

Creating a Data Transformation Pipeline with Cloud Dataprep

experiment Lab schedule 1 hour 15 minutes universal_currency_alt No cost

show_chart Intermediate



Lab instructions and tasks

expand_less

GSP430

Overview

Setup and requirements

Task 1. Open Dataprep in the Google Cloud console

Task 2. Creating a BigQuery dataset

Task 3. Connecting BigQuery data to Cloud Dataprep

Task 4. Exploring ecommerce data fields with the UI

Task 5. Cleaning the data

Task 6. Enriching the data

Task 7. Running Cloud Dataprep jobs to BigQuery

Congratulations!

This lab was developed with our partner, Alteryx. Your personal information may be shared with Alteryx, the lab sponsor, if you have opted-in to receive product updates, announcements, and offers in your Account Profile.

GSP430



Google Cloud Self-Paced Labs

Overview

Dataprep by Trifacta is an intelligent data service for visually exploring, cleaning, and preparing structured and unstructured data for analysis. In this lab, you explore the Dataprep user interface (UI) to build a data transformation pipeline that outputs results into BigQuery.

The dataset for this lab is an ecommerce dataset that has millions of Google Analytics session records for the Google Merchandise Store loaded into BigQuery. In the lab, you explore the available fields and rows and prepare the data for analysis.

What you'll do

In this lab, you learn how to perform these tasks:

- Connect BigQuery datasets to Dataprep
- Explore dataset quality with Dataprep
- Create a data transformation pipeline with Dataprep
- Run transformation jobs and send outputs to BigQuery

Setup and requirements

Note: to run this lab, you will need to use Google Chrome. Other browsers are currently not supported by Dataprep.

It is recommended that you take the Working with Cloud Dataprep on Google Cloud lab before attempting this lab.

Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

To complete this lab, you need:

- Access to a standard internet browser (Chrome browser recommended).

Note: Use an Incognito or private browser window to run this lab. This prevents any conflicts between your personal account and the Student account, which may cause extra charges incurred to your personal account.

- Time to complete the lab---remember, once you start, you cannot pause a lab.

Note: If you already have your own personal Google Cloud account or project, do not use it for this lab to avoid extra charges to your account.

How to start your lab and sign in to the Google Cloud console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:

- The **Open Google Cloud console** button
- Time remaining
- The temporary credentials that you must use for this lab
- Other information, if needed, to step through this lab

2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

Tip: Arrange the tabs in separate windows, side-by-side.

Note: If you see the **Choose an account** dialog, click **Use Another Account**.

3. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

student-00-b58d7196941b@qwiklabs.net

content_co

You can also find the **Username** in the **Lab Details** panel.

4. Click **Next**.

5. Copy the **Password** below and paste it into the **Welcome** dialog.

Aq946JtBVEK1

content_co

You can also find the **Password** in the **Lab Details** panel.

6. Click **Next**.

Important: You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.

Note: Using your own Google Cloud account for this lab may incur extra charges.

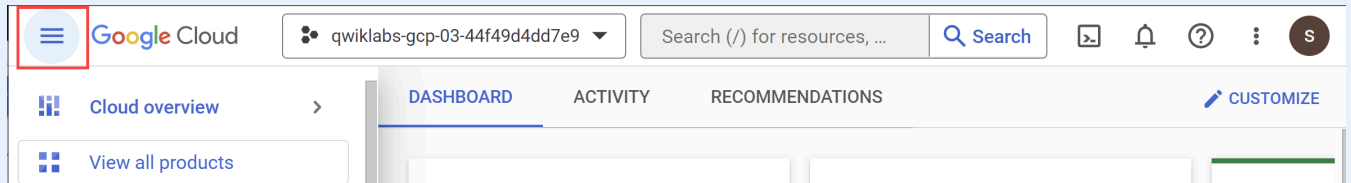
7. Click through the subsequent pages:

- Accept the terms and conditions.

- Do not add recovery options or two-factor authentication (because this is a temporary account).
- Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

Note: To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left.



Task 1. Open Dataprep in the Google Cloud console

1. Open **Cloud Shell** and run the following command:

```
gcloud beta services identity create --service=dataprep.googleapis.com content_co
```

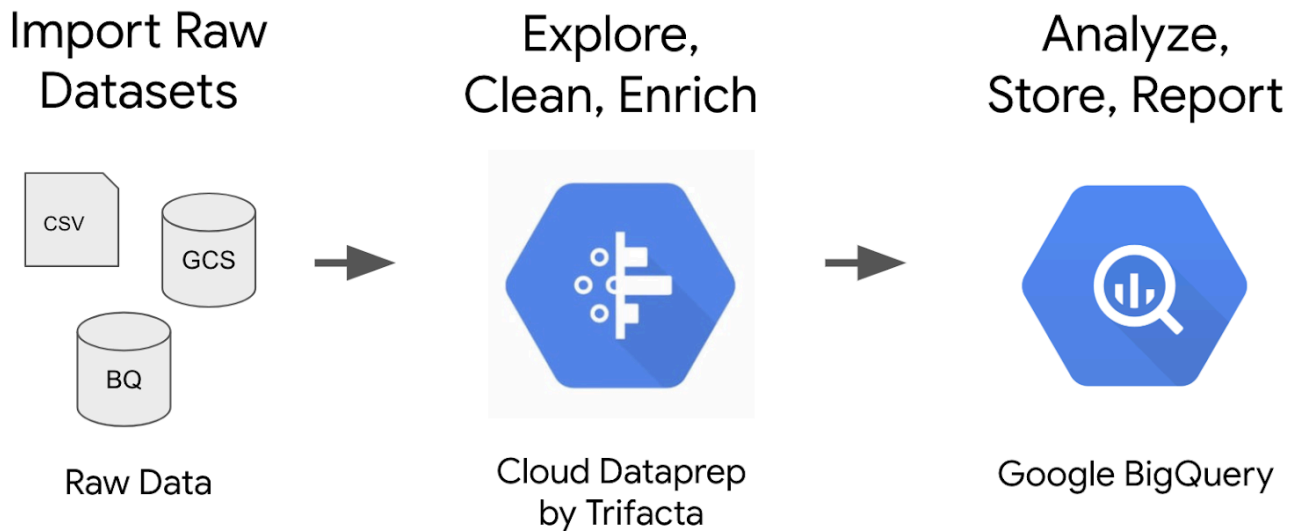
You should see a message saying the service identity was created.

