# Experiment 3

**Student Name: Jyoti Kumari**
**UID: 23BCS10877**
**Branch: BE-CSE**
**Section/Group: KRG-3B**
**Semester: 5**
**Date of performance: 22-08-2025**
**Subject name: ADBMS**
**Subject code: 23CSP-333**

**Aim:**

1. Max Value without Duplicates [EASY]

   • Create a table of Employee IDs.

   • Insert sample IDs (with duplicates).

   • Write a query to return the maximum EmpID excluding duplicate values using subqueries.

2. Department Salary Champions [MEDIUM]

   • Create dept and employee tables with a relationship.

   • Insert sample department and employee data.

   • Use subqueries to find the employee(s) with the highest salary in each department.

   • If multiple employees share the max salary in a department, include all.

3. Merging Employee Histories: Who Earned Least? [HARD]

   • Create two legacy tables (TableA and TableB).

   • Insert sample records (some overlapping).

   • Merge both tables and find the minimum salary per employee using subqueries.

**Code:**

```
--easy question

/*

GENERATE AN EMPLOYEE RELATIN WITH ONLY A ONE ATTRIBUTE I.E, EMP_ID
```

```
TASK: DIND THE MAX EMP_ID, BUT EXCLUDING THE DUPLICATES
*/
CREATE TABLE EMPLOYEE(
EMPID INT
);
INSERT INTO EMPLOYEE(EMPID) VALUES
(1),
(1),
(2),
(2),
(5),
(5),
(6),
(7),
(8),
(8);
SELECT MAX(EMPID) AS [MAX_UNIQUE] FROM Employee WHERE EmpID IN (SELECT
EmpID FROM Employee GROUP BY EmpID HAVING count(EmpID)=1);
CREATE TABLE TBL_PRODUCTS
(
        ID INT PRIMARY KEY IDENTITY,
        [NAME] NVARCHAR(50),
        [DESCRIPTION] NVARCHAR(250)
)
CREATE TABLE TBL_PRODUCTSALES
(
        ID INT PRIMARY KEY IDENTITY,
        PRODUCTID INT FOREIGN KEY REFERENCES TBL_PRODUCTS(ID),
```

```
        UNITPRICE INT,
        QUALTITYSOLD INT
)
INSERT INTO TBL_PRODUCTS VALUES ('TV','52 INCH BLACK COLOR LCD TV')
INSERT INTO TBL_PRODUCTS VALUES ('LAPTOP','VERY THIIN BLACK COLOR ACER LAPTOP')
INSERT INTO TBL_PRODUCTS VALUES ('DESKTOP','HP HIGH PERFORMANCE DESKTOP')

INSERT INTO TBL_PRODUCTSALES VALUES (3,450,5)
INSERT INTO TBL_PRODUCTSALES VALUES (2,250,7)
INSERT INTO TBL_PRODUCTSALES VALUES (3,450,4)
INSERT INTO TBL_PRODUCTSALES VALUES (3,450,9)

SELECT *FROM TBL_PRODUCTS
SELECT *FROM TBL_PRODUCTSALES
/*
TASK:FIND THE ID, NAME ,DESCRIPTION OF PRODUCT WHICH HAS NOT BEEN SOLD FOR ONCE.
*/
SELECT ID,[NAME],[DESCRIPTION] FROM TBL_PRODUCTS WHERE ID NOT IN
(SELECT DISTINCT PRODUCTID FROM TBL_PRODUCTSALES);

--USING JOIN
SELECT T.*,P.* FROM TBL_PRODUCTS AS T LEFT JOIN
TBL_PRODUCTSALES AS P
ON T.ID=P.PRODUCTID
WHERE PRODUCTID IS NULL;
```

