Experiment 3

Student Name: Rachit Kumar
UID: 23BCS11597
Branch: BE CSE
Section: 23BCS_KRG-2/A
Subject Name: Advanced Database. and Management System
Subject Code: 23CSP-333
Date :21/08/2025
Semester: 5th

1. Aim:

Q1: 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.

Q2: Merging Employee Histories: Who Earned Least? (Hard)
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.

2. Tools Used: SQL Server (One compiler)

3. Code:
Q1:

```
Q1: CREATE TABLE Employee (
    ID INT,
    NAME VARCHAR(50),
    SALARY INT,
    DEPT_ID INT
);

INSERT INTO Employee VALUES
(1, 'JOE', 70000, 1),
```

```
(2, 'JIM', 90000, 1),
(3, 'HENRY', 80000, 2),
(4, 'SAM', 60000, 2),
(5, 'MAX', 90000, 1);

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

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

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
    WHERE DEPT_ID = E.DEPT_ID
)
ORDER BY D.DEPT_NAME;
```

Q2:

```


CREATE TABLE A (
    EmpID INT,
    Ename VARCHAR(50),
    Salary INT
);

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

CREATE TABLE B (
    EmpID INT,
    Ename VARCHAR(50),
    Salary INT
);

INSERT INTO B VALUES
(2, 'BB', 400),
```

```
(3, 'CC', 100);


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

Output:

Output:

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

Output:

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

4. Learning Outcomes

- Write **correlated sub-queries** to identify top earners in each department.
- Apply **aggregate functions** (`MAX`, `MIN`) within sub-queries to extract meaningful insights.
- Handle **ties** (multiple employees with the same highest salary).
- Use **UNION / UNION ALL** to merge data from multiple sources into one consolidated view.
- Identify and select the **minimum value per group** using sub-queries and grouping.
- Strengthen understanding of **nested query execution** in SQL Server.