Experiment 1.3

Student Name: Yash Goel | **UID:** 23BCS11498

Branch: B.E-C.S.E | **Section/Group:** 23BCS_KRG-2B

Semester: 5th | **Date of Performance:** 18 August, 2025

Subject Name: ADBMS | **Subject Code:** 23CSP-333

MEDIUM - LEVEL

1. **Problem Title:** Department Salary Champions

2. **Procedure (Step-by-Step):** 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.

3. **SQL Commands:**

```
CREATE TABLE Departments (
    ID INT PRIMARY KEY IDENTITY(1,1),
    DEPT_NAME varchar(MAX)
);

INSERT INTO Departments(DEPT_NAME) VALUES ('IT'), ('SALES');

CREATE TABLE Employees(
    ID INT PRIMARY KEY IDENTITY(1,1),
    NAME varchar(MAX),
    SALARY INT,
    DEPT_ID INT
    FOREIGN KEY(DEPT_ID) REFERENCES Departments(ID)
);

INSERT INTO Employees (NAME, SALARY, DEPT_ID)
VALUES
('JOE', 70000, 1),
('JIM', 90000, 1),
('HENRY', 80000, 2),
('SAM', 60000, 2),
('MAX', 90000, 1);
```

```sql
SELECT D.DEPT_NAME, E.NAME, E.SALARY
FROM Departments AS D
INNER JOIN
Employees AS E
ON D.ID = E.DEPT_ID
WHERE E.SALARY IN
(
    SELECT MAX(E2.SALARY)
    FROM Employees AS E2
    WHERE E2.DEPT_ID = E.DEPT_ID
)
```

5. **Output:**

Results Messages

	ID	NAME	SALARY	DEPT_ID
1	1	JOE	70000	1
2	2	JIM	90000	1
3	3	HENRY	80000	2
4	4	SAM	60000	2
5	5	MAX	90000	1

	ID	DEPT_NAME
1	1	IT
2	2	SALES

	DEPT_NAME	NAME	SALARY
1	SALES	HENRY	80000
2	IT	MAX	90000
3	IT	JIM	90000

HARD - LEVEL

1. **Problem Title:** Merging Employee Histories

2. **Procedure (Step-by-Step):** 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.

3. **SQL Commands:**

```
CREATE TABLE A (EMPID INT PRIMARY KEY, ENAME VARCHAR(MAX), SALARY INT)

INSERT INTO A (EMPID, ENAME, SALARY) VALUES (1, 'AA', 1000), (2, 'BB', 300)

CREATE TABLE B (EMPID INT PRIMARY KEY, ENAME VARCHAR(MAX), SALARY INT)

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

SELECT EMPID, MIN(ENAME) AS ENAME, MIN(SALARY) AS SALARY
FROM
(
    SELECT * FROM A
    UNION ALL
    SELECT * FROM B
) AS INTERMEDIATE_RESULT
GROUP BY EMPID
```

4. **Output:**

Results | Messages

	EMPID	ENAME	SALARY
1	1	AA	1000
2	2	BB	300
3	3	CC	100