1. Product CRUD Application (with Transaction Handling)

```
// ProductCRUDApp.java
import java.sql.*;
import java.util.Scanner;

public class ProductCRUDApp {
    static final String URL = "jdbc:mysql://localhost:3306/your_database";
    static final String USER = "root";
    static final String PASS = "password";

    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(URL, USER, PASS);
             Scanner scanner = new Scanner(System.in)) {

            conn.setAutoCommit(false);

            while (true) {
                System.out.println("\n--- Product Management Menu ---");
                System.out.println("1. Add Product\n2. View Products\n3. Update Product\n4. Delete
Product\n5. Exit");
                System.out.print("Enter choice: ");
                int choice = scanner.nextInt();

                switch (choice) {
                    case 1:
                        System.out.print("Enter ProductID: ");
                        int id = scanner.nextInt();
                        scanner.nextLine();
                        System.out.print("Enter Product Name: ");
                        String name = scanner.nextLine();
                        System.out.print("Enter Price: ");
                        double price = scanner.nextDouble();
                        System.out.print("Enter Quantity: ");
                        int qty = scanner.nextInt();

                          try (PreparedStatement stmt = conn.prepareStatement("INSERT INTO Product
VALUES (?, ?, ?, ?)")) {
                            stmt.setInt(1, id);
                            stmt.setString(2, name);
                            stmt.setDouble(3, price);
                            stmt.setInt(4, qty);
                            stmt.executeUpdate();
                            conn.commit();
                            System.out.println("Product added successfully.");
                        } catch (SQLException e) {
                            conn.rollback();
                            System.out.println("Error: " + e.getMessage());
                        }
                        break;
                    case 2:
                        try (Statement stmt = conn.createStatement();
                             ResultSet rs = stmt.executeQuery("SELECT * FROM Product")) {
                            System.out.println("ProductID | ProductName | Price | Quantity");
                            while (rs.next()) {
                                System.out.printf("%d | %s | %.2f | %d\n",
                                    rs.getInt("ProductID"), rs.getString("ProductName"),
                                    rs.getDouble("Price"), rs.getInt("Quantity"));
                            }
```

```
                        }
                        break;
                    case 3:
                        System.out.print("Enter ProductID to update: ");
                        int pid = scanner.nextInt();
                        System.out.print("Enter new Price: ");
                        double newPrice = scanner.nextDouble();
                        System.out.print("Enter new Quantity: ");
                        int newQty = scanner.nextInt();

                        try (PreparedStatement stmt = conn.prepareStatement(
                                "UPDATE Product SET Price=?, Quantity=? WHERE ProductID=?")) {
                            stmt.setDouble(1, newPrice);
                            stmt.setInt(2, newQty);
                            stmt.setInt(3, pid);
                            stmt.executeUpdate();
                            conn.commit();
                            System.out.println("Product updated successfully.");
                        } catch (SQLException e) {
                            conn.rollback();
                            System.out.println("Error: " + e.getMessage());
                        }
                        break;
                    case 4:
                        System.out.print("Enter ProductID to delete: ");
                        int delId = scanner.nextInt();
                          try (PreparedStatement stmt = conn.prepareStatement("DELETE FROM Product
WHERE ProductID=?")) {
                            stmt.setInt(1, delId);
                            stmt.executeUpdate();
                            conn.commit();
                            System.out.println("Product deleted successfully.");
                        } catch (SQLException e) {
                            conn.rollback();
                            System.out.println("Error: " + e.getMessage());
                        }
                        break;
                    case 5:
                        System.out.println("Exiting.");
                        return;
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

2. Student Management - Model (Student.java)

```
// Student.java
public class Student {
    private int studentID;
    private String name;
    private String department;
    private int marks;

    public Student(int studentID, String name, String department, int marks) {
        this.studentID = studentID;
```