2. In the Cloud console, go to the **Navigation menu**, and under **Analytics** select **Dataprep**.
3. To get into Dataprep, check that you agree to Google Dataprep Terms of Service, and then click **Accept**.
4. Click the checkbox and then click **Agree and Continue** when prompted to share account information with Alteryx.
5. Click **Allow** to give Alteryx access to your project.
6. Select your Qwiklabs credentials to sign in and click **Allow**.
7. Check the box and click **Accept** to agree to Alteryx Terms of Service.

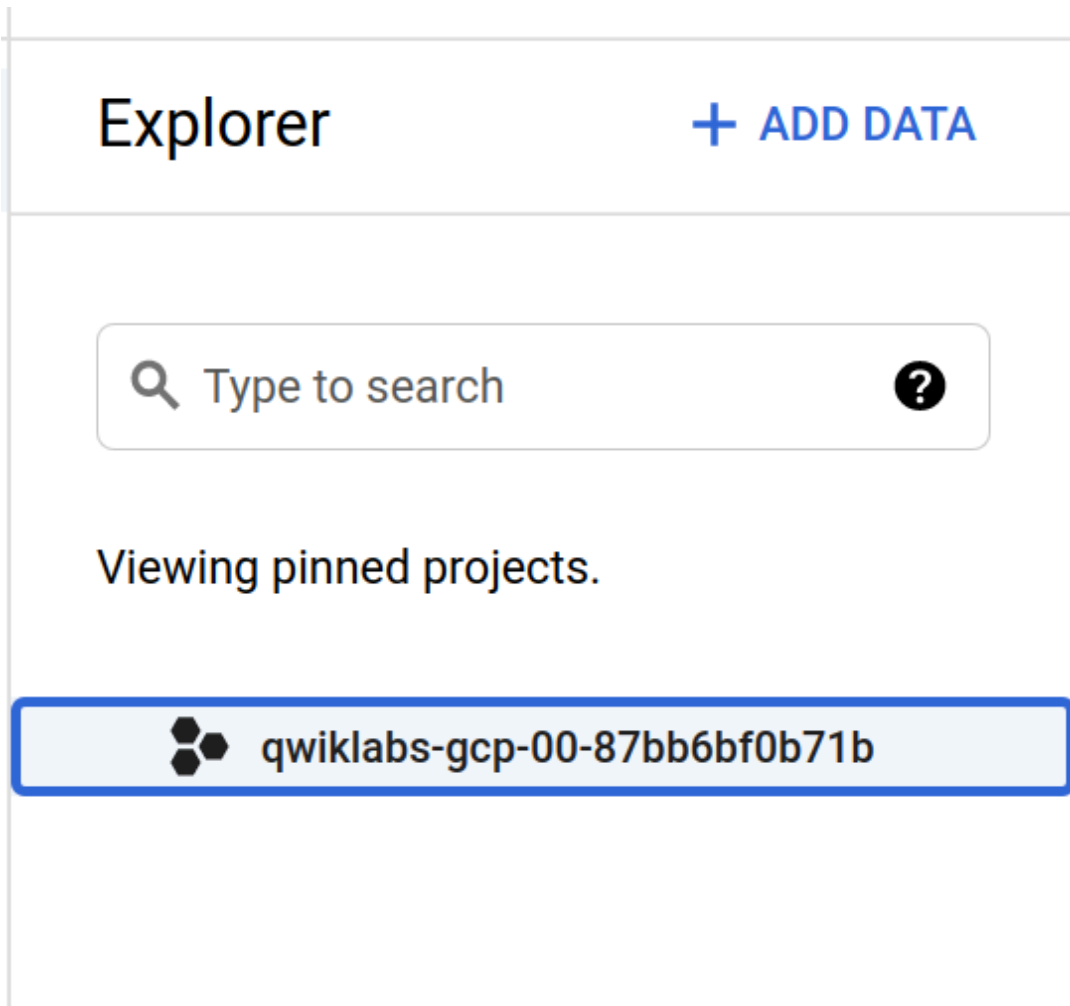
8. If prompted to use the default location for the storage bucket, click **Continue**.

Task 2. Creating a BigQuery dataset

Although this lab is largely focused on Cloud Dataprep, you need BigQuery as an endpoint for dataset ingestion to the pipeline and as a destination for the output when the pipeline is completed.



1. In the Cloud Console, select **Navigation menu** > **BigQuery**.
2. The **Welcome to BigQuery in the Cloud Console** message box opens. This message box provides a link to the quickstart guide and lists UI updates.
3. Click **Done**.
4. In the **Explorer** pane, select your project name:



5. In the left pane, under **Explorer** section, click on the **View actions** icon (⋮) to the right of your project ID, then click **Create dataset**.

- For **Dataset ID**, type `ecommerce`.
- Leave the other values at their defaults.

