# Week 8

## 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*;

Here are some examples showing different LIKE operators with '%' and '_'.

| LIKE Operator | Description |
|---|---|
| WHERE FirstName LIKE 'a%' | Finds any values that start with "a" |
| WHERE FirstName LIKE '%a' | Finds any values that end with "a" |
| WHERE FirstName LIKE '%or%' | Finds any values that have "or" in any position |
| WHERE FirstName LIKE '_r%' | Finds any values that have "r" in the second position |
| WHERE FirstName LIKE 'a_%' | Finds any values that start with "a" and are at least 2 characters in length |
| WHERE FirstName LIKE '_r%' | Finds any values that have "r" in the second position |
| WHERE FirstName LIKE 'a_%' | Finds any values that start with "a" and are at least 2 characters in length |
| WHERE FirstName LIKE 'a__%' | Finds any values that start with "a" and are at least 3 characters in length |
| WHERE FirstName LIKE 'a%o' | Finds any values that start with "a" and ends with "o" |

**Sample Table:**

| Id | FirstName | LastName | Email | City | GPA |
|----|-----------|----------|-------|------|-----|
| 1 | Saad | Iqbal | saad@gmail.com | Lahore | 3.2 |
| 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |
| 3 | Faraz | Shah | faraz@gmail.com | Lahore | 2.8 |
| 4 | Waseem | Khan | waseem@gmail.com | Karachi | 2.5 |
| 5 | Nimra | Zahid | nimra@gmail.com | Islamabad | 3.8 |
| 6 | Jasim | Khan | jasim@gmail.com | Lahore | 3.5 |

**Examples:**

1. The following SQL statement selects all students with a First Name starting with "f":

Showing rows 0 - 1 (2 total, Query took 0.0015 seconds.)

SELECT * FROM student WHERE FirstName LIKE 'f%'

☐ Show all | Number of rows: 25 ✓   Filter rows: Search this table   Sort by key:

+ Options

| ←T→ | | Id | FirstName | LastName | Email | City | GPA |
|-----|---|----|-----------|----------|-------|------|-----|
| ☐ 🖉 Edit ⬩ Copy ⊖ Delete | | 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |
| ☐ 🖉 Edit ⬩ Copy ⊖ Delete | | 3 | Faraz | Shah | faraz@gmail.com | Lahore | 2.8 |

2. The following SQL statement selects all students with a First Name ending with "d":

Showing rows 0 - 1 (2 total, Query took 0.0015 seconds.)

SELECT * FROM student WHERE FirstName LIKE '%d'

☐ Show all | Number of rows: 25 ✓   Filter rows: Search this table   Sort by key:

+ Options

| ←T→ | | Id | FirstName | LastName | Email | City | GPA |
|-----|---|----|-----------|----------|-------|------|-----|
| ☐ 🖉 Edit ⬩ Copy ⊖ Delete | | 1 | Saad | Iqbal | saad@gmail.com | Lahore | 3.2 |
| ☐ 🖉 Edit ⬩ Copy ⊖ Delete | | 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |

3. The following SQL statement selects all students with a First Name that have "i" in any position:

Showing rows 0 - 1 (2 total, Query took 0.0017 seconds.)

```
SELECT * FROM student WHERE FirstName LIKE '%i%'
```

☐ Show all | Number of rows: 25 ✔   Filter rows: Search this table   Sort by key:

+ Options

| | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 5 | Nimra | Zahid | nimra@gmail.com | Islamabad | 3.8 |
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 6 | Jasim | Khan | jasim@gmail.com | Lahore | 3.5 |

4. The following SQL statement selects all students with a First Name that have "a" in the second position:

Showing rows 0 - 4 (5 total, Query took 0.0017 seconds.)

```
SELECT * FROM student WHERE FirstName LIKE '_a%'
```

☐ Show all | Number of rows: 25 ✔   Filter rows: Search this table   Sort by key:

+ Options

| | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 1 | Saad | Iqbal | saad@gmail.com | Lahore | 3.2 |
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 3 | Faraz | Shah | faraz@gmail.com | Lahore | 2.8 |
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 4 | Waseem | Khan | waseem@gmail.com | Karachi | 2.5 |
| ☐ 🖉 Edit 📋 Copy ⊖ Delete | 6 | Jasim | Khan | jasim@gmail.com | Lahore | 3.5 |

5. The following SQL statement selects all students with a First Name that starts with "f" and are at least 3 characters in length:

Showing rows 0 - 1 (2 total, Query took 0.0016 seconds.)

