Experiment-3

Name: Deepanshu Saini **UID: 23BCS13189**
Section : 23BCS_KRG_01a **Subject Name : ADBMS**
Subject Code:23CSP-333 **Date:21/08/2025**

1. **Aim**: To design and manipulate a **University Database** using SQL that involves creating relational tables for Students, Courses, Enrollments, and Professors, inserting and retrieving data using SubQueries

[Medium]
[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 Management Studio

3. Code:

Easy Problem

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

drop table employee
-- Create Employee Table
CREATE TABLE employee (
    id INT,
    name VARCHAR(50),
    salary INT,
    department_id INT,
    FOREIGN KEY (department_id) REFERENCES department(id)
);


-- Insert into Department Table
INSERT INTO department (id, dept_name) VALUES
(1, 'IT'),
```

```
(2, 'SALES');

-- Insert into Employee Table
INSERT INTO employee (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 d.dept_name, e.name, e.salary
from Employee as e
join department as d
on e.department_id = d.id
where e.salary in
(
    select max(e2.salary)
    from employee as e2
    group by e2.department_id
);

select d.dept_name, e.name, e.salary
from Employee as e
join department as d
on e.department_id = d.id
where e.salary in
(
    select max(e2.salary)
    from employee as e2
    where e2.department_id = e.department_id
)
```

Medium Problem

```
create table a1(
    empID varchar(50),
    eName varchar(50),
    salary int,
)

create table b1(
    empID varchar(50),
    eName varchar(50),
    salary int,
```

```
)

INSERT INTO a1 (empID, eName, salary) VALUES
(1, 'AA', 1000),
(2, 'BB', 300);

INSERT INTO b1 (empID, eName, salary) VALUES
(2, 'BB', 400),
(3, 'CC', 100);

WITH c1 AS (
    SELECT * FROM a1
    UNION
    SELECT * FROM b1
)
SELECT *
FROM c1
WHERE salary IN (
    SELECT MIN(salary)
    FROM c1
    GROUP BY empID
);
```

4. Output:

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

	empID	eName	salary
1	3	CC	100
2	2	BB	300
3	1	AA	1000

5. Learning Outcomes:

- By the end of this experiment, students will:
- Understand how to design a relational schema for a real-world university system.
- Practice creating and linking tables using SQL.
- Use JOINs to query multi-table data meaningfully.
- Implement data access control using GRANT/REVOKE.
- Handle transactions safely using COMMIT and ROLLBACK.