6. Click **CREATE DATASET**. You will now see your dataset under your project in the left pane.

7. Copy and paste the following SQL query into the Query Editor:

```
#standardSQL
CREATE OR REPLACE TABLE ecommerce.all_sessions_raw_dataprep
OPTIONS(
  description="Raw data from analyst team to ingest into Cloud
Dataprep"
) AS
SELECT * FROM `data-to-insights.ecommerce.all_sessions_raw`
WHERE date = '20170801'; # limiting to one day of data 56k rows for
this lab
```

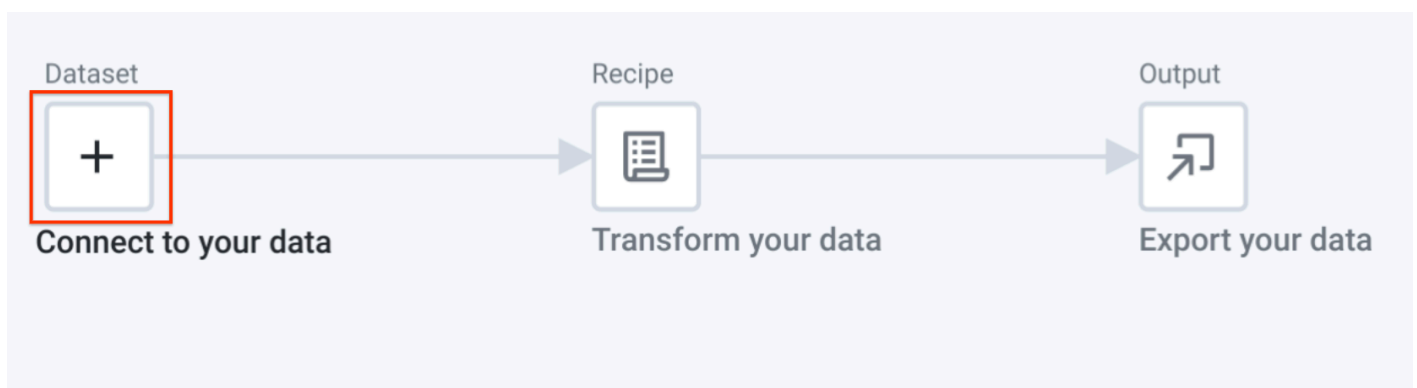
content_co

8. Click **RUN**. This query copies over a subset of the public raw ecommerce dataset (one day's worth of session data, or about 56 thousand records) into a new table named `all_sessions_raw_dataprep`, which has been added to your ecommerce dataset for you to explore and clean in Cloud Dataprep.
9. Confirm that the new table exists in your `ecommerce` dataset:

Task 3. Connecting BigQuery data to Cloud Dataprep

In this task, you will connect Cloud Dataprep to your BigQuery data source. On the Cloud Dataprep page:

1. Click **Create a flow** in the right corner.
2. Rename the **Untitled Flow** and specify these details:
 - For **Flow Name**, type `Ecommerce Analytics Pipeline`
 - For **Flow Description**, type `Revenue reporting table`
3. Click **Ok**.
4. If prompted with a `What's a flow?` popup, select **Don't show me any helpers**.
5. Click the **Add Icon** in the Dataset box.



6. In the **Add Datasets to Flow** dialog box, select **Import Datasets**.
7. In the left pane, click **BigQuery**.

8. When your **ecommerce** dataset is loaded, click on it.



9. Click on the **Create dataset** icon (+ sign) on the left of the `all_sessions_raw_dataprep` table.

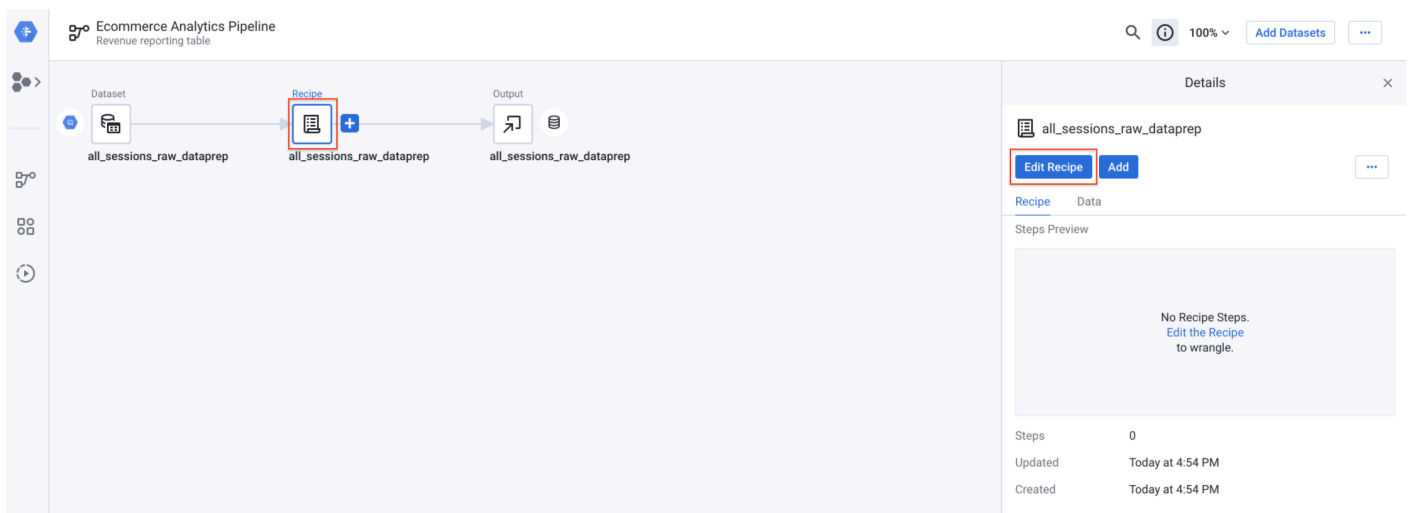
10. Click **Import & Add to Flow** in the bottom right corner.

The data source automatically updates. You are ready to go to the next task.

Task 4. Exploring ecommerce data fields with the UI

In this task, you will load and explore a sample of the dataset within Cloud Dataprep.

- Click the **Recipe icon** and then select **Edit Recipe**.

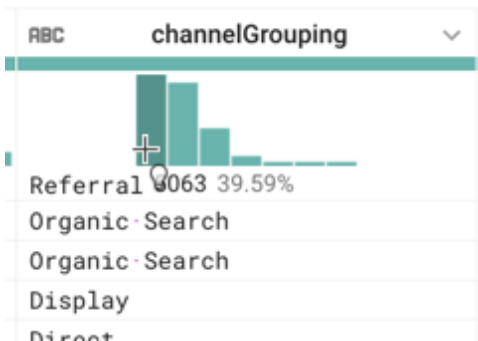


Cloud Dataprep loads a sample of your dataset into the Transformer view. This process might take a few seconds. You are now ready to start exploring the data!

