Ex no:3
30.1.2025

Join Operation

Aim:

To Execute Various type of join commands in SQL.

Join

A join is a command that combines rows from two or more tables into a single data set(Table).

Types of Join

1. Cross Join (or) Cartesian Product
2. Inner join
3. Natural Join
4. Outer Join
    a. Left Outer Join
    b. Right Outer Join
    c. Full Outer Join
5. Self Join
6. Equi Join
7. Theta Join

Creating Table

```
SQL> CREATE TABLE Faculty (
  2      FacultyID INT PRIMARY KEY,
  3      FacultyName VARCHAR(10),
  4      DepartmentID INT,
  5      Email VARCHAR(11)
  6  );

Table created.
SQL> CREATE TABLE Courses (

  2      CourseID INT PRIMARY KEY,

  3      CourseName VARCHAR(10),

  4      FacultyID INT,

  5      Credits INT,

  6      FOREIGN KEY (FacultyID) REFERENCES Faculty(FacultyID)

  7  );

Table created.
```

After inserting Values

```
 FACULTYID FACULTYNAME DEPARTMENTID EMAIL
---------- ----------- ------------ ---------------
         1 ASHA                   3 asha@gmail.com
         2 YAZHU                  3 yazhu@gmail.com
         3 RANI                   1 rani@gmail.com
         4 ABI                    2 abi@gmail.com
         5 SRI                    1 sri@gmail.com
         6 dhan                   4 dhan@gmail.com

6 rows selected.


SQL> select * from courses;

  COURSEID COURSENAME  FACULTYID    CREDITS
---------- ---------- ---------- ----------
         1 DBMS                1          4
         2 CO                  2          3
         3 ITC                 1          3
         4 DAA                 4          3
         5 DS                  3          4
5 rows selected.
```

Cartesian Product (Cross Product)

The **Cartesian Product** (or **Cross Product**) in SQL is the result of combining every row from one table with every row from another table.

```
 SQL> select * from faculty,courses;

   FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME   FACULTYID    CREDITS
   ---------- ---------- ----------- -------------- ---------- ---------- ----------

  1 ASHA                     3 asha@gmail.com           1 DBMS            1           4

  2 YAZHU                    3 yazhu@gmail.com          1 DBMS            1           4

  3 RANI                     1 rani@gmail.com           1 DBMS            1           4

  4 ABI                      2 abi@gmail.com            1 DBMS            1           4

  5 SRI                      1 sri@gmail.com            1 DBMS            1           4

  6 dhan                     4 dhan@gmail.com           1 DBMS            1           4

  1 ASHA                     3 asha@gmail.com           2 CO              2           3

  2 YAZHU                    3 yazhu@gmail.com          2 CO              2           3
```

```
3 RANI                  1 rani@gmail.com        2 CO                2          3

4 ABI                   2 abi@gmail.com         2 CO                2          3

5 SRI                   1 sri@gmail.com         2 CO                2          3

6 dhan                  4 dhan@gmail.com        2 CO                2          3

1 ASHA                  3 asha@gmail.com        3 IT                1          3

2 YAZHU                 3 yazhu@gmail.com       3 ITC               1          3

3 RANI                  1 rani@gmail.com        3 ITC               1          3

4 ABI                   2 abi@gmail.com         3 ITC               1          3

5 SRI                   1 sri@gmail.com         3 ITC               1          3

6 dhan                  4 dhan@gmail.com        3 ITC               1          3

1 ASHA                  3 asha@gmail.com        4 DAA               4          3

2 YAZHU                 3 yazhu@gmail.com       4 DAA               4          3

3 RANI                  1 rani@gmail.com        4 DAA               4          3

4 ABI                   2 abi@gmail.com         4 DAA               4          3

5 SRI                   1 sri@gmail.com         4 DAA               4          3

6 dhan                  4 dhan@gmail.com        4 DAA               4          3

1 ASHA                  3 asha@gmail.com        5 DS                3          4

2 YAZHU                 3 yazhu@gmail.com       5 DS                3          4

3 RANI                  1 rani@gmail.com        5 DS                3          4

4 ABI                   2 abi@gmail.com         5 DS                3          4

5 SRI                   1 sri@gmail.com         5 DS                3          4

6 dhan                  4 dhan@gmail.com        5 DS                3          4


30 rows selected.
```

