EXPERIMENT 2.2

Student Name: Pankaj Tayal	**UID:** 23BCS13706
Branch: CSE	**Section/Group:** krg-3b
Semester: 5th	**Date of Performance:** 22/07/25
Subject Name: ADBMS	**Subject Code:** 23CSP-333

1. Aim:

To create and manage a relational database that stores information about faculties and their respective subjects, and to retrieve faculties that offer more than two subjects.

2. Objective:

Create two related tables:

TBL_FACULTY: Stores faculty information (like Engineering, Mathematics, etc.).

TBL_SUBJECTS: Stores subjects offered under each faculty.

Link the two tables using a foreign key:

The FACULTY_REF column in the TBL_SUBJECTS table is a foreign key that refers to FACULTY_ID in the TBL_FACULTY table.

Insert sample data into both tables to simulate a real-world college or university faculty-subject

structure.

Use a JOIN and GROUP BY with HAVING clause to:

Count the number of subjects each faculty offers.

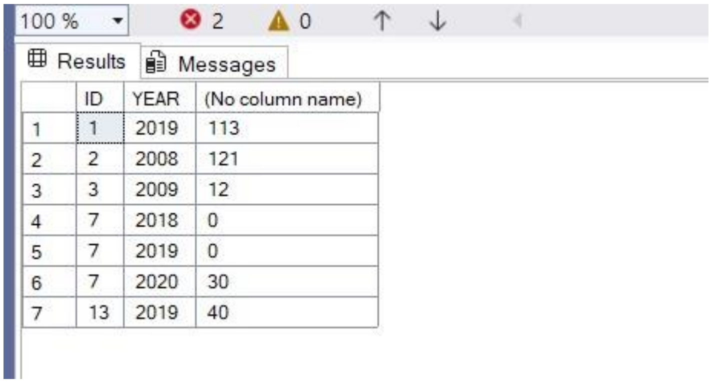
Show only those faculties that offer more than 2 subjects.

3. Code:

```
-- 1. Create table to hold actual NPV values

CREATE TABLE Year_tbl (
   ID INT,
   YEAR INT,
   NPV INT
);

-- 2. Create table for query requests
CREATE TABLE Queries (
   ID INT,
   YEAR INT
);
```

```
-- 3. Insert data into Year_tbl

INSERT INTO Year_tbl (ID, YEAR, NPV) VALUES
(1, 2018, 100),
(7, 2020, 30),
(13, 2019, 40),
(1, 2019, 113),
(2, 2008, 121),
(3, 2009, 12),
(11, 2020, 99),
(7, 2019, 0);

-- 4. Insert data into Queries

INSERT INTO Queries (ID, YEAR) VALUES
(1, 2019),
(2, 2008),
(3, 2009),
(7, 2018),
(7, 2019),
(7, 2020),
(13, 2019);

-- 5. Final query: Return (ID, YEAR) with NPV if available, else 0

SELECT
   Q.ID,
   Q.YEAR,
   ISNULL(Y.NPV, 0) AS NPV
FROM
   Queries AS Q
LEFT OUTER JOIN
   Year_tbl AS Y
ON
   Q.ID = Y.ID AND Q.YEAR = Y.YEAR;
```

4. Output:



	ID	YEAR	(No column name)
1	1	2019	113
2	2	2008	121
3	3	2009	12
4	7	2018	0
5	7	2019	0
6	7	2020	30
7	13	2019	40