Answer the following questions:

- How many columns are there in the dataset?

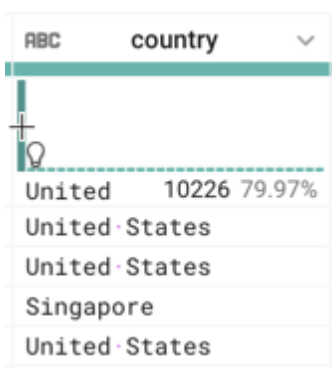
Hint: Find out by hovering your mouse cursor over the histogram under the `channelGrouping` column title.



Answer: Referral. A referring site is typically any other website that has a link to your content. An example here is a different website reviewed a product on our ecommerce website and linked to it. This is considered a different acquisition channel than if the visitor came from a search engine.

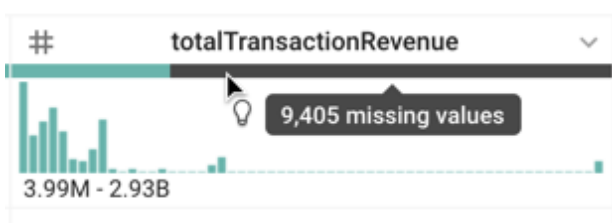
Note: When looking for a specific column, click the **Find column** icon (🔍) in the top right corner, then start typing the column's name in the **Find column** textfield, then click on the column's name. This will automatically scroll the grid to bring the column on the screen.

- What are the top three countries from which sessions are originated?




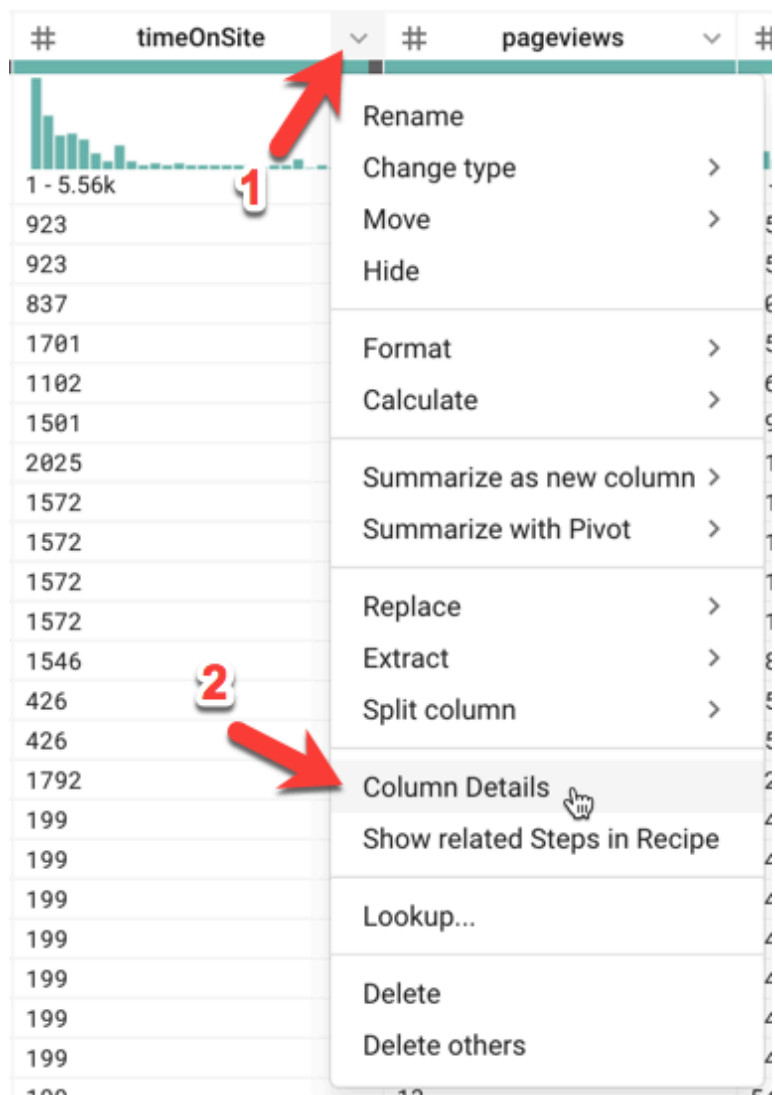
Answer: United States, India, United Kingdom

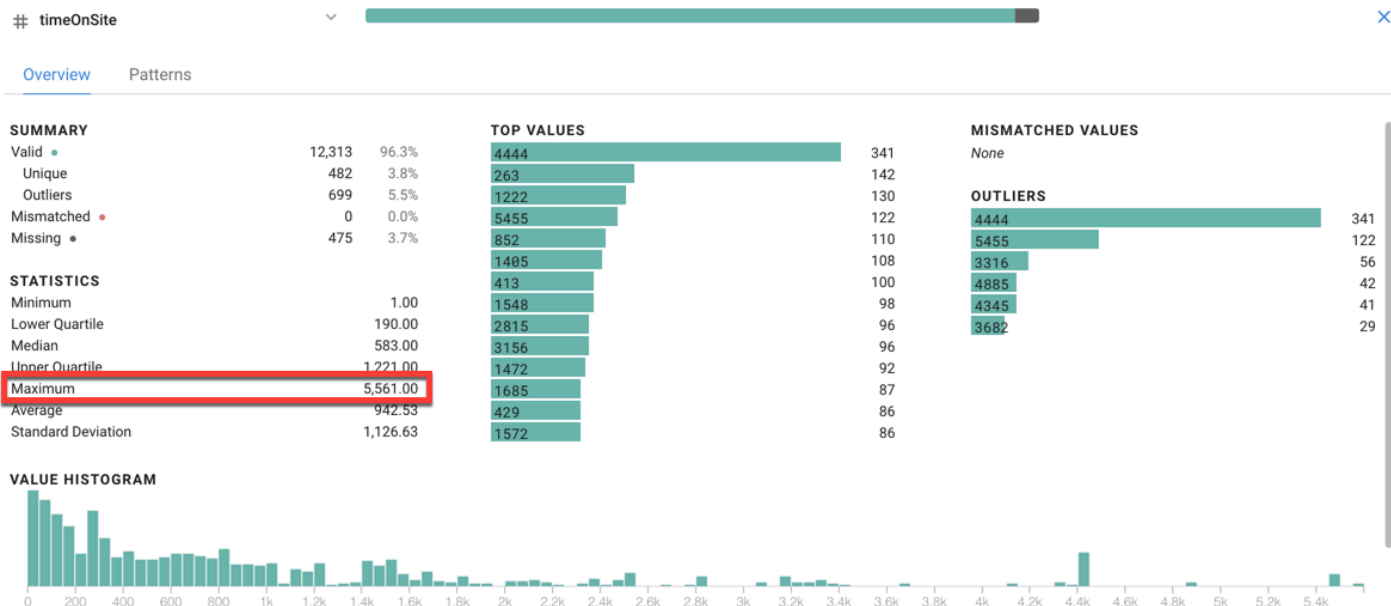
- What does the grey bar under `totalTransactionRevenue` represent?