Natural Join

Natural Join is a type of Join in which the common attributes in the resultant table appear **only once** . Then, the attributes of the first table followed by second table.

```
SQL> select * from faculty natural join courses;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL            COURSEID COURSENAME  CREDITS

   ---------- ----------- ------------ -------------- ---------- ----------

    1 ASHA                     3 asha@gmail.com          1 DBMS          4

    2 YAZHU                    3 yazhu@gmail.com         2 CO            3

    1 ASHA                     3 asha@gmail.com          3 ITC           3

    4 ABI                      2 abi@gmail.com           4 DAA           3

    3 RANI                     1 rani@gmail.com          5 DS            4
```

Inner Join

Inner Join is type of join in the which the common attribute in the resultant table **appear twice** .The resultant table only contains records that have common attribute in **both the table.**

Inner Join using Keyword

```
SQL> select * from faculty inner join courses on
faculty.facultyid=courses.facultyid;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME  FACULTYID    CREDITS
 ----------- ------------ -------------- ---------- ---------- ---------- ----------

  1 ASHA                     3 asha@gmail.com          1 DBMS          1    4

  2 YAZHU                    3 yazhu@gmail.com         2 CO            2    3

  1 ASHA                     3 asha@gmail.com          3 ITC           1    3

  4 ABI                      2 abi@gmail.com           4 DAA           4    3

  3 RANI                     1 rani@gmail.com          5 DS            3    4
```

Inner Join without using keyword

```
 SQL> select * from faculty,courses where faculty.facultyid=courses.facultyid;

FACULTYID FACULTYNAME DEPARTMENTID EMAILCOURSEID COURSENAME  FACULTYID CREDITS
---------- ----------- ------------ --------------- ---------- ----------

 1 ASHA                   3 asha@gmail.com           1 DBMS       1          4

 2 YAZHU                  3 yazhu@gmail.com          2 CO         2          3

 1 ASHA                   3 asha@gmail.com           3 ITC        1          3

 4 ABI                    2 abi@gmail.com            4 DAA        4          3

 3 RANI                   1 rani@gmail.com           5 DS         3          4
```

Inner Join with ‘using’ clause

```
SQL> select * from faculty join courses using(facultyid);

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS
---------- ----------- ------------ --------------- ---------- ----------

 1 ASHA                   3 asha@gmail.com           1 DBMS        1        4

 2 YAZHU                  3 yazhu@gmail.com          2 CO          2        3

 1 ASHA                   3 asha@gmail.com           3 ITC         1        3

 4 ABI                    2 abi@gmail.com            4 DAA         4        3

 3 RANI                   1 rani@gmail.com           5 DS          3        4
```

```
SQL> select * from faculty,courses where faculty.facultyid<>courses.facultyid;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME  FACULTYID CREDITS
--------- ----------- ------------ -------------- ---------- ----------
 2 YAZHU        3 yazhu@gmail.com          1 DBMS         1           4

 3 RAN I        1 rani@gmail.com           1 DBMS         1           4

 4 ABI          2 abi@gmail.com            1 DBMS         1           4

 5 SRI          1 sri@gmail.com            1 DBMS         1           4

 6 dhan         4 dhan@gmail.com           1 DBMS         1           4

 1 ASHA         3 asha@gmail.com           2 CO           2           3

 3 RANI         1 rani@gmail.com           2 CO           2           3

 4 ABI          2 abi@gmail.com            2 CO           2           3

 5 SRI          1 sri@gmail.com            2 CO           2           3

 6 dhan         4 dhan@gmail.com           2 CO           2           3

 2 YAZHU        3 yazhu@gmail.com          3 ITC          1           3

 3 RANI         1 rani@gmail.com           3 ITC          1           3

 4 ABI          2 abi@gmail.com            3 ITC          1           3

 5 SRI          1 sri@gmail.com            3 ITC          1           3

 6 dhan         4 dhan@gmail.com           3 ITC          1           3

 1 ASHA         3 asha@gmail.com           4 DAA          4           3

 2 YAZHU        3 yazhu@gmail.com          4 DAA          4           3

 3 RANI         1 rani@gmail.com           4 DAA          4           3

 5 SRI          1 sri@gmail.com            4 DAA          4           3

 6 dhan         4 dhan@gmail.com           4 DAA          4           3

 1 ASHA         3 asha@gmail.com           5 DS           3           4
```

```
2 YAZHU       3 yazhu@gmail.com          5 DS          3           4

4 ABI          2 abi@gmail.com            5 DS          3          4

5 SRI          1 sri@gmail.com            5 DS          3          4

6 dhan         4 dhan@gmail.com           5 DS          3          4


25 rows selected.
```

