Multiple Filters

by Sophia

WHAT'S COVERED

In this lesson, you will explore using AND and OR in the WHERE clause of SELECT statements to combine the filtering of conditions, in three parts. Specifically, this lesson will cover:

1. Using AND
2. Using OR
3. Combining Both

1. Using AND

The AND operator displays a record if all of the conditions separated by AND are true. You can think of this approach of using the AND operator as if you were searching for a car to rent. You may be looking for a four-door, with the color blue, having leather seats, and so forth. This kind of request (SQL SELECT) would be a great example of using the AND to ensure we consider all of the criteria.

If we wanted to search for customers from Manhattan in New York City, which is in the USA and has the area code 212, we would first need to identify the columns that have the data for us to do a search. The two columns we need are Country and Phone. The country column would contain 'USA' while the phone column would have the area code for the phone number. However, it is essential to note what comparison would be needed. Since the country is a specific value, we can use the equal sign to compare. For the area code, we would need to use LIKE with wildcards because there are more digits in the phone column than just the area code.

```
SELECT *
FROM customer
WHERE country ='USA'
AND phone LIKE '%(212)%';
```

Query Results

Row count: 1

customer_id	first_name	last_name	company	address	city	state	country	postal_code	phone	fax	email	support_rep_id
18	Michelle	Brooks		627 Broadway	New York	NY	USA	10012-2612	+1 (212) 221-3546	+1 (212) 221-4679	michelleb@aol.com	3

If we changed the criteria to finding American customers who spoke with a particular support representative, we would need to query the column for support_rep_id because that will have the information that we are looking for. We can query records that have the support_rep_id equal to 3 and the country USA. We can change the query as follows:

```
SELECT *
FROM customer
WHERE country ='USA'
AND support_rep_id = 3;
```

This would return a result set as follows:

Query Results

Row count: 3

customer_id	first_name	last_name	company	address	city	state	country	postal_code	phone	fax	email	support_rep_id
18	Michelle	Brooks		627 Broadway	New York	NY	USA	10012-2612	+1 (212) 221-3546	+1 (212) 221-4679	michelleb@aol.com	3
19	Tim	Goyer	Apple Inc.	1 Infinite Loop	Cupertino	CA	USA	95014	+1 (408) 996-1010	+1 (408) 996-1011	tgoyer@apple.com	3
24	Frank	Ralston		162 E Superior Street	Chicago	IL	USA	60611	+1 (312) 332-3232		fralston@gmail.com	3

We can have as many criteria as we want. If we extended the prior criteria to include those with the city starting with the letter C, we would adjust the query like this:

```
SELECT *
FROM customer
WHERE country ='USA'
AND support_rep_id = 3
AND city LIKE 'C%';
```

