```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DBConnection {
  private static final String URL = "jdbc:mysql://localhost:3306/employee_db";
  private static final String USER = "root";  // change if needed
  private static final String PASSWORD = "";  // change if needed

  public static Connection getConnection() throws SQLException {
    return DriverManager.getConnection(URL, USER, PASSWORD);
  }
}
public class Employee {
  private int id;
  private String name;
  private String position;
  private double salary;

  public Employee(String name, String position, double salary) {
    this.name = name;
    this.position = position;
    this.salary = salary;
  }

  public Employee(int id, String name, String position, double salary) {
    this(name, position, salary);
    this.id = id;
  }

  // Getters and Setters
```

```
}
import java.sql.*;
import java.util.*;

public class EmployeeDAO {

  public void addEmployee(Employee emp) throws SQLException {
    String sql = "INSERT INTO employee (name, position, salary) VALUES (?, ?, ?)";
    try (Connection conn = DBConnection.getConnection();
       PreparedStatement ps = conn.prepareStatement(sql)) {
      ps.setString(1, emp.getName());
      ps.setString(2, emp.getPosition());
      ps.setDouble(3, emp.getSalary());
      ps.executeUpdate();
      System.out.println("Employee added successfully.");
    }
  }

  public void viewEmployees() throws SQLException {
    String sql = "SELECT * FROM employee";
    try (Connection conn = DBConnection.getConnection();
       Statement stmt = conn.createStatement();
       ResultSet rs = stmt.executeQuery(sql)) {

      while (rs.next()) {
        System.out.printf("ID: %d, Name: %s, Position: %s, Salary: %.2f\n",
          rs.getInt("id"), rs.getString("name"),
          rs.getString("position"), rs.getDouble("salary"));
      }
    }
  }
```

```
    public void updateEmployee(Employee emp) throws SQLException {
        String sql = "UPDATE employee SET name=?, position=?, salary=? WHERE id=?";
        try (Connection conn = DBConnection.getConnection();
             PreparedStatement ps = conn.prepareStatement(sql)) {
            ps.setString(1, emp.getName());
            ps.setString(2, emp.getPosition());
            ps.setDouble(3, emp.getSalary());
            ps.setInt(4, emp.getId());
            ps.executeUpdate();
            System.out.println("Employee updated.");
        }
    }

    public void deleteEmployee(int id) throws SQLException {
        String sql = "DELETE FROM employee WHERE id=?";
        try (Connection conn = DBConnection.getConnection();
             PreparedStatement ps = conn.prepareStatement(sql)) {
            ps.setInt(1, id);
            ps.executeUpdate();
            System.out.println("Employee deleted.");
        }
    }
}
import java.sql.SQLException;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        EmployeeDAO dao = new EmployeeDAO();
        Scanner sc = new Scanner(System.in);
        int choice;

        do {
          System.out.println("\n1. Add Employee\n2. View Employees\n3. Update Employee\n4. Delete Employee\n5. Exit");
          choice = sc.nextInt();

          try {
            switch (choice) {
              case 1:
                System.out.print("Enter name: ");
                sc.nextLine(); // consume leftover newline
                String name = sc.nextLine();
                System.out.print("Enter position: ");
                String position = sc.nextLine();
                System.out.print("Enter salary: ");
                double salary = sc.nextDouble();
                dao.addEmployee(new Employee(name, position, salary));
                break;
              case 2:
                dao.viewEmployees();
                break;
              case 3:
                System.out.print("Enter ID to update: ");
                int id = sc.nextInt();
                sc.nextLine();
                System.out.print("Enter new name: ");
                name = sc.nextLine();
                System.out.print("Enter new position: ");
                position = sc.nextLine();
                System.out.print("Enter new salary: ");
```

```
                        salary = sc.nextDouble();
                        dao.updateEmployee(new Employee(id, name, position, salary));
                        break;
                    case 4:
                        System.out.print("Enter ID to delete: ");
                        id = sc.nextInt();
                        dao.deleteEmployee(id);
                        break;
                }
            } catch (SQLException e) {
                e.printStackTrace();
            }

        } while (choice != 5);

        sc.close();
    }
}
```