DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING

CU CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

Experiment 1

Student Name: Sukhjinder Singh **UID: 23BAI70078**

Branch: BE-AIT-CSE **Section/Group:** 23AIT-KRG-G2

Semester: 5th **Date of Performance:** 29th July, 2025

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

1. **AIM: To design and manipulate a University Database using SQL that involves creating relational tables for Students, Courses, Enrollments, and Professors, inserting and retrieving data using JOINs, managing access control with GRANT/REVOKE, and handling transaction control using COMMIT and ROLLBACK.**

2. **Tools Used:**
   MySQL Workbench in VS CODE.

3. **Experiment: 1. Easy-Level Problem: Author-Book Relationship Using Joins and Basic SQL Operations.**
   - **Design two tables — one for storing author details and the other for book details.**
   - **Ensure a foreign key relationship from the book to its respective author.**
   - **Insert at least three records in each table.**
   - **Perform an INNER JOIN to link each book with its author using the common**
   - **Select the book title, author name, and author's country.**

   Medium-Level Problem: Department-Course Subquery and Access Control.
   - **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.**

4. Solution:

Easy-Level

```
----------------EXPERIMENT-1------------

----------------EASY-LEVEL---------------

CREATE DATABASE ADBMS_1;
USE ADBMS_1;

create table author(
    AUTHOR_ID int primary key,
    AUTHOR_NAME varchar(20),
    AUTHOR_Age int,
    Author_Gender char(1)
);

create table book_table(
    BOOK_ID int primary key,
    BOOK_NAME varchar(20),
    AUTHOR_ID int,
    foreign key(AUTHOR_ID) references author(AUTHOR_ID)
);

insert into author values(554, 'Ruskin Bond', 43, 'M'), (130, 'Robert Greene', 37, 'M'), (145, 'Zadie Smith', 23, 'F'), (786, 'Arundhati Khan', 50, 'F');
insert into author values(250, 'Robert Frost', 60, 'M'), (120, 'Schewa Zaitsev', 25, 'F'), (200, 'J.K. Rowling', 55, 'F');
alter table author add COUNTRY varchar(20);
update author set COUNTRY = 'India' where AUTHOR_ID = 786;
update author set COUNTRY = 'USA' where AUTHOR_ID in (554, 130, 250);
update author set COUNTRY = 'Australia' where AUTHOR_ID in (145, 200);
update author SET COUNTRY = 'Russia' WHERE AUTHOR_ID = 120;
select A.AUTHOR_ID as 'Author Id', A.AUTHOR_NAME as 'Author Name', A.COUNTRY as 'Country' from author as A;
update author SET AUTHOR_NAME = 'Aruna Nair' WHERE AUTHOR_ID = 786;
select * from author;
```

Medium-Level

```


------------------MEDIUM-LEVEL------------------


CREATE TABLE dept (
    Dept_Id SMALLINT IDENTITY(1,1) PRIMARY KEY,
    Dept_Name VARCHAR(12) NOT NULL
);

create table course(
    Dept SMALLINT ,
    FOREIGN KEY(Dept) references dept(Dept_Id),
    Course varchar(12)
);

exec sp_help course;
insert into dept(Dept_Name) values('AI&ML'), ('CSE'), ('Bio-Tech'), ('Finance'), ('Psychology');
select * from dept;

ALTER TABLE course
ALTER COLUMN Course VARCHAR(30);
```

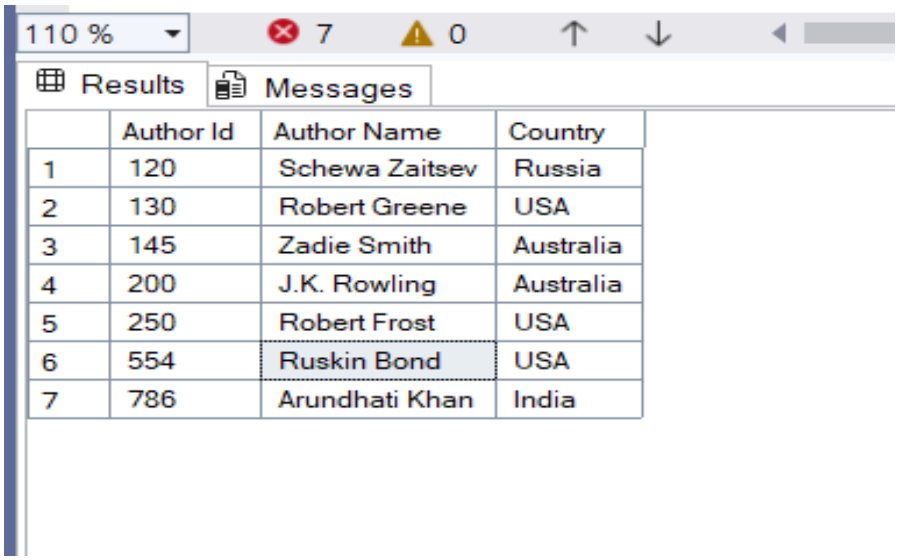
```sql

INSERT INTO course VALUES
(1, 'Data Science'), (1, 'Neural Networks'),
(1, 'Machine Learning'), (1, 'AI'),
(2, 'Data Analytics'), (2, 'Data Mining'),
(2, 'Full Stack Development'), (2, 'Web Development'),
(3, 'Cyber Security'), (3, 'Network Security'),
(3, 'Bioinformatics'), (3, 'Genetics'),
(3, 'Biology'),
(2, 'Full Stacks'),
(4, 'Economics'), (4, 'Socio-Psycho'),
(5, 'Socio-Psycho'), (5, 'Psychology');

SELECT
    C.Dept,
    C.Course,
    D.Dept_Name AS [Department Name]
FROM
    course AS C
LEFT JOIN
    dept AS D ON C.Dept = D.Dept_Id

UNION

SELECT
    C.Dept,
    C.Course,
    D.Dept_Name AS [Department Name]
FROM
    course AS C
RIGHT JOIN
```

```sql
SELECT
    D.Dept_Name AS Department,
    D.Dept_Id,
    C.CourseCount AS [COUNT]
FROM
    dept AS D
INNER JOIN (
    SELECT
        Dept,
        COUNT(Dept) AS CourseCount
    FROM
        course
    GROUP BY
        Dept
) AS C
ON D.Dept_Id = C.Dept;

SELECT
    D.Dept_Name AS Department,
    D.Dept_Id
FROM
    dept AS D
INNER JOIN (
    SELECT
        Dept,
        COUNT(Dept) AS [CourseCount]
    FROM
        course
    GROUP BY
        Dept
) AS C
ON D.Dept_Id = C.Dept
WHERE
```

```sql
WHERE
    C.CourseCount >= 2;

CREATE LOGIN SG_user WITH PASSWORD = '1234';

CREATE USER SG_user FOR LOGIN SG_user;

GRANT SELECT ON course TO SG_user;
```

