

## SQL Operators:

SQL (Structured Query Language) operators are the building blocks for working with data in a database. These operators help you manipulate, compare, and evaluate data, making it possible to perform a wide range of tasks, from simple calculations to complex queries. Let's break down some of the most common types of SQL operators:

- 1. Arithmetic Operators:** These operators handle basic math operations like addition (+), subtraction (-), multiplication (\*), and division (/). Use them to perform calculations directly within your queries.
- 2. Comparison Operators:** Use these to compare values. Examples include equal to (=), not equal to (<>), greater than (>), less than (<), greater than or equal to (>=), and less than or equal to (<=). They are essential for filtering and sorting data based on conditions.
- 3. Logical Operators:** Combine multiple conditions in your queries with logical operators like AND, OR, and NOT. These operators help refine your search results by defining complex conditions that must be met.
- 4. Set Operators:** Combine results from two or more queries using set operators such as UNION, INTERSECT. These are perfect when you need to merge data from different tables or queries into a single result set.

## General Structure & Examples of SQL Operators:

### *Example of Arithmetic Operators in MySQL*

Add (+):

```
Syntax: SELECT num1, num2, num1 + num2 AS sum FROM numbers;
```

Example:

num1	num2	sum
5	10	15
15	20	35

Subtraction (-):

```
Syntax: SELECT num1, num2, num1 - num2 AS difference FROM numbers;
```

Example:

num1	num2	difference
5	10	-5
15	20	-5

Multiply ( \* ):

```
Syntax: SELECT num1, num2, num1 * num2 AS product FROM numbers;
```

Example:

num1	num2	product
5	10	50
15	20	300

Division ( / ):

```
Syntax: SELECT num1, num2, num1 / num2 AS quotient FROM numbers;
```

Example:

num1	num2	quotient
5	10	0.5000
15	20	0.7500

Modulo ( % ):

```
Syntax: SELECT num1, num2, num1 % num2 AS remainder FROM numbers;
```

Example:

num1	num2	remainder
5	10	5
15	20	15

## SQL Comparison Operator Examples

Equal to ( = ):

Syntax: **SELECT \* FROM MATHS WHERE MARKS=50;**

Example:

ROLL_NUMBER	S_NAME	MARKS
5	MOHAN	50

Greater Than (>):

Syntax: **SELECT \* FROM MATHS WHERE MARKS>60;**

Example:

ROLL_NUMBER	S_NAME	MARKS
1	ABHI	70
2	RAVI	80
3	ARJUN	90
4	SAM	100

Less Than (<):

Syntax: **SELECT \* FROM MATHS WHERE MARKS<40;**

Example:

ROLL_NUMBER	S_NAME	MARKS
6	ROHAN	10
7	ROCKY	20
9	NEHA	30

Greater Than or Equal (>=):

Syntax: **SELECT \* FROM MATHS WHERE MARKS>=80;**

Example:

ROLL_NUMBER	S_NAME	MARKS
2	RAVI	80
3	ARJUN	90
4	SAM	100

Less Than or Equal ( <= ) :

Syntax: **SELECT \* FROM MATHS WHERE MARKS<=30;**

Example:

ROLL_NUMBER	S_NAME	MARKS
6	ROHAN	10
7	ROCKY	20
9	NEHA	30

Not Equal ( <> ) :

Syntax: **SELECT \* FROM MATHS WHERE MARKS<>70;**

Example:

ROLL_NUMBER	S_NAME	MARKS
2	RAVI	80
3	ARJUN	90
4	SAM	100
5	MOHAN	50
6	ROHAN	10
7	ROCKY	20
8	AYUSH	40
9	NEHA	30
10	KRITI	60

## Logical Operators

AND:

Syntax: SELECT \* FROM employee WHERE emp\_city = 'Allahabad' AND emp\_country = 'India';

Example:

emp_id	emp_name	emp_city	emp_country
104	Utkarsh Singh	Allahabad	India
105	Sudhanshu Yadav	Allahabad	India

OR :

Syntax: SELECT \* FROM employee WHERE emp\_city = 'Varanasi' OR emp\_country = 'India';

Example:

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
104	Utkarsh Singh	Allahabad	India
105	Sudhanshu Yadav	Allahabad	India
106	Ashutosh Kumar	Patna	India

LIKE :

Syntax: SELECT \* FROM employee WHERE emp\_city LIKE 'P%';

Example:

emp_id	emp_name	emp_city	emp_country
106	Ashutosh Kumar	Patna	India

NOT LIKE :

Syntax: SELECT \* FROM employee WHERE emp\_city NOT LIKE 'P%';

Example:

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
104	Utkarsh Singh	Allahabad	India
105	Sudhanshu Yadav	Allahabad	India

BETWEEN:

```
Syntax: SELECT * FROM employee WHERE emp_id BETWEEN 101 AND 104;
```

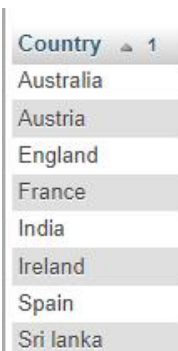
Example:

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
104	Utkarsh Singh	Allahabad	India

## SQL UNION Operator

UNION:

```
Syntax: SELECT Country FROM Emp1 UNION SELECT Country FROM Emp2 ORDER BY Country;
```



Country
Australia
Austria
England
France
India
Ireland
Spain
Sri lanka

Example:

INTERSECT:

```
Syntax: SELECT Country FROM Emp1 INTERSECT SELECT Country FROM Emp2 ORDER BY Country;
```

Example:



Country
India
Spain

Saif hossain Likhon

Id:21225103219

