SQL JOINS

- A **join** is used to combine rows from multiple tables.
- A join is performed whenever two or more tables is listed in the FROM clause of an SQL statement.

Different types of Joins:

- Cartesian product
- Cross joins
- Equijoins
- Outer Joins
  - ✓ Full Outer Join
  - ✓ Left Outer Join
  - ✓ Right Outer Joins
- Inner Joins
- Self Joins

Table Join Basics:

Five points should be noted down while joining tables:

1. **The query shows that the join is performed with the other WHERE conditions.**
2. **Adding an Analytical Function to a Query that Contains a Join (and Other WHERE Conditions).**
3. **Used a GROUP BY in a query with no ordering or analytical function.**
4. **Adding Ordering to the Query Containing the GROUP BY.**
5. **Supplying Table Aliases.**

By Dinesh

Basics of alias

An alias is used to rename a column or a table. This operation is useful when one wants to give a more "vivid" name to a column a or when one wants to handle a table more easily in particular when there are various conditions of join. Giving a name of alias to a table is mandatory in the event of self-joins.

Alias of column

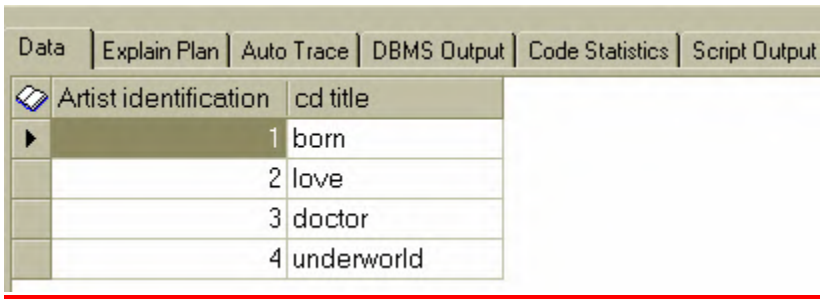
Syntax : Col1 **AS** "Column name".
In the result of the corresponding query, Col1 will be replaced by "Column name". The quotation marks are mandatory only if the name of alias comprises spaces.

Alias of table

Syntax : Table1 Alias1
In a query, Alias1 will be identical to Table1

Eg:

```
SELECT m.artist_id AS "Artist identification", c.title AS "cd title"
FROM music m, cd c
WHERE m.artist_id = c.artist_id;
```



Cartesian product

- A Cartesian join will get you a Cartesian product.
- A Cartesian join is when you join every row of one table to every row of another table.
- You can also get one by joining every row of a table to every row of itself.

By Dinesh

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> describe cart_join;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 NAME                                               CHAR(20)
 DOB                                                DATE
 CITY                                               CHAR(10)

SQL> describe cart_join2;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 NAME                                               CHAR(20)
 DOB                                                DATE
 CITY                                               CHAR(10)

