JOINS

- JOIN is used to retrieve data from multiple tables.
- Join query is used to combine rows from two or more tables and creates a new table.
- There must be one join condition for joining two tables. It compares two columns from the different tables and combines rows, for which join condition is true to form the result set.

JOINS

There are different types of joins:

- Inner join
- Left outer join
- Right outer join
- Full outer join
- Cross join
- Self join

Cross join

- CROSS JOIN of two tables makes a Cartesian product of the tables.
- CROSS JOIN combines all rows from first table with all of the rows of second table.
- If there are "x" rows in table1 and "y" rows in table2 then the CROSS JOIN result set have x*y rows.

Cross join

Syntax:

```
SELECT  <column_list>
FROM
table1  CROSS JOIN  table2;
```

ENO	ENAME	SAL	DNO
1	A	1000	10
2	B	2000	20
3	C	1000	10
4	D	3000	20

DNO	DNAME	LOCATION
10	PROD	HYD
20	FINAN	DEL

CROSS JOIN

Example:

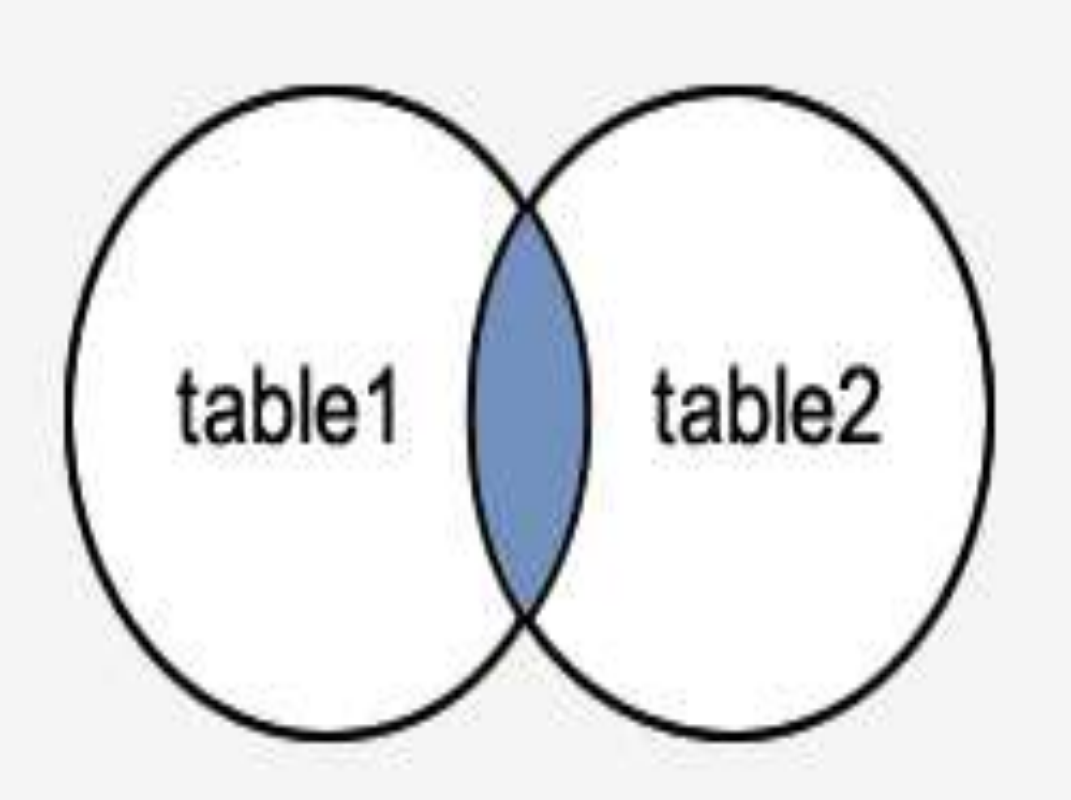
- SELECT * FROM EMP **CROSS JOIN** DEPT;

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
1	A	1000	10	10	PROD	HYD
2	B	2000	20	10	PROD	HYD
3	C	1000	10	10	PROD	HYD
4	D	3000	20	10	PROD	HYD
1	A	1000	10	20	FINAN	DEL
2	B	2000	20	20	FINAN	DEL
3	C	1000	10	20	FINAN	DEL
4	D	3000	20	20	FINAN	DEL

Inner join

- Inner Join is the simplest and most common type of join.
- It returns all rows from multiple tables where the join condition is met.
- INNER JOIN is used to retrieve matching rows from tables.

