

Table Calculations

Leigha Jarett

Welcome back! In this module, we'll focus on how you can use table calculations to create new metrics instantaneously.

Agenda

01 Introduction to Table Calculations

02 Types of Table Calculations

03 Example 6: Writing and Visualizing
Table Calculations

We'll begin with a general introduction to table calculations and how they can be used to create new metrics in Looker.

Agenda

- 01 Introduction to Table Calculations
- 02 Types of Table Calculations
- 03 Example 6: Writing and Visualizing Table Calculations

Then, we'll explore the different types of table calculations and walk through an example of using table calculations to create new metrics in Looker.

Agenda

04

Offset Functions

05

Example 7: Writing Offset Calculations

06

Lab 2: Working with Table Calculations and Offsets in Looker

Next, we'll discuss offsets, which are a subset of table calculations, and review an example of offset calculations in Looker.

Agenda

04

Offset Functions

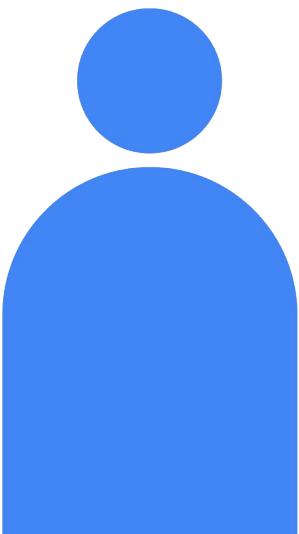
05

Example 7: Writing Offset Calculations

06

Lab 2: Working with Table Calculations and Offsets in Looker

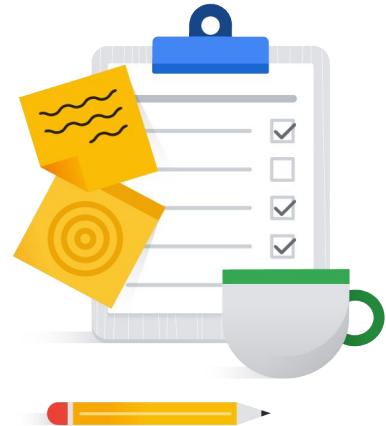
Last, you will practice working with table calculations and offsets in Looker using Qwiklabs.



Sounds pretty cool, right? Let's get started!

Table calculations

- 01 [Introduction to Table Calculations](#)
- 02 Types of Table Calculations
- 03 Example 6: Writing and Visualizing Table Calculations



Google Cloud

Typically, in Looker, data explorers are given access to one or more Explores predefined by LookML developers. At your own company, that would likely mean your data and analytics team, data engineers, or data analysts.

But sometimes you'll find you need particular logic for which they haven't provided a dimension or measure, maybe because you have a new kind of question or use case.

This is when you might need a table calculation.

What is a table calculation?

- A way to create on-the-fly metrics with formulas

The screenshot shows the 'Explore' interface in Google Cloud BigQuery. At the top, there's a search bar and tabs for 'All Fields' and 'In Use'. Below these are sections for 'Custom Fields' and 'TABLE CALCULATIONS'. Under 'TABLE CALCULATIONS', there is one entry: 'Percent of Flights Cancelled', which is highlighted with a green background.

Examples are based on fictional data.

Google Cloud

Table calculations are used to define what we call “on-the-fly metrics” because they run on your query results, instead of your whole database.

You'll need to start by generating results with at least one dimension or measure, and then you can incorporate these fields into spreadsheet software-like formulas to calculate a new metric.

What is a table calculation?

- A way to create on-the-fly metrics with formulas
- An option to create new fields, without waiting for a LookML developer

The screenshot shows the 'Explore' interface in Looker. At the top, there's a search bar and tabs for 'All Fields' and 'In Use'. Below that, a section titled 'Custom Fields' is expanded, showing a list of fields. One field, 'Percent of Flights Cancelled', is highlighted with a green background and is listed under 'TABLE CALCULATIONS'.

Field	Type	Count
Aircraft		1
Aircraft Destination		1
Aircraft Models		1
Aircraft Origin		1
Carriers		1
Flights		2
Flights Details		1

Examples are based on fictional data.

Google Cloud

With table calculations, you can also set up a custom field that isn't included in the set of dimensions and measures provided by your LookML developers.

For example, you can use a table calculation to create a new measure for percent of flights cancelled from existing measures on number of flights completed and number of flights cancelled.

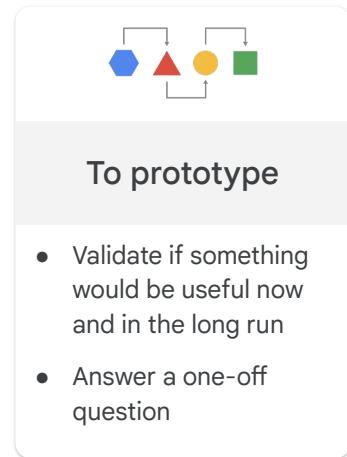
Table calculations are really cool and convenient. However, because they run after the query has been set up and are written on the fly, they can introduce differences across an organization, and they need to be recreated manually if they are not saved to a Look or dashboard.

When to use a table calculation

Google Cloud

So, we recommend using table calculations for a couple of specific use cases.

When to use a table calculation



Google Cloud

The first is to prototype a field or query, like if you want to validate whether it's useful enough to reuse. Similarly, if you need to answer a one-off question, then a table calculation is useful.

If you find yourself needing the same logic repeatedly, you should work with your LookML developer team to have that logic added to the data model as dimensions or measures that everybody can use.

When to use a table calculation



To prototype

- Validate if something would be useful now and in the long run
- Answer a one-off question



To support a results set

- Create a tile or visualization based purely on the query results

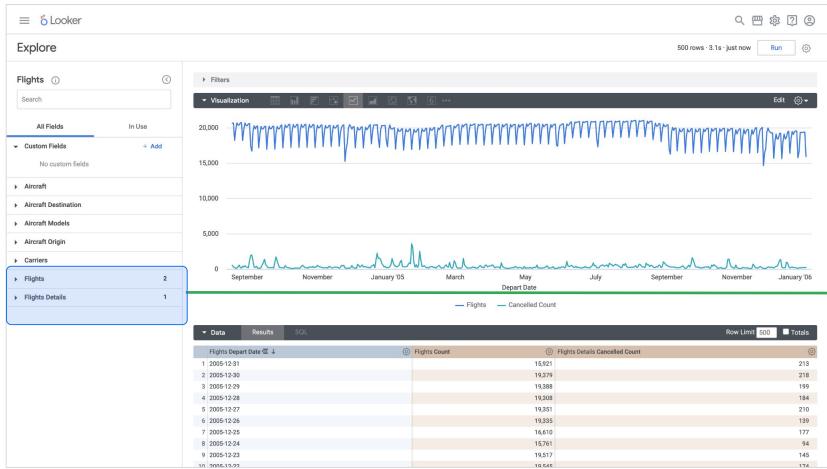
Google Cloud

Another scenario that might call for a table calculation is if you need to create something based purely on the results set and not on the overall data.

A specific example would be if you want to create a specific kind of visualization, like one not available in your current Looker version or for which you don't have the right types of fields in your data model.

In this case, you can use a table calculation to translate your data or visualization into the form you want.

Table calculations



1

Select the fields needed to calculate the desired metric

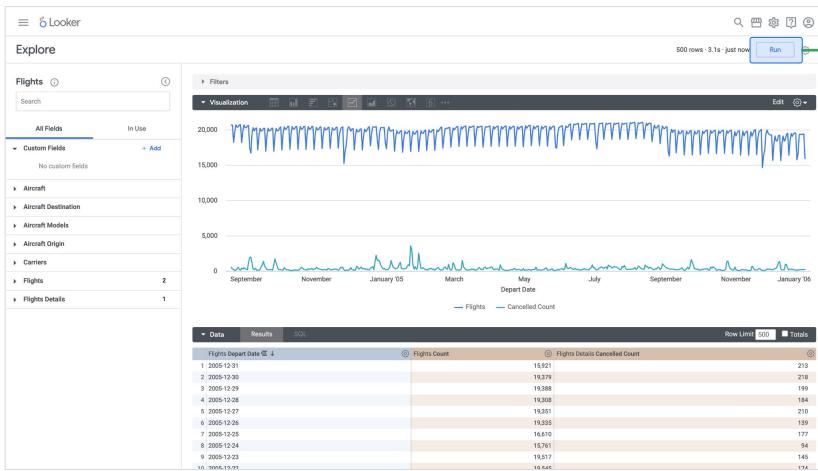
Examples are based on fictional data.

Google Cloud

Now, to understand how table calculations are created, let's walk through an example.

Since table calculations only operate on the results of your Explore query, you first need to bring in one or more dimensions or measures by selecting the fields you need to calculate the desired metric...

Table calculations



Examples are based on fictional data.

