.out.println("Table 'students' is ready.");
      } catch (SQLException e) {
         e.printStackTrace();
      }
   }

   public void menu() {
      while (true) {
         System.out.println("\n Select an operation:");
         System.out.println("1. Insert Student");
         System.out.println("2. Read Students");
         System.out.println("3. Update Student");
         System.out.println("4. Delete Student");
         System.out.println("5. Exit");
         System.out.print("Enter choice: ");

         int choice = scanner.nextInt();
         scanner.nextLine();

         switch (choice) {
            case 1:
               insertStudent();
               break;
            case 2:
               readStudents();
               break;
            case 3:
               updateStudent();
               break;
            case 4:
               deleteStudent();
               break;
```

```java
        case 5:
          System.out.println("Exiting...");
          return;
        default:
          System.out.println("Invalid choice. Try again.");
      }
    }
  }

  public void insertStudent() {
    Derived student = new Derived();
    student.getData(scanner);

    String query = "INSERT INTO students (rollNo, name, subject, marks) VALUES (?, ?, ?, ?)";
    try (PreparedStatement pstmt = con.prepareStatement(query)) {
      pstmt.setString(1, student.getRollNo());
      pstmt.setString(2, student.getName());
      pstmt.setString(3, student.getSubject());
      pstmt.setInt(4, student.getMarks());

      int rows = pstmt.executeUpdate();
      if (rows > 0) {
        System.out.println("Student added successfully.");
      }
    } catch (SQLException e) {
      e.printStackTrace();
    }
  }

  public void readStudents() {
    String query = "SELECT * FROM students";
    try (Statement stmt = con.createStatement();
      ResultSet rs = stmt.executeQuery(query)) {

      System.out.println("\n2.Student Records:");
      while (rs.next()) {
        System.out.println(
            "Roll No: " + rs.getString("rollNo") +
            ", Name: " + rs.getString("name") +
            ", Subject: " + rs.getString("subject") +
            ", Marks: " + rs.getInt("marks")
        );
      }
    } catch (SQLException e) {
      e.printStackTrace();
```

```java
        }
    }

    public void updateStudent() {
        System.out.print("Enter Roll No of student to update: ");
        String rollNo = scanner.nextLine();

        System.out.print("Enter new Marks: ");
        int newMarks = scanner.nextInt();
        scanner.nextLine();

        String query = "UPDATE students SET marks = ? WHERE rollNo = ?";
        try (PreparedStatement pstmt = con.prepareStatement(query)) {
            pstmt.setInt(1, newMarks);
            pstmt.setString(2, rollNo);

            int rows = pstmt.executeUpdate();
            if (rows > 0) {
                System.out.println("Student record updated.");
            } else {
                System.out.println("Student not found.");
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    public void deleteStudent() {
        System.out.print("Enter Roll No of student to delete: ");
        String rollNo = scanner.nextLine();

        String query = "DELETE FROM students WHERE rollNo = ?";
        try (PreparedStatement pstmt = con.prepareStatement(query)) {
            pstmt.setString(1, rollNo);

            int rows = pstmt.executeUpdate();
            if (rows > 0) {
                System.out.println("Student record deleted.");
            } else {
                System.out.println("Student not found.");
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
```

```java
    public void closeConnection() {
        try {
            if (con != null) {
                con.close();
                System.out.println("Connection closed.");
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
        scanner.close();
    }
}

class Base {
    private String rollNo;
    private String name;

    public void getData(Scanner scanner) {
        System.out.print("Enter Roll No: ");
        this.rollNo = scanner.nextLine();
        System.out.print("Enter Name: ");
        this.name = scanner.nextLine();
    }

    public String getRollNo() {
        return rollNo;
    }

    public String getName() {
        return name;
    }
}

class Derived extends Base {
    private String subject;
    private int marks;

    public void getData(Scanner scanner) {
        super.getData(scanner);
        System.out.print("Enter Subject: ");
        this.subject = scanner.nextLine();
        System.out.print("Enter Marks: ");
        this.marks = scanner.nextInt();
        scanner.nextLine();
```

```
    }

    public String getSubject() {
        return subject;
    }

    public int getMarks() {
        return marks;
    }
}
```

OUTPUT :

Connection Established Successfully
Table 'students' is ready.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 1
Enter Roll No: 32402
Enter Name: Sahil Amrutkar
Enter Subject: Maths
Enter Marks: 96
Student added successfully.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 1
Enter Roll No: 32406
Enter Name: Rahul Bhale

Enter Subject: AJP
Enter Marks: 91
Student added successfully.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 1
Enter Roll No: 32407
Enter Name: Ashwini Borse
Enter Subject: AJP
Enter Marks: 87
Student added successfully.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 1
Enter Roll No: 32250
Enter Name: Om Palwe
Enter Subject: SME
Enter Marks: 38
Student added successfully.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 1
Enter Roll No: 32221

Enter Name: Shreyash Abhange
Enter Subject: Science
Enter Marks: 100
Student added successfully.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 2

2.Student Records:
Roll No: 32221, Name: Shreyash Abhange, Subject: Science, Marks: 100
Roll No: 32250, Name: Om Palwe, Subject: SME, Marks: 38
Roll No: 32402, Name: Sahil Amrutkar, Subject: Maths, Marks: 96
Roll No: 32406, Name: Rahul Bhale, Subject: AJP, Marks: 91
Roll No: 32407, Name: Ashwini Borse, Subject: AJP, Marks: 87

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 3
Enter Roll No of student to update: 32221
Enter new Marks: 94
Student record updated.

 Select an operation:
1. Insert Student
2. Read Students
3. Update Student
4. Delete Student
5. Exit
Enter choice: 2

2.Student Records:

Roll No: 32221, Name: Shreyash Abhange, Subject: Science, Marks: 94

Roll No: 32250, Name: Om Palwe, Subject: SME, Marks: 38

Roll No: 32402, Name: Sahil Amrutkar, Subject: Maths, Marks: 96

Roll No: 32406, Name: Rahul Bhale, Subject: AJP, Marks: 91

Roll No: 32407, Name: Ashwini Borse, Subject: AJP, Marks: 87

Select an operation:

1. Insert Student

2. Read Students

3. Update Student

4. Delete Student

5. Exit

Enter choice: 4

Enter Roll No of student to delete: 32250

Student record deleted.

Select an operation:

1. Insert Student

2. Read Students

3. Update Student

4. Delete Student

5. Exit

Enter choice: 2

2.Student Records:

Roll No: 32221, Name: Shreyash Abhange, Subject: Science, Marks: 94

Roll No: 32402, Name: Sahil Amrutkar, Subject: Maths, Marks: 96

Roll No: 32406, Name: Rahul Bhale, Subject: AJP, Marks: 91

Roll No: 32407, Name: Ashwini Borse, Subject: AJP, Marks: 87

Select an operation:

1. Insert Student

2. Read Students

3. Update Student

4. Delete Student

5. Exit

Enter choice: 5
Exiting...
Connection closed.

SQL DATABASE After Performing all operations :-