SQL>
```

Cart_join:

Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

NAME	DOB	CITY
dinesh	1/6/1986	chennai
shovan	3/7/1985	ranchi
vinush	3/10/1984	mysore

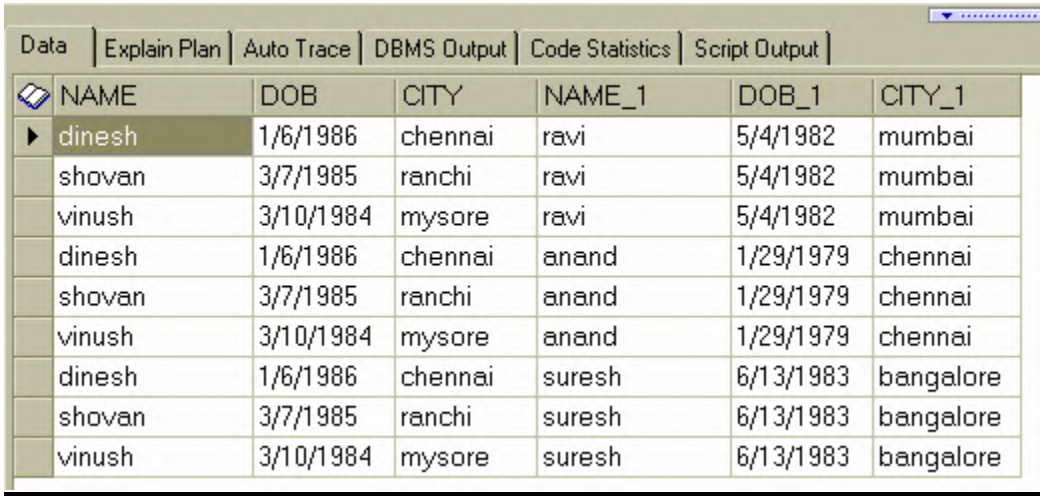
Cart_join2

Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

NAME	DOB	CITY
ravi	5/4/1982	mumbai
anand	1/29/1979	chennai
suresh	6/13/1983	bangalore

By Dinesh

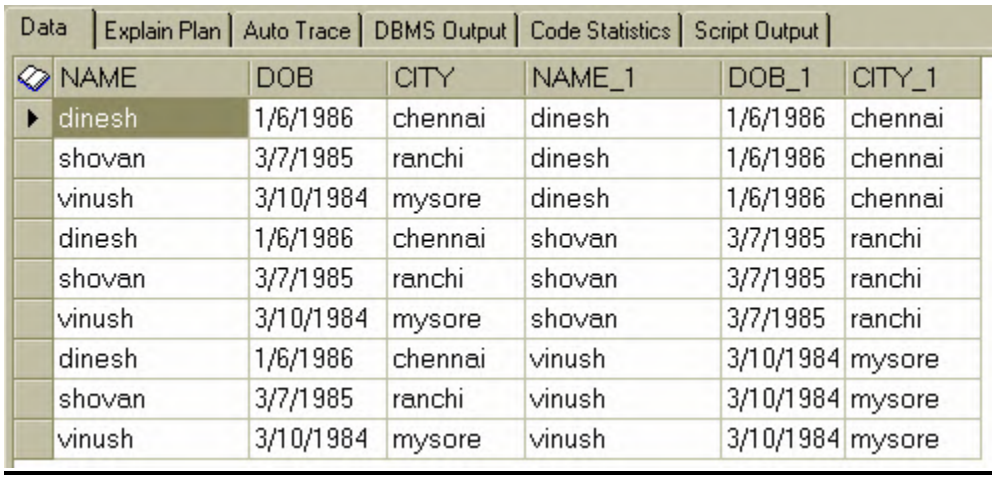
```
select * from cart_join, cart_join2
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

NAME	DOB	CITY	NAME_1	DOB_1	CITY_1
dinesh	1/6/1986	chennai	ravi	5/4/1982	mumbai
shovan	3/7/1985	ranchi	ravi	5/4/1982	mumbai
vinush	3/10/1984	mysore	ravi	5/4/1982	mumbai
dinesh	1/6/1986	chennai	anand	1/29/1979	chennai
shovan	3/7/1985	ranchi	anand	1/29/1979	chennai
vinush	3/10/1984	mysore	anand	1/29/1979	chennai
dinesh	1/6/1986	chennai	suresh	6/13/1983	bangalore
shovan	3/7/1985	ranchi	suresh	6/13/1983	bangalore
vinush	3/10/1984	mysore	suresh	6/13/1983	bangalore

```
select * from cart_join j1, cart_join j2
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

NAME	DOB	CITY	NAME_1	DOB_1	CITY_1
dinesh	1/6/1986	chennai	dinesh	1/6/1986	chennai
shovan	3/7/1985	ranchi	dinesh	1/6/1986	chennai
vinush	3/10/1984	mysore	dinesh	1/6/1986	chennai
dinesh	1/6/1986	chennai	shovan	3/7/1985	ranchi
shovan	3/7/1985	ranchi	shovan	3/7/1985	ranchi
vinush	3/10/1984	mysore	shovan	3/7/1985	ranchi
dinesh	1/6/1986	chennai	vinush	3/10/1984	mysore
shovan	3/7/1985	ranchi	vinush	3/10/1984	mysore
vinush	3/10/1984	mysore	vinush	3/10/1984	mysore

By Dinesh

Cross Join

- A cross join returns the cartesian product of the sets of records from the two joined tables.
- If A and B are two sets, then cross join = A X B.