Inner join



Inner join

Syntax:

SELECT <columns>

FROM

table1 **INNER JOIN** table2

ON <join condition> ;

ENO	ENAME	SAL	DNO
1	A	1000	10
2	B	2000	20
3	C	1000	10
4	D	3000	20
5	E	2000	-

DNO	DNAME	LOCATION
10	PROD	HYD
20	FINAN	DEL
40	SALES	MUM

Inner join

SELECT * FROM emp **INNER JOIN** dept

ON emp.dno = dept.dno;

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
1	A	1000	10	10	PROD	HYD
2	B	2000	20	20	FINAN	DEL
3	C	1000	10	10	PROD	HYD
4	D	3000	20	20	FINAN	DEL

Outer join

- An outer join is similar to innerjoin but it also gets the non-matched rows from the table.

1. Left Outer Join
2. Right Outer Join
3. Full Outer Join

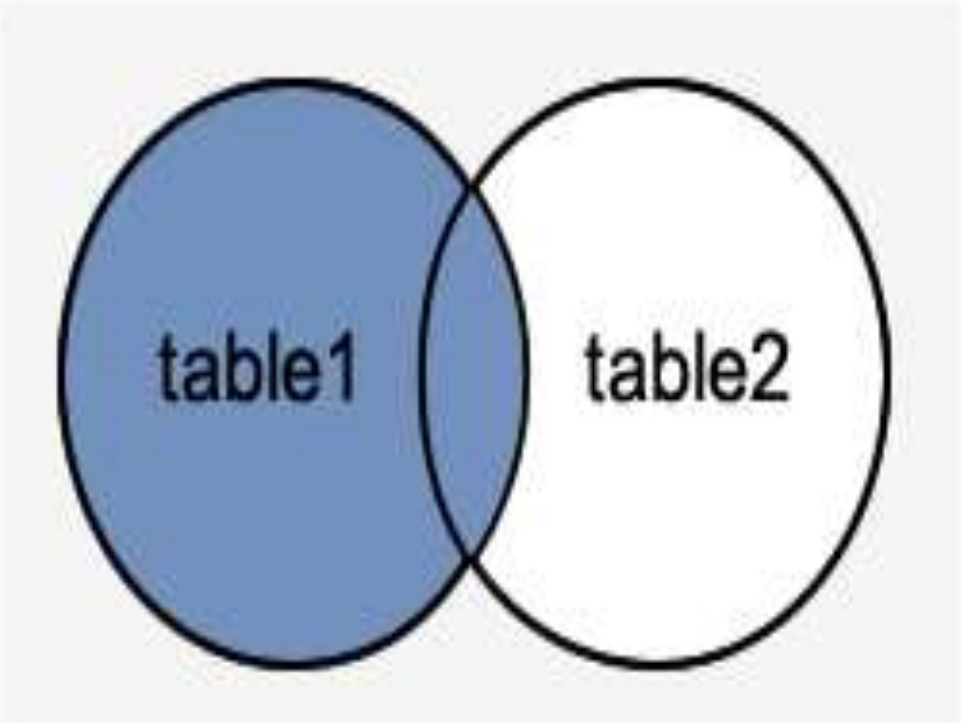
Left Outer Join

- **Left Outer Join** returns all rows from the left side table specified in the **ON** condition and only those rows from the right table where the join condition is met.
- It is used to retrieve all the matching records from both the tables as well as non-matching records from the left side table only.

Left Outer Join

- If there is no matching row found from the right table, the left join will have null values for the columns of the right table.

