## SQL LIKE Operator

The **LIKE** operator is used in a WHERE clause to search for a specified pattern in a column.

There are two **wildcards** often used in conjunction with the **LIKE** operator:

* The percent sign **%** represents zero, one, or multiple characters
* The underscore sign**\_** represents one, single character

**Syntax**

SELECT column1, column2, ...  
FROM table\_name  
WHERE columnN LIKE pattern;

## SQL Aggregate Functions

An aggregate function is a function that performs a calculation on a set of values, and returns a single value.

The most commonly used SQL aggregate functions are:

* **MIN()** - returns the smallest value within the selected column
* **MAX()** - returns the largest value within the selected column
* **COUNT**() - returns the number of rows in a set
* **SUM()** - returns the total sum of a numerical column
* **AVG()** - returns the average value of a numerical column

## SQL MIN() and MAX() Functions

SELECT MIN(column\_name)  
FROM table\_name  
WHERE condition;

SELECT MAX(column\_name)  
FROM table\_name  
WHERE condition;

SELECT MAX(GPA) AS HighestGPA  
FROM Student;

## SQL COUNT() Function

## Syntax

SELECT COUNT(column\_name)  
FROM table\_name  
WHERE condition;

## SQL SUM() Function

## Syntax

SELECT SUM(column\_name)  
FROM table\_name  
WHERE condition;

## SQL AVG() Function

## Syntax

SELECT AVG(column\_name)  
FROM table\_name  
WHERE condition;

## SQL GROUP BY Statement

The **GROUP BY** statement groups rows that have the same values into summary rows, like "find the number of customers in each country".

The **GROUP BY** statement is often used with aggregate functions (**COUNT(), MAX(), MIN(), SUM(), AVG())** to group the result-set by one or more columns.

### **GROUP BY Syntax**

SELECT column\_name(s)  
FROM table\_name  
WHERE condition  
GROUP BY column\_name(s)ORDER BY column\_name(s);

## SQL HAVING Clause

### **HAVING Syntax**

SELECT column\_name(s)  
FROM table\_name  
WHERE condition  
GROUP BY column\_name(s)HAVING conditionORDER BY column\_name(s);