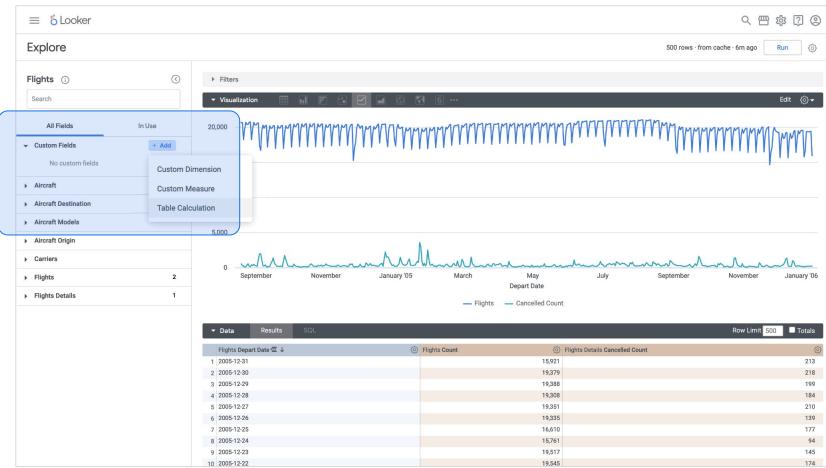
Google Cloud

... and then click **Run**.

For example, imagine that you want to calculate the percent of flights cancelled over a period of time. You can start by selecting the dimension for **Depart Date** and the measures for flight **Count** and **Cancelled Count**, and clicking **Run**.

Table calculations

From the “Custom Fields” section, add a new table calculation.



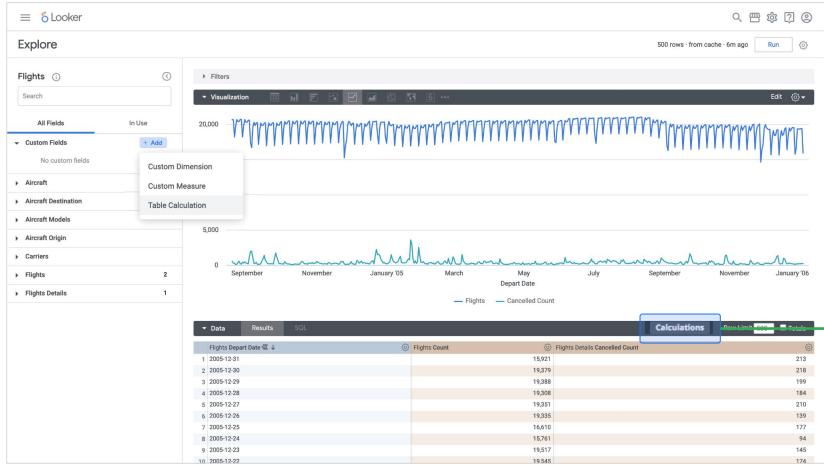
Examples are based on fictional data.

Google Cloud

After you select the dimensions and measures and run the initial results, you can now create a new table calculation from the results.

Next to **Custom Fields**, click the **Add** button, and select **Table Calculation**. A new popover window will appear in which to enter your formula.

Table calculations



Examples are based on fictional data.

4

If your admin does not have the Custom Fields feature enabled, table calculations will appear under **Calculations** at the top of the data table.

Google Cloud

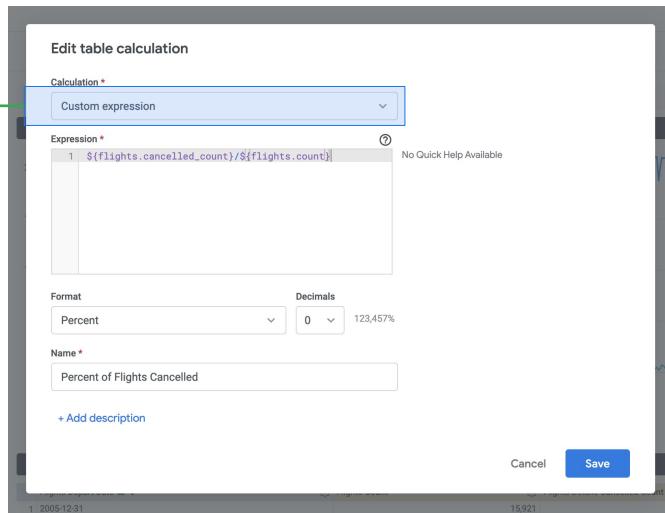
Now, one thing to be aware of here is that because administrators must enable this feature for end users, you might not have the **Custom Fields** view in the Explore of your organization's Looker instance, even if you do have permission to use table calculations.

In that case, you would see a **Calculations** button at the top of the data table.

Table calculations

5

Select a calculation type.

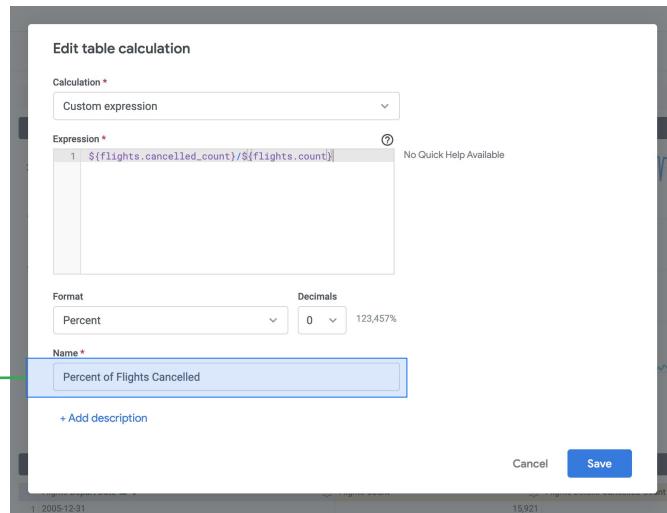


Examples are based on fictional data.

Google Cloud

Assuming Custom Fields are available to you, in the table calculation popover, you will need to select a calculation type, or leave the selected option for **Custom expression** to write ad hoc expressions.

Table calculations



6
Name the new calculation.

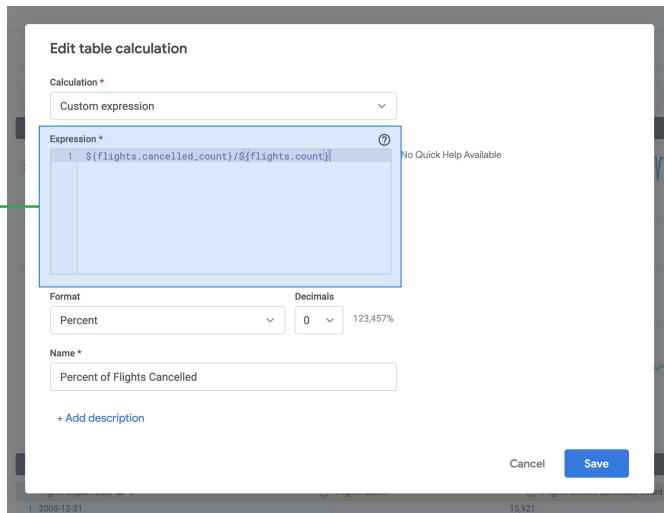
Examples are based on fictional data.

Google Cloud

Next, give your new table calculation a name such as Percent of Flights Cancelled. This name will show up as the column header and in the legend of the visualization.

Table calculations

7
Insert a formula.



Examples are based on fictional data.

Google Cloud

If using a custom expression for the table calculation, enter the formula you'd like to apply.

For example, you can calculate the percentage of cancelled flights by dividing the cancelled flight count by the overall count of flights.

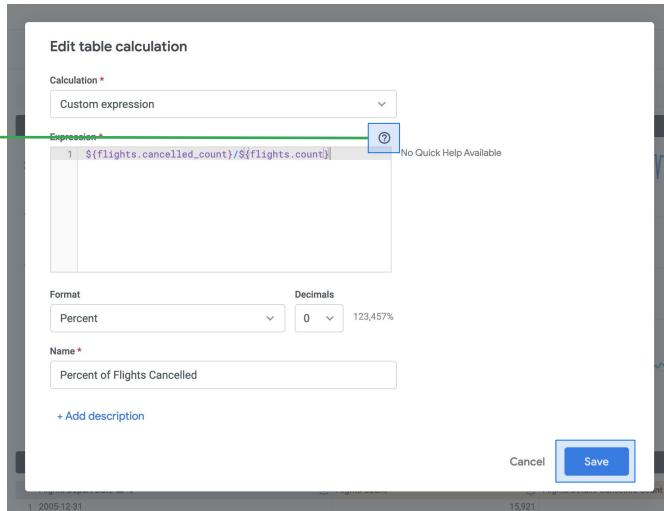
Note that Looker strives for parity with commonly used spreadsheet software, so if you simply type the letter **a**, you can see all the various functions that Looker supports.

You can get the absolute value of a number, add time intervals to a date, concatenate multiple values, write if-conditions, and so on.

Table calculations

8

Get help as needed.



Examples are based on fictional data.

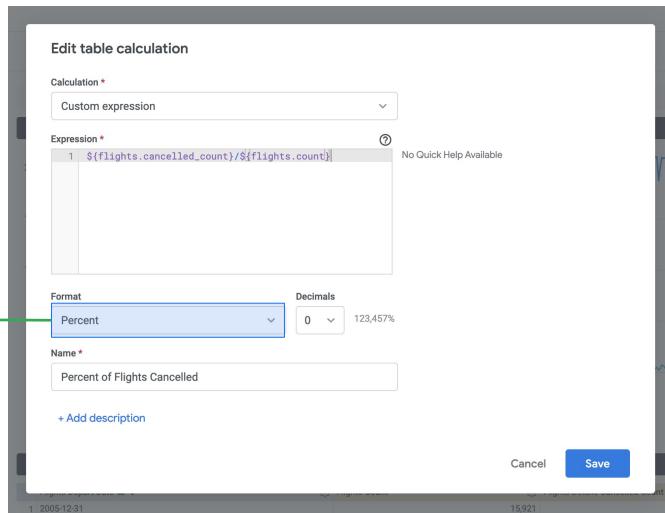
Google Cloud

If you want to see the full list of supported functions and some more information about them, you can click on the **Help + Syntax Reference** icon above the top right of the window, which takes you to our documentation.

