Task 13

1.Source code

```
import java.sql.*;

/*

 * Task 13: JDBC – MySQL Connectivity

 */

public class JdbcDemo {


   // DB credentials

   static final String URL = "jdbc:mysql://localhost:3306/internship_db";

   static final String USER = "root";

   static final String PASSWORD = "your_password";


   public static void main(String[] args) {


      Connection con = null;


      try {

         // 1. Load JDBC Driver

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


         // 2. Establish connection

         con = DriverManager.getConnection(URL, USER, PASSWORD);

         System.out.println("Database connected successfully.");
```

```
// ---------- INSERT ----------
String insertSQL = "INSERT INTO students(name, email, marks) VALUES (?, ?, ?)";
PreparedStatement psInsert = con.prepareStatement(insertSQL);
psInsert.setString(1, "Akshu");
psInsert.setString(2, "akshu@gmail.com");
psInsert.setInt(3, 90);
psInsert.executeUpdate();
System.out.println("Record inserted.");

// ---------- READ ----------
String selectSQL = "SELECT * FROM students";
PreparedStatement psSelect = con.prepareStatement(selectSQL);
ResultSet rs = psSelect.executeQuery();

System.out.println("\n--- Student Records ---");
while (rs.next()) {
   System.out.println(
         rs.getInt("id") + " | " +
         rs.getString("name") + " | " +
         rs.getString("email") + " | " +
         rs.getInt("marks")
   );
}

// ---------- UPDATE ----------
```

```
    String updateSQL = "UPDATE students SET marks=? WHERE name=?";
    PreparedStatement psUpdate = con.prepareStatement(updateSQL);
    psUpdate.setInt(1, 95);
    psUpdate.setString(2, "Akshu");
    psUpdate.executeUpdate();
    System.out.println("\nRecord updated.");

    // ---------- DELETE ----------
    String deleteSQL = "DELETE FROM students WHERE name=?";
    PreparedStatement psDelete = con.prepareStatement(deleteSQL);
    psDelete.setString(1, "Akshu");
    psDelete.executeUpdate();
    System.out.println("Record deleted.");

}
catch (ClassNotFoundException e) {
    System.out.println("JDBC Driver not found.");
}
catch (SQLException e) {
    System.out.println("Database error: " + e.getMessage());
}
finally {
    try {
        if (con != null) {
            con.close();
```

```
            System.out.println("Database connection closed.");
          }
        } catch (SQLException e) {
          System.out.println("Error closing connection.");
        }
      }
    }
}
```

2.Output