1. DISTINCT Keyword

- It is used to ensure that duplicate rows are removed from our resultset
- It returns only the unique rows.

SYNTAX:

```
SELECT DISTINCT column1, column2
FROM table_name;
```

Ex.Finding unique job roles in emp table

```
SELECT DISTINCT JOB
FROM EMP;
```

```
+-----------+
| JOB       |
+-----------+
| CLERK     |
| SALESMAN  |
| MANAGER   |
| ANALYST   |
| PRESIDENT |
+-----------+
```

This query returns all unique jobs from emp table.

Ex. Finding Distinct department numbers

```
SELECT DISTINCT DEPTNO
FROM EMP;
```

```
+--------+
| DEPTNO |
+--------+
|     20 |
|     30 |
|     10 |
+--------+
```

It will return unique department numbers

- Group By Clause
  It is used to group rows that share the same values in specified columns into summary rows.
  It is commonly used with aggregate functions like COUNT(), SUM(), AVG(), MIN() AND MAX().

  SYNTAX:
  SELECT column1, aggregate_function(column2)
  FROM table_name
  GROUP BY column;

  IMPs:
  - Columns in the SELECT clause not involved in aggregate functions must be included in the GROUP BY clause.
  - GROUP BY comes after the WHERE clause and before the ORDER BY clause(if used)

Ex. Total employees but with department

```
SELECT DEPTNO, COUNT(*) as total_employees
From emp
Group by deptno;
```

Ex. Total salary by job

```
SELECT JOB, SUM(SAL) AS TOTAL_SALARY
FROM EMP
GROUP BY JOB;
```

```
+-----------+--------------+
| JOB       | TOTAL_SALARY |
+-----------+--------------+
| CLERK     |      4150.00 |
| SALESMAN  |      5600.00 |
| MANAGER   |      8275.00 |
| ANALYST   |      6000.00 |
| PRESIDENT |      5000.00 |
+-----------+--------------+
```

Aggregate Functions***

Are used to perform calculations on set of values, It returns a summarized result.

Function	Description
COUNT()	Counts the number of rows

SUM()	Returns the total sum of numeric values in a column
AVG()	Calculate the average of numeric values
MIN()	Finds the smallest value in a column
MAX()	Finds the largest value in a column

1. COUNT()

- Counts the number of rows in a specified column

Ex. To count all employees

```
SELECT COUNT(*) AS total_employees
FROM emp;
```

```
+-----------------+
| total_employees |
+-----------------+
|              14 |
+-----------------+
```

This query counts all rows in emp table.

Ex. Count employees in each department.

```
SELECT DEPTNO, COUNT(*) as total_employees
From emp
Group by deptno;
```

```
+--------+-----------------+
| DEPTNO | total_employees |
+--------+-----------------+
|     20 |               5 |
|     30 |               6 |
|     10 |               3 |
+--------+-----------------+
```

This query groups employees by their department and counts how many employees are in each department.

Ex. Count distinct job roles.

```
mysql> select count(DISTINCT JOB) AS UNIQUE_JOBS FROM EMP;
+-------------+
| UNIQUE_JOBS |
+-------------+
|           5 |
+-------------+
1 row in set (0.00 sec)
```

Counts how many unique jobs exist in the emp table

```
mysql> select count(JOB) AS UNIQUE_JOBS FROM EMP;
+-------------+
| UNIQUE_JOBS |
+-------------+
```

```
|         14 |
+------------+
1 row in set (0.00 sec)
```

2. SUM()

- It calculates the total sum of numeric values in a column.

  Ex. To get the total salary of employees

```
SELECT SUM(sal) AS total_Salary
From emp;
```

```
+--------------+
| total_Salary |
+--------------+
|     29025.00 |
+--------------+
1 row in set (0.00 sec)
```

Total salary by department:

```
SELECT DEPTNO,SUM(sal) AS total_Salary
From emp
Group by deptno;
```

```
+--------+--------------+
| DEPTNO | total_Salary |
+--------+--------------+
|     20 |     10875.00 |
|     30 |      9400.00 |
|     10 |      8750.00 |
+--------+--------------+
```

Groups employees by dept and calculates the total salary paid in each dept.

3. AVG()

- To calculate the average of numeric values in a column.

Ex. To get Average salary of employees

```
SELECT AVG(SAL) AS AVG_SALARY
FROM EMP;
```

```
+-------------+
| AVG_SALARY  |
+-------------+
| 2073.214286 |
```

Ex. Average salary by job

```
Select job, avg(sal) as avg_salary
From emp
Group by job;
```

4. MIN() & MAX()

Finds the smallest value in a column
Finds the largest value in a column

Ex. To get the minimum and maximum salary.

```
SELECT MIN(SAL) AS MIN_SALARY, MAX(SAL) AS
MAX_SALARY
FROM EMP;
```

MIN_SALARY	MAX_SALARY
800.00	5000.00

Ex. To get minimum and maximum salary by department.

```
Select deptno, min(sal) as min_salary, max(sal) as max_salary
From emp
Group by deptno;
```

deptno	min_salary	max_salary
20	800.00	3000.00
30	950.00	2850.00
10	1300.00	5000.00