An important thing for people very familiar with spreadsheet software to know is that with Looker table calculations, you do *not* start the formula with an equals sign. That might feel a bit weird, but you'll get used to it.

Table calculations

9
Apply a format.

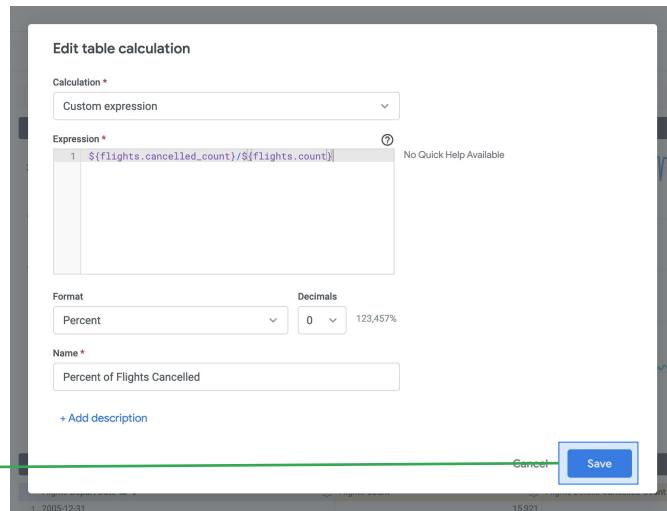


Examples are based on fictional data.

Google Cloud

Next, specify the format of the results if you'd like: decimals, percent, and so on.

Table calculations



10

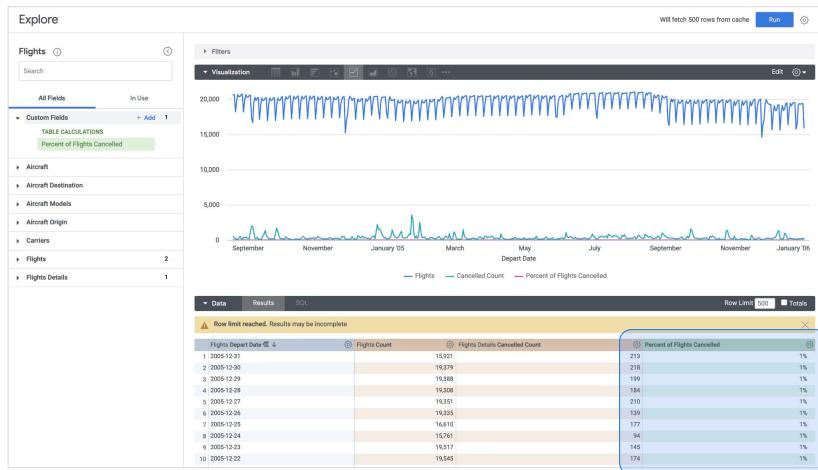
Save the calculation.

Examples are based on fictional data.

Google Cloud

Finally, click **Save** to save your table calculation.

Table calculations



11

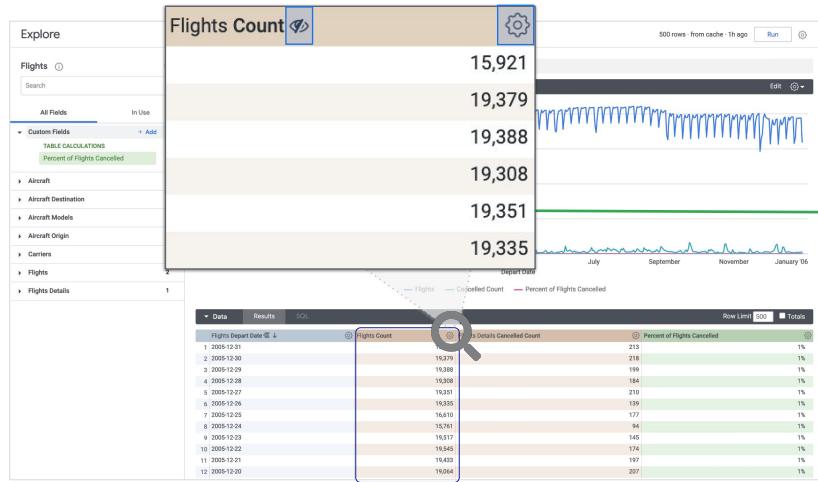
The new calculation displays as part of the Results and within the visualization.

Examples are based on fictional data.

Google Cloud

You will now see table calculations appear in your data table in green. The values will show up right away since you have already run your query with the dimensions and measures.

Table calculations



12

If only the result is required in the visualization, you can hide the components of the table calculation.

An icon (👁️) indicates that the column is hidden.

Examples are based on fictional data.

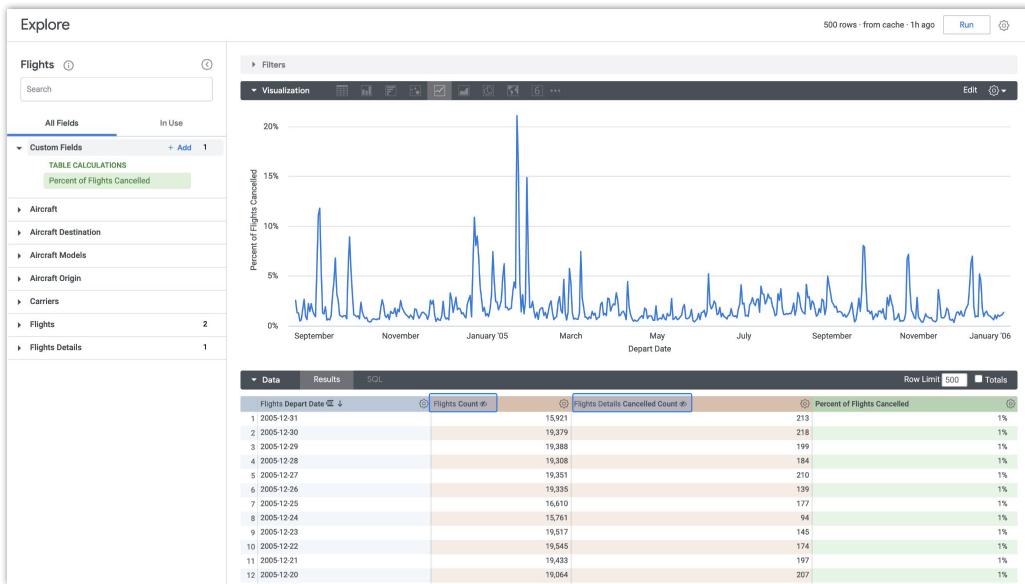
Google Cloud

Table calculations also get displayed in the visualization.

If you don't want a specific field or table calculation in your visualization, you can click on the gear icon in the column header and select **Hide from Visualization**.

More often than not, the table calculation is the thing you actually want to visualize; it might be the foundational dimensions or measures that you *don't* want, such as the flight count because you want the visualization to show only the percent of cancelled flights.

An icon of an eye with a line crossing over it will appear to indicate that the column is hidden.

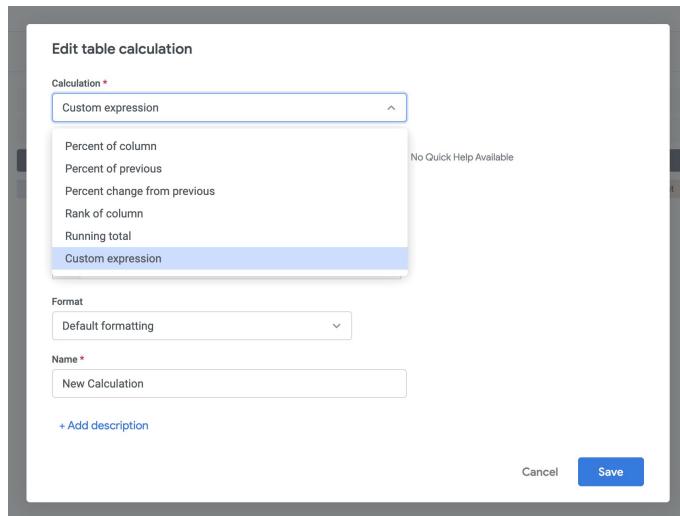


Examples are based on fictional data.

Google Cloud

So, here's how the final line chart would look if you keep the dimension and table calculation, but *hide* the two measures used in the table calculation formula.

Table calculations



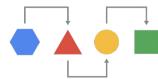
Examples are based on fictional data.

Google Cloud

In addition to writing custom expressions for table calculations, you can choose from some predefined options such as calculating the percent of column total, percent change from a previous row, running totals, and more.

With options for both predefined and custom expressions, Looker gives you the power to customize your visualizations and reports as needed.

