SQL Joins

Retrieving correlated data from a database

"Now I want to see the <u>customer name and their current balance</u> from my database" – Bank manager

Lets assume that the customer information including the customer name is in the **Customer table** and their account information is in **Accounts table**

We need to retrieve data from two related tables
- **Name** from **Customer** table
- **Curr_Balance** from **Accounts** table

Retrieving corelated information from different tables require an SQL **SELECT** statement with **JOIN clause**

Joins

Joins are used to combine the data from two or more tables based on a condition.

The different types of joins are

- **CROSS JOIN**   - Cartesan product of all the records in both tables

- **INNER JOIN**   - Matching records from both tables

- **LEFT OUTER JOIN**   - All the records from left table and matching

  records from right table

- **RIGHT OUTER JOIN** - All the records from right table and matching

  records from left table

- **FULL OUTER JOIN**   - Returns all rows from both tables

- **SELF JOIN**   - Used to join a table to itself

Consider the following two tables

```
SQL> select * from JJ_DOCTOR;

        ID NAME
---------- ------------------------------------------
       101 Aswathi J
       102 Kethu P
       103 Rahul N

SQL> select * from JJ_PATIENT;

ID      NAME                                        DOC_ID
------- ------------------------------------------- ----------
P01     Maya M                                         101
P02     Harish H                                       102
P03     Leeya P                                        105
```

CROSS JOIN

Syntax :

Select * From Table1, Table2

OR

Select * From Table1 Cross Join Table2

The cross join will combine each record of the first table with all records of the second table. The cross join does not have a join condition. If the table1 has 2 records and the table2 has 3 records, then the output will have 2* 3 = 6 records.

CROSS JOIN

```
SQL> select * from JJ_doctor d cross join JJ_patient p;

        ID NAME        ID    NAME          DOC_ID
---------- ----------- ----- ----------- ----------
       101 Aswathi J   P01   Maya M           101
       102 Kethu P     P01   Maya M           101
       103 Rahul N     P01   Maya M           101
       101 Aswathi J   P02   Harish H         102
       102 Kethu P     P02   Harish H         102
       103 Rahul N     P02   Harish H         102
       101 Aswathi J   P03   Leeya P          105
       102 Kethu P     P03   Leeya P          105
       103 Rahul N     P03   Leeya P          105

9 rows selected.
```

Note: - The JJ_Doctor table has 3 records. And the JJ_Patient table has 3 records. So the output of CROSS JOIN has 3*3 = 9 records.

INNER JOIN

Syntax :

Select * From Table1 JOIN Table2 ON Table1.column1 = Table2.column2

OR

Select * From Table1, Table2 Where Table1.column1 = Table2.column2

Inner Join will combine all records of the first table with the records of the second table as long as there is a match between the columns.

INNER JOIN

```
SQL> select * from JJ_doctor d join JJ_patient p on d.id = p.doc_id;

    ID NAME        ID  NAME            DOC_ID
---------- ---------- ----- ---------- ----------
   101 Aswathi J  P01  Maya M             101
   102 Kethu P    P02  Harish H           102
```

```
SQL> select * from JJ_doctor d, JJ_patient p where d.id = p.doc_id;

    ID NAME        ID  NAME            DOC_ID
---------- ---------- ----- ---------- ----------
   101 Aswathi J  P01  Maya M             101
   102 Kethu P    P02  Harish H           102
```

Note: - Here all the combination of records from the two tables which satisfy the join condition (d.id=p.doc_id) are returned

LEFT OUTER JOIN

Syntax :

Select * From Table1 LEFT JOIN Table2
ON Table1.column1 = Table2.column2

OR

Select * From Table1, Table2 Where Table1.column1 = Table2.column2(+)

Left outer join returns all the rows from the first table(table1) with the matching rows from the other table(table2). If there is no matching record from the second table, then the corresponding columns will have NULL values.

LEFT OUTER JOIN

```
SQL> select * from JJ_doctor d left join JJ_patient p on d.id = p.doc_id;

     ID NAME        ID    NAME        DOC_ID
---------- ---------- ----- ---------- ----------
    101 Aswathi J  P01   Maya M        101
    102 Kethu P    P02   Harish H      102
    103 Rahul N
```

```
SQL> select * from JJ_doctor d , JJ_patient p where d.id = p.doc_id(+);

     ID NAME        ID    NAME        DOC_ID
---------- ---------- ----- ---------- ----------
    101 Aswathi J  P01   Maya M        101
    102 Kethu P    P02   Harish H      102
    103 Rahul N
```

Note: - Here all the combination of records from the two tables which satisfy the join condition (d.id=p.doc_id) are returned. Along with those records, rest of the records from the left table JJ_doctor (for which no rows from the JJ_patient table satisfy the join condition) are also returned.

RIGHT OUTER JOIN

Syntax :

Select * From Table1 RIGHT JOIN Table2
ON Table1.column1 = Table2.column2

OR

Select * From Table1, Table2 Where Table1.column1(+) = Table2.column2

Right outer join returns matching rows from the first table(table1) with all the rows from the second table(table2). If there is no matching record from the first table, then the corresponding columns will have NULL values.

RIGHT OUTER JOIN

```
SQL> select * from JJ_doctor d right join JJ_patient p on d.id = p.doc_id;

    ID NAME        ID   NAME        DOC_ID
---------- ---------- ----- ---------- ----------
   101 Aswathi J  P01  Maya M         101
   102 Kethu P    P02  Harish H       102
                  P03  Leeya P        105
```

```
SQL> select * from JJ_doctor d , JJ_patient p where d.id(+) = p.doc_id;

    ID NAME        ID   NAME        DOC_ID
---------- ---------- ----- ---------- ----------
   101 Aswathi J  P01  Maya M         101
   102 Kethu P    P02  Harish H       102
                  P03  Leeya P        105
```

Note: - Here all the combination of records from the two tables which satisfy the join condition (d.id=p.doc_id) are returned. Along with those records, rest of the records from the right table JJ_patient (for which no rows from the JJ_doctor table satisfy the join condition) are also returned.

FULL OUTER JOIN

Syntax :

Select * From Table1 FULL JOIN Table2 ON Table1.column1 = Table2.column2

Full Outer Join will return all records of the first table with all the records of the second table. The records from both tables which doesn't have a matching record in the other table will also be returned with null values in the columns from the other table

FULL OUTER JOIN

```
SQL> select * from JJ_doctor d full join JJ_patient p on d.id = p.doc_id;

     ID NAME        ID   NAME            DOC_ID
---------- ---------- ----- ---------- ----------
    101 Aswathi J  P01  Maya M             101
    102 Kethu P    P02  Harish H           102
               P03  Leeya P            105

    103 Rahul N
```

Note: - Here all the combination of records from the two tables which satisfy the join condition (d.id=p.doc_id) are returned. Along with those records, rest of the records from the left table JJ_doctor and right table JJ_patient (for which no rows from the other table satisfied the join condition) are also returned.

SELF JOIN

Self Join is joining a table to itself. This table appears twice in the FROM clause and is followed by table aliases that qualify column names in the join condition.

Syntax :

Select * From Table1 A Join Table1 B On A.Col1 = B.Col1;

Example :

```
SQL> select * from JJ_doctor d1 join JJ_doctor d2 on d1.id = d2.id;

        ID NAME                ID NAME
---------- ----------  ---------- ----------
       101 Aswathi J          101 Aswathi J
       102 Kethu P            102 Kethu P
       103 Rahul N            103 Rahul N
```