Left Outer Join



Left Outer Join

Syntax:

SELECT <columns>

FROM table1 **LEFT [OUTER] JOIN** table2

ON table1.column = table2.column;

ENO	ENAME	SAL	DNO
1	A	1000	10
2	B	2000	20
3	C	1000	10
4	D	3000	20
5	E	2000	-

DNO	DNAME	LOCATION
10	PROD	HYD
20	FINAN	DEL
40	SALES	MUM

Left Outer Join

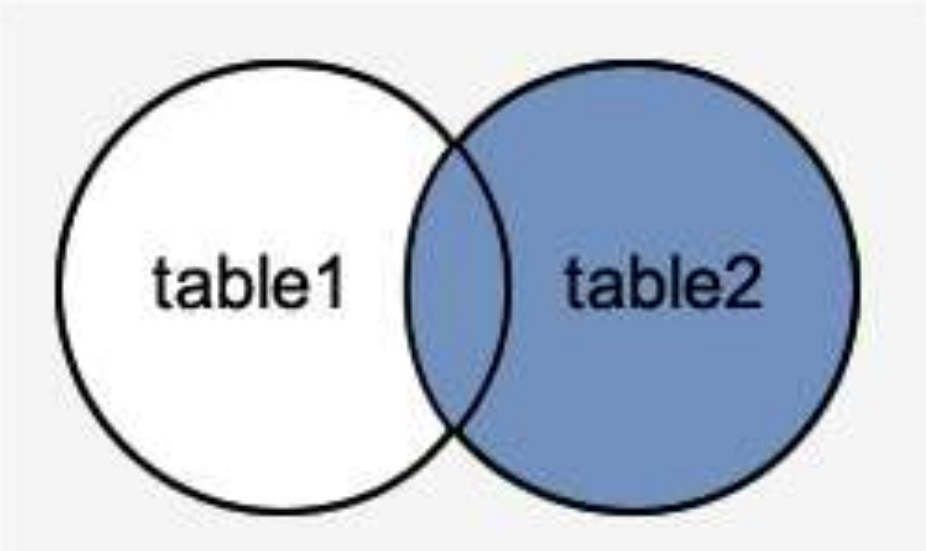
- SELECT * FROM emp LEFT OUTER JOIN dept ON emp.dno=dept.dno;

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
3	C	1000	10	10	PROD	HYD
1	A	1000	10	10	PROD	HYD
4	D	3000	20	20	FINAN	DEL
2	B	2000	20	20	FINAN	DEL
5	E	2000	-	-	-	-

Right Outer Join

- Right Outer Join returns all rows from the right-hand table specified in the ON condition and only those rows from the left table where the join condition is met.
- It is used to retrieve all the matching records from both the tables as well as non-matching records from the right side table only.

Right Outer Join



Right Outer Join

Syntax:

SELECT <columns>

FROM table1 **RIGHT [OUTER] JOIN** table2

ON table1.column = table2.column;

ENO	ENAME	SAL	DNO
1	A	1000	10
2	B	2000	20
3	C	1000	10
4	D	3000	20
5	E	2000	-

DNO	DNAME	LOCATION
10	PROD	HYD
20	FINAN	DEL
40	SALES	MUM

Right Outer Join

Example:

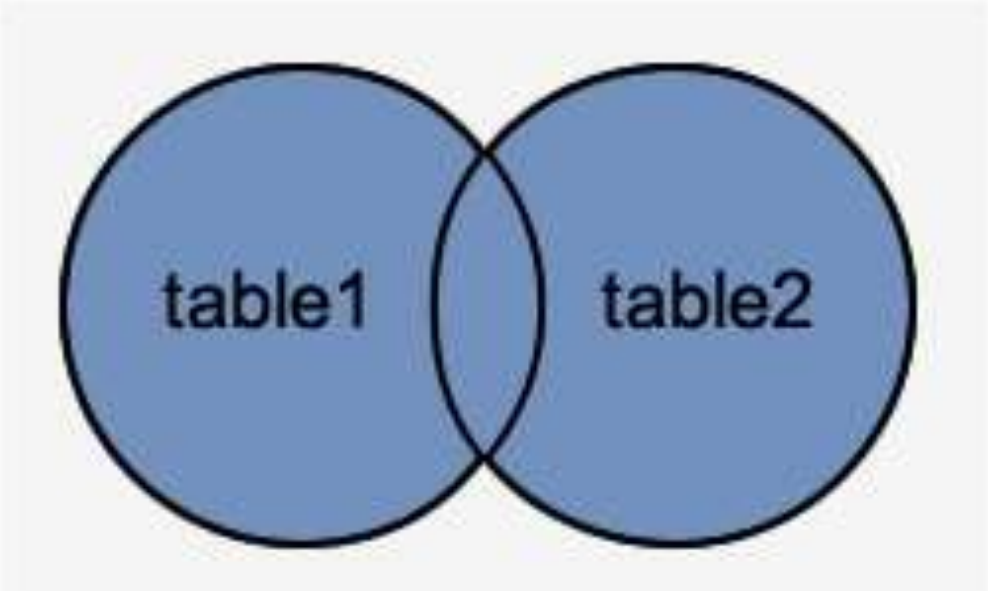
- SELECT * FROM emp RIGHT OUTER JOIN dept ON emp.dno=dept.dno;