Table calculations



To prototype

- Validate if something would be useful now and in the long run
- Answer a one-off question



To support a results set

- Create a tile or visualization based purely on the query results

Examples are based on fictional data.

Google Cloud

In summary, table calculations allow you to instantaneously create new metrics and fields, without needing to wait for a LookML developer to create them for you.

Enjoy using table calculations. Just remember that since they run after the query has been set up and are written on the fly, they can introduce differences across an organization, and they need to be recreated manually if they are not saved to a Look or dashboard.

For these reasons, table calculations are most useful for prototyping new metrics, answering one-off questions, or creating visualizations based purely on query results.

And with this knowledge, we hope you now look forward to playing around with table calculations in Looker!

Table calculations

- 01 Introduction to Table Calculations
- 02 [Types of Table Calculations](#)
- 03 Example 6: Writing and Visualizing Table Calculations



Google Cloud

In Looker, table calculations provide you with the ability to define new metrics instantaneously using custom formulas.

With table calculations, you can prototype new metrics or create one-off visualizations from your query results, without having to wait for a LookML developer to modify the options available in a particular Explore.

Types of table calculations

01

String

Google Cloud

There are four basic types of table calculations in Looker: string...

Types of table calculations

01

String

02

Mathematical

Google Cloud

There are four basic types of table calculations in Looker: string, mathematical...

Types of table calculations

01

String

02

Mathematical

03

Logical

Google Cloud

There are four basic types of table calculations in Looker: string, mathematical, logical...

Types of table calculations

01

String

02

Mathematical

03

Logical

04

Date and time

Google Cloud

There are four basic types of table calculations in Looker: string, mathematical, logical, and date & time.

01 String functions

- Lexical formulas operate on sentences, words or letters.
- Examples: concat(), upper(), substring()

Google Cloud

String functions operate on query results that are text. For example, they can be used to capitalize letters and words, extract parts of a phrase, check to see if a word or letter is in a phrase, or replace elements of a word or phrase.

01 String functions

- Lexical formulas operate on sentences, words or letters.
- Examples: concat(), upper(), substring()

Aircraft Category	Accidents Country	Accidents Count	Accident report
1 Airplane	US	1,313	1313 Airplane accidents in US
2 Helicopter	US	141	141 Helicopter accidents in US
3 Glider	US	33	33 Glider accidents in US
4 Airplane	Brazil	7	7 Airplane accidents in Brazil
5 Airplane	UK	6	6 Airplane accidents in UK

```
concat(${accidents.count}, " ",
${accidents.aircraft_category}, "accidents in",
${accidents.country})
```

Examples are based on fictional data.

Google Cloud

In this example table, there are dimensions for **Aircraft Category** and **Country**, and a **Count** measure. For some people, this table may not be very easy to read or understand.

To make the table easier to interpret, you can *concatenate* the values from these three fields with some additional text, and create a nice, user-friendly sentence in a new field called **Accident report**.

In your final visualization, you could then hide **Aircraft Category**, **Country**, and **Count**, and simply display the user-friendly sentence in the **Accident report** field that you created with a table calculation.

02 Mathematical functions

- Some functions return a distinct value for each and every row.
- Other functions, like averages and running totals, operate over many rows.
- Examples: `+`, `-`, `*`, `/`, `mean()`, `sum()`, `stddev_samp()`.

Google Cloud

With mathematical functions in Looker, you can apply any kind of math on numeric values, such as arithmetic, averages, sums—even running totals and standard deviations.

02 Mathematical functions

- Some functions return a distinct value for each and every row.
- Other functions, like averages and running totals, operate over many rows.
- Examples: +, -, *, /, mean(), sum(), stddev_samp().

Orders Create date	Orders Amount	Order Amount Rounded	Average Order Amount
1 2015-10-22	10.15	10	106.01
2 2015-10-22	13.95	14	106.01
3 2015-10-22	15.95	16	106.01
4 2015-10-22	17.70	18	106.01
5 2015-10-22	18.38	18	106.01

```
round(${orders.amount}, 0)
```

```
mean(${orders.amount})
```

Examples are based on fictional data.

Google Cloud

In this example, the **Order Amount Rounded** table calculation is rounding the **Amount** measure to the nearest decimal with 0 places (in other words, the nearest integer), while the **Average Order Amount** table calculation is calculating the mean of all the **Amount** values.

03 Logical functions

- Some functions return a distinct value for each and every row.
- Other functions, like averages and running totals, operate over many rows.
- Examples: if(), contains(), AND, OR, NOT

Google Cloud

With logical functions, you can check one or more conditions and execute different paths of logic, depending on the value.

For example, if-conditions are very popular for displaying different results depending on whether our particular condition is met.

03 Logical functions

- Some functions return a distinct value for each and every row.
- Other functions, like averages and running totals, operate over many rows.
- Examples: if(), contains(), AND, OR, NOT

Title	Title	Title Total Revenue	Title Total Budget	Profit Outcome
1	Example Movie 1	\$301.96M	250.00 M	Profitable
2	Example Movie 2	\$444.74 M	250.00 M	Profitable
3	Example Movie 3	\$255.12 M	250.00 M	Profitable
4	Example Movie 4	\$241.07 M	250.00 M	Lost Money
5	Example Movie 5	\$760.51 M	237.00 M	Profitable

```
if(${movie_revenue.total_revenue}>${movie_budget.total_budget}, "Profitable", "Lost Money")
```

Examples are based on fictional data.

Google Cloud

In this example, there is a **Title** dimension referring to box-office films and measures for **Total Revenue** and **Total Budget**.

The **Profit Outcome** table calculation evaluates whether the **Total Revenue** amount for each **Title** is greater than the **Total Budget**. If it is, the displayed value is the word “Profitable”.

However, if **Total Revenue** is /ess than or equal to **Total Budget**, the displayed value is “Lost Money”.

04 Date and time functions

- Date and time functions allow us to extract part of a date (like year), or calculate the difference between two date/times.
- Examples: extract_hours, extract_years, diff_hours, diff_years, now()

Google Cloud

Finally, date and time functions operate on datetime results and allow you to extract time periods and calculate date differences and additions.

You can also get the current date, time, or date and time, and create arbitrary date fields as needed.

04 Date and time functions

- Date and time functions allow us to extract part of a date (like year), or calculate the difference between two date/times.
- Examples: extract_hours, extract_years, diff_hours, diff_years, now()

Users Name	Users Created Date	Months a Customer	Join Month
1 Zoraida Ross	2014-09-11	13	September
2 Zoe Prince	2015-06-05	4	June
3 Zina Hazlewood	2015-07-24	3	July
4 Zelma Horton	2015-10-10	0	October
5 Zelda Alonzo	2013-01-01	33	January

```
diff_months(${user.created_date},now())
if(extract_months (${user.created_date})=8, "August",
if(extract_months (${user.created_date})=9, "September",
if(extract_months (${user.created_date})=10, "October",
if(extract_months (${user.created_date})=11, "November",
if(extract_months (${user.created_date})=12, "December",
"Other"))))))))))))
```

Examples are based on fictional data.

Google Cloud

In this example, there is a **Created Date** dimension showing when each user registered on our site.

The **Months a Customer** table calculation calculates the difference in months between a user's Created Date and today's date using the **diff_months** and **now** functions.

The **Join Month** table calculation extracts the month from the **Created Date** using **extract_months** and then applies logic to label the month name using an if-condition. For example, if the number extracted for month is 8, then the output is "August"; if it is 9, then the output is "September"; and so on.

These examples demonstrate how easy it is to combine different types of functions in one table calculation.

Types of table calculations - recap

01

String

02

Mathematical

03

Logical

04

Date and time

Google Cloud

In summary, table calculations allow you to create new metrics instantaneously and are incredibly useful for prototyping new metrics or creating one-off visualizations.

There are four main types of table calculations in Looker: string, mathematical, logical, and date & time.

String functions operate on text results, while mathematical functions operate on numeric results. Data & time functions operate on datetime results, and logical functions can be used to check one or more conditions and execute different paths of logic, depending on the value.

Last, you can combine different types of functions in one table calculation to customize your results even further and achieve your desired outcomes.

Table calculations

- 01 Introduction to Table Calculations
- 02 Types of Table Calculations
- 03 Example 6: Writing and Visualizing Table Calculations



Google Cloud

In this example, we will review how to write and visualize table calculations to instantaneously create new metrics.

Example: Writing and visualizing table calculations

What percentage of revenue per US state is from completed orders (out of revenue from all orders for the state) in the past 1 year?



Google Cloud

We'll work in the **Order Items** Explore to determine what percentage of revenue per US state is from completed orders (out of revenue from *all* orders for that state) in the past 1 year. Then, we'll create a map of the results.

The screenshot shows the Looker interface. On the left, there's a sidebar with a magnifying glass icon pointing to the 'Order Items' explore. The main area is titled 'Your organization's folders' and shows a list of items under 'Folders' and 'Looks'. The 'Order Items' explore is highlighted with a blue box. The 'Business Pulse' dashboard and 'Yearly Revenue' look are also visible.

Category	Name	Details
Folders	Business Pulse	34 Views, Created by Jenny
	Yearly Revenue	22 Views, 1 Favorite, Created by Brian
Looks	Business Pulse	34 Views, Created by Jenny
	Yearly Revenue	22 Views, 1 Favorite, Created by Brian

Examples are based on fictional data.

