Experiment 3

Student Name: Amit Kumar Ghosh **UID:** 23BCS10436
Branch: C.S.E. **Section/Group:** Krg-3A
Semester: 5th **Date:** 20th August , 2025
Subject Name: ADBMS **Subject Code:** 23CSH-333

1. **Aim** :

   a. Generate an employee relation with only one attribute i.e., EMP_ID.Then, find the max EMP_ID, but excluding the duplicates.

   b.Create two tables, Department(ID, name) and Employees(ID, name, salary, deptID).Then output the highest earners from each department.

   c. Create two tables A and B with the attributes (EmpID, EmpName, Salary) and output the lowest salary of each employee across the two tables.

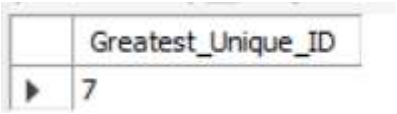
2. **Requirements (Hardware/Software):**
   My SQL server

3. **DBMS script and output**

   a.

```
-- a
CREATE TABLE TBL_EMPLOYEE(
   EMP_ID INT
);

INSERT INTO TBL_EMPLOYEE VALUES (2),(4),(4),(6),(6),(7),(8),(8);
SELECT MAX(EMP_ID) AS Greatest_Unique_ID
FROM TBL_EMPLOYEE
WHERE EMP_ID IN (
   SELECT EMP_ID
   FROM TBL_EMPLOYEE
   GROUP BY EMP_ID
   HAVING COUNT(EMP_ID) = 1
);
```

Output

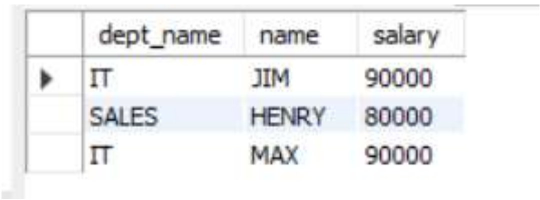


Greatest_Unique_ID
7

b.

```
-- b

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

CREATE TABLE employee (
    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 (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 e
INNER JOIN department d
    ON e.department_id = d.id
WHERE e.salary = (
    SELECT MAX(salary)
    FROM employee
    WHERE department_id = e.department_id
);
```

Output

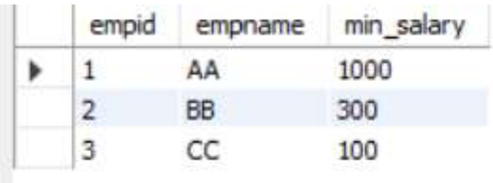


	dept_name	name	salary
▶	IT	JIM	90000
	SALES	HENRY	80000
	IT	MAX	90000

c.

```
--c
CREATE TABLE tbl_A (
    empid INT PRIMARY KEY,
    empname VARCHAR(20),
    salary INT
);

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

CREATE TABLE tbl_B (
    empid INT PRIMARY KEY,
    empname VARCHAR(20),
    salary INT
);

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

SELECT empid,
       MIN(empname) AS empname,
       MIN(salary) AS min_salary
FROM (
    SELECT * FROM tbl_A
    UNION
    SELECT * FROM tbl_B
) AS UNI
GROUP BY empid;
```

Output



empid	empname	min_salary
1	AA	1000
2	BB	300
3	CC	100

4. Learning Outcomes (What I have Learnt):

- Recognize how sub-queries help in breaking down and simplifying complex SQL operations.
- Implement sub-queries within SELECT, WHERE, and FROM clauses to extract precise data.
- Use sub-queries for tasks such as filtering, aggregation, and applying conditional logic.
- Evaluate the performance impact of using sub-queries compared to joins.