8/9/25

5. Join Quires, Equivalent and Recursive Quires.

Aim:- To implement and execute joins, Equivalent Queries and Recursive Queries in SQL.

Procedure:-

1. Create table DEPARTMENT & STUDENT 4.
2. Insert the values into tables
3. Perform join operation.
4. Perform Equivalent & Recursive Query
5. Display result.

```
CREATE TABLE DEPARTMENT4C
        DEPTID INT PRIMARY KEY,
        DEPTNAME VARCHAR(50);
CREATE TABLE STUDENT4C
        STU-ID INT PRIMARY KEY,
        NAME VARCHAR (50),
        AGE INT,
        DEPTID INT,
        FOREIGN KEY (DEPTID)
        REFERENCES DEPARTMENT4 (DEPTID)
        );
INSERT INTO DEPARTMENT4 VALUES
(201, 'Computer Science'),
(202, 'Electronics'),
(203, 'Mechanical'),
```

```
INSERT INTO STUDENT4 VALUES
(1, 'Ravi', 20, 201),
(2, 'sneha', 22, 201),
(3, 'Amit', 19, 202),
(4, 'Priya', 24, 203),
(5, 'kiran', 23, 201);
SELECT * FROM DEPARTMENT4;
```

	DEPTID	DEPTNAME
1	201	Computer science
2	202	Electronics
3	203	Mechanical.

```
SELECT * FROM STUDENT4;
```

	STU-ID	NAME	AGE	DEPT ID
1	1	RAVi	20	201
2	2	Sneha	22	201
3	3	Amit	19	202
4	4	Priya	24	203
5	5	kiran	23	201

```
SELECT S.NAME, S.AGE, D.DEPTNAME
FROM STUDENT4 S
INNER JOIN DEPARTMENT4 D
ON S.DEPT ID = D.DEPT ID;
--INNER JOIN
```

	NAME	AGE	DEPTNAME
1	RAVI	20	Computer science
2	Sneha	22	Computer science
3	Amit	19	Electronics
4	Priya	24	Mechanical
5	kiran	23	Computer science.

-- LEFTOUTER JOIN

```
SELECT S.NAME, S.AGE, D.DEPTNAME
FROM STUDENT4 AS
LEFT JOIN DEPARTMENT4 D
ON S.DEPT ID = D.DEPT ID;
```

	NAME	AGE	DEPTNAME
1	Ravi	20	Computer science
2	Sneha	22	Computer science
3	Amit	19	Electronics
4	Priya	24	Mechanics
5	kiran	23	Computer science

```
SELECT S.NAME, S.AGE, D. DEPARTNAME
FROM . STUDENT 4S
RIGHT JOIN DEPARTMENT4D
ON S. DEPT ID = D. DEPTID;
```

	NAME	AGE	DEPTNAME
1	Ravi	20	Computer science
2	Sneha	22	Computer science
3	kiran	23	Computer science
4	Amit	19	Electronics
5	Priya	24	Mechanical

```
SELECT TOP3 S.NAME, S.AGE, D. DEPT NAME
FROM STUDENT 4S
FULL OUTER JOIN DEPARTMENT 4D
ON S. DEPTID = D.DEPTID;
```

	NAME	AGE	DEPT NAME
1	Ravi	20	Computer science
2	Sneha.	22	Computer science
3	Amit	19	Electronics

-- Equivalent QUERIES
-- USING join

```
SELECT S.NAME, S.AGE
FROM STUDENT45
JOIN DEPARTMENT4D ON S.DEPT ID = D.DEPTID
WHERE D.DEPT NAME = 'Computer Science';
```

	NAME	AGE
1	Ravi	20
2	Sneha	22
3	kiran	23

-- RECURSIVE QUERIES

```
WITH COUNTCTEASC
    SELECT ASN
    UNION ALL
    SELECT N+1
    FROM COUNTCTE
    WHERE N<5
)
SELECT * FROM COUNTCTE;
```

	N
1	1
2	2
3	3
4	4
5	5

VELTECH	
EX No.	5
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	-
TOTAL (20)	15
SIGN WITH DATE	

Result: Thus, implementation of Join, Quires, Equivalent and Recursive Quires has successfully excuted and verified.