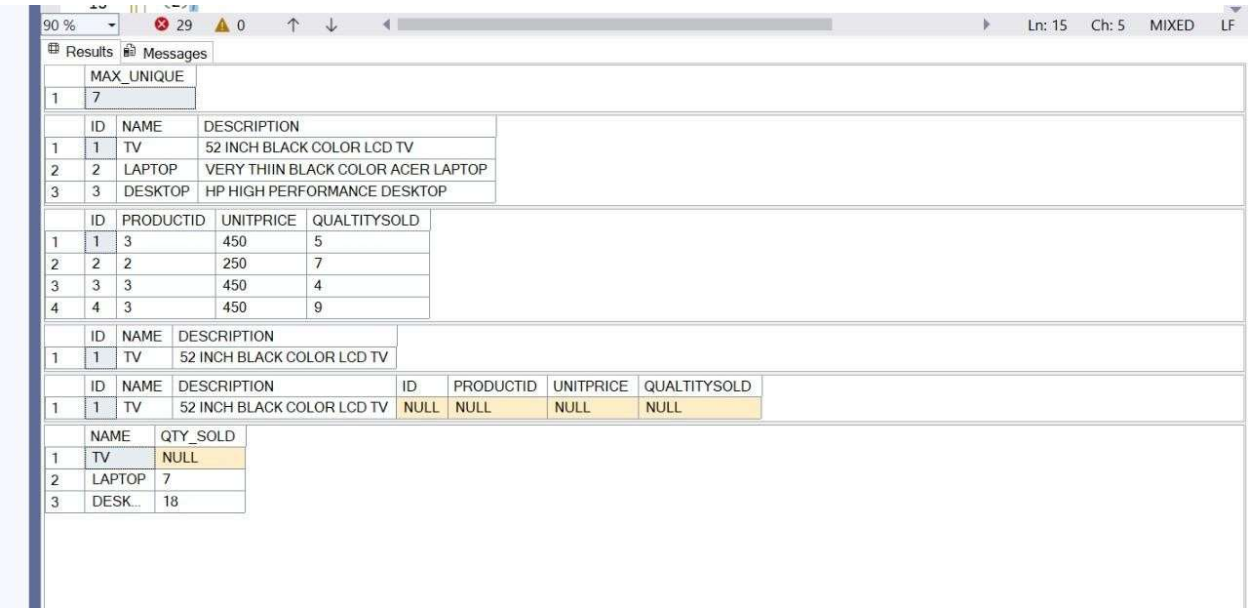
```
/*
TASK: FIND THE TOTAL QUANTITY SOLD FOR EACH RESPECTIVE PRODUCT
*/
SELECT T.NAME, (SELECT SUM(QUALTITYSOLD) FROM TBL_PRODUCTSALES WHERE PRODUCTID=T.ID) AS QTY_SOLD FROM TBL_PRODUCTS AS T;
create database exp4;
use exp4;
--medium
- ---------- EXPERIMENT 03: (MEDIUM LEVEL)
CREATE TABLE department (
    id INT PRIMARY KEY,
    dept_name VARCHAR(50)
);
-- 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);
--main
select d.dept_name,e.name,e.salary from employee as e
inner join department as d on d.id=e.department_id
where e.salary in(
select MAX(e2.salary)
from employee as e2
where e2.department_id=e.department_id
)
order by  dept_name;
-- group by approach
select d.dept_name,e.name,e.salary from employee as e
inner join department as d on d.id=e.department_id
where e.salary in(
select MAX(e2.salary)
from employee as e2
group by e2.department_id
);
--Hard Question
CREATE TABLE TableA (
    Empid INT,
    Ename VARCHAR(50),
    Salary INT
);
```

```
CREATE TABLE TableB (
    Empid INT,
    Ename VARCHAR(50),
    Salary INT
);
INSERT INTO TableA VALUES (1, 'AA', 1000), (2, 'BB', 300);
INSERT INTO TableB VALUES (2, 'BB', 400), (3, 'CC', 100);
--TIP; AFF OVER NUMBER DATA ONLY IS WRONG
--TAKE FIRST LETTER OF EMPNAME WILL CONVERT IN ASCII
select empid, ename ,MIN(salary) AS salary from(
select * from tableA as a
union all
select * from tableB as b
) as INTERMIDIATE_RESULT
group by empid,ename;
```