Google Cloud

Begin by opening the **Order Items** Explore under the **E-Commerce Training** heading.

The screenshot shows the Google Cloud BigQuery Explore interface. On the left, the 'Explore' sidebar lists various fields under 'Order Items'. Under 'DIMENSIONS', 'Created Date' and 'Delivered Date' are listed. Under 'MEASURES', 'Average Sale Price', 'Average Spend per User', 'Order Count', and 'Order Item Count' are listed. Two measures are highlighted with a blue border: 'Total Revenue' and 'Total Revenue from Completed Or...'. In the main area, the 'Data' tab is selected, showing the SQL query: 'SELECT SUM(revenue) AS "Order Items Total Revenue" FROM `Order Items` WHERE status = "Completed"'. Below the query, it says 'Order Items Total Revenue From Completed Orders'. At the top right, there are buttons for 'Run' and 'Row Limit 500'.

Examples are based on fictional data.

Google Cloud

From the **Order Items** view, select the **Total Revenue** and **Total Revenue from Completed Orders** measures.

The screenshot shows the Google Cloud BigQuery Explore interface. On the left, the sidebar lists dimensions and measures for 'Order Items'. A blue box highlights the 'Created Date' dimension under 'Dimensions'. A magnifying glass icon is placed over the 'Filter' button next to the 'Date' dimension. The main area shows a search bar with the query 'Order Items Total Revenue' and a filter bar below it. The filter bar has three fields: 'is in the past', '1', and 'complete years'. The results section displays the message 'No Results'.

Examples are based on fictional data.

Google Cloud

Click on the **Filter** button next to the **Date** dimension in the **Created Date** group.

Then, set the **Created Date** filter to “is in the past” “1” “complete years”.

The screenshot shows a data analysis interface with the following components:

- Search Bar:** "Order Items" with a placeholder "Search".
- Filter Bar:** "Filters (1)" with a single filter: "Order Items Created Date" set to "is in the past" for "1 complete years".
- Visualization Tabs:** "Data" (selected), "Results", and "SQL".
- Results Section:** "No Results".
- Left Sidebar:** "All Fields" (In Use) and "Custom Fields" (+ Add). Other sections include "Distribution Centers", "Inventory Items", "Order Items" (3), "Products", and "Users" (1).
 - Dimensions:** Age, City, Country, Created Date, Email, First Name, Gender, ID, Last Name, Latitude, Longitude, State, Traffic Source, Zip.
 - Measures:** Count.
- Annotations:** A magnifying glass icon is positioned over the "State" dimension in the sidebar, and a blue box highlights the "State" dimension in the visualization area.

Examples are based on fictional data.

Google Cloud

Next, in the **Users** view, select the **State** dimension.

The screenshot shows the Google Cloud BigQuery Explore interface. On the left, the sidebar lists dimensions and measures: All Fields, Custom Fields (+ Add), Distribution Centers, Inventory Items, Order Items (3), Products, and Users (2). Under Users, there are several dimensions: Age, City, Country, Created Date, Email, First Name, Gender, ID, Last Name, Latitude, Longitude, State, Traffic Source, and Zip. A magnifying glass icon is placed over the 'Country' dimension. The main area shows two filters applied: 'Order Items Created Date' (is in the past, 1 complete years) and 'Users Country' (is equal to USA). Below the filters, the visualization section shows 'Data' selected, with a query for 'Order Items Total Revenue'. The results pane displays the message 'No Results'.

Examples are based on fictional data.

Google Cloud

Then, click on the **Filter** button next to the **Country** dimension, and set the filter to “is equal to” **USA**. Click **Run**.

The screenshot shows the Google Cloud BigQuery Explore interface. On the left, there's a sidebar with 'All Fields' selected under 'Order Items'. A search bar at the top has 'Table Calculation' typed into it. A magnifying glass icon highlights the search term. In the center, a modal window titled 'Custom Dimension' is open, with three tabs: 'Custom Dimension', 'Custom Measure', and 'Table Calculation'. The 'Table Calculation' tab is highlighted with a blue box. To the right of the modal, a preview of a table titled 'Completed Orders' is shown, with columns like 'Order Items Total Revenue' and 'Order Items Created Date'. A 'Row Limit' dropdown is set to 500, and a 'Totals' checkbox is checked.

Examples are based on fictional data.

Google Cloud

Now that you have the initial query results, you can add a table calculation from the **Custom Fields** section.

Click **Add** next to Custom Fields, and then select **Table Calculation**.

The screenshot shows the Looker interface with the 'Edit table calculation' dialog box open. The dialog box has the following fields:

- Calculation ***: Custom expression
- Expression ***: `1 ${order_items.total_revenue_from_completed_orders} / ${order_items.total_revenue}`
- Format**: Percent
- Decimals**: 1
- Name ***: % Revenue from Completed Orders
- Save** button (highlighted)

In the background, a table of data is visible, showing columns for State and Revenue. The data includes rows for Missouri, Colorado, Tennessee, Indiana, Massachusetts, Virginia, Minnesota, Georgia, and Oklahoma, with revenue values ranging from \$53,225.09 to \$51,072.25.

Examples are based on fictional data.

Google Cloud

Use the Custom expression option and the following syntax for the expression:

`${order_items.total_revenue_from_completed_orders} / ${order_items.total_revenue}`

Next, change the format to **Percent** and decimals to **1**.

Name the table calculation **% Revenue from Completed Orders**, and then save the table calculation.

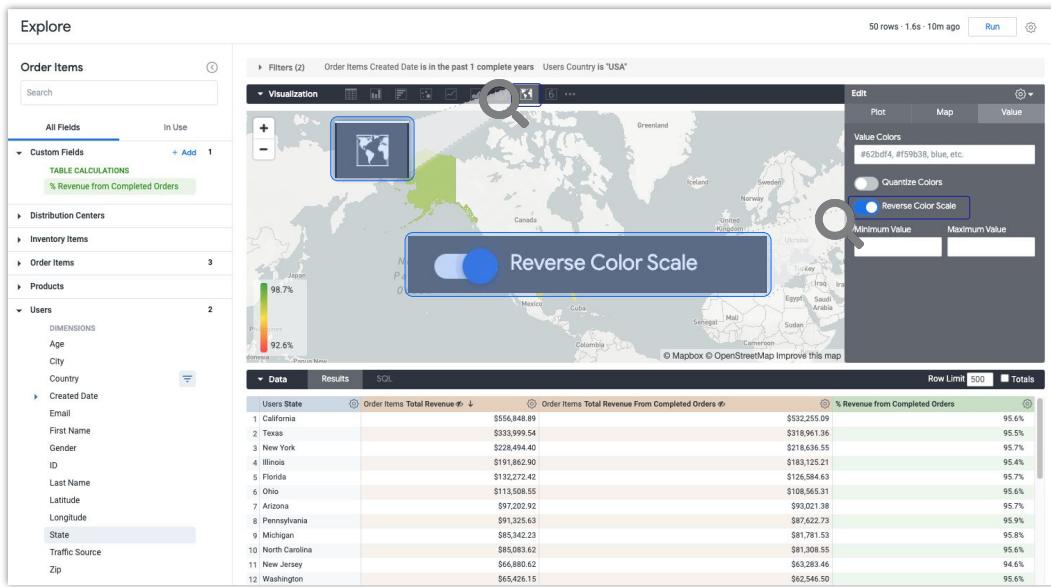
The screenshot shows a Looker dashboard titled "Orders". The main area displays a table of completed orders with the following columns: Order ID, User Country, Total Revenue, and % Revenue from Completed Orders. A context menu is open over the first row of the table, with the "Hide from Visualization" option highlighted. The sidebar on the left lists various dimensions and measures, including "Order Items", "Users", and "Dimensions".

Order ID	User Country	Total Revenue	% Revenue from Completed Orders
1 California	USA	\$63,283.46	95.6%
2 Texas	USA	\$62,546.50	95.5%
3 New York	USA	\$60,423.37	95.7%
4 Illinois	USA	\$60,520.13	95.4%
5 Florida	USA	\$57,398.27	95.7%
6 Ohio	USA	\$56,630.20	95.8%
7 Arizona	USA	\$54,553.52	95.6%
8 Pennsylvania	USA	\$53,715.06	95.7%
9 Michigan	USA	\$52,709.43	95.5%
10 North Carolina	USA	\$52,430.32	95.0%
11 New Jersey	USA	\$51,009.25	95.3%
12 Washington	USA	\$48,085.56	94.2%
13 Wisconsin	USA	\$46,195.57	97.9%
14 Missouri	USA	\$44,179.36	95.4%
15 Colorado	USA	\$43,577.14	95.8%
16 Tennessee	USA	\$41,681.94	95.5%
17 Indiana	USA	\$40,981.94	95.6%
18 Massachusetts	USA	\$39,747.08	95.7%
19 Virginia	USA	\$38,674.77	95.4%
20 Minnesota	USA	\$37,072.25	95.0%
21 Georgia	USA	\$36,307.38	95.3%
22 Oklahoma	USA	\$35,607.38	95.6%
23 Oregon	USA	\$34,907.38	95.7%
24 Alabama	USA	\$34,207.38	95.4%

Examples are based on fictional data.