Inner Join with logical operator

GREATER THAN:

```
SQL> select * from faculty,courses where faculty.facultyid>3;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS
---------- ----------- ------------ --------------- ---------- ----------

 4 ABI                 2 abi@gmail.com           1 DBMS     1        4

 4 ABI                 2 abi@gmail.com           2 CO       2        3

 4 ABI                 2 abi@gmail.com           3 ITC      1        3

 4 ABI                 2 abi@gmail.com           4 DAA      4        3

 4 ABI                 2 abi@gmail.com           5 DS       3        4

 5 SRI                 1 sri@gmail.com           1 DBMS     1        4

 5 SRI                 1 sri@gmail.com           2 CO       2        3

 5 SRI                 1 sri@gmail.com           3 ITC      1        3

 5 SRI                 1 sri@gmail.com           4 DAA      4        3

 5 SRI                 1 sri@gmail.com           5 DS       3        4

 6 dhan                4 dhan@gmail.com          1 DBMS     1        4

 6 dhan                4 dhan@gmail.com          2 CO       2        3

 6 dhan                4 dhan@gmail.com          3 ITC      1        3
 6 dhan                4 dhan@gmail.com           4 DAA     4         3

 6 dhan                4 dhan@gmail.com           5 DS      3         4

 15 rows selected.
```

LESS THAN OPERATOR:

```
SQL> select * from faculty,courses where faculty.facultyid<courses.facultyid;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS
---------- ----------- ------------ --------------- ---------- ----------
 1 ASHA               3 asha@gmail.com           2 CO       2         3

 1 ASHA               3 asha@gmail.com           4 DAA      4         3

 2 YAZHU              3 yazhu@gmail.com          4 DAA      4         3

 3 RANI               1 rani@gmail.com           4 DAA      4         3

 1 ASHA               3 asha@gmail.com           5 DS       3         4

 2 YAZHU              3 yazhu@gmail.com          5 DS       3         4


6 rows selected.
```

LESS THAN OR EQUAL TO :

```
SQL> select * from faculty,courses where faculty.facultyid<=courses.facultyid;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS

---------- ----------- ------------ --------------- ---------- ----------

1 ASHA               3 asha@gmail.com           1 DBMS     1         4

1 ASHA               3 asha@gmail.com           2 CO       2         3

2 YAZHU              3 yazhu@gmail.com          2 CO       2         3

1 ASHA               3 asha@gmail.com           3 ITC      1         3

1 ASHA               3 asha@gmail.com           4 DAA      4         3
```

```
2 YAZHU                  3 yazhu@gmail.com          4 DAA      4        3

3 RANI                   1 rani@gmail.com           4 DAA      4        3

4 ABI                    2 abi@gmail.com            4 DAA      4        3

1 ASHA                   3 asha@gmail.com           5 DS       3        4

2 YAZHU                  3 yazhu@gmail.com          5 DS       3        4

3 RANI                   1 rani@gmail.com           5 DS       3        4

11 rows selected.
```

GREATER THAN OR EQUALT TO:

```
SQL> select facultyname,courseid from faculty,courses where faculty.facultyid>4
and faculty.facultyid>=courses.facultyid;

 FACULTYNAME   COURSEID
 ----------- ----------
   SRI                  1
   SRI                  2
   SRI                  3
   SRI                  4
   SRI                  5
   dhan                 1
   dhan                 2
   dhan                 3
   dhan                 4
   dhan                 5

10 rows selected.
```

Inner Join with Combination logical operator

```
SQL> select * from faculty,courses where faculty.facultyid >3 and
faculty.facultyid=courses.facultyid;


FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS
---------- ----------- ------------ -------------- ---------- ----------

   4 ABI          2 abi@gmail.com            4 DAA          4          3
```

Viewing Table before outer Join

```
SQL> select * from faculty;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL
---------- ----------- ------------ ---------------
         1 ASHA                   3 asha@gmail.com
         2 YAZHU                  3 yazhu@gmail.com
         3 RANI                   1 rani@gmail.com
         4 ABI                    2 abi@gmail.com
         5 SRI                    1 sri@gmail.com
         6 dhan                   4 dhan@gmail.com

6 rows selected.


SQL> select * from courses;

  COURSEID COURSENAME  FACULTYID    CREDITS
---------- ---------- ---------- ----------
         1 DBMS                1          4
         2 CO                  2          3
         3 ITC                 1          3
         4 DAA                 4          3
         5 DS                  3          4
```

Outer Join

An **OUTER JOIN** in SQL returns all records from one or both tables, including unmatched rows. It has three types:

- **LEFT OUTER JOIN**: Returns all records from the left table and matching records from the right table.
- **RIGHT OUTER JOIN**: Returns all records from the right table and matching records from the left table.
- **FULL OUTER JOIN**: Returns all records from both tables, with NULLs for unmatched rows.

