



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment 1

Student Name: Mandeep Kaur

UID: 23BCS10854

Branch: BE-CSE

Section/Group: KRG_2B

Semester: 5th

Date of Performance: 28/7/25

Subject Name: Advanced Database and Management System

Subject Code: 23CSP-333

1. Aim:

Easy-Problem Title: Author-Book Relationship Using Joins and Basic SQL Operations.

Procedure (Step-by-Step):

1. Design two tables — one for storing author details and the other for book details.
2. Ensure a foreign key relationship from the book to its respective author.
3. Insert at least three records in each table.
4. Perform an INNER JOIN to link each book with its author using the common author ID.
5. Select the book title, author name, and author's country.

Sample Output Description:

When the join is performed, we get a list where each book title is shown along with its author's name and their country.

Medium-Problem Title: Department-Course Subquery

Procedure (Step-by-Step):

1. Design normalized 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.

Sample Output Description:

The result shows the names of departments which are associated with more than two courses in the system.

2. Objective: The main objective is to design normalized relational database schemas for managing author-book and department-course data using foreign key relationships, enabling meaningful data retrieval through INNER JOINS and subqueries, and demonstrating basic access control by granting SELECT-only permissions.

3. Expected Results-

Easy Problem-

Author table-

| author_id | author_name | author_country |
|-----------|-----------------|----------------|
| 1 | Chetan Bhagat | India |
| 2 | Haruki Murakami | Japan |
| 3 | Yann Martel | Canada |

Books table-

| book_id | book_title | author_id(FK) |
|---------|--------------------|---------------|
| 101 | Five Point Someone | 1 |
| 102 | 2 States | 1 |
| 103 | Norwegian Wood | 2 |
| 104 | Life of Pi | 3 |
| 105 | Kafka on the Shore | 2 |



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Output table-

| book_title | author_name | author_country |
|--------------------|-----------------|----------------|
| Five Point Someone | Chetan Bhagat | India |
| 2 States | Chetan Bhagat | India |
| Norwegian Wood | Haruki Murakami | Japan |
| Life of Pi | Yann Martel | Canada |
| Kafka on the Shore | Haruki Murakami | Japan |

Medium Level Problem -

Department Table-

| dept_id | dept_name |
|---------|------------------|
| 1 | Computer science |
| 2 | Mathematics |
| 3 | Physics |
| 4 | History |
| 5 | Psychology |



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Courses Table-

| course_id | course_name | dept_id |
|-----------|-----------------------|---------|
| 101 | Data structures | 1 |
| 102 | Operating systems | 1 |
| 103 | Database Systems | 1 |
| 104 | Calculus | 2 |
| 105 | Linear algebra | 2 |
| 106 | Quantum Mechanics | 3 |
| 107 | Optics | 3 |
| 108 | Modern History | 4 |
| 109 | Ancient History | 4 |
| 110 | Cognitive Pshychology | 5 |

