8/9/25

TASK 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 and STUDENT
2. Insert the values into tables
3. Perform join operation
4. Perform equivalent recursive query
5. Display result

```
CREATE TABLE DEPARTMENT
        DEPT ID INT PRIMARY KEY,
        DEPT NAME VARCHAR(50));

CREATE TABLE STUDENT(
        STU-ID INT PRIMARY KEY,
        NAME VARCHAR(50),
        AGE INT,
          FOREIGN KEY(DEPT ID)
        REFERENCE DEPARTMENT (DEPT)
        );

INSERT INTO DEPERTMENT 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 DEPARTMENT;

	DEPT ID	DEPT NAME
1	201	computer science
2	202	electronics
3	203	Mechinal

SELECT * FROM STUDENT4;

	STUID	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.DEPARTMENT NAME
FROM STUDENT S
INNER JOIN DEPARTMENT 2D
ON S.DEPTMENT4D
-- INNER JOIN

	Name	AGE	DEPT_NAME
1	Ravi	20	computer science
2	sneha	22	computer science
3	Amit	19	electronics
4	Priya	24	Mechanical
5	Kiran	23	computer science.

— LEFT OUTER JOIN

```
SELECT S.Name, S.AGE, D.DEPT NAME FROM STUDENTS.
LEFT JOIN DEPARTMENT D.
  ON S.DEPT_ID = D.DEPT_ID;
```

	NAME	AGE	DEPT_NAME
1	Ravi	20	computer science
2	sneha	22	computer science
3	Amit	19	computer science
4	Priya	24	electronics
5	Kiran	23	Mechanical

```
Select TOP 3.SNAMES, S.AGE, D.DEPTNAME
FROM STUDENTS
FULL OUTER JOIN DEPARTMENT ID.
ON S.DEPTID = D.DEPT D;
```

	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 STUDENTS
JOIN DEPARTMENT AS D ON S.DEPT ID = D.DEPT ID
WHERE D.DEPT NAME = "computer science";
```

	NAME	AGE
1	Ravi	20
2	sneha	22
3	Kiran	25

-- RECURSIVE QUERIES

-- Using Join.

```
(SELECT S.NAME, S.AGE
        SELECT 1 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

VEL TECH - CSE	
EX NO.	5
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	20
SIGN WITH DATE	8/9/24

Result:- Thus, Implementation of Join QUERIES, Equivalent and Recursive queries has successfully executed and verified.