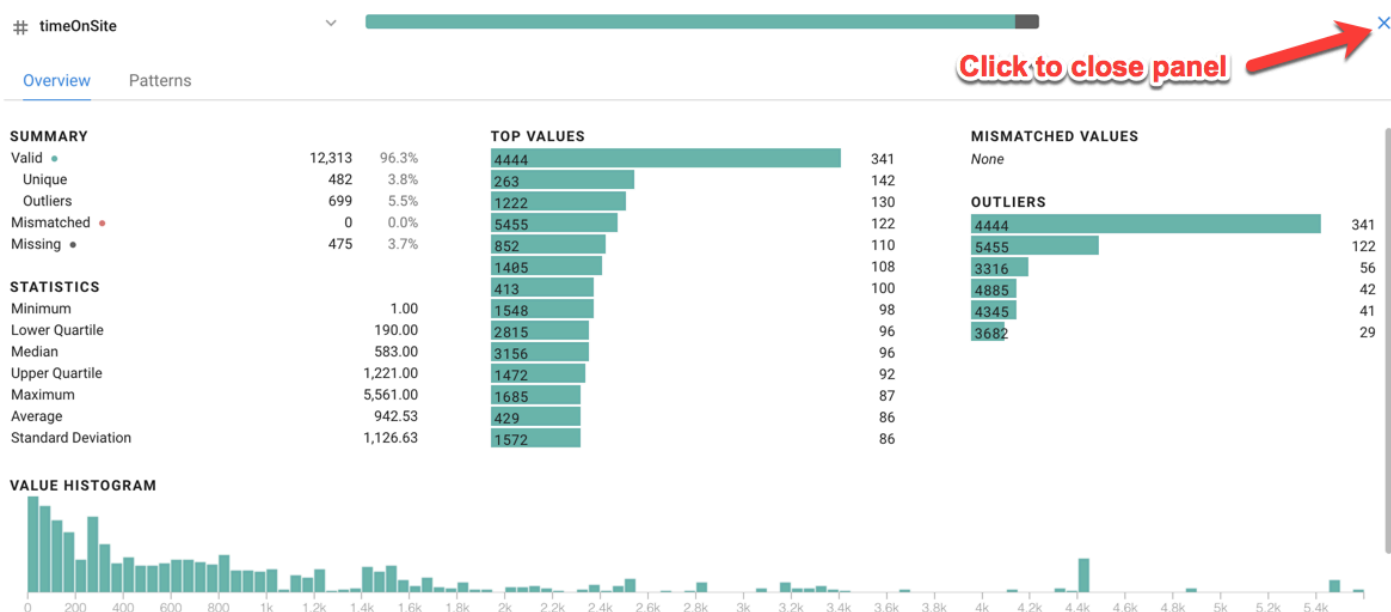
Answer: Missing values for the `totalTransactionRevenue` field. This means that a lot of sessions in this sample did not generate revenue. Later, we will filter out these values so our final table only has customer transactions and associated revenue.

- What is the maximum `timeOnSite` in seconds, maximum `pageviews`, and maximum `sessionQualityDim` for the data sample? (Hint: Open the menu to the right of the `timeOnSite` column by clicking  the **Column Details** menu)





To close the details window, click the **Close Column Details (X)** button in the top right corner. Then repeat the process to view details for the `pageviews` and `sessionQualityDim` columns.



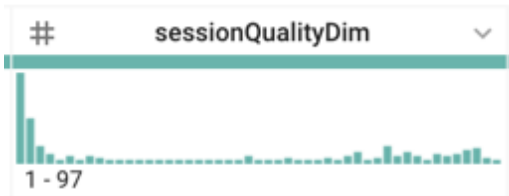
Answers:

- **Maximum Time On Site:** 5,561 seconds (or 92 minutes)
- **Maximum Pageviews:** 155 pages
- **Maximum Session Quality Dimension:** 97

Note: Your answers for maximums may vary slightly due to the data sample used by Cloud Dataprep.

Note on averages: Use extra caution when performing aggregations like averages over a column of data. We need to first ensure fields like `timeOnSite` are only counted once per session. We'll explore the uniqueness of visitor and session data in a later lab.

- Looking at the histogram for `sessionQualityDim`, are the data values evenly distributed?

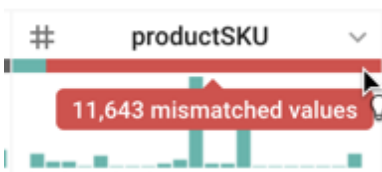


Answer: No, they are skewed to lower values (low quality sessions), which is expected.

