Experiment 1

Student Name: Riya Mittal   **UID: 23BCS10539**

Branch: B.E.CSE   **Section/Group: 23BCS-KRG-3B**

Semester: 5th   **Date of Performance:28-07-25**

Subject Name: ADBMS   **Subject Code:23CSP-333**

1. Aim:

- Problem Title: Department-Course Subquery and Access Control

-- Procedure (Step-by-Step):

-- Design normalized tables for departments and the courses they offer, maintaining a foreign key relationship.

-- Insert five departments and at least ten courses across those departments.

-- Use a subquery to count the number of courses under each department.

-- Filter and retrieve only those departments that offer more than two courses.

-- Grant SELECT-only access on the courses table to a specific user

2. DBMS script and output:

```
CREATE TABLE Departments (
    DeptID INT PRIMARY KEY,
    DeptName VARCHAR(100)
);


CREATE TABLE Courses (
    CourseID INT PRIMARY KEY,
    CourseName VARCHAR(100),
    DeptID INT,
    FOREIGN KEY (DeptID) REFERENCES Departments(DeptID)
);
```

```sql
INSERT INTO Departments VALUES
(1, 'Computer Science'),
(2, 'Mechanical Engineering'),
(3, 'Electrical Engineering'),
(4, 'Civil Engineering'),
(5, 'Mathematics');


INSERT INTO Courses VALUES
(101, 'Data Structures', 1),
(102, 'Operating Systems', 1),
(103, 'Database Systems', 1),
(104, 'Thermodynamics', 2),
(105, 'Fluid Mechanics', 2),
(106, 'Circuit Analysis', 3),
(107, 'Digital Electronics', 3),
(108, 'Structural Engineering', 4),
(109, 'Calculus', 5),
(110, 'Linear Algebra', 5);


SELECT DeptName
FROM Departments
WHERE DeptID IN (
    SELECT DeptID
    FROM Courses
    GROUP BY DeptID
    HAVING COUNT(*) > 2
);


GRANT SELECT ON Courses TO [username];
```

3. Output:

DeptName
Computer Science

4. Learning outcomes:

- You will be able to write basic SQL queries.

- You will learn to perform JOINS in SQL.
- You will understand how to implement foreign k