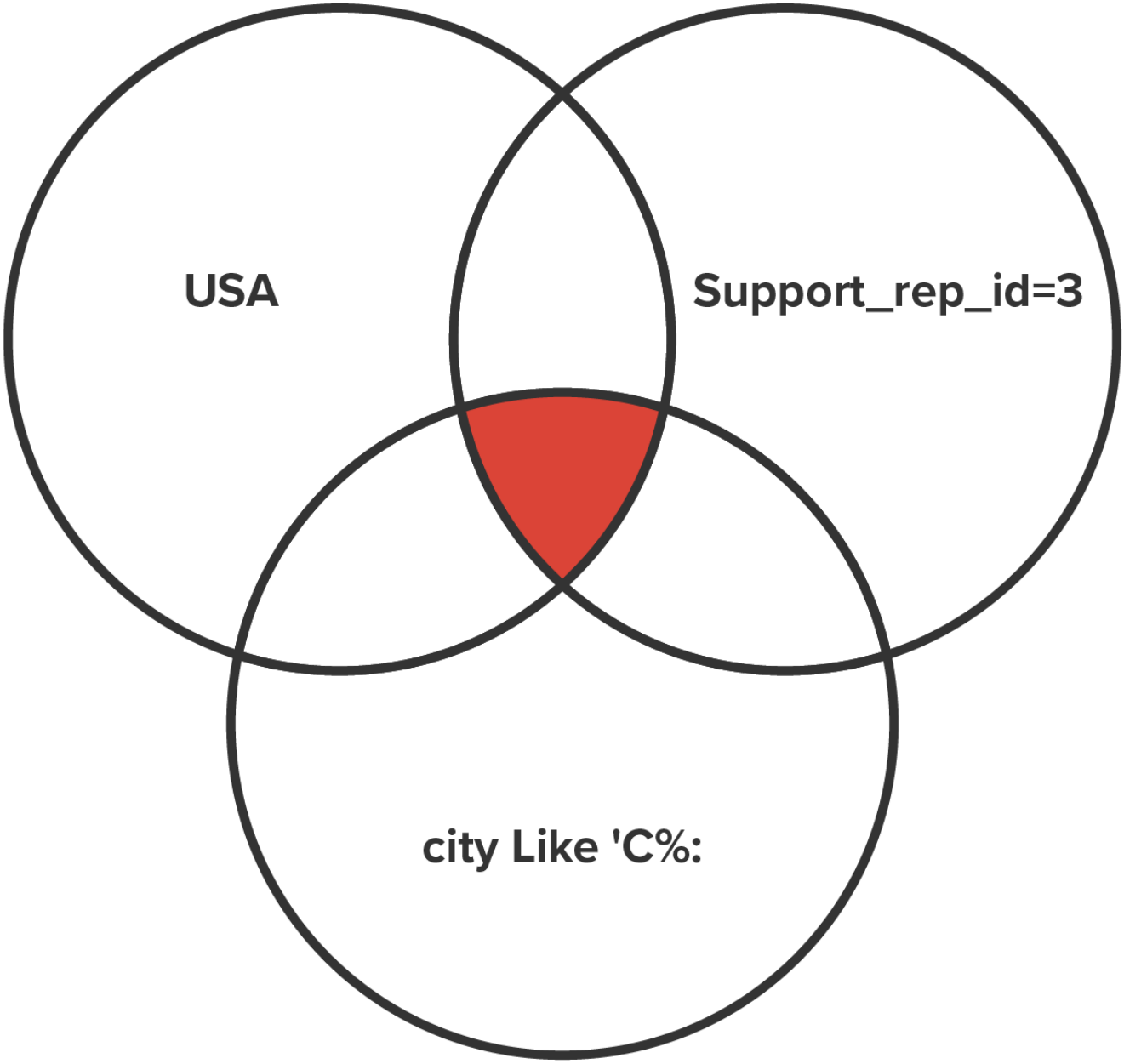
This would return the following result set:

Query Results

Row count: 2

customer_id	first_name	last_name	company	address	city	state	country	postal_code	phone	fax	email	support_rep_id
19	Tim	Goyer	Apple Inc.	1 Infinite Loop	Cupertino	CA	USA	95014	+1 (408) 996-1010	+1 (408) 996-1011	tgoyer@apple.com	3
24	Frank	Ralston		162 E Superior Street	Chicago	IL	USA	60611	+1 (312) 332-3232		fralston@gmail.com	3

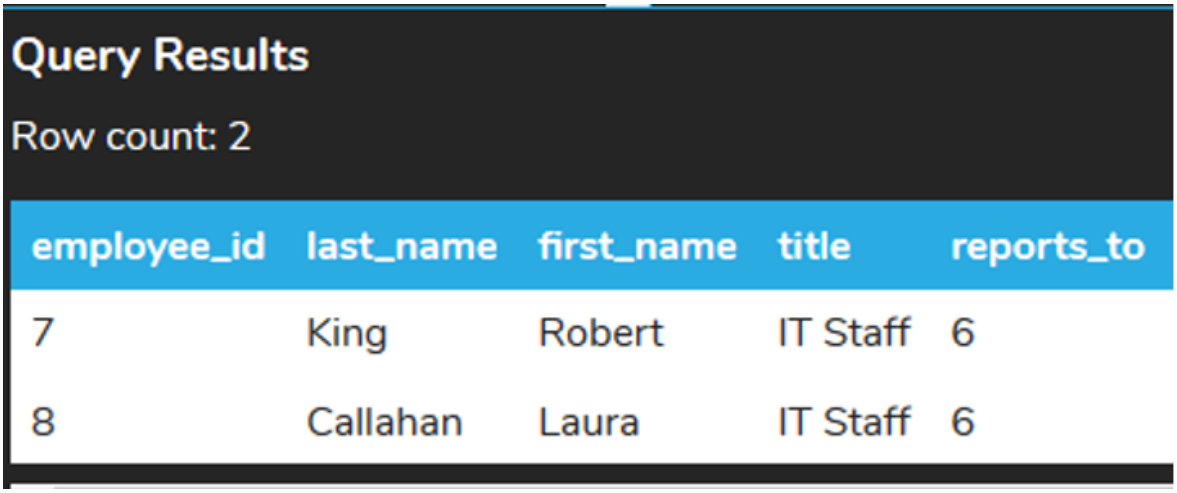
In essence, the query would first take the customers and retrieve those in the USA. In that subset, it would find those that had the support_rep_id equal to 3. Next, it would retrieve those that had the city that started with the letter C. You can think of this as the intersection of all of the criteria being returned.