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
1	A	1000	10	10	PROD	HYD
2	B	2000	20	20	FINAN	DEL
3	C	1000	10	10	PROD	HYD
4	D	3000	20	20	FINAN	DEL
				40	SALES	MUM

Full Outer Join

- **Full Outer Join** returns all rows from the left hand table and right hand table with the matching rows AND non matching rows from **both sides**.
- It places **NULL** where the join condition is not met.

Full Outer Join



Full Outer Join

Syntax:

SELECT <columns>

FROM table1 **FULL [OUTER] JOIN** table2

ON table1.column = table2.column;

ENO	ENAME	SAL	DNO
1	A	1000	10
2	B	2000	20
3	C	1000	10
4	D	3000	20
5	E	2000	-

DNO	DNAME	LOCATION
10	PROD	HYD
20	FINAN	DEL
40	SALES	MUM

Full Outer Join

- **SELECT * FROM emp FULL OUTER JOIN dept ON emp.dno=dept.dno;**

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
3	C	1000	10	10	PROD	HYD
1	A	1000	10	10	PROD	HYD
4	D	3000	20	20	FINAN	DEL
2	B	2000	20	20	FINAN	DEL
5	E	2000				
				40	SALES	MUM

EQUIJOIN

- **EQUIJOIN** contains only equality operator in join condition.
- Equi join returns the matching rows based on join condition(=) of the associated tables.
- When we use EQUI join between two or more tables, there should be a common column.
- Common column names need not be the same name but datatype must be matched.

EQUIJOIN

Syntax:

SELECT <column_list> **FROM** table1, table2
WHERE table1.column =table2.column;

(or)

SELECT <column_list> **FROM** table1 **JOIN** table2
ON table1.column =table2.column;

EQUIJOIN

SELECT * FROM emp **JOIN** dept

ON emp.dno **=** dept.dno;

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
1	A	1000	10	10	PROD	HYD
2	B	2000	20	20	FINAN	DEL
3	C	1000	10	10	PROD	HYD
4	D	3000	20	20	FINAN	DEL

EQUIJOIN

SELECT * FROM emp, dept

WHERE emp.dno **=** dept.dno;

ENO	ENAME	SAL	DNO	DNO	DNAME	LOCATION
1	A	1000	10	10	PROD	HYD
2	B	2000	20	20	FINAN	DEL
3	C	1000	10	10	PROD	HYD
4	D	3000	20	20	FINAN	DEL

Self Join

- In Self Join, a table is joined with itself.
- A self join specifies that each rows of a table is combined with itself and every other row of the table.
- To perform self join, use table alias with different names for table in the query.

Self Join

Syntax:

SELECT A1.column_name, A2.column_name...
FROM table1 **A1**, table1 **A2**
WHERE A1.common_filed = A2.common_field;

Self Join

ENO	ENAME	SAL	MGR_NO
1	A	1000	-
2	B	2000	1
3	C	1000	1
4	D	3000	5
5	E	2000	-

Self Join

- **Find the names of employees and their manager names.**

```
SELECT E1.ENAME  AS EMPNAME ,
                E2.ENAME AS MANAGER
                FROM EMP E1,EMP E2
WHERE E1.MGR_NO = E2.ENO;
```

EMPNAME	MANAGER
C	A
B	A
D	E

Self Join

- **Find the employees whose salary is > their manager salary.**