```
Oracle SQL*Plus
File  Edit  Search  Options  Help
SQL> describe music
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ARTIST_ID                                          NUMBER(38)

SQL> describe cd;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ARTIST_ID                                          NUMBER(38)
 TITLE                                              CHAR(20)
 TYPE                                               CHAR(10)

SQL>
```

Music:

Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID
1
2
3
4

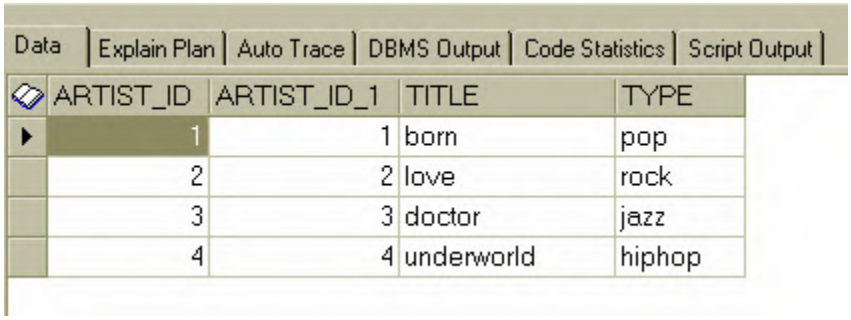
CD:

Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	TITLE	TYPE
1	born	pop
4	underworld	hiphop
3	doctor	jazz
2	love	rock

By Dinesh

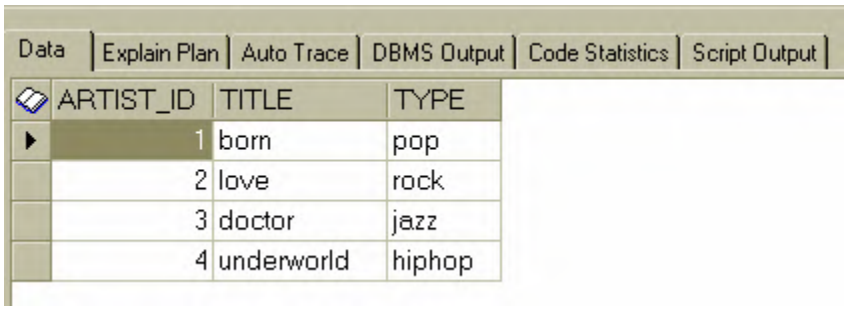
```
select * from music m, cd c where m.artist_id = c.artist_id
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	ARTIST_ID_1	TITLE	TYPE
1	1	born	pop
2	2	love	rock
3	3	doctor	jazz
4	4	underworld	hiphop

```
select m.artist_id, c.title, c.type from music m, cd c where
m.ARTIST_ID = c.ARTIST_ID
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	TITLE	TYPE
1	born	pop
2	love	rock
3	doctor	jazz
4	underworld	hiphop

Equi-Join:

- An **equi-join** also known as an **equijoin**, a specific type of comparator-based join, or *theta join*, uses only equality comparisons in the join-predicate.
- This operation allows to connect, with a relation of equality, the tables which have at least a common attribute. One must have n-1 conditions of join, n being the number of tables which intervene in the query.

If no condition of join is specified, the corresponding query will realize the *Cartesian product* of the implied tables.

Note:

- **Using other comparison operators (such as <) disqualifies a join as an equi-join.**
- **If no condition of join is specified, the corresponding query will realize the *Cartesian product* of the implied tables.**

```
select * from music m, cd c where m.artist_id = c.artist_id
(write the outputabove)
```

By Dinesh

Outer Join:

A "standard" join between 2 tables, or *inner join*, returns rows only if the column of join of a table is equal to the column of join of the other table.

It can be useful, in certain circumstances, to display all the rows of a particular table if there is or not matching with the other table.

The columns for which there is no matching are filled with the value NULL. This operation is called an **outer join.**

There are 3 types of outer joins and the way an outer join is performed depends on the position of the tables compared to the join instruction.

- **Left Outer Join**
- **Right Outer Join**
- **Full Outer Join**

Left Outer Join:

Syntax:

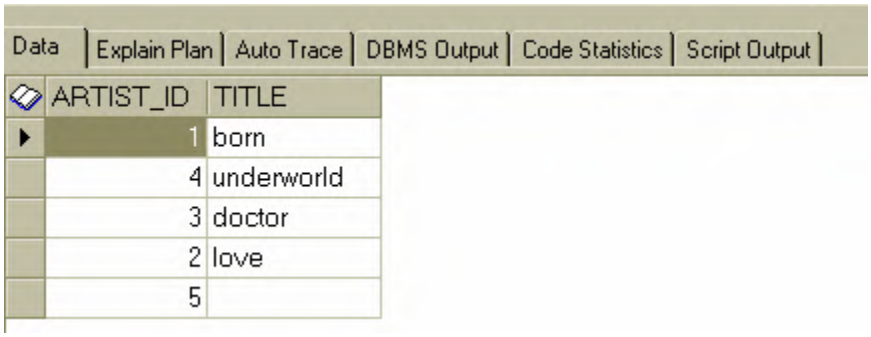
SELECT
FROM *table1* [*alias*]
LEFT OUTER JOIN *table2* [*alias*]
ON <Join conditions>

NOTE:

All the rows of *table1* will be displayed even if < Join conditions > is not realized in *table2*.

By Dinesh