Expected Output Table -

| |
|------------------|
| Department Name |
| Computer science |



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

4. SQL QUERY AND OUTPUTS -

Easy Problem -

```
CREATE TABLE AUTHORS_DATA(  
AUTHOR_ID INT PRIMARY KEY IDENTITY(1,1), AUTHOR_NAME VARCHAR(MAX), AUTHOR_COUNTRY  
VARCHAR(MAX)  
);  
  
INSERT INTO AUTHORS_DATA(AUTHOR_NAME, AUTHOR_COUNTRY) VALUES('CHETAN BHAGAT', 'INDIA');  
INSERT INTO AUTHORS_DATA(AUTHOR_NAME, AUTHOR_COUNTRY) VALUES('HARUKI  
MURAKAMI', 'JAPAN');  
INSERT INTO AUTHORS_DATA(AUTHOR_NAME, AUTHOR_COUNTRY) VALUES('YANN MARTEL', 'CANADA');  
  
CREATE TABLE BOOKS_DATA(  
BOOK_ID INT PRIMARY KEY, BOOK_TITLE VARCHAR(20), AUTHORID INT,  
FOREIGN KEY (AUTHORID) REFERENCES AUTHORS_DATA(AUTHOR_ID)  
);  
  
INSERT INTO BOOKS_DATA(AUTHORID, BOOK_ID, BOOK_TITLE) VALUES (1,101, 'FIVE POINT  
SOMEONE');  
INSERT INTO BOOKS_DATA(AUTHORID, BOOK_ID, BOOK_TITLE) VALUES (1,102, '2 STATES');  
INSERT INTO BOOKS_DATA(AUTHORID, BOOK_ID, BOOK_TITLE) VALUES (2,103, 'NORWEGIAN WOOD');  
INSERT INTO BOOKS_DATA(AUTHORID, BOOK_ID, BOOK_TITLE) VALUES (3,104, 'LIFE OF PI');  
INSERT INTO BOOKS_DATA(AUTHORID, BOOK_ID, BOOK_TITLE) VALUES (2,105, 'KAFKA ON THE  
SHORE');  
  
SELECT * FROM AUTHORS_DATA  
SELECT * FROM BOOKS_DATA  
  
SELECT B.BOOK_TITLE AS [BOOK TITLE], A.AUTHOR_NAME AS [AUTHOR NAME], A.AUTHOR_COUNTRY  
AS [AUTHOR COUNTRY]  
FROM AUTHORS_DATA AS A  
INNER JOIN
```

