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

Experiment 1.1

Student Name: Anvesha **UID: 23BAI70081**

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

Semester: 5th **Date of Performance: 23 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:**

```
use happy;
create table tab_emp
(
    EMP_ID int identity(101,2),
    EMP_NAME varchar(10)
);
insert into tab_emp (EMP_NAME) values('A');
insert into tab_emp (EMP_NAME) values('B');
insert into tab_emp (EMP_NAME) values('C');
select * from tab_emp;

create table tbl_author
(
    author_id int primary key,
    author_name varchar(max),
    country varchar(max)
)

create table tbl_book
(
    book_id int primary key,
    book_title varchar(max),
    authorid int
    foreign key (authorid) references tbl_author(author_id)
)

insert into tbl_book(book_id , book_title , authorid) values (1,'A',1),
                                                             (2,'B',2),
                                                             (3,'C',3)

insert into tbl_author(author_id , author_name , country) values (1,'Harsh','India'),
                                                                 (2,'Tushar','USA'),
                                                                 (3,'Nikhil','UK')
select * from tbl_author

select b.book_title,a.author_name,a.country
from tbl_author as a
```

```
inner join tbl_book as b
on a.author_id= b.authorid
create table department
(
    department_id int primary key,
    department_name varchar(max)
)

create table courses
(
    course_id int primary key,
    course_name varchar(max),
    department_id int foreign key (department_id) references
department(department_id)
)

insert into department (department_id,department_name) values
                                                    (1,'Computer Science'),
                                                    (2,'Civil Engineering'),
                                                    (3,'Biotechnology'),
                                                    (4,'Mechenical Engineering'),
                                                    (5,'Electrical Engineering')

insert into courses(course_id,course_name,department_id) values
                                                    (101, 'Data Structures', 1),
                                                    (102, 'Algorithms', 1),
                                                    (103, 'Database Systems', 1),
                                                    (201, 'Thermodynamics', 2),
                                                    (301, 'Circuit Theory', 3),
                                                    (302, 'Power Systems', 3),
                                                    (401, 'Structural Analysis', 4),
                                                    (402, 'Building Materials', 4),
                                                    (403, 'Surveying', 4),
                                          (404, 'Geotechnical Engineering', 4),
                                                    (501, 'Molecular Biology', 5)
select * from department
select * from courses

SELECT department.department_name
FROM department
JOIN courses  ON department.department_id = courses.department_id
```

```
GROUP BY department.department_id, department.department_name
HAVING COUNT(*) >= 2;
GRANT SELECT ON courses TO Avin
```

4. Output:

100 % 3 0

Results Messages

	EMP_ID	EMP_NAME
1	101	A
2	103	B
3	105	C

100 % 3 0

Results Messages

	author_id	author_name	country
1	1	Harsh	India
2	2	Tushar	USA
3	3	Nikhil	UK

100 % 3 0

Results Messages

	book_title	author_name	country
1	A	Harsh	India
2	B	Tushar	USA
3	C	Nikhil	UK

100 % 3 0

Results Messages

	department_id	department_name
1	1	Computer Science
2	2	Civil Engineering
3	3	Biotechnology
4	4	Mechenical Engineering
5	5	Electrical Engineering

100 % 3 0

Results Messages

	course_id	course_name	department_id
1	101	Data Structures	1
2	102	Algorithms	1
3	103	Database Systems	1
4	201	Thermodynamics	2
5	301	Circuit Theory	3
6	302	Power Systems	3
7	401	Structural Analysis	4
8	402	Building Materials	4
9	403	Surveying	4
10	404	Geotechnical Engineering	4
11	501	Molecular Biology	5

100 % 3 0

Results Messages

	department_name
1	Computer Science
2	Biotechnology
3	Mechenical Engineering

5. **Learning Outcomes:**
   - Understanding of Table Design and Relationships
   - Proficiency in SQL JOIN Operations
   - Mastery of Subqueries for Filtering Data