Google Cloud

Next, hide the **Total Revenue** measure and the **Total Revenue from Completed Orders** measure from the visualization by clicking on each measure's gear menu, and selecting **Hide from Visualization**.

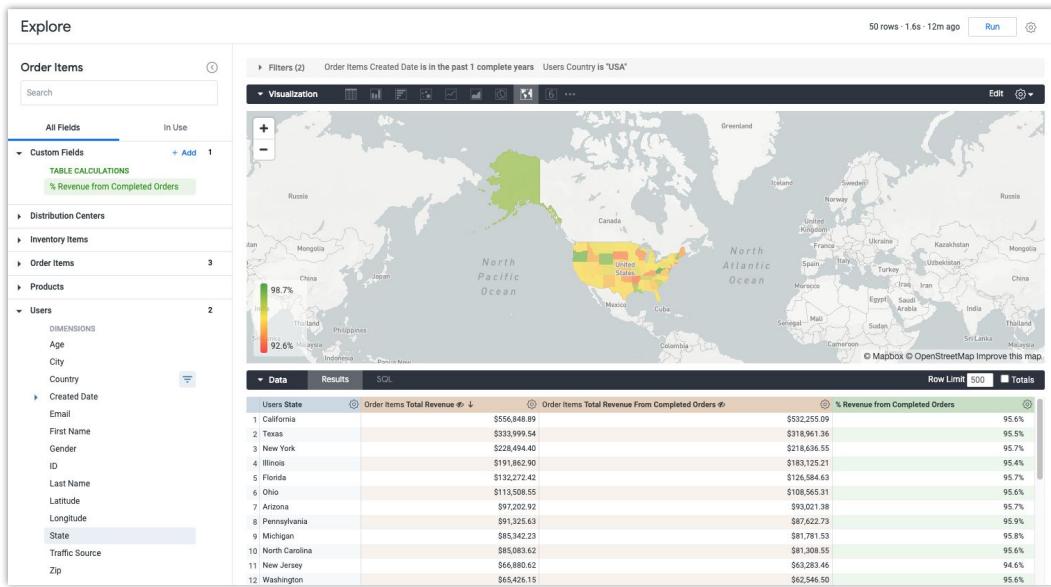


Examples are based on fictional data.

Google Cloud

Last, change the visualization type to **Map**.

You can also change the default color order in the legend by selecting **Reverse Color Scale** in the **Values** tab of the visualization settings, so that higher values are green and lower values are red.



Examples are based on fictional data.

Google Cloud

You now have a map visualization of percentage of revenue per US state from completed orders in the past 1 complete year.

Table calculations

04

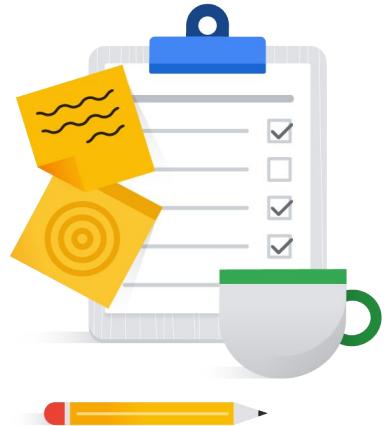
Offset Functions

05

Example 7: Writing Offset Calculations

06

Lab 2: Working with Table Calculations and Offsets in Looker



Google Cloud

Offset functions are a subset of table calculation functions. They allow you to programmatically reference values from other rows or columns in your query results to calculate new values.

Types of offset functions

01

`offset()`

02

`pivot_offset()`

03

`offset_list()`

Google Cloud

There are three main types of offset functions: regular `offset()`, `offset_list()`, and `pivot_offset()`.

offset()

- Used to reference a value that is in a higher or lower position in a column.
- Uses include computing period-over-period change and rolling averages.

Google Cloud

A regular **offset()** function is used when you want to reference a value from a higher or lower row of your results.

Note that changing the order of the rows by sorting your results can turn the previous row into the subsequent row, and vice versa.

offset()

- Used to reference a value that is in a higher or lower position in a column.
- Uses include computing period-over-period change and rolling averages.

Orders Create Month	Orders Total Revenue	Previous Month Total	Subsequent Month Total
1 2015-10	\$118,579.91	\$127,280.05	–
2 2015-09	\$127,280.05	\$132,223.17	\$118,579.91
3 2015-08	\$132,223.17	\$123,295.45	\$127,280.05
4 2015-07	\$123,295.45	\$112,443.68	\$132,223.17
5 2015-06	\$112,443.68	\$118,574.69	\$123,295.45

```
offset(${orders.total_revenue}, 1)
offset(${orders.total_revenue}, -1)
```

Examples are based on fictional data.

Google Cloud

In this example, there are results for a series of months in descending order—October, September, and so on—with the **Total Revenue** for each month.

The **Previous Month Total** is a table calculation that offsets the **Total Revenue** column by one. The calculation starts with October, which is in row 1. Then, it adds 1 to its row number, and captures the Total Revenue from row 2, which is September.

The **Subsequent Month Total** is a table calculation that offsets in the other direction; it does a negative-one offset, so it captures the value from the previous row.

There isn't a value for October, since that is already the first row of our results set. For September, it goes back from row 2 to row 1, and gets October's Total Revenue. For August, it goes back from row 3 to row 2, and gets September's Total Revenue.

As you can imagine, the sort order of your data is critical here. If you are sorting your data from newest to oldest, as seen here, you need to do a *positive* offset to get the previous month's data. But if you are sorting from oldest to newest, you need to do a *negative* offset to get the previous month's data.

With these new values, you can create yet another table calculation to calculate the percent difference from one month to the next, to see if business is improving or not.

pivot_offset()

- Used to reference the values in a column to the left or right.
- Uses include computing period-over-period change and rolling averages.

Google Cloud

The **pivot_offset()** function is used to reference values from a column to the left or the right when you have a pivot table.

pivot_offset()

- Used to reference the values in a column to the left or right.
- Uses include computing period-over-period change and rolling averages.

Orders Created Year	2012		2013		
	Orders Created Month Num	Orders Total Revenue	Last Year Revenue	Orders Total Revenue	Last Year Revenue
1 1	\$2,922.37	–	\$4,813.02	\$2,922.37	
2 2	\$5,231.76	–	\$8,491.73	\$5,231.76	
3 3	\$7,077.91	–	\$9,430.26	\$7,077.91	
4 4	\$11,804.10	–	\$15,250.35	\$11,804.10	
5 5	\$13,881.63	–	\$18,386.91	\$13,881.63	

```

pivot_offset(${orders.total_revenue}, -1)

(${orders.total_revenue}-${last_year_revenue})/${last_yea
r_revenue}

```

Examples are based on fictional data.

Google Cloud

In this example, there is a **Created Month Number** dimension with a pivot on **Created Year**.

From these results, maybe you want to know: how did revenue from January 2013 compare against revenue from January 2012? How did February 2013 compare to February 2012?

To determine this, the **Last Year Revenue** table calculation looks to the **-1** column, or one pivot column to the left, and pulls the **Total Revenue** from there. It is blank for 2012, since that is the very first pivot column in the results.

The sort order matters here, too, to determine whether you should be offsetting by a positive or negative value.

Once again, you could create yet another table calculation to calculate the percent difference from one year to the next, to see if business is improving or not.

offset_list()

- Used to combine values from multiple rows into a single cell.
- Uses include computing period-over-period change and rolling averages.

Google Cloud

The **offset_list()** function moves up or down a column of rows defined by a first provided value, and then grabs another number of rows' worth of data defined by a second provided value. The two numbers can be the same or different, depending on the results you want.

offset_list()

- Used to combine values from multiple rows into a single cell.
- Uses include computing period-over-period change and rolling averages.

	Orders Create Date	Orders Count	Offset List Results	Rolling Average
1	2016-03-23	391	"	-
2	2015-03-24	359	,391	391.00
3	2015-03-25	388	,391,359	375.00
4	2015-03-26	385	391,359,388	379.33
5	2015-03-27	376	359,388,385	377.33

```
offset_list(${orders.count}, -3,3)
mean(offset_list(${orders.count}, -3,3))
```

Examples are based on fictional data.

Google Cloud

In this example, the table calculation using `offset_list` references 3 rows back, or `-3` rows, and starting at that point, gets the order count for next 3 rows.

This means the first row where it's possible to get a full string of values is now row 4, where we can go back to row 1, and starting there, grab the three values of 391, 359, and 388 for rows 1, 2, and 3.

While this type of offset is not used as often as the first two types, it's perfect for calculating rolling averages. You may want to know, what is the rolling average number of users over the past 7 days? Or what is the rolling average number of purchases over the past 30 days?

That's where it might be useful to create one table calculation with the `offset_list()`, and another table calculation to determine something new, like the mean of that resulting list.

Types of offset functions - recap

01

`offset()`

02

`pivot_offset()`

03

`offset_list()`

Google Cloud

In summary, offsets are a subset of table calculation functions and allow you to programmatically reference values from other rows or columns in your query results to calculate new values.

There are three main types of offset functions: regular `offset()` for referencing rows higher or lower in a column, `offset_list()` for referencing values in a column to the left or right, and `pivot_offset()` for referencing multiple rows and combining their values into one new value.

With these new values, you can create even more table calculations to calculate new metrics such as the percent difference from one month to the next, one year to the next, or even a rolling average.

Very powerful features!

