Joins

Joins in SQL are the statements or clauses using which we can combine two or more tables, based on some common fields present among the tables

- Types of Join
  - Cartesian Product or Cross join
  - Equi Join
  - Natural Join
  - Non-Equi Join
  - Self Join
  - Left Outer Join
  - Right Outer Join

Cartesian Product/Cross Join

Cartesian Product operation combines tuples from two relations. It results in all pairs of rows from the two input relations, regardless of whether or not they have the same values on common attributes. It is denoted as 'X'.

Mathematics

SNo	Name	Class
1	Ananya	6A
2	Sara	6A
3	Nashim	7B
4	Sanjay	7A

Computer

SNo	Name	Class
1	Tanya	6A
2	Sharukh	7A
3	Mithun	8A
4	Aliya	9B

SNo	Name	Class	SNo	Name	Class
1	Ananya	6A	1	Tanya	6A
2	Sara	6A	1	Tanya	6A
3	Nashim	7B	1	Tanya	6A
4	Sanjay	7A	1	Tanya	6A
1	Ananya	6A	2	Sharukh	7A
2	Sara	6A	2	Sharukh	7A
3	Nashim	7B	2	Sharukh	7A
4	Sanjay	7A	2	Sharukh	7A
1	Ananya	6A	3	Mithun	8A
2	Sara	6A	3	Mithun	8A
3	Nashim	7B	3	Mithun	8A
4	Sanjay	7A	3	Mithun	8A
1	Ananya	6A	4	Aliya	9B
2	Sara	6A	4	Aliya	9B
3	Nashim	7B	4	Aliya	9B
4	Sanjay	7A	4	Aliya	9B

```
select * from mathematics,computer;
```

```
select * from mathematics cross join
computer;
```

Q. Display all the possible combinations of tuples of relations mathematics and computer.

```
select * from mathematics,computer;
```

Mathematics

SNo	Name	Class
1	Ananya	6A
2	Sara	6A
3	Nashim	7B
4	Sanjay	7A

Computer

SNo	Name	Class
1	Tanya	6A
2	Sharukh	7A
3	Mithun	8A
4	Aliya	9B

SNo	Name	Class	SNo	Name	Class
1	Ananya	6A	1	Tanya	6A
2	Sara	6A	1	Tanya	6A
3	Nashim	7B	1	Tanya	6A
4	Sanjay	7A	1	Tanya	6A
1	Ananya	6A	2	Sharukh	7A
2	Sara	6A	2	Sharukh	7A
3	Nashim	7B	2	Sharukh	7A
4	Sanjay	7A	2	Sharukh	7A
1	Ananya	6A	3	Mithun	8A
2	Sara	6A	3	Mithun	8A
3	Nashim	7B	3	Mithun	8A
4	Sanjay	7A	3	Mithun	8A
1	Ananya	6A	4	Aliya	9B
2	Sara	6A	4	Aliya	9B
3	Nashim	7B	4	Aliya	9B
4	Sanjay	7A	4	Aliya	9B

Degree and Cardinality

If Table 1 has degree d1 and cardinality c1 and table2 has degree d2 and cardinality c2, their Cartesian Product has degree d=d1+d2 and cardinality c=c1*c2;

Equi-Join

names

SNo	Name
1	Roji
2	Sara
3	Nashim
4	Kajol

class

SNo	Class
1	6A
2	7A
3	8A
4	9B

Explicit use of Join Clause

Select *from names n,class c where n.SNo=c.SNo;

