

# Video-8

# Topics to cover:

- Order By
- Like Operations

## # ORDER BY Clause

The ORDER BY clause in SQL is used to sort the result set of a query by one or more columns, either in ascending (ASC) or descending (DESC) order.

### **Syntax:**

```
SELECT column1, column2, ...
```

```
FROM table_name
```

```
ORDER BY column1 [ASC|DESC], column2 [ASC|DESC], ...;
```

- ASC (default): Sorts from lowest to highest (A–Z, 0–9).
- DESC: Sorts from highest to lowest (Z–A, 9–0).

## # Creating employee table to understand different Commands

Create table employee (

Id int,

Name varchar(50),

Age int,

Hiring\_date date,

Salary int

);

insert into employee values(1,"Ankit", 24, "2021-08-10", 10000);

insert into employee values(2,"Arya", 25, "2021-08-10", 20000);

insert into employee values(3,"Sunny", 22, "2021-08-11", 11000);

insert into employee values(4,"Amit", 25, "2021-08-11", 12000);

insert into employee values(5,"Mohit", 26, "2021-08-12", 50000);

insert into employee values(6,"Sumit", 26, "2021-08-12", 50000);

Select \* from employee;

## # Examples

### # 1. Sort by one column (ascending)

```
SELECT * FROM employee  
ORDER BY name;
```

### # 2. Sort by one column (descending)

```
SELECT * FROM employee  
ORDER BY name DESC;
```

### # 3. Sort by multiple columns

```
SELECT * FROM employee  
ORDER BY age DESC, name ASC;
```

This sorts first by age (highest first), then by name (A–Z) within each age;

## # Examples

### # 4. You can use column aliases in ORDER BY

```
SELECT *, salary * 12 AS annual_salary  
FROM employee  
ORDER BY annual_salary DESC;
```

### # 5. You can use column positions in ORDER BY

```
SELECT name, age FROM employee  
ORDER BY 2 DESC; -- Sorts by the second column (age)
```

## # LIKE Operations

The **LIKE** operator in SQL is used to search for a specific pattern in a column's text data, often with wildcards like **%** and **\_** to match partial strings.

### Syntax

```
SELECT column1, column2, ...  
FROM table_name  
WHERE column_name LIKE pattern;
```

### LIKE operation

Wildcard	Meaning
%	Matches zero, one, or many characters
_	Matches exactly one character

## # Examples

**# 1. Get all those employees whose name starts with "A".**

```
SELECT * FROM employee  
WHERE name LIKE "A%";
```

**# 2. Get all those employees whose name starts with "SU".**

```
SELECT * FROM employee  
WHERE name LIKE "SU%";
```

**# 3. Get all those employees whose name ends with "T".**

```
SELECT * FROM employee  
WHERE name LIKE "%T";
```

**# 4. Get all those employees whose name starts with "A" and ends with "T".**

```
SELECT * FROM employee  
WHERE name LIKE "A%T";
```



## # Examples

**# 5. Get all those employees whose name will have exact 5 characters.**

```
SELECT * FROM employee  
WHERE name LIKE "_____";
```

**# 6. Return all those employees whose name contains at least 5 characters.**

```
SELECT * FROM employee WHERE name LIKE "%_____";  
SELECT * FROM employee WHERE name LIKE "_____%";  
SELECT * FROM employee WHERE name LIKE "%_____%";
```