Table calculations

04

Offset Functions

05

Example 7: Writing Offset Calculations

06

Lab 2: Working with Table Calculations and Offsets in Looker



Google Cloud

In Looker, offset functions are a subset of table calculations that you can use to programmatically reference values from other rows or columns to calculate new values.

Example: Writing offset calculations

Calculate the percent change in the number of users created month over month in the year 2018.



Google Cloud

In this example, we'll use an offset function to calculate changes in values from one row to the next row.

Specifically, we'll apply the offset function in the **Order Items** Explore to find the percent change in the number of users created month over month in the year 2018.

The screenshot shows the Looker Explore interface. On the left, there's a sidebar with a search bar and a tree view of categories: E-Commerce Training (selected), Events, Looker Basics, and System Activity. Under E-Commerce Training, 'Order Items' is highlighted with a blue box and a magnifying glass icon. The main area is titled 'Your organization's folders' and contains two sections: 'Folders' and 'Looks'. In the 'Folders' section, there's a 'Human Resources' folder. In the 'Looks' section, there's a 'Business Pulse' look. Both sections have sorting and filtering options.

Examples are based on fictional data.

Google Cloud

From the **Explore** tab, locate **Order Items** under the **E-Commerce Training** heading.

The screenshot shows the Google Cloud BigQuery Explore interface. On the left, the sidebar lists various datasets and tables under 'Order Items'. Under the 'Users' section, the 'Created Date' dimension group is expanded, showing options like Date, Month, Quarter, Time, Week, and Year. The 'Month' option is highlighted with a blue box and a magnifying glass icon. The main panel displays a query editor with the text 'Users Created Month'. At the top right, it says 'Will process 3.44 MB' and has a 'Run' button. Below the query editor, there are tabs for 'Data', 'Results', and 'SQL', with 'Data' selected. A note at the bottom says 'Press "Run" to explore this data.'

Examples are based on fictional data.

Google Cloud

Using the **Order Items** Explore, you want to know the number of users created month over month.

So begin by expanding the **Users** view, and selecting **Month** under the **Created Date** dimension group.

The screenshot shows the Google Cloud BigQuery Explore interface. On the left, there's a sidebar with a tree view of datasets and tables. The main area shows a query editor with a single filter applied:

```
Users Created Year is in the year 2018
```

The interface includes a search bar, a results table, and various navigation and configuration buttons at the top.

Examples are based on fictional data.

Google Cloud

Then, you need to create a filter on the **Year** dimension using “is in the year” 2018.

The screenshot shows the Looker Explore interface. On the left, the sidebar lists dimensions and measures. Dimensions include All Fields, Distribution Centers, Inventory Items, Order Items, Products, and Users (with 2 selected). Measures include Count. A note at the bottom of the sidebar says "92 fields". In the center, a query is being built with a single filter: "Users Created Year is in the year 2018". The visualization section shows a bar chart for "Users Count" over "Users Created Month". At the top right, it says "Will process 3.44 MB" and has a "Run" button. Below the Run button is a "Custom Filter" checkbox. The bottom right of the interface says "Press 'Run' to explore this data.".

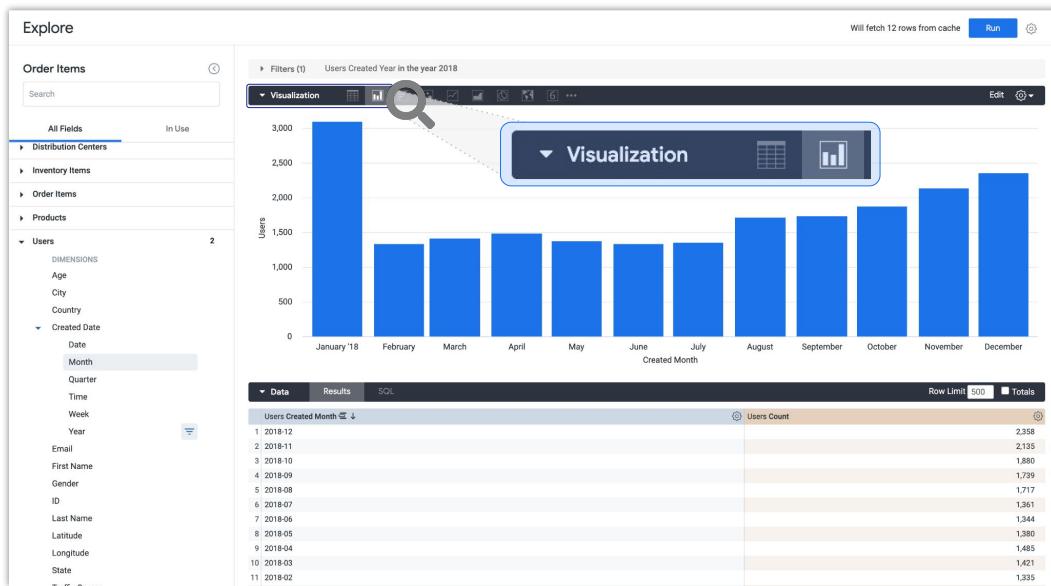
Examples are based on fictional data.

Google Cloud

You want the number of users, so you need to select the **Users Count** measure. Then, click **Run**.

Although you can choose to add table calculations before or after clicking the Run button, we recommend clicking Run first when working with offsets.

That way, you can see or decide on the data sort order *before* writing your offset function. This is important because the decision to offset by a positive or negative number depends on the sort order of your results set.

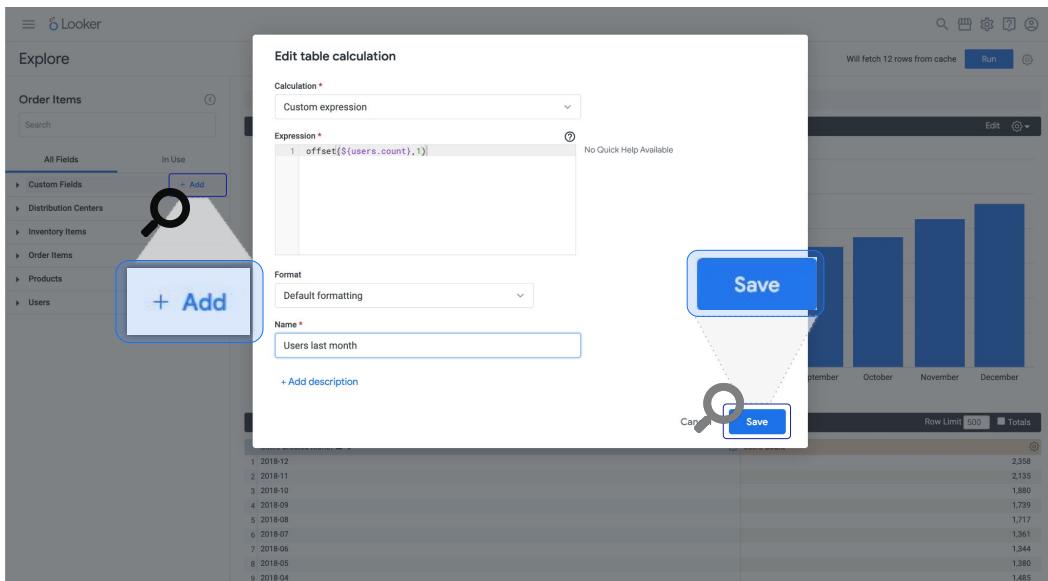


Examples are based on fictional data.

Google Cloud

Next, select column chart for the visualization type.

Notice that by default Looker is sorting by the **Created Month** descending, which is appropriate for our upcoming calculations. To ensure accurate table calculations, be sure to not sort these results manually.



Examples are based on fictional data.

Google Cloud

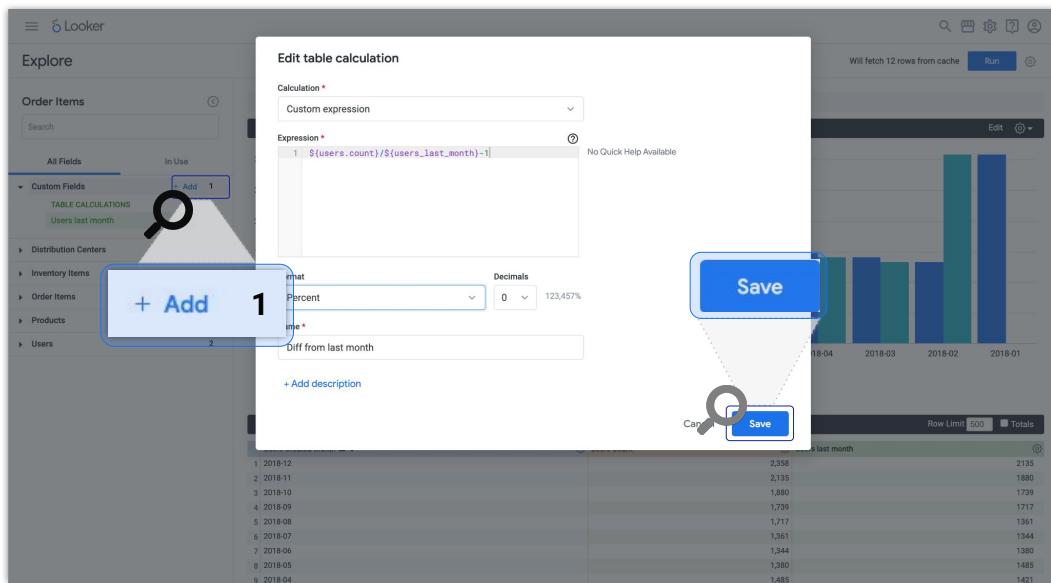
Next, you can create a new table calculation. Next to **Custom Fields**, click **Add**, and select **Table Calculation**.