- What is the **date** range for the dataset? Hint: Look at **date** field

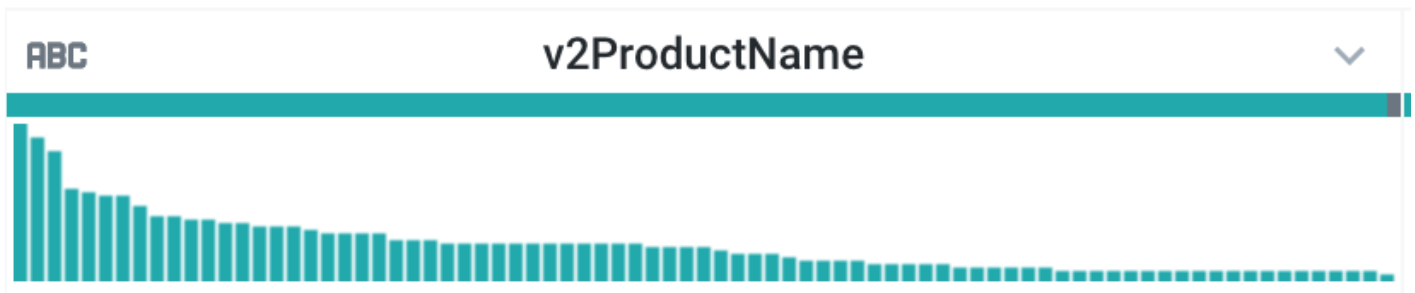
Answer: 8/1/2017 (one day of data)

- You might see a red bar under the `productSKU` column. If so, what might that mean?



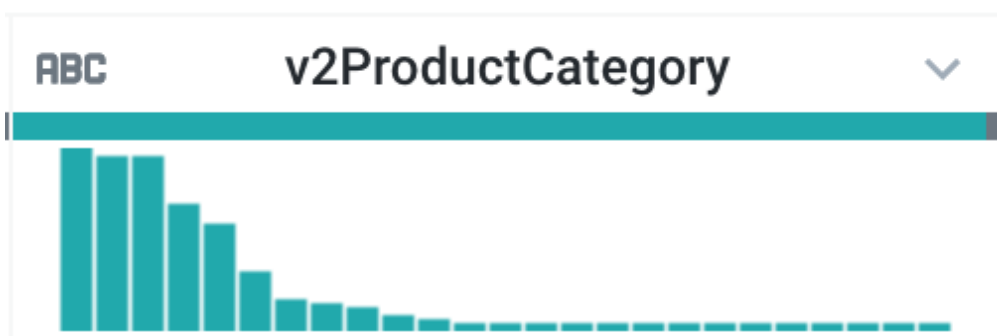
Answer: A red bar indicates mismatched values. While sampling data, Cloud Dataprep attempts to automatically identify the type of each column. If you do not see a red bar for the `productSKU` column, then this means that Cloud Dataprep correctly identified the type for the column (i.e. the String type). If you do see a red bar, then this means that Cloud Dataprep found enough number values in its sampling to determine (incorrectly) that the type should be Integer. Cloud Dataprep also detected some non-integer values and therefore flagged those values as mismatched. In fact, the `productSKU` is not always an integer (for example, a correct value might be "GGOEGOCD078399"). So in this case, Cloud Dataprep incorrectly identified the column type: it should be a string, not an integer. You will fix that later in this lab.

- Looking at the `v2ProductName` column, what are the most popular products?



Answer: Nest products

- Looking at the `v2ProductCategory` column, what are some of the most popular product categories?



Answers:

The most popular product categories are:

- Nest**
- Bags**
- (not set)** (which means that some sessions are not associated with a category)
- True or False? The most common `productVariant` is `COLOR`.

Answer: False. It's **(not set)** because most products do not have variants (80%+)

- What are the two values in the **type** column?

Answer: `PAGE` and `EVENT`

A user can have many different interaction types when browsing your website. Types include recording session data when viewing a `PAGE` or a special `EVENT` (like "clicking on a product") and other types.

Multiple hit types can be triggered at the exact same time so you will often filter on type to avoid double counting. We'll explore this more in a later analytics lab.

- What is the maximum `productQuantity`?

Answer: 100 (your answer may vary)

`productQuantity` indicates how many units of that product were added to cart. 100 means 100 units of a single product was added.

- What is the dominant `currencyCode` for transactions?

Answer: USD (United States Dollar)

- Are there valid values for `itemQuantity` or `itemRevenue`?

Answer: No, they are all `NULL` (or missing) values.

Note: After exploration, in some datasets you may find duplicative or deprecated columns. We will be using ``productQuantity`` and ``productRevenue`` fields instead and dropping the ``itemQuantity`` and ``itemRevenue`` fields later in this lab to prevent confusion for our report users.

- What percentage of `transactionId` values are valid? What does this represent for our `ecommerce` dataset?

Overview

Patterns

SUMMARY

Valid ●	582	4.6%
Unique	97	0.8%
Outliers	0	0.0%
Mismatched ●	0	0.0%
Missing ●	12,206	95.4%

STRING LENGTH STATISTICS

Minimum	15.00
Lower Quartile	15.00
Median	15.00
Upper Quartile	15.00
Maximum	15.00
Average	15.00
Standard Deviation	0.00

- Answer: About 4.6% of transaction IDs have a valid value, which represents the average conversion rate of the website (4.6% of visitors transact).
- How many `eCommerceAction_type` values are there, and what is the most common value?

Hint: Count the distinct number of histogram columns.



Answers: There are seven values found in our sample. The most common value is zero `0` which indicates that the type is unknown. This makes sense as the majority of the web sessions on our website will not perform any ecommerce actions as they are just browsing.

- Using the schema, what does `eCommerceAction_type = 6` represent?

Hint: Search for `eCommerceAction` type and read the description for the mapping


Answer: 6 maps to "Completed purchase". Later in this lab we will ingest this mapping as part of our data pipeline.

