EXPERIMENT- 03

Student Name: Ritesh Kumar **UID: 23BCS12809**
Branch: BE-CSE **Section/Group: KRG 1(B)**
Semester: 05 **Date of Performance: 19/08/25**
Subject Name: ADBMS **Subject Code: 23CSP-333**

Medium Level - Department Salary Champions

1. **Aim:**

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.

2. **Objective:**

- **Understanding Subqueries:** Learned how to use subqueries to perform intermediate calculations (like finding the maximum salary per department) and use that result in the main query.
- **Handling Ties and Multiple Results:** Gained the skill to fetch all employees sharing the top salary within a department, not just one, ensuring accurate and fair results.
- **Data Integration & Presentation:** Practiced joining multiple tables (employees and departments) and arranging results logically, which reinforces skills in combining and presenting relational data efficiently.

3. DBMS script:

```
CREATE TABLE department_new (
    dept_id INT PRIMARY KEY,
    dept_title VARCHAR(50)
);

CREATE TABLE staff (
    emp_id INT PRIMARY KEY,
    emp_name VARCHAR(50),
    emp_salary INT,
    dept_ref INT,
    FOREIGN KEY (dept_ref) REFERENCES department_new(dept_id)
);

INSERT INTO department_new VALUES
(1, 'IT'),
(2, 'SALES');

INSERT INTO staff VALUES
(1, 'JOE', 70000, 1),
(2, 'JIM', 90000, 1),
(3, 'HENRY', 80000, 2),
(4, 'ABC', 90000, 1);
```

-- i. Query Using Subquery with GROUP BY

```
SELECT s.emp_id, s.emp_name, d.dept_title, s.emp_salary
FROM department_new AS d
JOIN staff AS s
ON d.dept_id = s.dept_ref
```

```
WHERE s.emp_salary IN
(
    SELECT MAX(emp_salary)
    FROM staff
    GROUP BY dept_ref
);
```

-- ii. Query Using Correlated Subquery

```
SELECT s.emp_id, s.emp_name, d.dept_title, s.emp_salary
FROM department_new AS d
JOIN staff AS s
ON d.dept_id = s.dept_ref
WHERE s.emp_salary =
(
    SELECT MAX(s2.emp_salary)
    FROM staff AS s2
    WHERE s2.dept_ref = s.dept_ref
);
```

4. Output:

110 % No issues found

Results Messages

	emp_id	emp_name	dept_title	emp_salary
1	2	JIM	IT	90000
2	4	ABC	IT	90000
3	3	HENRY	SALES	80000

Hard Level - Merging Employee Histories: Who Earned Least?

1. Aim:

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.

i. Combine two tables A and B.
ii. Return each EmpID with their **lowest salary,** and the corresponding **Ename.**

2. Objective:

- **Merging Data:** Learned how to combine multiple tables using UNION to handle overlapping employee records.
- **Finding Minimum Salary:** Practiced using aggregation to determine the lowest salary for each employee.
- **Handling Duplicates & Retrieval:** Reinforced skills in managing duplicate entries and retrieving associated information like employee name accurately.

3. DBMS script:

```
CREATE TABLE A(
EmpID int primary key,
Ename varchar(50),
Salary int
);
```

```
CREATE TABLE B(
EmpID int primary key,
Ename varchar(50),
Salary int
);

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

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


SELECT EmpID, Ename, min(Salary) as Min_Salary
FROM
(SELECT* FROM A
UNION
SELECT* FROM B) AS X
 GROUP BY EmpID, Ename;
```

4. Output:

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