Left Outer Join

```
SQL> select * from faculty natural left join courses;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL  COURSEID COURSENAME   CREDITS
---- ----------- ------------ --------------- ---------- ---------- ----------

 1 ASHA                   3 asha@gmail.com           1 DBMS          4

 2 YAZHU                  3 yazhu@gmail.com          2 CO            3

 1 ASHA                   3 asha@gmail.com           3 ITC           3

 4 ABI                    2 abi@gmail.com            4 DAA           3

 3 RANI                   1 rani@gmail.com           5 DS            4

 6 dhan                   4 dhan@gmail.com

 5 SRI                    1 sri@gmail.com


 7 rows selected.
```

Right Outer Joint

```
    SQL> select * from faculty natural right outer join courses;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL             COURSEID COURSENAME CREDITS
---------- ----------- ------------ --------------- ---------- ----------

     1 ASHA                   3 asha@gmail.com           3 ITC         3

     1 ASHA                   3 asha@gmail.com           1 DBMS        4

     2 YAZHU                  3 yazhu@gmail.com          2 CO          3

     3 RANI                   1 rani@gmail.com           5 DS            4

     4 ABI                    2 abi@gmail.com            4 DAA          3
```

Full Outer Join

```
SQL> select * from faculty natural full outer join courses;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME  CREDITS
---------- ----------- ------------ -------------- ---------- ----------

 1 ASHA                     3 asha@gmail.com          1 DBMS           4

 2 YAZHU                    3 yazhu@gmail.com         2 CO             3

 1 ASHA                     3 asha@gmail.com          3 ITC            3

 4 ABI                      2 abi@gmail.com           4 DAA            3

 3 RANI                     1 rani@gmail.com          5 DS             4

 6 dhan                     4 dhan@gmail.com

 5 SRI                      1 sri@gmail.com


7 rows selected.
```

Right Outer Join Without using keyword

```
SQL> select * from faculty,courses where faculty.facultyid(+) =
courses.facultyid;

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS
---------- ----------- ------------ -------------- ---------- ----------
  1 ASHA        3 asha@gmail.com          3 ITC          1          3
  1 ASHA        3 asha@gmail.com          1 DBMS         1          4
  2 YAZHU       3 yazhu@gmail.com         2 CO           2          4
  3 RANI        1 rani@gmail.com          5 DS           3          4
  4 ABI         2 abi@gmail.com           4 DAA          4          3
```

Left Outer Joint without Using Keyword

```
SQL> select * from faculty,courses where faculty.facultyid =
courses.facultyid(+);

FACULTYID FACULTYNAME DEPARTMENTID EMAIL COURSEID COURSENAME FACULTYID CREDITS
---------- ----------- ------------ -------------- ---------- ----------
 1 ASHA                     3 asha@gmail.com          1 DBMS    1          4

 2 YAZHU                    3 yazhu@gmail.com         2 CO      2          3
```

```
  1 ASHA                   3 asha@gmail.com           3 ITC     1          3

  4 ABI                    2 abi@gmail.com            4 DAA     4          3

  3 RANI                   1 rani@gmail.com           5 DS      3          4

  6 dhan                   4 dhan@gmail.com

  5 SRI                    1 sri@gmail.com


   7 rows selected.
```

Self Join

A **self-join** is a type of **join** where a table is joined with itself. It is useful when dealing with **hierarchical relationships** such as employee-manager structures, where employees and managers are stored in the same table.

```
SQL> create table teachers(tid number(2) primary key,tname
varchar2(20),mentorid number(2));

Table created.

SQL> select * from teachers;
       TID TNAME                  MENTORID
---------- -------------------- ----------
         1 RAJESH                        3
         2 ABARNA                        1
         3 YAZHU                         5
         4 RANI                          2
         5 ASHA                          4

SQL> select t.tid as teach_id,t.tname from teachers t,teachers t1 where
t.tid=t1.mentorid;

  TEACH_ID TNAME
---------- --------------------
         3 YAZHU
         1 RAJESH
         5 ASHA
         2 ABARNA
         4 RANI
```

