# LIKE Wildcards

*by Sophia*

WHAT'S COVERED

In this lesson, you will explore using the LIKE operator with more complex options through wildcards. This lesson will be addressed in two parts. Specifically, this lesson will cover:

1. Combining Wildcards
2. Complex Comparisons

## 1. Combining Wildcards

We can add more complexity to the LIKE operator with precise criteria. For example, if we explore the data in the customer table, you may see various phone numbers in different formats. If we want to find the phone numbers that have the format +1 (###) ###-####, we can use the underscore wildcard (_) to represent each digit:

```
SELECT *
FROM customer
WHERE phone like '+1 (___) ___-____';
```

Query Results

Row count: 21

| customer_id | first_name | last_name | company | address | city | state | country | postal_code | phone |
|---|---|---|---|---|---|---|---|---|---|
| 3 | François | Tremblay | | 1498 rue Bélanger | Montréal | QC | Canada | H2G 1A7 | +1 (514) 721-4711 |
| 14 | Mark | Philips | Telus | 8210 111 ST NW | Edmonton | AB | Canada | T6G 2C7 | +1 (780) 434-4554 |
| 15 | Jennifer | Peterson | Rogers Canada | 700 W Pender Street | Vancouver | BC | Canada | V6C 1G8 | +1 (604) 688-2255 |
| 16 | Frank | Harris | Google Inc. | 1600 Amphitheatre Parkway | Mountain View | CA | USA | 94043-1351 | +1 (650) 253-0000 |
| 17 | Jack | Smith | Microsoft Corporation | 1 Microsoft Way | Redmond | WA | USA | 98052-8300 | +1 (425) 882-8080 |
| 18 | Michelle | Brooks | | 627 Broadway | New York | NY | USA | 10012-2612 | +1 (212) 221-3546 |
| 19 | Tim | Goyer | Apple Inc. | 1 Infinite Loop | Cupertino | CA | USA | 95014 | +1 (408) 996-1010 |
| 20 | Dan | Miller | | 541 Del Medio Avenue | Mountain View | CA | USA | 94040-111 | +1 (650) 644-3358 |
| 21 | Kathy | Chase | | 801 W 4th Street | Reno | NV | USA | 89503 | +1 (775) 223-7665 |
| 22 | Heather | Leacock | | 120 S Orange Ave | Orlando | FL | USA | 32801 | +1 (407) 999-7788 |
| 23 | John | Gordon | | 69 Salem Street | Boston | MA | USA | 2113 | +1 (617) 522-1333 |
| 24 | Frank | Ralston | | 162 E Superior Street | Chicago | IL | USA | 60611 | +1 (312) 332-3232 |
| 25 | Victor | Stevens | | 319 N. Frances Street | Madison | WI | USA | 53703 | +1 (608) 257-0597 |
| 26 | Richard | Cunningham | | 2211 W Berry Street | Fort Worth | TX | USA | 76110 | +1 (817) 924-7272 |
| 27 | Patrick | Gray | | 1033 N Park Ave | Tucson | AZ | USA | 85719 | +1 (520) 622-4200 |
| 28 | Julia | Barnett | | 302 S 700 E | Salt Lake City | UT | USA | 84102 | +1 (801) 531-7272 |
| 29 | Robert | Brown | | 796 Dundas Street West | Toronto | ON | Canada | M6J 1V1 | +1 (416) 363-8888 |
| 30 | Edward | Francis | | 230 Elgin Street | Ottawa | ON | Canada | K2P 1L7 | +1 (613) 234-3322 |
| 31 | Martha | Silk | | 194A Chain Lake Drive | Halifax | NS | Canada | B3S 1C5 | +1 (902) 450-0450 |
| 32 | Aaron | Mitchell | | 696 Osborne Street | Winnipeg | MB | Canada | R3L 2B9 | +1 (204) 452-6452 |
| 33 | Ellie | Sullivan | | 5112 48 Street | Yellowknife | NT | Canada | X1A 1N6 | +1 (867) 920-2233 |

We could also use this to find email addresses that have a domain name extension with exactly two characters:

```
SELECT *
FROM customer
WHERE email like '%.__';
```