`SELECT * FROM student WHERE FirstName LIKE 'f__%'`

Show all | Number of rows: 25 ∨    Filter rows: Search this table    Sort by key:

+ Options

| | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|
| Edit Copy Delete | 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |
| Edit Copy Delete | 3 | Faraz | Shah | faraz@gmail.com | Lahore | 2.8 |

6. The following SQL statement selects all students with a Firs Name that starts with "n" and ends with "a":

Showing rows 0 - 0 (1 total, Query took 0.0032 seconds.)

`SELECT * FROM student WHERE FirstName LIKE 'n%a'`

Show all | Number of rows: 25 ∨    Filter rows: Search this table

+ Options

| | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|
| Edit Copy Delete | 5 | Nimra | Zahid | nimra@gmail.com | Islamabad | 3.8 |

Check all    With selected: Edit Copy Delete Export

7. The following SQL statement selects all students with a First Name that does NOT start with "f":



Showing rows 0 - 3 (4 total, Query took 0.0018 seconds.)

SELECT * FROM student WHERE FirstName NOT LIKE 'f%'

| Id | FirstName | LastName | Email | City | GPA |
|----|-----------|----------|-------|------|-----|
| 1 | Saad | Iqbal | saad@gmail.com | Lahore | 3.2 |
| 4 | Waseem | Khan | waseem@gmail.com | Karachi | 2.5 |
| 5 | Nimra | Zahid | nimra@gmail.com | Islamabad | 3.8 |
| 6 | Jasim | Khan | jasim@gmail.com | Lahore | 3.5 |

# SQL IN Operator

The IN operator allows you to specify multiple values in a WHERE clause.

The IN operator is a shorthand for multiple OR conditions.

**Syntax 1:**

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (value1, value2, ...);
```

**Syntax 2:**

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (SELECT STATEMENT);
```

**Examples:**

1.  The following SQL statement selects all students that are located in "Lahore" or "Karachi":

✔ Showing rows 0 - 3 (4 total, Query took 0.0178 seconds.)

```sql
SELECT * FROM student WHERE City IN ('Lahore', 'Karachi')
```

☐ Show all | Number of rows: 25 ✔   Filter rows: Search this table   Sort by key:

+ Options

| | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|
| ☐ 🖉 Edit ⬚ Copy ⊖ Delete | 1 | Saad | Iqbal | saad@gmail.com | Lahore | 3.2 |
| ☐ 🖉 Edit ⬚ Copy ⊖ Delete | 3 | Faraz | Shah | faraz@gmail.com | Lahore | 2.8 |
| ☐ 🖉 Edit ⬚ Copy ⊖ Delete | 4 | Waseem | Khan | waseem@gmail.com | Karachi | 2.5 |
| ☐ 🖉 Edit ⬚ Copy ⊖ Delete | 6 | Jasim | Khan | jasim@gmail.com | Lahore | 3.5 |

2.  The following SQL statement selects all students that are NOT located in "Lahore" or "Karachi":

✔ Showing rows 0 - 1 (2 total, Query took 0.0181 seconds.)

```sql
SELECT * FROM student WHERE City NOT IN ('Lahore', 'Karachi')
```

☐ Show all | Number of rows: 25 ✔   Filter rows: Search this table   Sort by key:

+ Options

| | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|
| ☐ 🖉 Edit ⬚ Copy ⊖ Delete | 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |
| ☐ 🖉 Edit ⬚ Copy ⊖ Delete | 5 | Nimra | Zahid | nimra@gmail.com | Islamabad | 3.8 |

3. The following SQL statement selects all students that are from the same countries as the teachers:

Showing rows 0 - 4 (5 total, Query took 0.0472 seconds.)

