**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**

import java.sql.\*;

import java.util.Scanner;

public class CourseJDBC {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

try (

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/course\_db", "root", "password");

Statement stmt = con.createStatement()

) {

while (true) {

System.out.println("\nCourse 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 an option: ");

int choice = sc.nextInt();

if (choice == 1) {

System.out.print("Enter Course ID: ");

int id = sc.nextInt();

sc.nextLine();

System.out.print("Enter Course Name: ");

String name = sc.nextLine();

System.out.print("Enter Faculty: ");

String faculty = sc.nextLine();

System.out.print("Enter Credits: ");

int credits = sc.nextInt();

String sql = "INSERT INTO courses VALUES (?, ?, ?, ?)";

try (PreparedStatement ps = con.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 successfully!");

}

} else if (choice == 2) {

ResultSet rs = stmt.executeQuery("SELECT \* FROM courses");

System.out.println("Course List:");

while (rs.next()) {

System.out.printf("%d | %s | %s | %d\n",

rs.getInt("course\_id"),

rs.getString("course\_name"),

rs.getString("faculty"),

rs.getInt("credits"));

}

} else if (choice == 3) {

System.out.print("Enter Course ID to update: ");

int id = sc.nextInt();

sc.nextLine();

System.out.print("Enter new Faculty: ");

String faculty = sc.nextLine();

System.out.print("Enter new Credits: ");

int credits = sc.nextInt();

String sql = "UPDATE courses SET faculty=?, credits=? WHERE course\_id=?";

try (PreparedStatement ps = con.prepareStatement(sql)) {

ps.setString(1, faculty);

ps.setInt(2, credits);

ps.setInt(3, id);

ps.executeUpdate();

System.out.println("Course updated successfully!");

}

} else if (choice == 4) {

System.out.print("Enter Course ID to delete: ");

int id = sc.nextInt();

String sql = "DELETE FROM courses WHERE course\_id=?";

try (PreparedStatement ps = con.prepareStatement(sql)) {

ps.setInt(1, id);

ps.executeUpdate();

System.out.println("Course deleted successfully!");

}

} else if (choice == 5) {

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

break;

} else {

System.out.println("Invalid choice.");

}

}

} catch (SQLException e) {

e.printStackTrace();

}

}

}

**2. Product Inventory System**

**Objective: Track product stock in a retail store.**

**Table Structure:**

**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**

import java.sql.\*;

import java.util.Scanner;

public class InventorySystem {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

try {

// 1. Connect to MySQL database

Connection conn = DriverManager.getConnection(

"jdbc:mysql://localhost:3306/inventory\_db", "root", "password");

while (true) {

// 2. Show Menu

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("Enter choice: ");

int choice = input.nextInt();

if (choice == 1) {

// Add product

System.out.print("Enter product ID: ");

int id = input.nextInt();

input.nextLine(); // clear buffer

System.out.print("Enter product name: ");

String name = input.nextLine();

System.out.print("Enter quantity: ");

int qty = input.nextInt();

System.out.print("Enter price: ");

double price = input.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) {

// View products

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("SELECT \* FROM products");

System.out.println("Product List:");

while (rs.next()) {

System.out.println(rs.getInt("product\_id") + " | " +

rs.getString("product\_name") + " | " +

rs.getInt("quantity") + " | " +

rs.getDouble("price"));

}

} else if (choice == 3) {

// Update quantity

System.out.print("Enter product ID to update: ");

int id = input.nextInt();

System.out.print("Enter new quantity: ");

int qty = input.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) {

// Delete product

System.out.print("Enter product ID to delete: ");

int id = input.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) {

System.out.println("Goodbye!");

break;

} else {

System.out.println("Invalid option!");

}

}

conn.close(); // Close connection

} catch (Exception e) {

e.printStackTrace();

}

}

}