When you write your table calculation, you need to use the following expression:

`offset(${users.count}, 1)`

This translates to: Get the count of users from each current row number +1.

Don't forget to give it a meaningful name, such as **Users last month**, and click **Save**.



Examples are based on fictional data.

Google Cloud

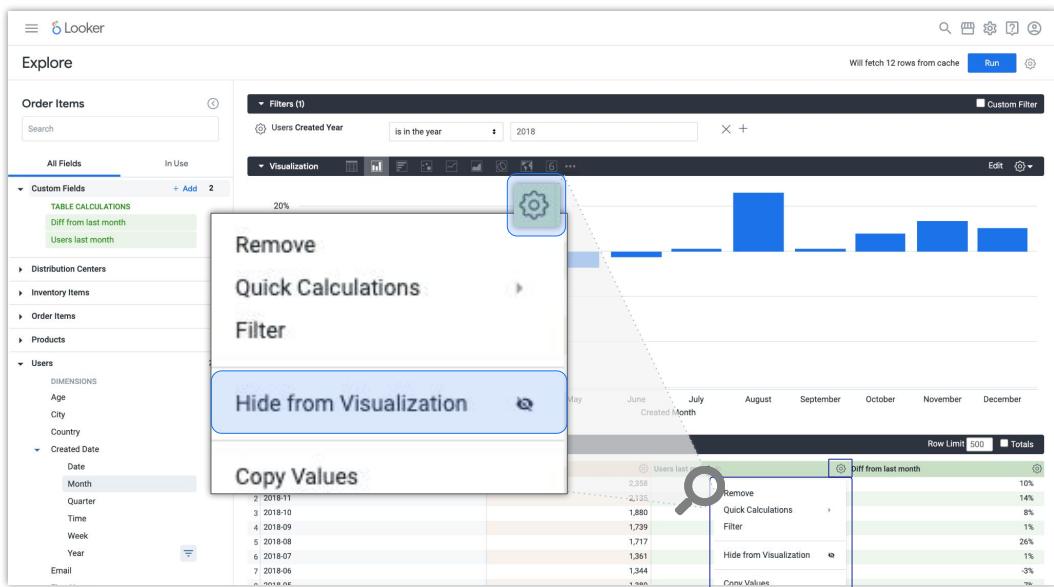
Now you need a second table calculation to calculate the percent change from the prior month to this one. Next to **Custom Fields**, click **Add**, and select **Table Calculation**.

For this table calculation, you can use `${users.count}` to get the current number of users, add the forward slash to divide, and then select the name of your first table calculation **Users last month** from the suggestions drop-down, minus one.

The formula will look like this:

`${users.count}/ ${users.last_month}-1`

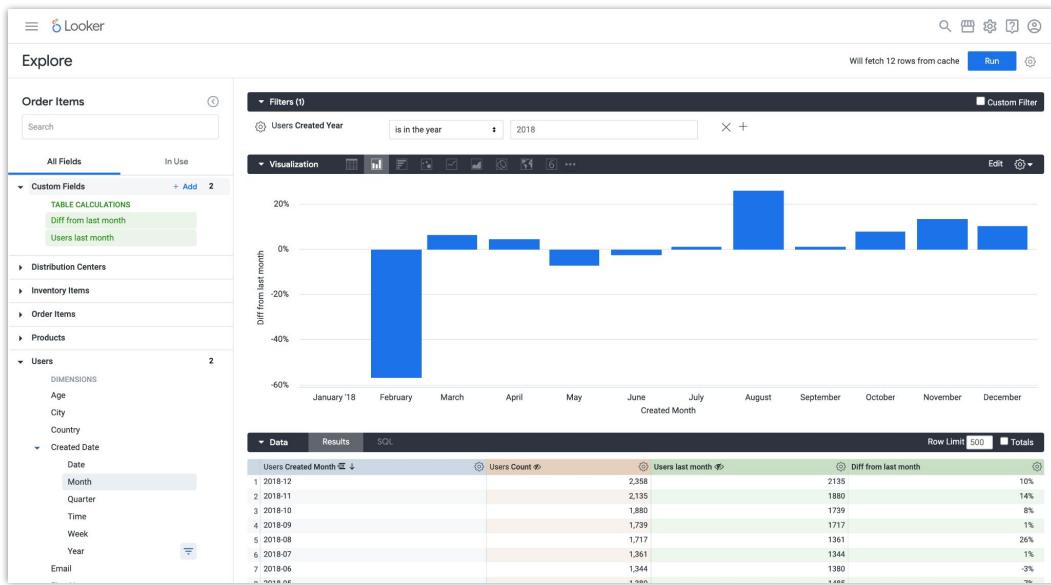
Give your table calculation a name such as **Diff from last month**, apply a Percent (0) format, and click **Save**.



Examples are based on fictional data.

Google Cloud

To finalize your visualization, hide the **Count** measure and the first table calculation for **Users last month** by clicking on the column's gear menu, and selecting **Hide from Visualization**.



Examples are based on fictional data.

Google Cloud

Now you can see why we recommend the column chart for percent change visualizations like this one. Notice that it clearly shows you which months were above or below 0, meaning which months resulted in gains or losses of users from the previous month.

And with that, your visualization of percent change in the number of users created month over month in the year 2018 is now complete.

Table calculations

04

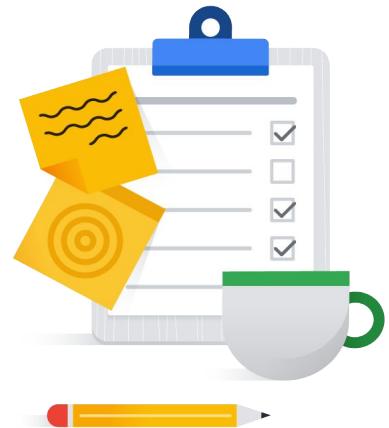
Offset Functions

05

Example 7: Writing Offset Calculations

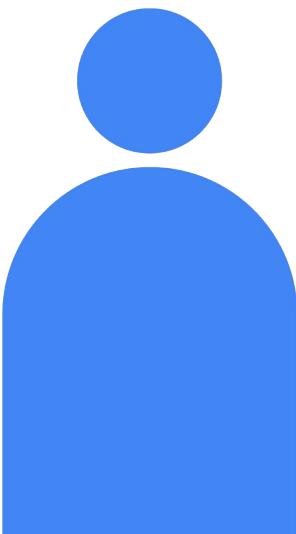
06

Lab 2: Working with Table Calculations and Offsets in Looker



Google Cloud

By now, we've spent a considerable time reviewing how you can analyze and visualize data in Looker with dimensions, measures, filters, and pivots, and how you can create your own custom metrics with table calculations and offsets.



In this next lab, you will get more hands-on practice with dimensions, measures, filters, and pivots, and learn how to implement table calculations and offsets. You will also learn how to edit dashboards, which contain multiple visualizations.

Enjoy the lab!

Summary

[Introduction to table calculations](#)

Types of table calculations

Example: Writing and visualizing table calculations

Types of offset functions

Example: Writing offset calculations

Lab: Working with table calculations and offsets in Looker

In this module, we started with introduction to table calculations, which allow you to easily and instantaneously create new metrics in Looker.

Summary

Introduction to table calculations

[Types of table calculations](#)

[Example: Writing and visualizing table calculations](#)

Types of offset functions

Example: Writing offset calculations

Lab: Working with table calculations and offsets in Looker

We then discussed the various types of table calculations that are available in Looker and reviewed an example of using table calculations to create new metrics in Looker.

Summary

Introduction to table calculations

Types of table calculations

Example: Writing and visualizing table calculations

[Types of offset functions](#)

[Example: Writing offset calculations](#)

Lab: Working with table calculations and offsets in Looker

Next, we explored offsets, which are a subset of table calculations, and reviewed an example of offset calculations in Looker.

Summary

Introduction to table calculations

Types of table calculations

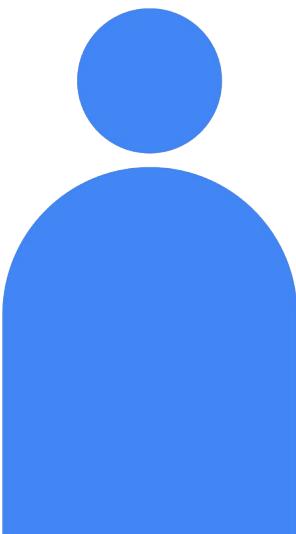
Example: Writing and visualizing table calculations

Types of offset functions

Example: Writing offset calculations

Lab: Working with table calculations and offsets in Looker

Last, you completed a hands-on lab in which you used table calculations and offsets to define new metrics in Looker.



In the next module, we will focus on creating and sharing content in Looker, specifically on Looks, which are single visualization reports, and dashboards, which contain multiple visualizations.

Hope you're excited to keep going!