`SELECT * FROM student WHERE City IN (SELECT City FROM teacher)`

Show all | Number of rows: 25 ✓  Filter rows: Search this table  Sort by key:

+ Options

| | | | | Id | FirstName | LastName | Email | City | GPA |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | Edit | Copy | Delete | 1 | Saad | Iqbal | saad@gmail.com | Lahore | 3.2 |
| ☐ | Edit | Copy | Delete | 2 | Fahad | Khan | fahad@gmail.com | Peshawar | 3.1 |
| ☐ | Edit | Copy | Delete | 3 | Faraz | Shah | faraz@gmail.com | Lahore | 2.8 |
| ☐ | Edit | Copy | Delete | 4 | Waseem | Khan | waseem@gmail.com | Karachi | 2.5 |
| ☐ | Edit | Copy | Delete | 6 | Jasim | Khan | jasim@gmail.com | Lahore | 3.5 |

# SQL Alter Table Statement

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.

The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

## ALTER TABLE - ADD Column

To add a column in a table, use the following syntax:

```
ALTER TABLE table_name
ADD column_name datatype;
```

The following SQL adds "Pay" column to the "Teacher" table:



✔ MySQL returned an empty result set (i.e. zero rows). (Query took 0.1454 seconds.)

ALTER TABLE teacher ADD Pay varchar(255)

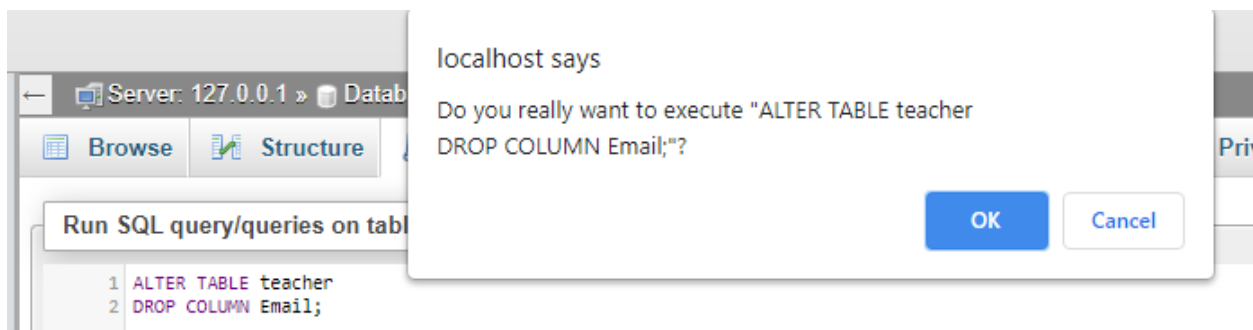| Id | Name | Email | City | Pay |
|----|------|-------|------|-----|
| 1 | Faheem | faheem@gmail.com | Lahore | NULL |
| 2 | Adnan | adnan@gmail.com | Peshawar | NULL |
| 3 | Arshad | arshad@gmail.com | Karachi | NULL |

*Note: You can add values to "Pay" using SQL update statement.*

## ALTER TABLE - DROP COLUMN

To delete a column in a table, use the following syntax (notice that some database systems don't allow deleting a column):

```
ALTER TABLE table_name
DROP COLUMN column_name;
```

The following SQL deletes the "Email" column from the "teacher" table:



localhost says

Do you really want to execute "ALTER TABLE teacher DROP COLUMN Email;"?

OK    Cancel

Server: 127.0.0.1 » Datab

Browse    Structure    Priv

Run SQL query/queries on tabl

```
1  ALTER TABLE teacher
2  DROP COLUMN Email;
```

✔ MySQL returned an empty result set

ALTER TABLE teacher DROP COLUMN Email

| Id | Name | City | Pay |
|----|------|------|-----|
| 1 | Faheem | Lahore | NULL |
| 2 | Adnan | Peshawar | NULL |
| 3 | Arshad | Karachi | NULL |

## ALTER TABLE - ALTER/MODIFY COLUMN

To change the data type of a column in a table, use the following syntax:

ALTER TABLE *table_name*
ALTER COLUMN *column_name datatype*;

Now we want to change the data type of the column named "GPA" in the "Student" table.

We use the following SQL statement:

✔ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0475 seconds.)

ALTER TABLE student MODIFY COLUMN GPA varchar(255)

# THE END