



## Experiment 1

**Student Name:** ROHIT KUMAR

**Branch:** CSE

**Semester:** 5th

**Subject Name:** ADBMS

**UID:** 23BCS12640

**Section/Group:** KRG 3-A

**Date of Performance:** 24/07/2025

**Subject Code:** 23CSP-333

**1. Aim:** University Database System helps in managing student enrollments, course allocations, and professor assignments effectively. The system also demonstrates secure access control and transaction safety. This includes CRUD operations, JOIN queries, and database-level user permission management.

- a. Author-Book Relationship Using Joins and Basic SQL Operations
- b. Department-Course Subquery and Access Control

## **2. Objective:**

- To create and manage relational databases LibraryDB and UniversityDB using SQL.
- To define tables with appropriate primary and foreign key constraints.
- To insert sample data into author, book, department, and course tables.
- To retrieve related data using **INNER JOIN** and **subqueries** with GROUP BY and HAVING.
- To manage user access by granting **SELECT privileges** on specific tables.

## **3. DBMS script and output:**

### **Solution-(a)**

```
-- Create the database
CREATE DATABASE LibraryDBM;
USE LibraryDBM;

-- Create the Authors table
CREATE TABLE Authors (
    authorID INT PRIMARY KEY,
    fullName VARCHAR(100),
    nationality VARCHAR(50)
);

-- Create the Books table
CREATE TABLE Books (
    bookID INT PRIMARY KEY,
    bookTitle VARCHAR(100),
    writtenBy INT,
    FOREIGN KEY (writtenBy) REFERENCES Authors(authorID)
);

-- Insert authors
INSERT INTO Authors (authorID, fullName, nationality) VALUES
(1, 'Paulo Coelho', 'Brazil'),
(2, 'Chimamanda Ngozi Adichie', 'Nigeria'),
(3, 'Kazuo Ishiguro', 'United Kingdom');

-- Insert books
INSERT INTO Books (bookID, bookTitle, writtenBy) VALUES
(101, 'The Alchemist', 1),
(102, 'Half of a Yellow Sun', 2);
```

# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

```
(103, 'Never Let Me Go', 3);
```

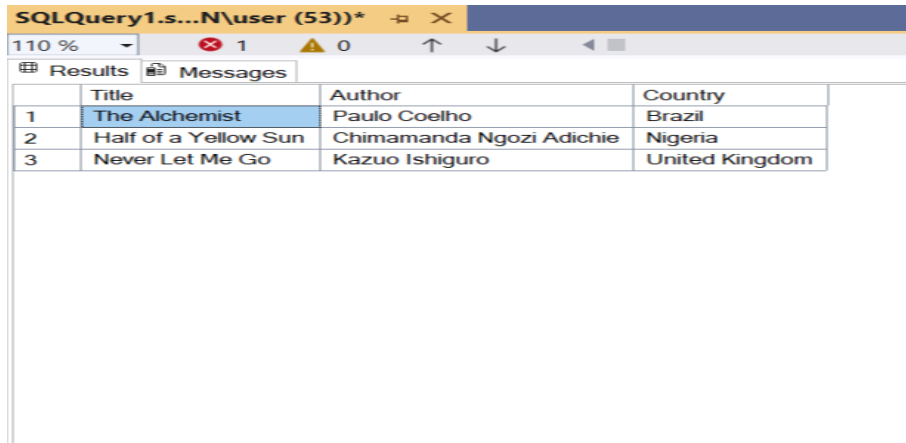
```
-- Join and display results
```

```
SELECT
```

```
    B.bookTitle AS Title,  
    A.fullName AS Author,  
    A.nationality AS Country
```

```
FROM
```

```
    Books B  
  INNER JOIN Authors A ON B.writtenBy = A.authorID;
```



	Title	Author	Country
1	The Alchemist	Paulo Coelho	Brazil
2	Half of a Yellow Sun	Chimamanda Ngozi Adichie	Nigeria
3	Never Let Me Go	Kazuo Ishiguro	United Kingdom

## Solution-(b)

```
CREATE DATABASE UniversityDB;
```

```
USE UniversityDB;
```

```
CREATE TABLE Departments (  
    departmentId INT PRIMARY KEY,  
    departmentName VARCHAR(100)  
);
```

```
CREATE TABLE Courses (  
    courseId INT PRIMARY KEY,  
    courseTitle VARCHAR(100),  
    departmentId INT,  
    FOREIGN KEY (departmentId) REFERENCES Departments(departmentId)  
);
```

```
INSERT INTO Departments (departmentId, departmentName) VALUES  
(1, 'Information Technology'),  
(2, 'Aerospace Engineering'),  
(3, 'Chemical Engineering'),  
(4, 'Architecture'),  
(5, 'Statistics');
```

```
INSERT INTO Courses (courseId, courseTitle, departmentId) VALUES  
(201, 'Web Development', 1),  
(202, 'Computer Networks', 1),  
(203, 'Cloud Computing', 1),  
(204, 'Flight Mechanics', 2),  
(205, 'Propulsion Systems', 2),  
(206, 'Organic Chemistry', 3),  
(207, 'Process Control', 3),  
(208, 'Building Design', 4),  
(209, 'Probability Theory', 5),  
(210, 'Regression Analysis', 5);
```

```
CREATE LOGIN studentUser WITH PASSWORD = 'securePass123';  
USE UniversityDB;  
CREATE USER studentUser FOR LOGIN studentUser;
```

# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

GRANT SELECT ON dbo.Courses TO studentUser;

```
SELECT departmentName
FROM Departments
WHERE departmentId IN (
    SELECT departmentId
    FROM Courses
    GROUP BY departmentId
    HAVING COUNT(courseId) > 2
);
```

The screenshot shows a SQL IDE with a query editor and a results pane. The query editor contains the following SQL code:

```
9 );
10
11 CREATE TABLE Courses (
12     courseId INT PRIMARY KEY,
13     courseTitle VARCHAR(100),
14     departmentId INT,
15     FOREIGN KEY (departmentId) REFERENCES
16 );
17
18
19 INSERT INTO Departments (departmentId, d
```

The results pane shows a table with the following data:

	departmentName
1	Information Technology

#### **4. Learning Outcomes (What I have Learnt):**

- Understand how to design relational databases using **primary and foreign key** constraints.
- Gain hands-on experience with **SQL DDL and DML** commands for creating and manipulating tables.
- Learn to use **INNER JOINS** to combine data from related tables.
- Apply **subqueries** with aggregation (GROUP BY, HAVING) to filter complex data sets.
- Learn how to **grant user privileges** using the GRANT statement for controlled access.