Experiment – 3

Student Name: Gauri Singh	Student UID 23BCS13437
Branch: BE-CSE	Section/Group: KRG_1A
Semester: 5th	Date of Performan 21-08-25
Subject Name: ADBMS	Subject Code: 23CSP-333

1. **Aim:**

Q1) To understand and implement subqueries in MySQL for solving complex database queries.

Q2) 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.

Q3) legacy HR systems ,a and b, have seperate records of employee salaries. these records may overlap. management wnts to merge these 2 datasets and identify each unique employee(by empid) along with lowest recorded salary across both systems.

   objective:
   1) combine both tables a and b using join
   2) return each empid with their lowest salary and corresponding ename

2. **DBMS code:**

   Q1)

```
--MADE BY GAURI SINGH

USE KRG_1A

create table employee (
    emp_id int,
    empname varchar(50),
    gender varchar(10),
    salary int,
    city varchar(50),
```

```sql
        dept_id int
);

insert into employee(emp_id, empname, gender, salary, city, dept_id)
values
(1, 'amit', 'male', 50000, 'delhi', 2),
(2, 'priya', 'female', 60000, 'mumbai', 1),
(3, 'rajesh', 'male', 45000, 'agra', 3),
(4, 'sneha', 'female', 55000, 'delhi', 4),
(5, 'anil', 'male', 52000, 'agra', 2),
(6, 'sunita', 'female', 48000, 'mumbai', 1),
(7, 'vijay', 'male', 47000, 'agra', 3),
(8, 'ritu', 'female', 62000, 'mumbai', 2),
(8, 'alok', 'male', 51000, 'delhi', 1),
(9, 'neha', 'female', 53000, 'agra', 4),
(9, 'simran', 'female', 33000, 'agra', 3);

SELECT MAX(emp_id) AS id
FROM (
    SELECT emp_id
    FROM employee
    GROUP BY emp_id
    HAVING COUNT(emp_id) = 1
      ) AS unique_emp_ids;
--MADE BY YASH SHARMA
```

Q2)

```sql
--MADE BY GAURI SINGH
USE KRG_1A

CREATE TABLE employees (
    id INT,
    name VARCHAR(50),
    salary INT,
    dept_id INT
);

CREATE TABLE departments (
    id INT,
    dept_name VARCHAR(50)
);

INSERT INTO employees VALUES (1, 'joe', 70000, 1);
INSERT INTO employees VALUES (2, 'jim', 90000, 1);
INSERT INTO employees VALUES (3, 'henry', 80000, 2);
INSERT INTO employees VALUES (4, 'sam', 60000, 2);
INSERT INTO employees VALUES (4, 'max', 90000, 1);

INSERT INTO departments VALUES (1, 'it');
INSERT INTO departments VALUES (2, 'sales');

SELECT d.dept_name, e.name, e.salary
FROM employees e
JOIN departments d ON e.dept_id = d.id
WHERE e.salary = (
```

```sql
        SELECT MAX(salary)
        FROM employees e2
        WHERE e2.dept_id = e.dept_id
    )
        ORDER BY d.dept_name;
    --MADE BY YASH SHARMA
```

Q3)

```sql
    --MADE BY GAURI SINGH
    USE KRG_1A
    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, 'xyza', 1000);
    INSERT INTO A VALUES (2, 'bcde', 300);

    INSERT INTO B VALUES (2, 'bcde', 400);
    INSERT INTO B VALUES (1, 'xyza', 100);

    SELECT EmpID, Ename, MIN(Salary) AS Salary
    FROM (
        SELECT * FROM A
        UNION ALL
        SELECT * FROM B
    ) merged
    GROUP BY EmpID, Ename
    ORDER BY EmpID;
    --MADE BY YASH SHARMA
```

3. **Output:**

Q1)

	id
1	7

✓ Query executed successfully.

Q2)

	dept_name	name	salary
1	it	jim	90000
2	it	max	90000
3	sales	henry	80000

✓ Query executed successfully.

Q3)

Results | Messages

	EmpID	Ename	Salary
1	1	xyza	100
2	2	bcde	300

✓ Query executed successfully.