2. Using OR

The OR operator displays a record if any of the conditions separated by OR are true. The result would be returned as long as one of the conditions is met. If we wanted to get employees that had the title of either IT Staff or IT Manager, we could run it as two separate statements. First, we would query for IT Staff, as shown below, and then we would query for IT Manager. Because we are looking for text strings, it is important to put everything in quotes. That way, it will look for an exact match.

```
SELECT *
FROM employee
WHERE title = 'IT Staff';
```



Query Results

Row count: 2

employee_id	last_name	first_name	title	reports_to
7	King	Robert	IT Staff	6
8	Callahan	Laura	IT Staff	6

```
SELECT *
FROM employee
WHERE title = 'IT Manager';
```

Query Results

Row count: 1

employee_id	last_name	first_name	title	reports_to
6	Mitchell	Michael	IT Manager	1

Using the OR operator, we have the ability to combine the sets together like this:

```
SELECT *
FROM employee
WHERE title = 'IT Manager'
OR title = 'IT Staff'
```

Query Results

Row count: 3

employee_id	last_name	first_name	title	reports_to
6	Mitchell	Michael	IT Manager	1
7	King	Robert	IT Staff	6
8	Callahan	Laura	IT Staff	6

The OR operator behaves differently from the AND operator. With the AND operator, the more times we use it, the smaller (or the same) the result set becomes. Using the OR operator, the more times we use it, the larger (or the same) the result set becomes.

Here's another example. In this case, the query references two different columns. If we look at employees that either have the title IT Staff or report to the supervisor whose ID is 6, the query would look like this:

```
SELECT *
FROM employee
WHERE title = 'IT Staff'
OR reports_to = 6;
```

Query Results

Row count: 2

employee_id	last_name	first_name	title	reports_to
7	King	Robert	IT Staff	6
8	Callahan	Laura	IT Staff	6

However, running each independently would return the same rows because it just so happens that all the people with an IT Staff job title report to the same supervisor.

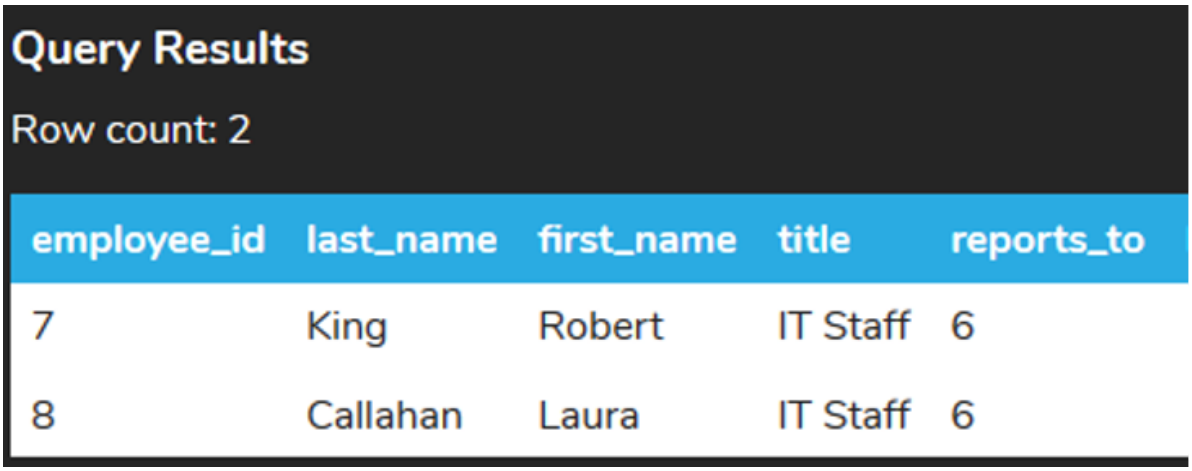
```
SELECT *
FROM employee
WHERE title = 'IT Staff';
```

