Experiment 3

Student Name: Sohit Kumar **UID: 23BCS12492**
Branch: CSE **Section/Group: KRG-3A**
Semester: 5th **Date of Performance:21AUG**
Subject Name: ADBMS **Subject Code: 23CSP-333**

1. Problem Statement & SQL Code:

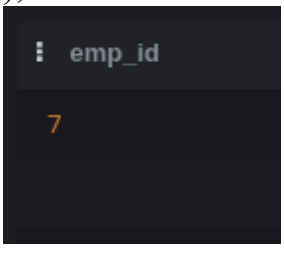
Q1)

Consider the following employees table , Write a query to find the maximum employee id that is not duplicated in the table (i.e., the largest id that occurs only once).

Solution:

```
CREAT TABLE employees (
        id INT
);
INSERT INTO employees VALUES (2),(4),(4),(6),(6),(7),(8),(8);

SELECT MAX(id) AS emp_id
FROM employees
WHERE id not in (
        SELECT id
        FROM employees
        GROUP BY  id
        HAVING COUNT(*) > 1
);
```



Q2) Department Salary Champions

In a bustling corporate organization, each department strives to retain the most talented (and well-compensated) employees. You have access to two key records: one lists every employee along with their salary and department, while the other details the names of each department. Your task is to identify the top earners in every department.

If multiple employees share the same highest salary within a department, all of them should be celebrated equally. The final result should present the department name, employee name, and salary of these top-tier professionals arranged by department.

Solution:

```
CREATE TABLE Department (
    ID INT PRIMARY KEY,
    DEPT_NAME VARCHAR(50)
);

CREATE TABLE Employee (
    ID INT,
    NAME VARCHAR(50),
    SALARY INT,
    DEPT_ID INT,
    FOREIGN KEY (DEPT_ID) REFERENCES Department(ID)
);

INSERT INTO Employee (ID, NAME, SALARY, DEPT_ID) VALUES
(1, 'JOE', 70000, 1),
(2, 'JIM', 90000, 1),
(3, 'HENRY', 80000, 2),
(4, 'SAM', 60000, 2),
(5, 'MAX', 90000, 1);
```

```
SELECT d.DEPT_NAME, e.NAME, e.SALARY
FROM Employee e
JOIN
Department d
ON e.DEPT_ID = d.ID
WHERE e.SALARY = (
        SELECT MAX(SALARY)
        FROM Employee e2
        WHERE e2.DEPT_ID = e.DEPT_ID
) ORDER BY d.DEPT_NAME;
```

OUTPUT:

DEPT_NAME	NAME	SALARY
IT	MAX	90000
IT	JIM	90000
SALES	HENRY	80000

Q3) Merging Employee Histories: Who Earned Least?

Two legacy HR systems (A and B) have separate records of employee salaries. These records may overlap. Management wants to merge these datasets and identify each unique employee (by EmpID) along with their lowest recorded salary across both systems.

Objective
1. Combine two tables A and B.
2. Return each EmpID with their lowest salary, and the corresponding Ename.

Solution:

```
CREATE TABLE A (
    EmpID INT PRIMARY KEY,
    Ename VARCHAR(23),
    Salary INT
);
```

```
CREATE TABLE B (
    EmpID INT PRIMARY KEY,
    Ename VARCHAR(23),
    Salary  INT
);

INSERT INTO A (EmpID, Ename, Salary) VALUES
(1, 'AA', 1000),
(2, 'BB', 300);

INSERT INTO B (EmpID, Ename, Salary) VALUES
(2, 'BB', 400),
(3, 'CC', 100);

SELECT EmpID, Ename, MIN(Salary) AS Salary
FROM (
    SELECT EmpID, Ename, Salary FROM A
    UNION ALL
    SELECT EmpID, Ename, Salary FROM B
) AS Intermediate_Result
GROUP BY EmpID, Ename;
```

OUTPUT:

EmpID	Ename	Salary
1	AA	1000
2	BB	300
3	CC	100