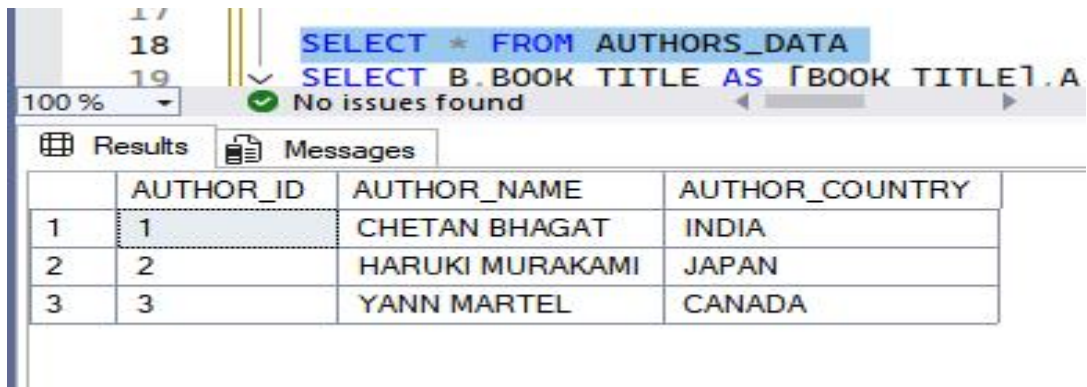
BOOKS_DATA AS B

ON

A . AUTHOR_ID=B . AUTHORID

OUTPUTS OBTAINED -

Easy Problem-

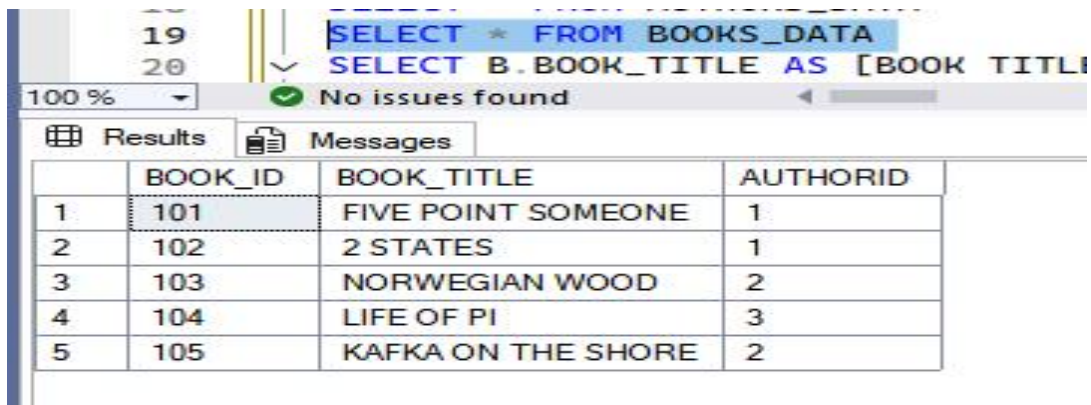


The screenshot shows a SQL query window with the following text:

```
SELECT * FROM AUTHORS_DATA
SELECT B.BOOK TITLE AS [BOOK TITLE].A
```

The query is executed, and the Results tab displays the following table:

| | AUTHOR_ID | AUTHOR_NAME | AUTHOR_COUNTRY |
|---|-----------|-----------------|----------------|
| 1 | 1 | CHETAN BHAGAT | INDIA |
| 2 | 2 | HARUKI MURAKAMI | JAPAN |
| 3 | 3 | YANN MARTEL | CANADA |



The screenshot shows a SQL query window with the following text:

```
SELECT * FROM BOOKS_DATA
SELECT B.BOOK_TITLE AS [BOOK TITLE]
```

The query is executed, and the Results tab displays the following table:

| | BOOK_ID | BOOK_TITLE | AUTHORID |
|---|---------|--------------------|----------|
| 1 | 101 | FIVE POINT SOMEONE | 1 |
| 2 | 102 | 2 STATES | 1 |
| 3 | 103 | NORWEGIAN WOOD | 2 |
| 4 | 104 | LIFE OF PI | 3 |
| 5 | 105 | KAFKA ON THE SHORE | 2 |

```

20 SELECT B.BOOK_TITLE AS [BOOK TITLE], A.AUTHOR_NAME AS [AUTHOR]
21 FROM AUTHORS_DATA AS A
22 INNER JOIN
23 BOOKS_DATA AS B
24 ON
25 A.AUTHOR_ID=B.AUTHORID

```

100 % No issues found Ln: 25 Ch: 23 TABS CRL

Results Messages

| | BOOK TITLE | AUTHOR NAME | AUTHOR COUNTRY |
|---|--------------------|-----------------|----------------|
| 1 | FIVE POINT SOMEONE | CHETAN BHAGAT | INDIA |
| 2 | 2 STATES | CHETAN BHAGAT | INDIA |
| 3 | NORWEGIAN WOOD | HARUKI MURAKAMI | JAPAN |
| 4 | LIFE OF PI | YANN MARTEL | CANADA |
| 5 | KAFKA ON THE SHORE | HARUKI MURAKAMI | JAPAN |