```
select m.artist_id,c.title from music m
left outer join cd c
on m.ARTIST_ID = c.ARTIST_ID
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	TITLE
1	born
4	underworld
3	doctor
2	love
5	

Right Outer Join:

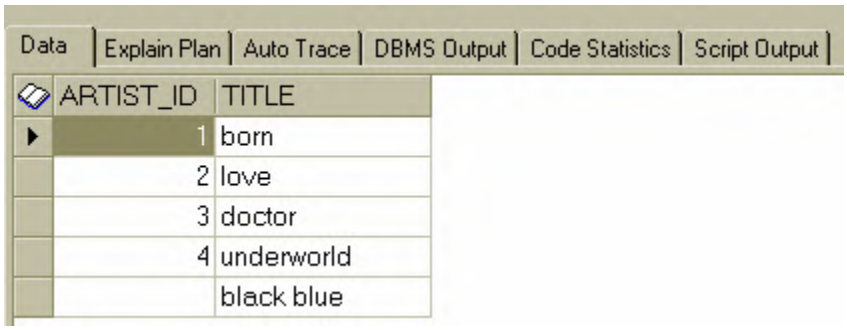
Syntax:

SELECT ...
FROM *table1* [*alias*]
RIGHT OUTER JOIN *table2* [*alias*]
ON <Join conditions>

Note:

All the rows of *table2* will be displayed even if < Join conditions > is not realized in *table1*.

```
select m.artist_id,c.title from music m
right outer join cd c
on m.ARTIST_ID = c.ARTIST_ID
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	TITLE
1	born
2	love
3	doctor
4	underworld
	black blue

By Dinesh

Full Outer Join:

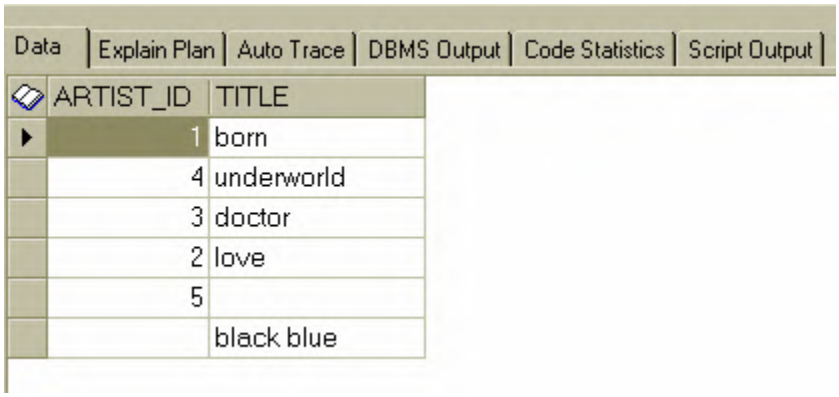
Syntax:

SELECT
FROM *table1* [*alias*]
FULL OUTER JOIN *table2* [*alias*]
ON <Join conditions>

Note:

All the rows of *table1* and *table2* will be displayed and the columns for which there is no matching will be filled with value NULL.

```
select m.artist_id,c.title from music m
full outer join cd c
on m.ARTIST_ID = c.ARTIST_ID
```

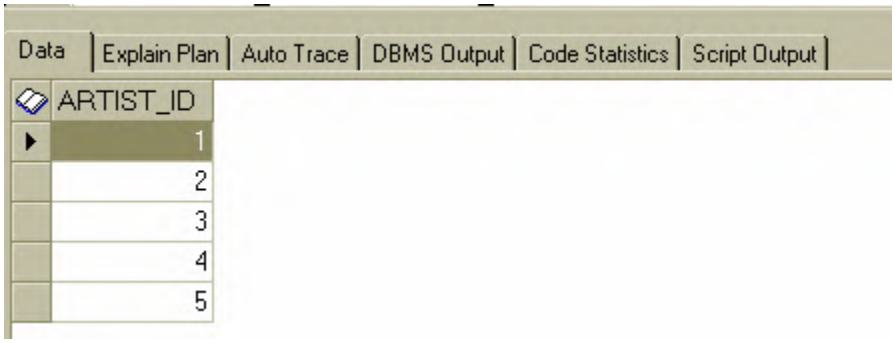


ARTIST_ID	TITLE
1	born
4	underworld
3	doctor
2	love
5	
	black blue

Self Join:

It is the join of a table with itself. This operation is useful when one wishes to connect attributes which are inside the same table.

