Case Study 1: Online Course Registration System

Objective:

Allow students to register/unregister for courses and view course details.

Table Structure:

```
CREATE DATABASE course_db;

USE course_db;

CREATE TABLE courses (  course_id
        INT PRIMARY KEY,
        course_name VARCHAR(100),
        faculty VARCHAR(100),
         credits INT );
```

JDBC Operations:

- INSERT: Add new courses.
- SELECT: List available courses.
- UPDATE: Modify faculty or credit values.
- DELETE: Remove obsolete courses.

```
package Jdbccasestudy;

import java.sql.Connection; import
java.sql.DriverManager; import
java.sql.PreparedStatement; import
java.sql.ResultSet; import
java.sql.Statement; import
java.util.Scanner;

public class CourseRegistrationSystem {
 static final String DB_URL = "jdbc:mysql://localhost:3306/course_db";
 static final String USER = "root";
```

```
static final String PASS = "root";

public static void main(String[] args) {

      Scanner sc = new Scanner(System.in);
       try
(
         Connection conn = DriverManager.getConnection(DB_URL, USER, PASS);
         Statement stmt = conn.createStatement();
      ) {
         while (true) {
            System.out.println("\n--- Course Registration System ---");
            System.out.println("1. Add Course");
            System.out.println("2. View Courses");
            System.out.println("3. Update Course");
            System.out.println("4. Delete Course");
            System.out.println("5. Exit");
System.out.print("Choose: ");              int
choice = sc.nextInt();
sc.nextLine();


            if (choice == 1) {
               System.out.print("Course ID: ");
int id = sc.nextInt();                  sc.nextLine();
               System.out.print("Course Name: ");
               String name = sc.nextLine();
               System.out.print("Faculty: ");
               String faculty = sc.nextLine();
System.out.print("Credits: ");                int
credits = sc.nextInt();
```

```
            String sql = "INSERT INTO courses VALUES (?, ?, ?, ?)";
PreparedStatement ps = conn.prepareStatement(sql);
            ps.setInt(1, id);
ps.setString(2, name);
ps.setString(3, faculty);
ps.setInt(4, credits);
ps.executeUpdate();
            System.out.println("Course added!");

        } else if (choice == 2) {
            ResultSet rs = stmt.executeQuery("SELECT * FROM courses");
System.out.println("\nCourses:");              while (rs.next()) {
                System.out.println(rs.getInt(1) + " | " + rs.getString(2) +
                        " | " + rs.getString(3) + " | Credits: " + rs.getInt(4));
            }

        } else if (choice == 3) {
            System.out.print("Enter Course ID to Update: ");
int id = sc.nextInt();              sc.nextLine();
            System.out.print("New Faculty: ");
            String faculty = sc.nextLine();
System.out.print("New Credits: ");              int
credits = sc.nextInt();

            String sql = "UPDATE courses SET faculty = ?, credits = ? WHERE course_id = ?";
PreparedStatement ps = conn.prepareStatement(sql);
            ps.setString(1, faculty);
ps.setInt(2, credits);              ps.setInt(3,
id);              ps.executeUpdate();
System.out.println("Course updated!");
```

```
            } else if (choice == 4) {
                System.out.print("Enter Course ID to Delete: ");
int id = sc.nextInt();
                String sql = "DELETE FROM courses WHERE course_id = ?";
PreparedStatement ps = conn.prepareStatement(sql);
                ps.setInt(1, id);
ps.executeUpdate();
                System.out.println("Course deleted!");

            } else if (choice == 5) {
break;
            } else {
                System.out.println("Invalid choice");
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
  }

}
```

Case Study 2: Product Inventory System

Objective:

Track product stock in a retail store.

Table Structure:C

CREATE DATABASE inventory_db;

USE inventory_db;

```
CREATE TABLE products (  product_id
        INT PRIMARY KEY,
        product_name VARCHAR(100),
        quantity INT,
        price DECIMAL(10,2)
);
```

JDBC Operations:

• INSERT: Add new products to inventory.

• SELECT: View stock levels and prices.

• UPDATE: Update quantity after sale/purchase.

• DELETE: Remove discontinued products

```
package Jdbccasestudy;

import java.sql.Connection; import
java.sql.DriverManager; import
java.sql.PreparedStatement; import
java.sql.ResultSet; import
java.util.Scanner;

import java.sql.Statement;

public class ProductInventorySystem {
    static final String DB_URL = "jdbc:mysql://localhost:3306/inventory_db";
    static final String USER = "root";
    static final String PASS ="root";

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        try (
```

```
        Connection conn = DriverManager.getConnection(DB_URL, USER, PASS);
        Statement stmt = conn.createStatement();
    ) {
        while (true) {
            System.out.println("\n--- Product Inventory ---");
            System.out.println("1. Add Product");
            System.out.println("2. View Products");
            System.out.println("3. Update Quantity");
            System.out.println("4. Delete Product");
            System.out.println("5. Exit");
System.out.print("Choose: ");              int
choice = sc.nextInt();
sc.nextLine();

            if (choice == 1) {
                System.out.print("Product ID: ");
                int id = sc.nextInt();
sc.nextLine();
                System.out.print("Product Name: ");
                String name = sc.nextLine();
System.out.print("Quantity: ");                int
qty = sc.nextInt();
System.out.print("Price: ");
double price = sc.nextDouble();

                String sql = "INSERT INTO products VALUES (?, ?, ?, ?)";
PreparedStatement ps = conn.prepareStatement(sql);
                ps.setInt(1, id);
ps.setString(2, name);
ps.setInt(3, qty);
```

```
ps.setDouble(4, price);
ps.executeUpdate();
                System.out.println("Product added!");

            } else if (choice == 2) {
                ResultSet rs = stmt.executeQuery("SELECT * FROM products");
System.out.println("\nInventory:");
                while (rs.next()) {
                   System.out.println(rs.getInt(1) + " | " + rs.getString(2) +
                         " | Qty: " + rs.getInt(3) + " | " + rs.getDouble(4));
                }

            } else if (choice == 3) {
                System.out.print("Enter Product ID: ");
int id = sc.nextInt();
                System.out.print("New Quantity: ");
int qty = sc.nextInt();

                String sql = "UPDATE products SET quantity = ? WHERE product_id = ?";
PreparedStatement ps = conn.prepareStatement(sql);
                ps.setInt(1, qty);
ps.setInt(2, id);
ps.executeUpdate();
                System.out.println("Quantity updated!");

            } else if (choice == 4) {
                System.out.print("Enter Product ID to Delete: ");
                int id = sc.nextInt();
                String sql = "DELETE FROM products WHERE product_id = ?";
PreparedStatement ps = conn.prepareStatement(sql);
                ps.setInt(1, id);
```

```
                ps.executeUpdate();
                System.out.println("Product deleted!");

            } else if (choice == 5) {
break;
            } else {
                System.out.println("Invalid choice");
            }
          }
      } catch (Exception e) {
e.printStackTrace();
      }
   }
}
```