1.Create procedure or functions for employee table

1. **Add 5000 bonus to all employee**
2. **Print same name employees**
3. **Print highest and lowest salary from employee table**

```
CREATE TABLE employee (
   id INT PRIMARY KEY,
   name VARCHAR(50),
   salary INT
);
-- Insert sample data
INSERT INTO employee VALUES
(1, 'Jayanth', 50000),
(2, 'Srihari', 55000),
(3, 'Ram', 60000),
(4, 'Raghu', 70000),
(5, 'Nandu', 45000);
DELIMITER $$

CREATE PROCEDURE add_bonus()
BEGIN
   UPDATE employee
   SET salary = salary + 5000;
END $$

DELIMITER ;

DELIMITER $$
```

```
CREATE PROCEDURE print_duplicate_names()
BEGIN
    SELECT emp_name, COUNT(*) as count
    FROM employee
    GROUP BY emp_name
    HAVING COUNT(*) > 1;
END $$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE salary_stats()
BEGIN
    SELECT
        MAX(salary) AS highest_salary,
        MIN(salary) AS lowest_salary
    FROM employee;
END $$

DELIMITER ;
```

Program:

```
package JDBC_conn;

import java.sql.*;
```

```
public class EmployeeProcedure {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/java";
        String user = "root";
        String password = "root";
        try (Connection con = DriverManager.getConnection(url, user, password))
{
            System.out.println("Connected to DB ");
            CallableStatement bonusStmt = con.prepareCall("{CALL
addBonusToAll()}");
            bonusStmt.execute();
            System.out.println(" ₹5000 bonus added to all employees.\n");
            CallableStatement sameNameStmt = con.prepareCall("{CALL
printSameNameEmployees()}");
            ResultSet rs = sameNameStmt.executeQuery();

            System.out.println(" Employees with same name:");
            System.out.println("ID\tName\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%-15s\t%d\n", id, name, salary);
            }
            CallableStatement rangeStmt = con.prepareCall("{? = CALL
getSalaryRange()}");
            rangeStmt.registerOutParameter(1, Types.VARCHAR);
            rangeStmt.execute();

            String range = rangeStmt.getString(1);
            System.out.println("\n Salary Range:");
            System.out.println(range);

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

OutPut:

```
Connected to DB
₹5000 bonus added to all employees.

 Employees with same name:
ID      Name          Salary
------------------------------

💰 Salary Range:
Lowest Salary = 70000, Highest Salary = 95000
```

2. Create procedure or functions for Hospital table

1. **print avg patient count on daily basis**
2. **print all the patients whose belong to same ward**
3. **arrange the patients list according their admission date**

```
CREATE TABLE hospital (
   patient_id INT PRIMARY KEY,
   name VARCHAR(100),
   ward_no INT,
   admission_date DATE
);
drop table hospital;
INSERT INTO hospital VALUES
(1, 'Akhil', 'A1', '2025-08-01'),
(2, 'Nikhil', 'A2', '2025-08-01'),
(3, 'Raju', 'A1', '2025-08-02'),
(4, 'Balu', 'A3', '2025-08-02'),
(5, 'Manish', 'A1', '2025-08-03');
DELIMITER $$

CREATE PROCEDURE avg_patient_count_daily()
BEGIN
   SELECT AVG(cnt) AS avg_patient_per_day
   FROM (
      SELECT admission_date, COUNT(*) AS cnt
      FROM hospital
```

```
        GROUP BY admission_date
    ) AS sub;
END $$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE patients_same_ward()
BEGIN
    SELECT h1.*
    FROM hospital h1
    JOIN (
        SELECT ward_no
        FROM hospital
        GROUP BY ward_no
        HAVING COUNT(*) > 1
    ) h2 ON h1.ward_no = h2.ward_no;
END $$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE patients_by_admission()
BEGIN
```

```
    SELECT * FROM hospital ORDER BY admission_date;
END $$

DELIMITER ;
```

Java Program:

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

public class Hospital_data{
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/mydb";
        String user = "root";
        String password = "root";

        try (Connection con = DriverManager.getConnection(url, user, password)) {
            System.out.println("Connected to DB...");

            // 1. Average patient count daily
            CallableStatement cs1 = con.prepareCall("{CALL avg_patient_count_daily()}");
            ResultSet rs1 = cs1.executeQuery();
            while (rs1.next()) {
                System.out.println("Average patients per day: " + rs1.getDouble("avg_patient_per_day"));
            }
            CallableStatement cs2 = con.prepareCall("{CALL
patients_same_ward()}");
            ResultSet rs2 = cs2.executeQuery();
            System.out.println("\nPatients in same ward:");
            while (rs2.next()) {
                System.out.println(rs2.getInt("patient_id") + " - " +
rs2.getString("name") +
                                   " (Ward: " + rs2.getInt("ward_no") + ")");
            }
            CallableStatement cs3 = con.prepareCall("{CALL
patients_by_admission()}");
            ResultSet rs3 = cs3.executeQuery();
            System.out.println("\nPatients by admission date:");
            while (rs3.next()) {
                System.out.println(rs3.getInt("patient_id") + " - " +
rs3.getString("name") +
                                   " (Admitted: " + rs3.getDate("admission_date") + ")");
            }

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

OutPut:

```
Connected
```

```
Avg patient count per day: 1.6667


 Patients in same wards:
1       Akhil   A1      2025-08-01
3       Raju    A1      2025-08-02
5       Manish          A1      2025-08-03


 Patients sorted by admission date:
1       Akhil   A1      2025-08-01
2       Nikhil  A2      2025-08-01
3       Raju    A1      2025-08-02
4       Balu    A3      2025-08-02
5       Manish          A1      2025-08-03
```