Set Operations

In SQL, Set operations are used to combine multiple SELECT statements. There are mainly four types of set operations in SQL -

1. Union
2. Intersect
3. Minus

- **What is the difference between JOIN and Set Operations(UNION)?**

- **UNION -**
  This operator is used to combine two or more more SELECT statements.

  General Form -
  SELECT column(s) FROM
  table1 UNION
  SELECT column(s) FROM table2;
  Ex- Below we have two tables

  Table - **Ninja**

Ninja_id	Ninja_Name	City
1	Saksham	Dehradun
2	Sachin	Vadodara
3	Utkarsh	Varanasi

Table - **Course**

Course_id	Course_Name	City
1	Database	Lucknow
2	OS	Mumbai
3	Networks	Delhi
4	ML	Varanasi

```
SELECT City FROM Ninja
UNION
SELECT City FROM Course;
```

Output:

City
Dehradun
Vadodara
Varanasi
Lucknow
Mumbai
Delhi

What if we replace UNION with INNER JOIN in the above query?

```
SELECT City FROM Ninja
INNER JOIN
SELECT City FROM Course;
```

Output:

City
Dehradun
Vadodara
Varanasi
Lucknow
Mumbai
Delhi
Varanasi

As you can see there is recurrence in case of INNER JOIN. Hence, let us understand the difference between JOIN and Set Operations.

- **What is the difference between JOIN and Set Operations(UNION)?**

JOIN	UNION(Set Operations)
JOIN Combines multiple table depending upon the matched condition	UNION(& other Set Operations) Combines is the resulting set from two or more SELECT statements
Combination is column-wise	Combination is row-wise
Datatypes of corresponding columns from each table could be different.	Datatypes of corresponding columns from each table should be the same.
May or maynot return distinct row(s)	Returns distinct row(s)
From each table the number of column(s) selected may or maynot be the same.	From each table the number of column(s) selected should be the same.

Note:- Above difference is valid for other set operations and join(inner, left, right) as well.

- **INTERSECT-**

This operator returns the common values of the tables.

Ex- Below we have two tables -

Ninja -

Ninja_id	Ninja_Name	City
1	Saksham	Dehradun
2	Sachin	Vadodara
3	Utkarsh	Varanasi

Course-

Course_id	Course_Name	City
1	Database	Lucknow
2	OS	Mumbai
3	Networks	Delhi
4	ML	Varanasi

```
SELECT DISTINCT City
FROM Ninja
INNER JOIN Course
USING(City);
```

Output:

City
Varanasi

- **MINUS-**

  This operator returns the distinct row from the first table that does not occur in the second table.

  Ex- Below we have two tables

  Table- **Ninja**

Ninja_id	Ninja_Name	City
1	Saksham	Dehradun
2	Sachin	Vadodara
3	Utkarsh	Varanasi

  Table - **Course**

Course_id	Course_Name	City
1	Database	Lucknow
2	OS	Mumbai
3	Networks	Delhi
4	ML	Varanasi

  Query:

```
SELECT City FROM Ninja
LEFT JOIN Course USING(City)
WHERE Course.City IS NULL;
```

Output:

City
Dehradun
Vadodara