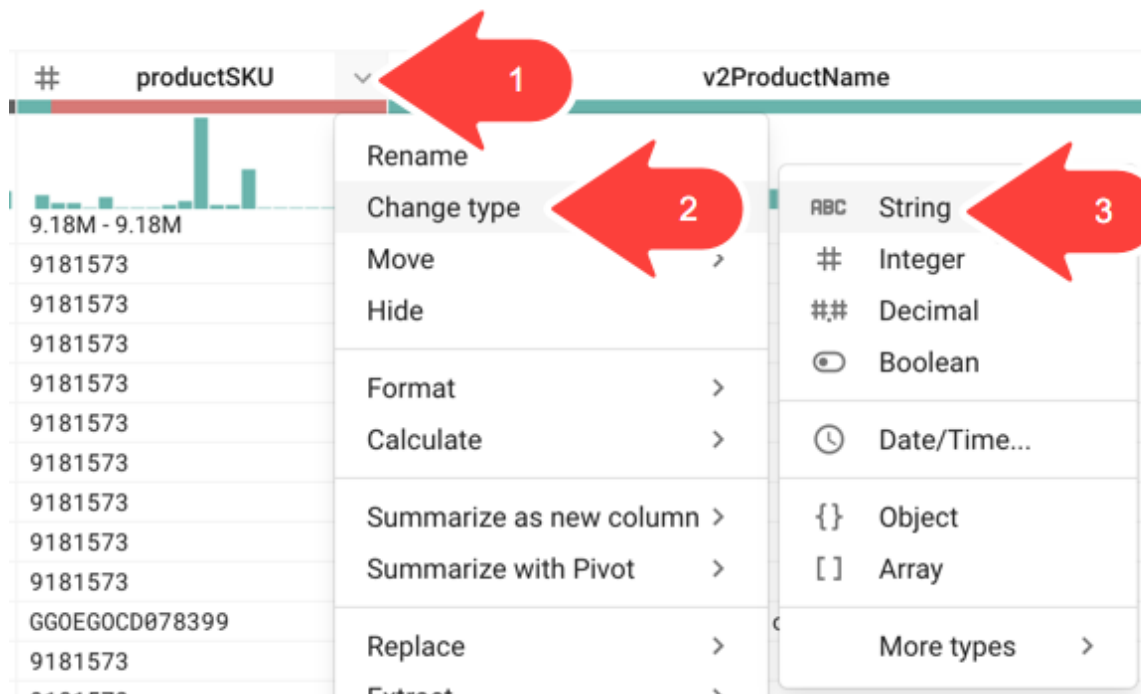
<code>commerceAction.action_type</code>	STRING	The action type. Click through of product lists = 1, Product detail views = 2, Add product(s) to cart = 3, Remove product(s) from cart = 4, Check out = 5, Completed purchase = 6, Refund of purchase = 7, Checkout options = 8, Unknown = 0.
---	--------	---

Task 5. Cleaning the data

In this task, you will clean the data by deleting unused columns, eliminating duplicates, creating calculated fields, and filtering out unwanted rows.

Converting the productSKU column data type

1. To ensure that the **productSKU** column type is a string data type, open the menu to the right of the **productSKU** column by clicking , then click **Change type > String**.

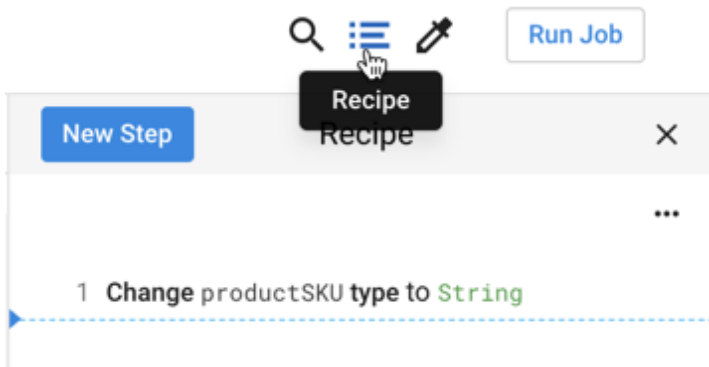


#	productSKU	v2ProductName
9.18M - 9.18M		
9181573		
9181573		
9181573		
9181573		
9181573		
9181573		
9181573		
9181573		
9181573		
GG0EG0CD078399		
9181573		
9181573		

Rename
Change type
Move
Hide
Format
Calculate
Summarize as new column
Summarize with Pivot
Replace
Extract

