Aggregate functions are used to **perform calculations on multiple rows of a table** and return a single value.

**1. COUNT() – Count rows or non-null values**

**Purpose:**

* Counts the number of rows or non-null values in a column.

**Syntax:**

COUNT(\*) -- Counts all rows

COUNT(column) -- Counts non-null values in a column

**Examples:**

**a) Count all rows**

SELECT COUNT(\*) AS TotalEmployees

FROM Employees;

*Output:* Total number of employees in the table.

**b) Count non-null values**

SELECT COUNT(Email) AS EmployeesWithEmail

FROM Employees;

*Output:* Counts only rows where Email is NOT NULL.

**c) Count distinct values**

SELECT COUNT(DISTINCT DepartmentID) AS TotalDepartments

FROM Employees;

*Output:* Counts unique departments.

**2. SUM() – Sum of numeric values**

**Purpose:**

* Adds all values in a numeric column.

**Syntax:**

SUM(column)

**Examples:**

**a) Total salary of all employees**

SELECT SUM(Salary) AS TotalSalary

FROM Employees;

**b) Total salary by department**

SELECT DepartmentID, SUM(Salary) AS DeptSalary

FROM Employees

GROUP BY DepartmentID;

**3. AVG() – Average of numeric values**

**Purpose:**

* Calculates the mean of numeric values.

**Syntax:**

AVG(column)

**Examples:**

**a) Average salary of all employees**

SELECT AVG(Salary) AS AvgSalary

FROM Employees;

**b) Average salary by department**

SELECT DepartmentID, AVG(Salary) AS AvgDeptSalary

FROM Employees

GROUP BY DepartmentID;

**4. MIN() – Minimum value**

**Purpose:**

* Returns the smallest value in a column.

**Syntax:**

MIN(column)

**Examples:**

**a) Minimum salary**

SELECT MIN(Salary) AS MinSalary

FROM Employees;

**b) Minimum salary by department**

SELECT DepartmentID, MIN(Salary) AS MinDeptSalary

FROM Employees

GROUP BY DepartmentID;

**5. MAX() – Maximum value**

**Purpose:**

* Returns the largest value in a column.

**Syntax:**

MAX(column)

**Examples:**

**a) Maximum salary**

SELECT MAX(Salary) AS MaxSalary

FROM Employees;

**b) Maximum salary by department**

SELECT DepartmentID, MAX(Salary) AS MaxDeptSalary

FROM Employees

GROUP BY DepartmentID;

**6. GROUP BY – Aggregate by groups**

**Purpose:**

* Groups rows based on one or more columns before applying aggregate functions.

**Example: Count employees in each department**

SELECT DepartmentID, COUNT(\*) AS EmployeeCount

FROM Employees

GROUP BY DepartmentID;

**Example: Average salary by department**

SELECT DepartmentID, AVG(Salary) AS AvgSalary

FROM Employees

GROUP BY DepartmentID;

**7. HAVING – Filter aggregated results**

* WHERE cannot be used with aggregate functions.
* HAVING is used to filter after aggregation.

**Example: Departments with more than 5 employees**

SELECT DepartmentID, COUNT(\*) AS EmployeeCount

FROM Employees

GROUP BY DepartmentID

HAVING COUNT(\*) > 5;

**Example: Departments with average salary > 50000**

SELECT DepartmentID, AVG(Salary) AS AvgSalary

FROM Employees

GROUP BY DepartmentID

HAVING AVG(Salary) > 50000;

**8. Multiple Aggregates in one query**

SELECT DepartmentID,

COUNT(\*) AS TotalEmployees,

SUM(Salary) AS TotalSalary,

AVG(Salary) AS AvgSalary,

MIN(Salary) AS MinSalary,

MAX(Salary) AS MaxSalary

FROM Employees

GROUP BY DepartmentID;

✅ This gives a **complete summary of each department**.

**Key Points to Remember**

1. Aggregate functions **ignore NULL values** (except COUNT(\*)).
2. Use GROUP BY to aggregate by categories.
3. Use HAVING to filter aggregated data.
4. You can use multiple aggregate functions together.