DAY 11

1) **StudentReportGenerator**

```
package demo_JDBC;
import java.sql.*;

public class StudentReportGenerator {
        public static void main(String[] args) {

        String url = "jdbc:mysql://localhost:3306/db";
        String user = "root"; // Changed to a placeholder
        String password = "1752"; // Changed to a placeholder

        try (Connection con = DriverManager.getConnection(url, user, password)) {

                CallableStatement cst = con.prepareCall("{CALL getStudentBymark00()}");

                ResultSet rs = cst.executeQuery();

                System.out.println("Students with 0 Marks");
                System.out.println("ID\tName\t\t\tMarks");
                System.out.println("------------------------------------");

                while (rs.next()) {
                        int id = rs.getInt("id");
                        String name = rs.getString("name");
                        int marks = rs.getInt("marks");

                        System.out.printf("%d\t%-20s\t%d\n", id, name, marks);
                }

                rs.close();

        } catch (SQLException e) {
                e.printStackTrace();
        }
}
}
```

2) **StudentDataUpdater**

```
package demo_JDBC;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class StudentDataUpdater {
  public static void main(String[] args) throws SQLException {
    String url = "jdbc:mysql://localhost:3306/db";
    String user = "root";
    String password = "1752";

    String createTableSQL = "CREATE TABLE Student (rollno INT, name VARCHAR(50), per INT, email
VARCHAR(50))";
    String insertSQL = "INSERT INTO Student (rollno, name, per, email) VALUES " +
             "(101, ‘Harshu’, 97, harshu@gmail.com'), " +
             "(102, ‘Thejas’, 48, thejas@gmail.com'), " +
                "(103, ‘dhanya’, 52, dhanya@gmail.com'), " +
                "(104, 'Sandhya', 66, 'sandhya@gmail.com'), " +
                "(105, 'dhaivik', 56, 'dhaivik@gmail.com')";
String updateSQL = "UPDATE Student SET per=90 WHERE rollno = 102";
String selectAllSQL = "SELECT * FROM Student";
String selectMaxSQL = "SELECT MAX(per) FROM Student";
String selectAscSQL = "SELECT * FROM Student ORDER BY per ASC";

try (Connection con = DriverManager.getConnection(url, user, password);
    Statement stmt = con.createStatement()) {

   stmt.executeUpdate("DROP TABLE IF EXISTS Student"); // Added for safe re-run
   stmt.executeUpdate(createTableSQL);
   System.out.println("Student table created.");

   int rowInserted = stmt.executeUpdate(insertSQL);
   if (rowInserted > 0) {
      System.out.println(rowInserted + " new student records inserted.");
   }

   int updatedRows = stmt.executeUpdate(updateSQL);
   if (updatedRows > 0) {
      System.out.println(updatedRows + " record updated.");
   }



   ResultSet rs = stmt.executeQuery(selectAllSQL);
   System.out.println("\nAll Student Records:");
   System.out.println("rollno\t name\t\t per\t Email");
   while (rs.next()) {
      int rollno = rs.getInt("rollno");
      String name = rs.getString("name");
      int per = rs.getInt("per");
      String email = rs.getString("email");
      System.out.printf("%-6d\t %-12s %-5d\t %s\n", rollno, name, per, email);
   }
   rs.close();

   ResultSet rsMax = stmt.executeQuery(selectMaxSQL);
   if (rsMax.next()) {
      int maxvalue = rsMax.getInt(1);
      System.out.println("\nMax percentage is " + maxvalue);
   }
   rsMax.close();

            ResultSet rasc = stmt.executeQuery(selectAscSQL);
   System.out.println("\nStudents in ascending order by percentage:");
   System.out.println("rollno\t name\t\t per\t Email");
   while (rasc.next()) {
      int rollno = rasc.getInt("rollno");
      String name = rasc.getString("name");
      int per = rasc.getInt("per");
      String email = rasc.getString("email");
      System.out.printf("%-6d\t %-12s %-5d\t %s\n", rollno, name, per, email);
   }
   rasc.close();

} catch (SQLException e) {
```

```
            System.out.println("A SQL exception occurred.");
            e.printStackTrace();
        }
    }
}
```

