DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
Discover. Learn. Empower.

Experiment 1

Student Name: Palash Mathur **UID: 23BAI70673**

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

```
CREATE DATABASE ADBMS;
USE ADBMS;

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

```
create table dept(
Dept_Id smallint AUTO_INCREMENT PRIMARY KEY ,
Dept_Name varchar(12) NOT NULL
);

drop table course;
drop table dept;
create table course(
    Dept SMALLINT ,
    FOREIGN KEY(Dept) references dept(Dept_Id),
    Course varchar(12)
);
insert into dept(Dept_Name) values('AI&ML'), ('CSE'), ('Bio-Tech'), ('Finance'),
('Psychology');
select * from dept;
```

```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
    dept AS D ON C.Dept = D.Dept_Id;

SELECT
    D.Dept_Name AS Department,
    D.Dept_Id,
    C.COUNT AS `COUNT`
FROM
    dept AS D
INNER JOIN (
    SELECT
        Dept,
        COUNT(Dept) AS COUNT
    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 `COUNT`
    FROM
        course
    GROUP BY
        Dept
) AS C
ON D.Dept_Id = C.Dept
WHERE
    C.COUNT >= 2;


CREATE USER 'SG_user'@'localhost' IDENTIFIED BY '1234';
GRANT SELECT ON course TO 'SG_user'@'localhost';
FLUSH PRIVILEGES;
```

5. **Output:**
   Easy-Level

AUTHOR_ID	AUTHOR_NAME	AUTHOR_Age	Author_Gender
Filter...	Filter...	Filter...	Filter...
120	Schewa Zaitsev	25	F
130	Robert Greene	37	M
145	Zadie Smith	23	F
200	J.K. Rowling	55	F
250	Robert Frost	60	M
554	Ruskin Bond	43	M
786	Aruna Nair	50	F

Author Id	Author Name	Country
Filter...	Filter...	Filter...
120	Schewa Zaitsev	Russia
130	Robert Greene	USA
145	Zadie Smith	Australia
200	J.K. Rowling	Australia
250	Robert Frost	USA
554	Ruskin Bond	USA
786	Aruna Nair	India

Medium-Level

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

Dept	Course
Filter...	Filter...
1	Data Science
1	Neural Networks
1	Machine Learning
1	AI
2	Data Analytics
2	Data Mining
2	Full Stack Developm...
2	Web Development
3	Cyber Security
3	Network Security
3	Bioinformatics

Dept	Course	Department Na...
Filter...	Filter...	Filter...
1	Data Science	AI&ML
1	Neural Networks	AI&ML
1	Machine Learning	AI&ML
1	AI	AI&ML
2	Data Analytics	CSE
2	Data Mining	CSE
2	Full Stack Developm...	CSE
2	Web Development	CSE
3	Cyber Security	Bio-Tech
3	Network Security	Bio-Tech
3	Bioinformatics	Bio-Tech

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

6. **Learning 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