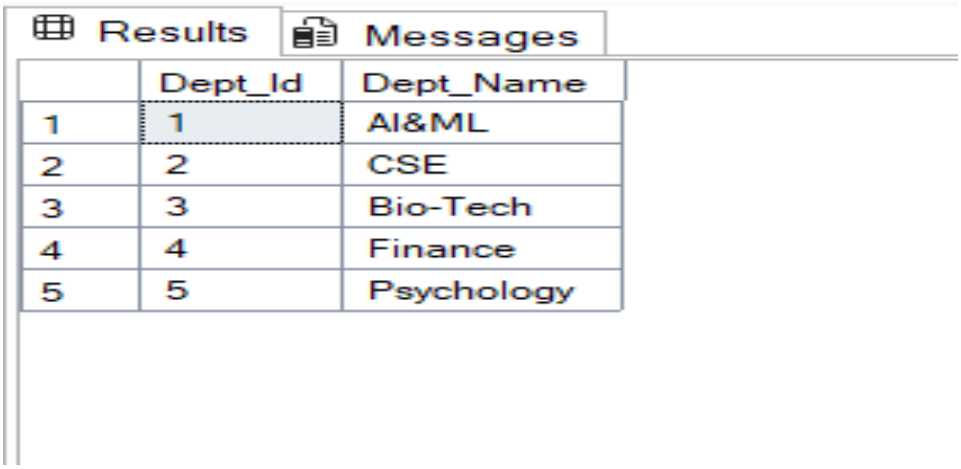
5. **Output:**

Easy-Level



	Author Id	Author Name	Country
1	120	Schewa Zaitsev	Russia
2	130	Robert Greene	USA
3	145	Zadie Smith	Australia
4	200	J.K. Rowling	Australia
5	250	Robert Frost	USA
6	554	Ruskin Bond	USA
7	786	Arundhati Khan	India

Medium-Level



	Dept_Id	Dept_Name
1	1	AI&ML
2	2	CSE
3	3	Bio-Tech
4	4	Finance
5	5	Psychology

Results | Messages

	Dept	Course	Department Name
1	1	AI	AI&ML
2	1	Data Science	AI&ML
3	1	Machine Learning	AI&ML
4	1	Neural Networks	AI&ML
5	2	Data Analytics	CSE
6	2	Data Mining	CSE
7	2	Full Stack Development	CSE
8	2	Full Stacks	CSE
9	2	Web Development	CSE
10	3	Bioinformatics	Bio-Tech
11	3	Biology	Bio-Tech
12	3	Cyber Security	Bio-Tech
13	3	Genetics	Bio-Tech
14	3	Network Security	Bio-Tech
15	4	Economics	Finance
16	4	Socio-Psycho	Finance
17	5	Psychology	Psychology

Query executed successfully.

Results | Messages

	Department	Dept_Id	COUNT
1	AI&ML	1	4
2	CSE	2	5
3	Bio-Tech	3	5
4	Finance	4	2
5	Psychology	5	2

Results | Messages

	Department	Dept_Id
1	AI&ML	1
2	CSE	2
3	Bio-Tech	3
4	Finance	4
5	Psychology	5

6. Outcomes:

- Learnt about SQL Basic Operations.
- Learnt about various types of JOINS such as FULL JOIN, INNER JOIN, LEFT & RIGHT JOIN.
- Learnt about foreign key and its implementation in actual scenario.
- Learnt how to perform subquery and implement filter along with subquery