3) EmployeeDataRetriever

```
package demo_JDBC;
import java.sql.*;

public class EmployeeDataRetriever {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/db";
        String username = "root";
        String password = "1752";

        try (Connection con = DriverManager.getConnection(url, username, password)) {

            CallableStatement cst = con.prepareCall("{CALL samename()}");
            CallableStatement cst1 = con.prepareCall("{CALL descending()}");

            ResultSet rs = cst.executeQuery();
            ResultSet rs1 = cst1.executeQuery();

            System.out.println("Employees with the same name:");
            System.out.println("ID\tName\t\t\tSalary");
            System.out.println("------------------------------------");

            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                int salary = rs.getInt("salary");
                System.out.printf("%d\t%-20s\t%d\n", id, name, salary);
            }

            System.out.println("\nEmployees sorted by salary in descending order:");
            System.out.println("ID\tName\t\t\tSalary");
            System.out.println("------------------------------------");

            while (rs1.next()) {
                int id = rs1.getInt("id");
                String name = rs1.getString("name");
                int salary = rs1.getInt("salary");
                System.out.printf("%d\t%-20s\t%d\n", id, name, salary);
            }

            rs.close();
            cst.close();
            rs1.close();
            cst1.close();

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

4) HospitalDataAnalyzer

```
package demo_JDBC;
import java.sql.*;

public class HospitalDataAnalyzer {
  public static void main(String[] args) {
    String url = "jdbc:mysql://localhost:3306/db";
    String username = "root";
    String password = "1752";

    try (Connection con = DriverManager.getConnection(url, username, password);
       CallableStatement cstAvgWages = con.prepareCall("{CALL avg_daily_wages()}");
       CallableStatement cstSameWard = con.prepareCall("{CALL same_ward()}");
       CallableStatement cstAscending = con.prepareCall("{CALL ascending_()}")
    ) {

      System.out.println("Average number of patients per day:");
      try (ResultSet rsAvg = cstAvgWages.executeQuery()) {
        if (rsAvg.next()) {
          System.out.println(rsAvg.getDouble(1));
        }
      }

      System.out.println("\nPatients in the same ward:");
      System.out.println("P_ID\tP_Name\tDaily_wages\tWard_no\tJoining_Date");
      System.out.println("--------------------------------------------------");
      try (ResultSet rsSameWard = cstSameWard.executeQuery()) {
        while (rsSameWard.next()) {
          int patient_id = rsSameWard.getInt("patient_id");
          String p_name = rsSameWard.getString("p_name");
          int daily_wages = rsSameWard.getInt("daily_wages");
          int ward_no = rsSameWard.getInt("ward_no");
          String joining_date = rsSameWard.getString("joining_date");
          System.out.printf("%d\t%-10s\t%d\t\t%d\t%s\n", patient_id, p_name, daily_wages, ward_no,
joining_date);
        }
      }

      System.out.println("\nPatients in ascending order by some criteria:");
      System.out.println("P_ID\tP_Name\tDaily_wages\tWard_no\tJoining_Date");
      System.out.println("--------------------------------------------------");
      try (ResultSet rsAsc = cstAscending.executeQuery()) {
        while (rsAsc.next()) {
          int patient_id = rsAsc.getInt("patient_id");
          String p_name = rsAsc.getString("p_name");
          int daily_wages = rsAsc.getInt("daily_wages");
          int ward_no = rsAsc.getInt("ward_no");
          Date joining_date = rsAsc.getDate("joining_date");
          System.out.printf("%d\t%-10s\t%d\t\t%d\t%s\n", patient_id, p_name, daily_wages, ward_no,
joining_date);
        }
      }

    } catch (SQLException e) {
      e.printStackTrace();
    }
  }}
```

5) Simple_JDBC_Query