Equi Join

An **Equi Join** is a type of join where we use the **equality (=) operator** to match values between two tables. It retrieves only those records that have matching values in both tables.

```
SQL> select * from faculty;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL
---------- ----------- ------------ ---------------
         1 ASHA                   3 asha@gmail.com
         2 YAZHU                  3 yazhu@gmail.com
         3 RANI                   1 rani@gmail.com
         4 ABI                    2 abi@gmail.com
         5 SRI                    1 sri@gmail.com
         6 dhan                   4 dhan@gmail.com

 6 rows selected.

 SQL> select * from courses;

  COURSEID COURSENAME  FACULTYID    CREDITS
---------- ---------- ---------- ----------
         1 DBMS                1          4
         2 CO                  2          3
         3 ITC                 1          3
         4 DAA                 4          3
         5 DS                  3          4


 SQL> select f.facultyname,f.email from faculty f,courses where
 f.facultyid=courses.facultyid;

 FACULTYNAME EMAIL
 ----------- ---------------
 ASHA        asha@gmail.com
 ASHA        asha@gmail.com
 YAZHU       yazhu@gmail.com
 RANI        rani@gmail.com
 ABI         abi@gmail.com
```

Theta Join

Theta Join is a type of **join condition** where the condition involves a comparison operator other than just equality (=).

```
SQL> select * from faculty;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL
---------- ----------- ------------ ---------------
         1 ASHA                   3 asha@gmail.com
         2 YAZHU                  3 yazhu@gmail.com
         3 RANI                   1 rani@gmail.com
         4 ABI                    2 abi@gmail.com
         5 SRI                    1 sri@gmail.com
         6 dhan                   4 dhan@gmail.com

6 rows selected.


SQL> select * from fac;

       FID        SAL DEPT
---------- ---------- ----
         1      50000 IT
         2      30000 CSE
         3      70000 AIML
         4      50000 AIDS
         5      56000 EIE
```

```
SQL> select facultyid,facultyname,sal from faculty,fac where sal>50000
and faculty.facultyid=fac.fid;

 FACULTYID FACULTYNAME        SAL
---------- ----------- ----------
         3 RANI             70000
         5 SRI              56000
```

JOIN USING THREE TABLES:

```
SQL> select * from faculty inner join courses on faculty.facultyid =
courses.facultyid inner join fac on courses.facultyid = fac.fid;
```

```
 FACULTYID FACULTYNAME DEPARTMENTID EMAILCOURSEID COURSENAME
---------- ----------- ------------ -------------- ---------- ----------
 FACULTYID    CREDITS        FID        SAL DEPT
---------- ---------- ---------- ---------- ----
         1 ASHA                   3 asha@gmail.com           1 DBMS
         1          4          1      50000 IT
         2 YAZHU                  3 yazhu@gmail.com          2 CO
         2          3          2      30000 CSE

         1 ASHA                   3 asha@gmail.com           3 ITC
         1          3          1      50000 IT

         4 ABI                    2 abi@gmail.com            4 DAA
         4          3          4      50000 AIDS

         3 RANI                   1 rani@gmail.com           5 DS
         3          4          3      70000 AIML



 SQL> select * from faculty,courses,fac where faculty.facultyid =
 courses.facultyid and  courses.facultyid = fac.fid;

 FACULTYID FACULTYNAME DEPARTMENTID EMAIL             COURSEID COURSENAME
---------- ----------- ------------ -------------- ---------- ----------
 FACULTYID    CREDITS        FID        SAL DEPT
---------- ---------- ---------- ---------- ----
         1 ASHA                   3 asha@gmail.com           1 DBMS
         1          4          1      50000 IT

         2 YAZHU                  3 yazhu@gmail.com          2 CO
         2          3          2      30000 CSE

         1 ASHA                   3 asha@gmail.com           3 ITC
         1          3          1      50000 IT

         4 ABI                    2 abi@gmail.com            4 DAA
         4          3          4      50000 AIDS

         3 RANI                   1 rani@gmail.com           5 DS
         3          4          3      70000 AIML
```

CONTENTS	MARKS ALLOTE	MARKS OBTAINE
Aim, Algorithm, SQL, PL/SQL	30	
Execution and Result	20	
Viva	10	
Total	60	

RESULT

Thus Data Definition Language commands and Integrity Constraints were executed