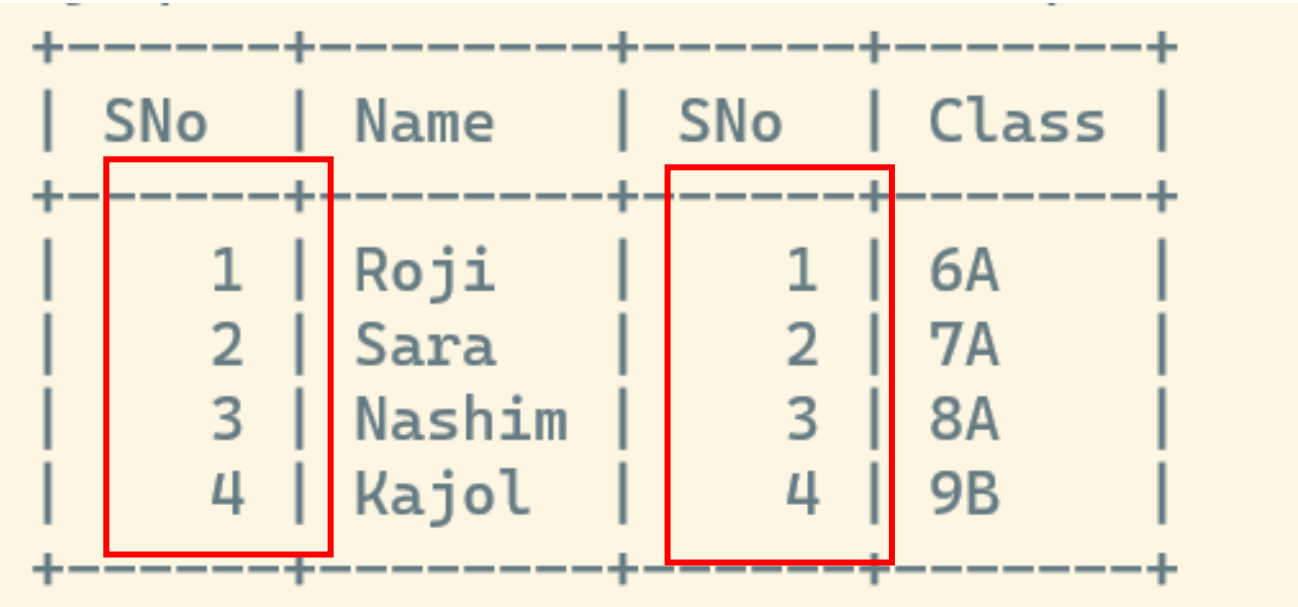
Select * from names n join class c on n.SNo=c.SNo;

SNo	Name	SNo	Class
1	Roji	1	6A
2	Sara	2	7A
3	Nashim	3	8A
4	Kajol	4	9B

Natural Join

The output of the previous queries have repetitive column Sno having exactly the same values. This redundant column is of no use. So, there is an extension of JOIN operation called NATURAL JOIN which works similar to JOIN clause in SQL but removes the redundant attribute. This operation can be used to join the contents of two tables if there is one common attribute in both the tables.

A natural join combines rows from both tables based on columns with the same name and data type.



SNo	Name	SNo	Class
1	Roji	1	6A
2	Sara	2	7A
3	Nashim	3	8A
4	Kajol	4	9B

Natural Join

select * from names natural join class;

names

SNo	Name
1	Roji
2	Sara
3	Nashim
4	Kajol

class

SNo	Class
1	6A
2	7A
3	8A
4	9B

SNo	Name	Class
1	Roji	6A
2	Sara	7A
3	Nashim	8A
4	Kajol	9B

Contd..

Employee

EMP_ID	EMP_NAME	DEPT_NAME
1	SUMIT	HR
2	JOEL	IT
3	BISWA	MARKETING
4	VAIBHAV	IT
5	SAGAR	SALES

Department

DEPT_NAME	MANAGER_NAME
IT	ROHAN
SALES	RAHUL
HR	TANMAY
FINANCE	ASHISH
MARKETING	SAMAY

```
SELECT *
FROM employee
NATURAL JOIN department;
```

EMP_ID	EMP_NAME	DEPT_NAME	MANAGER_NAME
1	SUMIT	HR	TANMAY
2	JOEL	IT	ROHAN
3	BISWA	MARKETING	SAMAY
4	VAIBHAV	IT	ROHAN
5	SAGAR	SALES	RAHUL

- Difference between Equi-Join vs Natural Join

Equi-Join	Natural Join
Join performed on equality of value of the columns	Join is performed on column haing common name.
Where clause is used to specify the condition	There is no need to use where clause
Both columns from tables are displayed in the result.	Common column is displayed only once