```
package demo_JDBC;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class Simple_JDBC_Query
{
   public static void main(String[] args) throws ClassNotFoundException, SQLException {
      String url = "jdbc:mysql://localhost:3306/db";
      String user = "root";
      String password = "1752";

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

      try (Connection con = DriverManager.getConnection(url, user, password);
          Statement stmt = con.createStatement();
          ResultSet rs = stmt.executeQuery("SELECT * FROM emp")) {

         System.out.println("Connected to the database Successfully .");
         System.out.println("id\t name\t salary\t age");

         while (rs.next()) {
            int id = rs.getInt("id");
            String name = rs.getString("name");
            int salary = rs.getInt("salary");
            int age = rs.getInt("age");
            System.out.println(id + "\t " + name + "\t " + salary + "\t " + age);
         }

      } catch (SQLException e) {
         System.out.println(" exception(SQL) occurred.");
         e.printStackTrace();
      }
   }
}
```

6) Product

```
package day11;
import java.sql.*;
import java.util.*;

class ProductData implements Comparable<ProductData> {
   private int id;
   private String name;
   private double unitPrice;

   public ProductData(int id, String name, double unitPrice) {
      this.id = id;
      this.name = name;
      this.unitPrice = unitPrice;
   }

   public double getUnitPrice() {
      return unitPrice;
   }
   @Override
   public int compareTo(ProductData other) {
            return Double.compare(this.unitPrice, other.unitPrice);
   }

   @Override
   public String toString() {
      return "ID: " + id + ", Name: " + name + ", Unit Price: " + unitPrice;
   }
}
public class DataSorterApp {
   private static final String URL = "jdbc:mysql://localhost:3306/dbb";
   private static final String USER = "root";
   private static final String PASSWORD = "1752";

   public static void main(String[] args) {
      List<ProductData> productList = new ArrayList<>();

            try (Connection conn = DriverManager.getConnection(URL, USER, PASSWORD);
          Statement stmt = conn.createStatement()) {
         stmt.executeUpdate("DROP TABLE IF EXISTS merchandise");
         stmt.executeUpdate("CREATE TABLE merchandise (id INT, name VARCHAR(50), unit_price DOUBLE)");
         stmt.executeUpdate("INSERT INTO merchandise VALUES (1, 'Laptop', 1200.00), (2, 'Mouse', 25.50), (3,
'Keyboard', 75.00), (4, 'Monitor', 300.75)");

         System.out.println("Database setup complete. The 'merchandise' table is ready with sample data.");

         String fetchQuery = "SELECT * FROM merchandise";
         try (ResultSet rs = stmt.executeQuery(fetchQuery)) {
                    while (rs.next()) {
               int id = rs.getInt("id");
               String name = rs.getString("name");
               double unitPrice = rs.getDouble("unit_price");
               productList.add(new ProductData(id, name, unitPrice));
            }
         }

         System.out.println("\n--- Product List (Unsorted as fetched from DB) ---");
         for (ProductData product : productList) {
            System.out.println(product);
         }
         Collections.sort(productList); // Sorts the list using the compareTo method in the ProductData class

         System.out.println("\n--- Product List (Sorted by Unit Price) ---");
         for (ProductData product : productList) {
            System.out.println(product);
         }

      } catch (SQLException | ClassNotFoundException e) {
         e.printStackTrace();
      }
   }
}
```

7) MyDbConnection

```
package day11;

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

public class MySQLConnection {
    public static void main(String[] args) {

        String url = "jdbc:mysql://localhost:3306/dbb";
        String user = "root";
        String password = "1752”;
        try {

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

            Connection connection = DriverManager.getConnection(url, user, password);

            System.out.println("Connected to the database successfully!");

            connection.close();
        } catch (ClassNotFoundException e) {
            System.out.println("Couldn’t Find MySQL JDBC Driver .");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Failed to Connect Database.");
            e.printStackTrace();
        }
    }
}
```