Medium Problem-

```
CREATE TABLE DEPARTMENT(
```

```
DEPT_ID INT PRIMARY KEY IDENTITY(1,1),DEPT_NAME VARCHAR(MAX)
```

```
);
```

```
INSERT INTO DEPARTMENT(DEPT_NAME) VALUES ('COMPUTER SCIENCE')
```

```
INSERT INTO DEPARTMENT(DEPT_NAME) VALUES ('MATHEMATICS')
```

```
INSERT INTO DEPARTMENT(DEPT_NAME) VALUES ('PHYSICS')
```

```
INSERT INTO DEPARTMENT(DEPT_NAME) VALUES ('HISTORY')
```

```
INSERT INTO DEPARTMENT(DEPT_NAME) VALUES ('PSHYCHOLOGY')
```

```
CREATE TABLE COURSES(
```

```
COURSE_ID INT IDENTITY(101,1),COURSE_NAME VARCHAR(MAX),DEPTID INT,FOREIGN KEY  
(DEPTID) REFERENCES DEPARTMENT(DEPT_ID)
```

```
);
```

```
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('DATA STRUCTURES',1)
```

```
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('OPERATING SYSTEMS',1)
```

```
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('DATABASE SYSTEMS',1)
```

```
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('CALCULUS',2)
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('LINEAR ALGEBRA',2)
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('QUANTUM MECHANICS',3)
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('OPTICS',3)
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('MODERN HISTORY',4)
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('ANCIENT HISTORY',4)
INSERT INTO COURSES(COURSE_NAME,DEPTID) VALUES('COGNITIVE PSYCHOLOGY',5)
SELECT * FROM DEPARTMENT;
SELECT * FROM COURSES;
```

```
SELECT D.DEPT_NAME,COUNT(C.COURSE_ID) AS [NUMBER OF COURSES]
FROM DEPARTMENT AS D
INNER JOIN
COURSES AS C
ON D.DEPT_ID=C.DEPTID
GROUP BY D.DEPT_NAME
```

```
SELECT DEPT_NAME
FROM DEPARTMENT
WHERE DEPT_ID IN
(
SELECT DEPTID
FROM COURSES GROUP BY DEPTID HAVING COUNT(*)>2
)
```


OUTPUTS OBTAINED-

Medium problem-

```

23  INSERT INTO COURSES(COURSE_N
24  SELECT * FROM DEPARTMENT;
25  SELECT * FROM COURSES;
26
27  SELECT D.DEPT_ID,D.DEPT_NAME

```

100 % No issues found

| | DEPT_ID | DEPT_NAME |
|---|---------|------------------|
| 1 | 1 | COMPUTER SCIENCE |
| 2 | 2 | MATHEMATICS |
| 3 | 3 | PHYSICS |
| 4 | 4 | HISTORY |
| 5 | 5 | PSHYCHOLOGY |

```

24  SELECT * FROM DEPARTMENT;
25  SELECT * FROM COURSES;
26
27  SELECT D.DEPT_ID,D.DEPT_NAME,COUR

```

100 % No issues found

| | COURSE_ID | COURSE_NAME | DEPTID |
|----|-----------|-----------------------|--------|
| 1 | 101 | DATA STRUCTURES | 1 |
| 2 | 102 | OPERATING SYSTEMS | 1 |
| 3 | 103 | DATABASE SYSTEMS | 1 |
| 4 | 104 | CALCULUS | 2 |
| 5 | 105 | LINEAR ALGEBRA | 2 |
| 6 | 106 | QUANTUM MECHANICS | 3 |
| 7 | 107 | OPTICS | 3 |
| 8 | 108 | MODERN HISTORY | 4 |
| 9 | 109 | ANCIENT HISTORY | 4 |
| 10 | 110 | COGNITIVE PSHYCHOLOGY | 5 |

```
27 SELECT D.DEPT_NAME, COUNT(C.COURSE_ID) AS [NUMBER OF COURSE
28 FROM DEPARTMENT AS D
29 INNER JOIN
30 COURSES AS C
31 ON D.DEPT_ID=C.DEPTID
32 GROUP BY D.DEPT_NAME
33
```

100 % No issues found Ln: 27 Ch: 1 TABS CRI

Results Messages

| | DEPT_NAME | NUMBER OF COURSES |
|---|------------------|-------------------|
| 1 | COMPUTER SCIENCE | 3 |
| 2 | HISTORY | 2 |
| 3 | MATHEMATICS | 2 |
| 4 | PHYSICS | 2 |
| 5 | PSYCHOLOGY | 1 |

```
34 SELECT DEPT_NAME
35 FROM DEPARTMENT
36 WHERE DEPT_ID IN
37 (
38 SELECT DEPTID
39 FROM COURSES GROUP BY DEPTID HAVING COUNT(*)>2
40 )
41
42
```

100 % No issues found Ln: 34 Ch: 1

Results Messages

| | DEPT_NAME |
|---|------------------|
| 1 | COMPUTER SCIENCE |