**OUTPUT:**

**EASY:**



90 % 29 0 Ln: 15 Ch: 5 MIXED LF

Results Messages

| | MAX_UNIQUE |
|---|---|
| 1 | 7 |

| | ID | NAME | DESCRIPTION |
|---|---|---|---|
| 1 | 1 | TV | 52 INCH BLACK COLOR LCD TV |
| 2 | 2 | LAPTOP | VERY THIIN BLACK COLOR ACER LAPTOP |
| 3 | 3 | DESKTOP | HP HIGH PERFORMANCE DESKTOP |

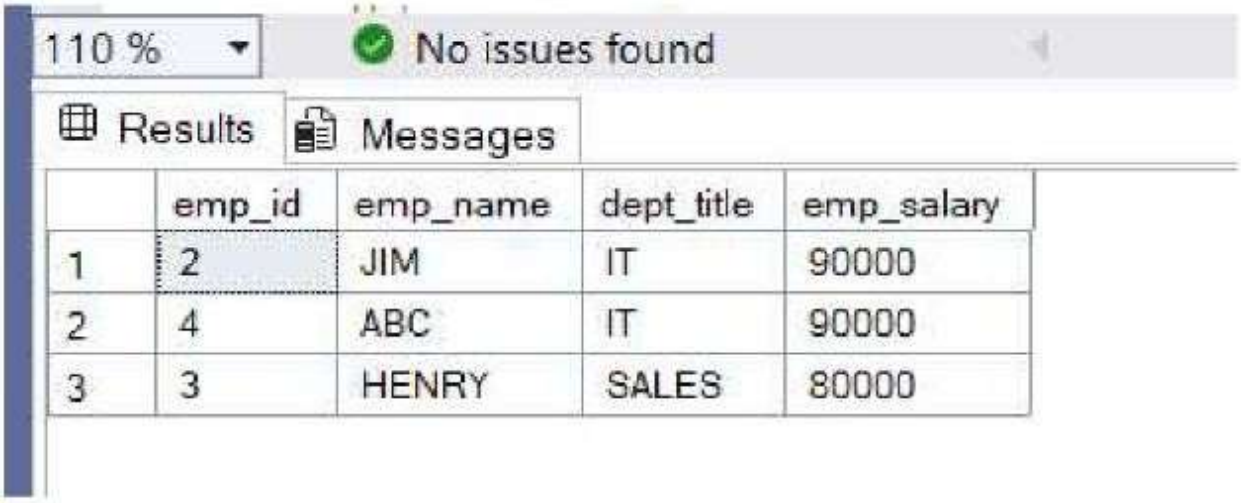
| | ID | PRODUCTID | UNITPRICE | QUALTITYSOLD |
|---|---|---|---|---|
| 1 | 1 | 3 | 450 | 5 |
| 2 | 2 | 2 | 250 | 7 |
| 3 | 3 | 3 | 450 | 4 |
| 4 | 4 | 3 | 450 | 9 |

| | ID | NAME | DESCRIPTION |
|---|---|---|---|
| 1 | 1 | TV | 52 INCH BLACK COLOR LCD TV |

| | ID | NAME | DESCRIPTION | ID | PRODUCTID | UNITPRICE | QUALTITYSOLD |
|---|---|---|---|---|---|---|---|
| 1 | 1 | TV | 52 INCH BLACK COLOR LCD TV | NULL | NULL | NULL | NULL |

| | NAME | QTY_SOLD |
|---|---|---|
| 1 | TV | NULL |
| 2 | LAPTOP | 7 |
| 3 | DESK... | 18 |

**MEDIUM**




| | emp_id | emp_name | dept_title | emp_salary |
|---|---|---|---|---|
| 1 | 2 | JIM | IT | 90000 |
| 2 | 4 | ABC | IT | 90000 |
| 3 | 3 | HENRY | SALES | 80000 |

**HARD**

| | EmpID | Ename | Min_Salary |
|---|---|---|---|
| 1 | 1 | AA | 1000 |
| 2 | 2 | BB | 300 |
| 3 | 3 | CC | 100 |