



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment 3

Name: Harshit Gupta
Branch: BE-CSE
Semester: 5th
SubjectCode: 23CSP-333

UID: 23BCS10228
Section/Group: Krg-3A
Subject: ADBMS
Date: 21 / 08 / 2025

Aim: **EASY LEVEL**

YOU ARE GIVEN WITH EMPLOYEE RELATION HAVING ONLY ONE ATTRIBUTE NAMED AS EMP_ID. YOUR TASK IS TO FETCH THE MAXIMUM EMP_ID, BUT EXCLUDING THE DUPLICATE VALUES.

INPUT: EMPLOYEE (EMP_ID)
 2
 4
 4
 6
 6
 7
 8
 8
 8

OUTPUT: 7

Code:

```
CREATE TABLE EMPLOYEE(  
EMP_ID INT  
);
```

```
INSERT INTO EMPLOYEE  
VALUES  
(2),  
(4),  
(4),  
(6),  
(7),
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

(8),
(8),
(8);

```
SELECT MAX(EMP_ID) FROM EMPLOYEES WHERE EMP_ID  
NOT IN  
(SELECT EMP_ID FROM EMPLOYEE  
GROUP BY EMP_ID  
HAVING COUNT(EMP_ID) > 1 )
```

Output:



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

DEPARTMENT SALARY CHAMPIONS (MEDIUM)

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.

Code:

-- Create Department Table

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

-- Create Employee Table

```
CREATE TABLE employee_new (  
    id INT,  
    name VARCHAR(50),  
    salary INT,  
    department_id INT,  
    FOREIGN KEY (department_id) REFERENCES department_new(id)  
);
```

-- Insert into Department Table

```
INSERT INTO department_new (id, dept_name) VALUES  
(1, 'IT'),  
(2, 'SALES');
```

-- Insert into Employee Table

```
INSERT INTO employee_new (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);
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

-- CO RELATED SUB QUERY

```
SELECT D.DEPT_NAME,E.NAME, E.SALARY
FROM Employee_new as E
INNER JOIN
Department_new as D
ON E.department_id=D.id
WHERE E.SALARY IN -- 90000
(
  SELECT MAX(SALARY)
  FROM employee_new
  WHERE department_id = E.department_id -- 90000
)

ORDER BY D.dept_name;
```

Output:

100 %	4	0	↑	↓
Results	Messages			
	DEPT_NAME	NAME	SALARY	
1	IT	MAX	90000	
2	IT	JIM	90000	
3	SALES	HENRY	80000	



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

MERGING EMPLOYEE HISTORY: 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.

Code:

```
CREATE TABLE A(  
  EMPID INT,  
  ENAME VARCHAR(20),  
  Salary INT  
);
```

```
CREATE TABLE B(  
  EMPID INT,  
  ENAME VARCHAR(20),  
  Salary INT  
);
```

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

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



```
SELECT * FROM A  
UNION ALL  -- HERE, WE CAN ALSO USE UNION  
SELECT * FROM B  
WHERE SALARY IN  
(SELECT MIN(SALARY) FROM B);
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Output:

 Results		 Messages	
	EMPID	ENAME	Salary
1	1	AA	1000
2	2	BB	300
3	3	CC	100