```
        this.name = name;
        this.department = department;
        this.marks = marks;
    }

    public int getStudentID() { return studentID; }
    public String getName() { return name; }
    public String getDepartment() { return department; }
    public int getMarks() { return marks; }
}
```

3. Student Management - Controller (StudentController.java)

```
// StudentController.java
import java.sql.*;
import java.util.ArrayList;
import java.util.List;

public class StudentController {
    private Connection conn;

    public StudentController() throws SQLException {
          conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/your_database", "root",
"password");
    }

    public void addStudent(Student s) throws SQLException {
        String sql = "INSERT INTO Student VALUES (?, ?, ?, ?)";
        try (PreparedStatement stmt = conn.prepareStatement(sql)) {
            stmt.setInt(1, s.getStudentID());
            stmt.setString(2, s.getName());
            stmt.setString(3, s.getDepartment());
            stmt.setInt(4, s.getMarks());
            stmt.executeUpdate();
        }
    }

    public List<Student> getAllStudents() throws SQLException {
        List<Student> list = new ArrayList<>();
        String sql = "SELECT * FROM Student";
        try (Statement stmt = conn.createStatement();
             ResultSet rs = stmt.executeQuery(sql)) {
            while (rs.next()) {
                list.add(new Student(
                    rs.getInt("StudentID"),
                    rs.getString("Name"),
                    rs.getString("Department"),
                    rs.getInt("Marks")
                ));
            }
        }
        return list;
    }

    public void updateStudentMarks(int id, int newMarks) throws SQLException {
        String sql = "UPDATE Student SET Marks=? WHERE StudentID=?";
        try (PreparedStatement stmt = conn.prepareStatement(sql)) {
            stmt.setInt(1, newMarks);
            stmt.setInt(2, id);
```

```
            stmt.executeUpdate();
        }
    }

    public void deleteStudent(int id) throws SQLException {
        String sql = "DELETE FROM Student WHERE StudentID=?";
        try (PreparedStatement stmt = conn.prepareStatement(sql)) {
            stmt.setInt(1, id);
            stmt.executeUpdate();
        }
    }
}
```

4. Student Management - View (StudentApp.java)

```
// StudentApp.java
import java.util.Scanner;

public class StudentApp {
    public static void main(String[] args) {
        try (Scanner scanner = new Scanner(System.in)) {
            StudentController controller = new StudentController();

            while (true) {
                System.out.println("\n--- Student Management Menu ---");
                 System.out.println("1. Add Student\n2. View Students\n3. Update Student\n4. Delete
Student\n5. Exit");
                System.out.print("Enter choice: ");
                int choice = scanner.nextInt();

                switch (choice) {
                    case 1:
                        System.out.print("Enter StudentID: ");
                        int id = scanner.nextInt();
                        scanner.nextLine();
                        System.out.print("Enter Name: ");
                        String name = scanner.nextLine();
                        System.out.print("Enter Department: ");
                        String dept = scanner.nextLine();
                        System.out.print("Enter Marks: ");
                        int marks = scanner.nextInt();

                        Student s = new Student(id, name, dept, marks);
                        controller.addStudent(s);
                        System.out.println("Student added.");
                        break;

                    case 2:
                        for (Student st : controller.getAllStudents()) {
                            System.out.printf("%d | %s | %s | %d\n",
                                              st.getStudentID(), st.getName(), st.getDepartment(),
st.getMarks());
                        }
                        break;

                    case 3:
                        System.out.print("Enter StudentID to update: ");
                        int sid = scanner.nextInt();
                        System.out.print("Enter new Marks: ");
```

```
                    int newMarks = scanner.nextInt();
                    controller.updateStudentMarks(sid, newMarks);
                    System.out.println("Student updated.");
                    break;

                case 4:
                    System.out.print("Enter StudentID to delete: ");
                    int del = scanner.nextInt();
                    controller.deleteStudent(del);
                    System.out.println("Student deleted.");
                    break;

                case 5:
                    System.out.println("Exiting.");
                    return;
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
  }
}
```