

# OPERATORS IN SQL

1. ARITHMETIC OPERATORS :- ( + , - , \* , / )
2. CONCATENATION OPERATOR :- ( || )
3. COMPARISON OPERATORS :- ( = , != or <> )
4. RELATIONAL OPERATOR :- ( > , < , >= , <= )
5. LOGICAL OP : ( AND , OR , NOT )
6. SPECIAL OPERATOR :-
  1. IN
  2. NOT IN
  3. BETWEEN
  4. NOT BETWEEN
  5. IS
  6. IS NOT
  7. LIKE
  8. NOT LIKE
7. SUBQUERY OPERATORS:-
  1. ALL
  2. ANY
  3. EXISTS
  4. NOT EXISTS

## CONCATENATION Operator :

" It is used to join the strings ".

Symbol :
----------

Example :   SELECT ENAME  
              FROM EMP  
              WHERE JOB ='MANAGER' ;

<u>Ename</u>
ALLEN
MARTIN
SMITH

```
SELECT 'Hi ' || ename
FROM EMP
WHERE JOB ='MANAGER' ;
```

<u>Ename</u>
Hi ALLEN
Hi MARTIN
Hi SMITH

- WAQTD name and deptno of the employees hiredAfter '01-JAN-87' .

```
SELECT ENAME, DEPTNO FROM EMP
WHERE HIREDATE > '01-JAN-1987' ;
```

- WAQTD name and hiredate of the employees hiredbefore 31-JUL-88

```
SELECT ENAME , HIREDATEFROM EMP
WHERE HIREDATE < '31-JUL-88' ;
```

## **LOGICAL OPERATORS**

1. AND
2. OR
3. NOT

### **We use logical operators to write multiple conditions .**

1. WAQTD name and deptno along with job for theemployee working in dept 10 .

```
SELECT ENAME , DEPTNO , JOB
FROM EMP
WHERE DEPTNO = 10 ;
```

2. WAQTD name and deptno along with job for theemployee working as manager in dept 10 .

```
SELECT ENAME , DEPTNO , JOB
FROM EMP
WHERE JOB ='MANAGER' AND DEPTNO = 10 ;
```

3. WAQTD name , deptno , salary of the employee working in dept 20 and earning less than 3000 .

```
SELECT ENAME, DEPTNO , SAL
FROM EMP
WHERE DEPTNO = 20 AND SAL < 3000 ;
```

4. WAQTD name and salary of the employee if emp earns More than 1250 but less than 3000 .

```
SELECT ENAME , SAL
FROM EMP
WHERE SAL > 1250 AND SAL < 3000 ;
```

5. WAQTD name and deptno of the employees if the works in dept 10 or 20 .

```
SELECT ENAME , DEPTNO
FROM EMP
```

WHERE DEPTNO = 10 OR DEPTNO = 20 ;

6. WAQTD name and sal and deptno of the employees. If emp gets more than 1250 but less than 4000 and works in dept 20.

```
SELECT ENAME , SAL , DEPTNO
FROM EMP
WHERE SAL > 1250 AND SAL < 4000 AND DEPTNO = 20 ;
```

7. WAQTD name , job , deptno of the employees working as a manager in dept 10 or 30 .

```
SELECT ENAME , JOB , DEPTNO
FROM EMP
WHERE JOB ='MANAGER' AND ( DEPTNO = 10 OR DEPTNO = 20 ) ;
```

8. WAQTD name , deptno , job of the employees working in dept 10 or 20 or 30 as a clerk .

```
SELECT ENAME , JOB , DEPTNO
FROM EMP
WHERE JOB ='CLERK' AND ( DEPTNO = 10 OR DEPTNO = 20 AND
DEPTNO = 30 ) ;
```

9. WAQTD name , job and deptno of the employees working as clerk or manager in dept 10 .

```
SELECT ENAME , JOB , DEPTNO
FROM EMP
WHERE ( JOB = 'CLERK' OR JOB ='MANAGER' ) AND DEPTNO = 10 ;
```

10. WAQTD name , job , deptno , sal of the employees working as clerk or salesman in dept 10 or 30 and earning more than 1800 .

```
SELECT ENAME , JOB , SAL
FROM EMP
WHERE ( JOB ='CLERK' OR JOB ='SALESMAN' ) AND ( DEPTNO = 10
OR DEPTNO = 30 ) AND SAL >1800 ;
```

## SPECIAL OPERATORS

1. IN
2. NOT IN
3. BETWEEN
4. NOT BETWEEN
5. IS
6. IS NOT
7. LIKE
8. NOT LIKE

1. **IN** : It is a multi-valued operator which can accept multiple values At the RHS .

