

Experiment 1.1

Student Name: Raj Gupta UID: 23BAI70387

Branch: BE-AIT-CSE Section/Group: 23AML-1 (B)

Semester: 5th Date of Performance: 22 July 2025

Subject Name: ADBMS Subject Code: 23CSP-333

1. Experiment Name:

To design and manipulate a University Database using SQL that involves creating relations Tables for Students, Courses, Enrollments and Professors, inserting and retrieving data Using JOINs, managing access control with GRANT/REVOKE, and handling transactions Control using COMMIT and ROLLBACK.

2. Objective:

Easy-Level Problem

Problem Title: Author-Book Relationship Using Joins and Basic SQL Operations Procedure (Step-by-Step):

Design two tables — one for storing author details and the other for book details.

- 1. Ensure a foreign key relationship from the book to its respective author.
- 2. Insert at least three records in each table.
- 3. Perform an INNER JOIN to link each book with its author using the common author ID.
- 4. Select the book title, author name, and author's country.

Medium-Level Problem

Problem Title: Department-Course Subquery and Access Control Procedure (Step-by-Step):

- 1. Design normalised tables for departments and the courses they offer, maintaining a foreign key relationship.
- 2. Insert five departments and at least ten courses across those departments.

- 3. Use a subquery to count the number of courses under each department.
- 4. Filter and retrieve only those departments that offer more than two courses.
- 5. Grant SELECT-only access on the courses table to a specific user.

```
3. Code:
 ----EASY LEVEL PROBLEM CODE----
    USE DB DEMO
    CREATE TABLE Author (
      AuthorID INT PRIMARY KEY,
      AuthorName VARCHAR(100),
      Country VARCHAR(100)
    );
    CREATE TABLE Book (
    BookID INT PRIMARY KEY,
    Title VARCHAR(100),
    AuthorID INT,
    FOREIGN KEY (AuthorID) REFERENCES Author(AuthorID)
    );
    INSERT INTO Author (AuthorID, AuthorName, Country) VALUES
    (1, 'J.K. Rowling', 'United Kingdom'),
    (2, 'George R.R. Martin', 'United States'),
    (3, 'Haruki Murakami', 'Japan');
    INSERT INTO Book (BookID, Title, AuthorID) VALUES
    (101, 'Harry Potter and the Sorcerer's Stone', 1),
    (102, 'A Game of Thrones', 2),
    (103, 'Norwegian Wood', 3);
    SELECT
    Book.Title AS BookTitle,
    Author.AuthorName,
    Author.Country
    FROM
    Book
    INNER JOIN
    Author ON Book.AuthorID = Author.AuthorID;
----MEDIUM LEVEL PROBLEM CODE----
   CREATE TABLE Department (
   DepartmentID INT PRIMARY KEY,
   DepartmentName VARCHAR(100)
```

);

```
CREATE TABLE Course (
CourseID INT PRIMARY KEY,
CourseName VARCHAR(100),
DepartmentID INT,
FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
);
-- Insert departments
INSERT INTO Department (DepartmentID, DepartmentName) VALUES
(1, 'Computer Science'),
(2, 'Mechanical Engineering'),
(3, 'Electrical Engineering'),
(4, 'Mathematics'),
(5, 'Civil Engineering');
-- Insert courses
INSERT INTO Course (CourseID, CourseName, DepartmentID) VALUES
(101, 'Data Structures', 1),
(102, 'Operating Systems', 1),
(103, 'DBMS', 1),
(104, 'Thermodynamics', 2),
(105, 'Fluid Mechanics', 2),
(106, 'Circuit Theory', 3),
(107, 'Digital Electronics', 3),
(108, 'Linear Algebra', 4),
(109, 'Calculus', 4),
(110, 'Structural Analysis', 5);
SELECT
  DepartmentName,
  (SELECT COUNT(*)
  FROM Course
  WHERE Course.DepartmentID = Department.DepartmentID) AS CourseCount
FROM Department;
SELECT
  DepartmentName
FROM
  Department
WHERE
  (SELECT COUNT(*)
  FROM Course
  WHERE Course.DepartmentID = Department.DepartmentID) > 2;
GRANT SELECT ON Course TO readonly user;
```

-- Course table with foreign key to Department

4.Output:

----easy level problem output----

⊞ Re	esults 🗐 Me	essages	
	AuthorID	AuthorName	Country
1	1	J.K. Rowling	United Kingdom
2	2	George R.R. Martin	United States
3	3	Haruki Murakami	Japan

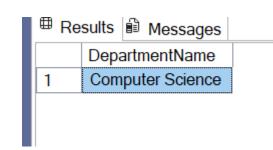
		BookID	Title	AuthorID
l	1	101	Harry Potter and the Sorcerer's Stone	1
l	2	102	A Game of Thrones	2
	3	103	Norwegian Wood	3

■ Results			
	BookTitle	AuthorName	Country
1	Harry Potter and the Sorcerer's Stone	J.K. Rowling	United Kingdom
2	A Game of Thrones	George R.R. Martin	United States
3	Norwegian Wood	Haruki Murakami	Japan

----medium level problem output----

⊞ R	esults 🖺 Me	essag	jes		
	CourseID	Cou	ırseName	DepartmentID	
1	101	Dat	ta Structures	1	
2	102	Ope	erating Systems	1	
3	103	DB	MS	1	
4	104	The	ermodynamics	2	
5	105	Flui	id Mechanics	2	
6	106	Circ	cuit Theory	3	
7	107	Dig	ital Electronics	3	
8	108	Line	ear Algebra	4	
9	109	Calculus		4	
10	110	Structural Analysis		5	
	Departmen	tID	DepartmentNar	ne	
1	1	Computer Scien		nce	
2	2		Mechanical Eng	gineering	
3	3		Electrical Engin	eering	
4	4		Mathematics		

⊞ Re	esults 🛍 Messages	
	DepartmentName	CourseCount
1	Computer Science	3
2	Mechanical Engineering	2
3	Electrical Engineering	2
4	Mathematics	2
5	Civil Engineering	1



4.Learning Outcomes:

- Understanding Database Normalization:
- Mastering Subqueries in SQL:
- Filtering Data Using Conditions:
- Data Access Control:
- Working with One-to-Many Relationships: