ROUND to Round Numbers

by Sophia Tutorial

☰	WHAT'S COVERED

This tutorial explores using the ROUND function to round numbers in three parts:

1. Using the ROUND Function
2. Many Ways To Round
3. Rounding Averages

1. Using the ROUND Function

Although we have looked at how to cast values to numeric values in PostgreSQL, this is a feature that doesn't exist in all databases. The ROUND function is one that does exist in most databases. It rounds a numeric value to the nearest integer, or within a number of decimal places.

The ROUND function takes in two arguments. The first is the number, column or expression to be rounded, and the second is the number of decimal places to be rounded to. If you do not include the second argument, it defaults to 0, meaning it will round to the nearest integer.

Let's run a quick script to make some updates to the track table so that we can see how the ROUND function works:

```
UPDATE track SET unit_price = .49 WHERE track_id = 1;
UPDATE track SET unit_price = .51 WHERE track_id = 2;
UPDATE track SET unit_price = .55 WHERE track_id = 3;
UPDATE track SET unit_price = .01 WHERE track_id = 4;
UPDATE track SET unit_price = .91 WHERE track_id = 5;
UPDATE track SET unit_price = .95 WHERE track_id = 6;
```

We can now see what this would look like when ordered by track_id. The LIMIT clause can be used to only display the first six rows.

```
SELECT track_id, unit_price
FROM track
ORDER BY track_id
LIMIT 6;
```

Query Results

Row count: 6

track_id	unit_price
1	0.49
2	0.51
3	0.55
4	0.01
5	0.91
6	0.95

2. Many Ways To Round

Let's see how our results would change if we rounded based on various parameters:

SELECT track_id, unit_price, ROUND(unit_price), ROUND(unit_price,1),ROUND(unit_price,2),ROUND(unit_price,3)
FROM track
ORDER BY track_id
LIMIT 6;

Query Results

Row count: 6

track_id	unit_price	round	round	round	round
1	0.49	0	0.5	0.49	0.490
2	0.51	1	0.5	0.51	0.510
3	0.55	1	0.6	0.55	0.550
4	0.01	0	0.0	0.01	0.010
5	0.91	1	0.9	0.91	0.910
6	0.95	1	1.0	0.95	0.950

The first round takes the unit price to the closest integer. Any value of 0.50 or higher is rounded to 1, whereas any value less than 0.50 is rounded to 0. The second round takes the unit price to the closest tenth. Note how the 3rd and 6th rows round up. The third round takes the unit price to the nearest hundredth. No changes with the data appear here. The fourth round takes the unit price the nearest thousandth.

3. Rounding Averages

The round can be most useful within calculations that use the AVG function, to simplify the data. For example, if we want to get the average total of invoices for each customer, the query would look like the following:

SELECT customer_id, AVG(total)
FROM invoice

GROUP BY customer_id
ORDER BY customer_id;
Here is the result:

Query Results

Row count: 59

customer_id	avg
1	5.7142857142857143
2	5.4285714285714286
3	5.7142857142857143
4	5.7142857142857143
5	5.8571428571428571
6	7.1428571428571429

Since we're looking at currency, it makes sense to round it to the nearest cent (two decimal places). We could do that by adding the ROUND function around the AVG function.

SELECT customer_id, ROUND(AVG(total),2)
FROM invoice
GROUP BY customer_id
ORDER BY customer_id;

Query Results

Row count: 59

customer_id	round
1	5.71
2	5.43
3	5.71
4	5.71
5	5.86
6	7.14
7	6.14
8	5.43
9	5.43

Video Transcription

The round function can be extremely useful, especially when it comes to aggregate calculations, like the average. As you'll see here, we're actually selecting the customer ID, as well as their average total from the invoice based on the customer ID. And what the results of it-- it doesn't really make sense, as there's multiple different decimal places, which is hard to read. Being that it's a currency value, it makes most sense to be able to round to the nearest hundredth column.

So, what we can do in this case here is add the round function. The round function goes around a specific value-- a column, or a calculation. In this case here, it's the average total. And as a parameter, we can pass in the number of decimal places we want to store. So, if we have it at two, it makes the most sense in this case here where it's round to the nearest hundredth. However, and we can certainly change it, or even remove it. If we remove it, it'll round to the nearest integer in this case here. And we can always add in additional decimal places as needed to make the most sense of the data set itself.

☑	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

The ROUND function can be used to round numbers, columns and calculations to the nearest decimal or integer.

Source: Authored by Vincent Tran