**Syntax:** Column\_Name / Exp **IN** ( v1 , v2 , . . Vn )

- WAQTD name and deptno of the employees working in dept 10 or 30 .

```
SELECT ENAME , DEPTNO FROM EMP
WHERE DEPTNO = 10 OR DEPTNO = 30 ;
```

```
SELECT ENAME , DEPTNO FROM EMP
WHERE DEPTNO IN ( 10 , 30 ) ;
```

- WAQTD name and job of the employee working as a clerk or manager Or salesman .

```
SELECT ENAME , JOB
FROM EMP
WHERE JOB IN ('CLERK' , 'MANAGER' , 'SALESMAN' ) ;
```

- WAQTD empno , ename and salary of the employees whose empno is 7902 or 7839 and getting salary more than 2925.

```
SELECT EMPNO , ENAME , SAL
FROM EMP
WHERE EMPNO IN ( 7902 , 7839 ) AND SAL > 2925 ;
```

2. **NOT IN** : It is a multi-valued operator which can accept multiple values At the RHS . It is similar to IN op instead of selecting it Rejects the values .

**Syntax:** Column\_Name / Exp **NOT IN** ( v1 , v2 , . . vn )

- WAQTD name and deptno of all the employees except the emp Working in dept 10 or 40 .

```
SELECT ENAME , DEPTNO FROM EMP
WHERE DEPTNO NOT IN ( 10 , 40 ) ;
```

- WAQTD name , deptno and job of the employee working in dept 20 but not as a clerk or manager.

```
SELECT ENAME , DEPTNO
FROM EMP
WHERE DEPTNO = 20 AND JOB NOT IN ( 'CLERK' , 'MANAGER' ) ;
```

**3.BETWEEN :** *"It is used whenever we have range of values "*  
[ Start value and Stop Value ] .

Syntax:

Column_Name BETWEEN Lower_Range AND Higher_Range ;
--

- Between Op works including the range .

- WAQTD name and salary of the employees if the emp is earning Salary in the range 1000 to 3000 .

```
SELECT ENAME , SAL FROM EMP
WHERE SAL BETWEEN 1000 AND 3000 ;
```

- WAQTD name and deptno of the employees working in dept 10 And hired during 2019 (the entire year of 2019) .

```
SELECT ENAME, DEPTNO
FROM EMP
WHERE DEPTNO = 10 AND HIREDATE BETWEEN '01-JAN-2019' AND '31-DEC-2019' ;
```

- WAQTD name , sal and hiredate of the employees hired during 2017 into dept 20 with a salary greater than 2000 .

```
SELECT ENAME , SAL , HIREDATE
FROM EMP
WHERE DEPTNO = 20 AND SAL > 2000 AND HIREDATE BETWEEN
'01-JAN-2017' AND '31-DEC-2017' ;
```

**4.NOT BETWEEN :** It is Opposite of Between .

Syntax:

Column_Name NOT BETWEEN Lower_Range AND Higher_Range ;
--

- WAQTD name and salary of the employees if the emp is not earning Salary in the range 1000 to 3000 .

```
SELECT ENAME , SAL FROM EMP
WHERE SAL NOT BETWEEN 1000 AND 3000 ;
```

- WAQTD name and deptno of the employees working in dept 10 And not hired during 2019 .

```
SELECT ENAME , DEPTNO
FROM EMP
WHERE DEPTNO = 10 AND HIREDATE NOT BETWEEN '01-JAN-2019' AND '31-DEC-2019' ;
```