Query Results

Row count: 2

employee_id	last_name	first_name	title	reports_to
7	King	Robert	IT Staff	6
8	Callahan	Laura	IT Staff	6

```
SELECT *
FROM employee
WHERE reports_to = 6;
```



3. Combining Both

We can also combine AND and OR operations in a single query. It's important to be aware of the order in which you place the operators because you will get different results. In most cases, the AND operators are performed first, and then the OR statements are executed. Let's look at the difference by comparing the results when the order is switched. In the following query:

```
SELECT *
FROM employee
WHERE title = 'IT Staff'
OR reports_to = 6
AND phone like '%1';
```

Query Results

Row count: 2

employee_id	last_name	first_name	title	reports_to	birth_date	hire_date	address	city	state	country	postal_code	phone
7	King	Robert	IT Staff	6	1970-05-29T00:00:00.000Z	2004-01-02T00:00:00.000Z	590 Columbia Boulevard West	Lethbridge	AB	Canada	T1K 5N8	+1 (403) 456-9986
8	Callahan	Laura	IT Staff	6	1968-01-09T00:00:00.000Z	2004-03-04T00:00:00.000Z	923 7 ST NW	Lethbridge	AB	Canada	T1H 1Y8	+1 (403) 467-3351

We would first get the result set of the AND statement as if it were:

```
SELECT *
FROM employee
WHERE reports_to = 6
AND phone like '%1';
```

Query Results

Row count: 1

employee_id	last_name	first_name	title	reports_to	birth_date	hire_date	address	city	state	country	postal_code	phone
8	Callahan	Laura	IT Staff	6	1968-01-09T00:00:00.000Z	2004-03-04T00:00:00.000Z	923 7 ST NW	Lethbridge	AB	Canada	T1H 1Y8	+1 (403) 467-3351

Then we would combine it with:

```
SELECT *
FROM employee
WHERE title = 'IT Staff';
```

Query Results

Row count: 2

employee_id	last_name	first_name	title	reports_to	birth_date	hire_date	address	city	state	country	postal_code	phone
7	King	Robert	IT Staff	6	1970-05-29T00:00:00.000Z	2004-01-02T00:00:00.000Z	590 Columbia Boulevard West	Lethbridge	AB	Canada	T1K 5N8	+1 (403) 456-9986
8	Callahan	Laura	IT Staff	6	1968-01-09T00:00:00.000Z	2004-03-04T00:00:00.000Z	923 7 ST NW	Lethbridge	AB	Canada	T1H 1Y8	+1 (403) 467-3351

This would return different results than if the OR operator were performed first. If we did want to have the OR operator first, we would use parentheses to form complex expressions. The query would look like this:

```
SELECT *
FROM employee
WHERE (title = 'IT Staff'
OR reports_to = 6)
AND phone like '%1';
```

This would force the query to do the OR statement first and then find the intersection between that result and the phone like ‘%1’. This statement would return:

Query Results

Row count: 1

employee_id	last_name	first_name	title	reports_to	birth_date	hire_date	address	city	state	country	postal_code	phone
8	Callahan	Laura	IT Staff	6	1968-01-09T00:00:00.000Z	2004-03-04T00:00:00.000Z	923 7 ST NW	Lethbridge	AB	Canada	T1H 1Y8	+1 (403) 467-3351

As you see, you can customize the result set through the use of filters with the AND and OR operators while using parentheses to control the ordering.

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 about the logical operators **AND** and **OR** in PostgreSQL. AND and OR enable you to **combine both conditions** in the WHERE clause of a query. An AND operator specifies that all conditions must be true in order for a row to appear in the result set. Multiple conditions can be combined with AND to create more precise and restrictive queries. You also learned that the OR operator allows for broader queries since it specifies that at least one of the conditions must be met for a row to be included in the results. Any of the specified conditions can be used to search for records matching the specified conditions, thus expanding the scope of the query.

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