```
select m1.artist_id from music m1, music m2 where m1.artist_id =
m2.artist_id
```



ARTIST_ID
1
2
3
4
5

Inner Join:

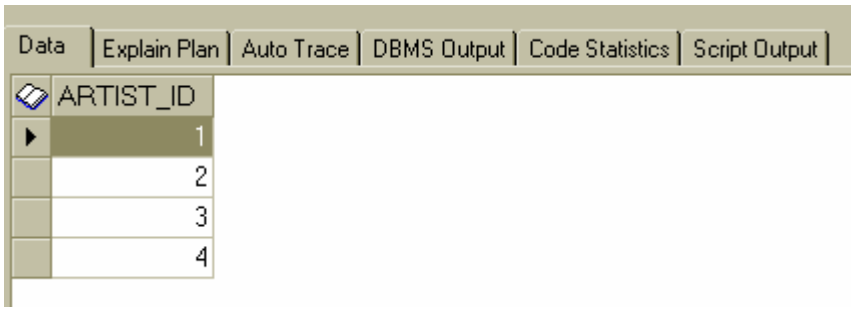
We should use the SQL INNER JOIN when you only want to return records where there is at least one row in both tables that match the join condition.

The **INNER JOIN** operation can be used in any **FROM** clause to combine records from two tables. It is, in fact, the most common type of join.

NOTE:

- An **INNER JOIN** cannot be nested inside a **LEFT JOIN** or **RIGHT JOIN**
- There must be a matching value in a field common to both tables.

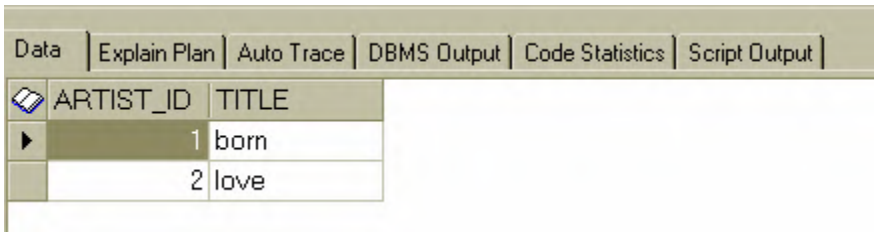
```
select m.artist_id from
music m inner join cd c
on m.artist_id = c.artist_id
```



ARTIST_ID
1
2
3
4

By Dinesh

```
select m.artist_id, c.title from
music m inner join cd c
on m.artist_id = c.artist_id
where m.artist_id = 2 or c.TYPE like 'pop'
```

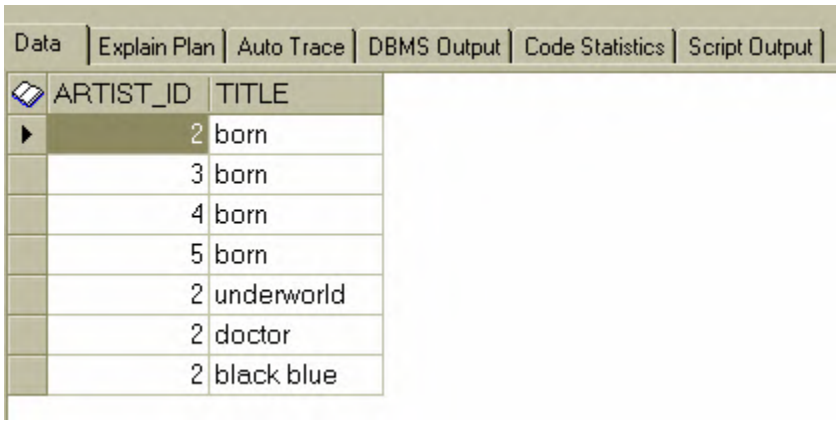


Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	TITLE
1	born
2	love

With the INNER JOIN operation any relational comparison operator can be used in the ON clause: =, <, >, <=, >=, or <>.

```
select m.artist_id, c.title from
music m inner join cd c
on m.artist_id <> c.artist_id
where m.artist_id = 2 or c.TYPE like 'pop'
```



Data | Explain Plan | Auto Trace | DBMS Output | Code Statistics | Script Output

ARTIST_ID	TITLE
2	born
3	born
4	born
5	born
2	underworld
2	doctor
2	black blue

By Dinesh