- WAQTD name , sal and hiredate of the employees who were not hired during 2017 into dept 20 with a salary greater than 2000 .

```
SELECT ENAME , SAL , HIREDATE
FROM EMP
WHERE DEPTNO = 20 AND SAL > 2000 AND HIREDATE NOT BETWEEN
'01-JAN-2017' AND '31-DEC-2017' ;
```

#### **5. IS :** "It is used to compare only NULL "

Syntax: Column\_Name **IS** NULL ;

Example :

<b><u>EID</u></b>	<b><u>ENAME</u></b>	<b><u>SAL</u></b>	<b><u>COMM</u></b>
1	A	1000	100
2	B	null	null
3	C	null	200
4	D	2000	null

- WAQTD name of the employee who is not getting salary .

```
SELECT ENAME
FROM EMP
WHERE SAL IS NULL ;
```

- WAQTD name of the emp who doesn't get commission .

```
SELECT ENAME
FROM EMP
WHERE COMM IS NULL ;
```

- WAQTD name , sal and comm of the emp if the emp doesn't earn both .

```
SELECT ENAME , SAL , COMM
FROM EMP
WHERE COMM IS NULL AND SAL IS NULL ;
```

**6.IS NOT :** *"It is used to compare the values with NOT NULL "*.

Syntax: Column_Name <b>IS NOT</b> NULL ;
--

- WAQTD name of the employee who is getting salary .

```
SELECT ENAME
FROM EMP
WHERE SAL IS NOT NULL ;
```

- WAQTD name of the emp who gets commission .

```
SELECT ENAME
FROM EMP
WHERE COMM IS NOT NULL ;
```

- WAQTD name , sal and comm of the emp if the emp doesn't earn commission but gets salary .

```
SELECT ENAME , SAL , COMM
FROM EMP
WHERE COMM IS NULL AND SAL IS NOT NULL ;
```

**7.LIKE :** *"It is used for Pattern Matching "*.

Syntax: Column_Name LIKE 'pattern' ;
--------------------------------------

**To achieve pattern matching we use special characters:**

- Percentile (%) – Takes any no. of characters, any type of characters and also no characters.
- Underscore ( \_ ) – takes only one character but any character.

Examples :

- WAQTD details of an employee whose name is SMITH .

```
SELECT *
```

```
FROM EMP
WHERE ENAME ='SMITH' ;
```

- WAQTD details of the employee who's name starts with 'S' .

```
SELECT *
FROM EMP
WHERE ENAME LIKE 'S%';
```

- WAQTD details of the employee who's name ends with 'S'.

```
SELECT *
FROM EMP
WHERE ENAME LIKE '%S';
```

- WAQTD names of the employees who have character 'S' in theirnames .

```
SELECT *
WHERE ENAME LIKE '%S%';
```

- WAQTD names that starts with 'J' and ends with 'S' .

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE 'J%S';
```

- WAQTD names of the employee if the emp has char 'A' as hissecond character .

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE '_A%';
```

- WAQTD names of the employee if the emp has char 'A' as his Thirdcharacter .

```
SEELCT ENAME
FROM EMP
WHERE ENAME LIKE '__A%';
```

- WAQTD names of the employee if the emp has char 'A' as second character and 'S' is last character .

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE '_AS';
```



- WAQTD names of the employee if the emp has char 'A' present at atleast 2 times .

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE '%A%A%';
```

- WAQTD names of the employee if the emp name starts with 'A' and ends with 'A' .

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE 'A%A';
```

- WAQTD names of the employee if the emp's salary's last 2 digit is 50 rupees .

```
SELECT ENAME
FROM EMP
WHERE SAL LIKE '%50';
```

- WAQTD names of the employees hired in November.

```
SELECT ENAME
WHERE HIREDATE LIKE '%NOV%';
```

**8. NOT LIKE:** Opposite of Like .

Syntax: Column_Name <b>NOT LIKE</b> 'pattern' ;
---

- WAQTD names of the employees who were not hired in November.

```
SELECT ENAME
WHERE HIREDATE LIKE '%NOV%';
```

- WAQTD names of the employee if the emp doesn't have char 'A' as second character and 'S' is last character .

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE '_A%S';
```