The “%.__” would look for any number of characters, followed by the dot and ending with two characters. We can use wildcards to find values of a certain length with a specific format. For example, if we wanted to find customers with the last_name starting with the letter S and having four characters, we can do the following with an S and three underscore characters:

```
SELECT *
FROM customer
WHERE last_name like 'S___';
```

Query Results

Row count: 1

| customer_id | first_name | last_name | company | address | city | state | country |
|---|---|---|---|---|---|---|---|
| 31 | Martha | Silk | | 194A Chain Lake Drive | Halifax | NS | Canada |

If we were to expand this to find the customers starting with the letter S and having at least four characters (and maybe more), we could add the % to the end of the clause:

```
SELECT *
FROM customer
WHERE last_name like 'S___%';
```

# 2. Complex Comparisons

You can also combine both types of wildcards to find specific content. For example, we can split up a marketing campaign to get all the email addresses that start with the letter m and have a domain starting with the letter a. One might think you could use the following query:

```
SELECT *
FROM customer
WHERE email like 'm%a%';
```

However, this would return the following result that simply looks for the letter m, then zero or more characters later, it looks for the letter a, and then can end with zero or more characters.

**Query Results**

Row count: 1

| customer_id | first_name | last_name | company | address | city | state | country |
|---|---|---|---|---|---|---|---|
| 31 | Martha | Silk | | 194A Chain Lake Drive | Halifax | NS | Canada |

This would return values that begin with the letter m and then include the letter a at any other point.

With an email, you may want to have specific character checks with the @ and the dot (.) that will change the search criteria. We must include these characters as part of the comparison in a particular order as defined by our problem.

```
SELECT *
FROM customer
WHERE email like 'm%@a%.%';
```

In the parameter, we are looking for the letter m, then any number of characters before the @ sign followed by the letter a. Then any number of characters before the dot (.) followed by any number of characters.

**Query Results**

Row count: 7

| customer_id | first_name | last_name | company | address | city | state | country | postal_code | phone | fax | email |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 14 | Mark | Philips | Telus | 8210 111 ST NW | Edmonton | AB | Canada | T6G 2C7 | +1 (780) 434-4554 | +1 (780) 434-5565 | mphilips12@shaw.ca |
| 18 | Michelle | Brooks | | 627 Broadway | New York | NY | USA | 10012-2612 | +1 (212) 221-3546 | +1 (212) 221-4679 | michelleb@aol.com |
| 31 | Martha | Silk | | 194A Chain Lake Drive | Halifax | NS | Canada | B3S 1C5 | +1 (902) 450-0450 | | marthasilk@gmail.com |
| 35 | Madalena | Sampaio | | Rua dos Campeões Europeus de Viena, 4350 | Porto | | Portugal | | +351 (225) 022-448 | | masampaio@sapo.pt |
| 41 | Marc | Dubois | | 11, Place Bellecour | Lyon | | France | 69002 | +33 04 78 30 30 30 | | marc.dubois@hotmail.com |
| 55 | Mark | Taylor | | 421 Bourke Street | Sidney | NSW | Australia | 2010 | +61 (02) 9332 3633 | | mark.taylor@yahoo.au |
| 58 | Manoj | Pareek | | 12,Community Centre | Delhi | | India | 110017 | +91 0124 39883988 | | manoj.pareek@rediff.com |

WATCH

TRY IT

Your turn! Open the SQL tool by clicking on the LAUNCH DATABASE button below. Then, enter in one of the examples above and see how it works. Next, try your own choices for which columns you want the query to provide.

## SUMMARY

In this lesson, you learned how to **combine wildcards** to create complex criteria for a LIKE operator. You saw several examples of **complex comparisons**, combining percent (%) and underscore (_) wildcard symbols with fixed values such as characters and symbols to precisely identify the criteria that records must meet in order to be included in the results set.

Source: THIS TUTORIAL WAS AUTHORED BY DR. VINCENT TRAN, PHD (2020) AND Faithe Wempen (2024) FOR SOPHIA LEARNING. PLEASE SEE OUR **TERMS OF USE**.