# Experiment 3

**Student Name:** om prakash Reddy **UID:** 23BAI70296

**Branch:** BE-AIT-CSE **Section/Group:** 23AIT-KRG-G2

**Semester:** 5th **Date of Performance:** 19th Aug, 2025

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

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 & 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. **Tools Used:**

   SQL Server Management Studio 21 (SSMS) code editor.

3. **Experiment:**

### Medium-Level

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.

### Hard-Level

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.

## 4. Solution:

### Medium-Leve

```
CREATE TABLE department (
    id INT PRIMARY KEY,
    dept_name VARCHAR(50)
);

-- Create Employee Table
CREATE TABLE employee8 (
    id INT,
    name VARCHAR(50),
    salary INT,
    department_id INT,
    FOREIGN KEY (department_id) REFERENCES department(id)
);
INSERT INTO department (id, dept_name) VALUES
(1, 'IT'),
(2, 'SALES');

-- Insert into Employee Table
INSERT INTO employee8 (id, name, salary, department_id) VALUES
(1, 'JOE', 70000, 1),
(2, 'JIM', 90000, 1),
(3, 'HENRY', 80000, 2),
(4, 'SAM', 60000, 2),
(5, 'MAX', 90000, 1);

select E.salary,e.name,d.dept_name
from
employee8 as E
inner join
department as d
on
e.department_id=d.id
where salary in
```

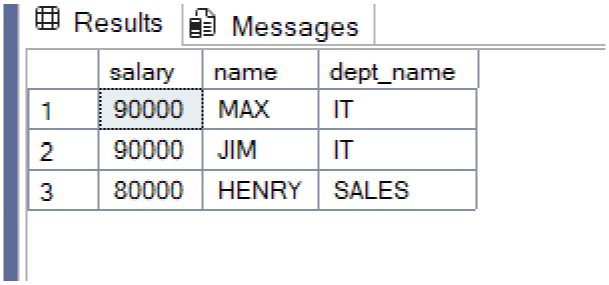
```
(

select max(e2.salary)

from employee8 as e2

where e2.department_id=e.department_id

)

order by d.dept_name
```

### Hard-Level

```
CREATE TABLE emply1 (
    emp_id INT PRIMARY KEY,
    emp_name VARCHAR(50),
    emp_salary int
)

CREATE TABLE emply3 (
    emp_id INT PRIMARY KEY,
    emp_name VARCHAR(50),
    emp_salary int
)

INSERT INTO emply1 (emp_id, emp_name,emp_salary) VALUES
(1, 'AA',1000),
(2, 'BB',300);

INSERT INTO emply3 (emp_id, emp_name,emp_salary) VALUES
(2, 'BB',600),
(3, 'CC',500);


with res_set as
(     select a.* from emply1 a
      UNION ALL
      select b.* from emply3 b
)
select Emp_ID, Emp_name, min(emp_Salary) as Salary from res_set r
      group by Emp_ID, Ename;
```

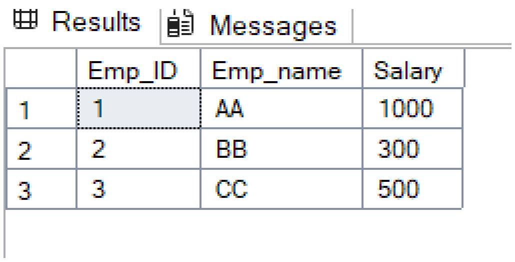
## 5. Output:

### Medium-Leve



| | salary | name | dept_name |
|---|---|---|---|
| 1 | 90000 | MAX | IT |
| 2 | 90000 | JIM | IT |
| 3 | 80000 | HENRY | SALES |

## Hard-Level



| | Emp_ID | Emp_name | Salary |
|---|---|---|---|
| 1 | 1 | AA | 1000 |
| 2 | 2 | BB | 300 |
| 3 | 3 | CC | 500 |