ABC String
Integer
Decimal
Boolean
Date/Time...
Object
Array
More types

2. Verify that the first step in your data transformation pipeline was created by clicking on the **Recipe** icon:

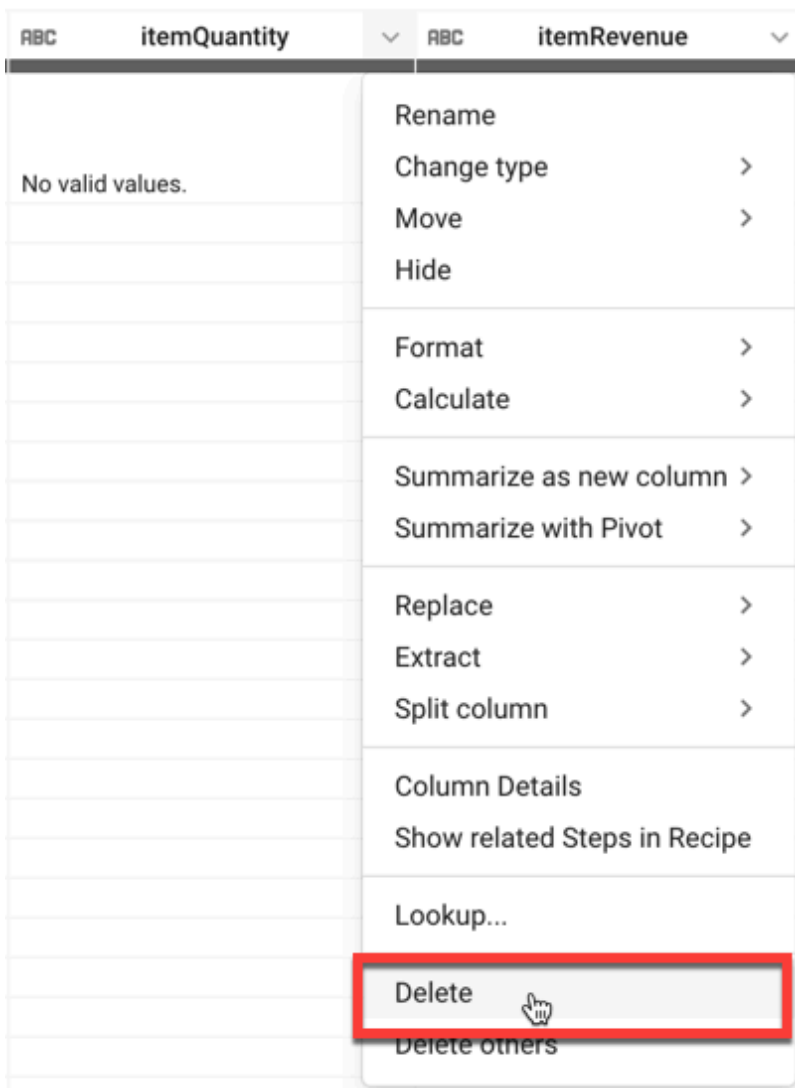


Note: If you are seeing the receipe is Locked, select the receipe, click edit and uncheck the option to lock the column type.

Deleting unused columns

As we mentioned earlier, we will be deleting the **itemQuantity** and **itemRevenue** columns as they only contain NULL values and are not useful for the purpose of this lab.

1. Open the menu for the **itemQuantity** column, and then click **Delete**.

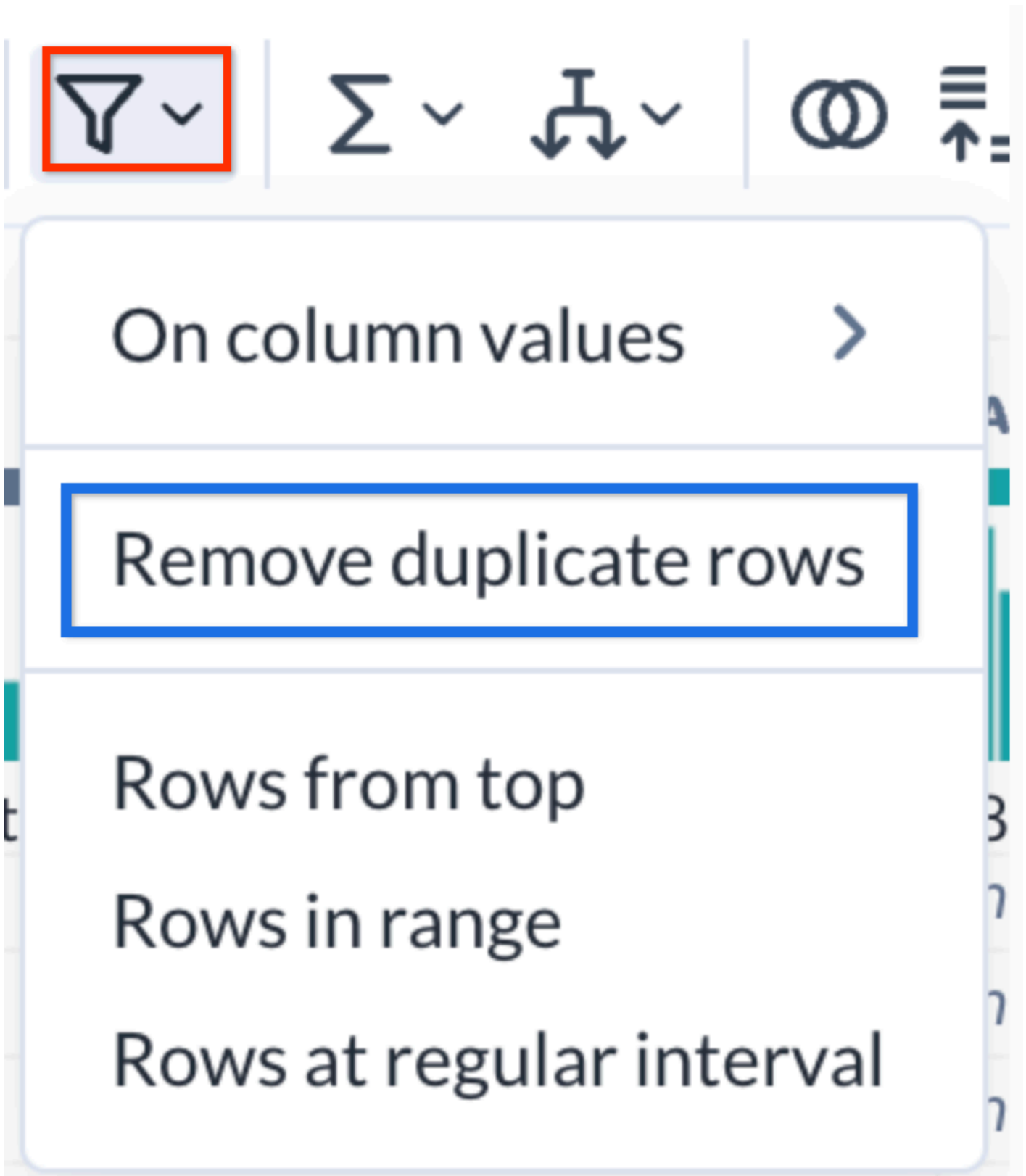


2. Repeat the process to delete the **itemRevenue** column.

Deduplicating rows

Your team has informed you there may be duplicate session values included in the source dataset. Let's remove these with a new deduplicate step.

1. Click the **Filter rows** icon in the toolbar, then click **Remove duplicate rows**.



2. Click **Add** in the right-hand panel.

3. Review the recipe that you created so far, it should resemble the following:

The dataset contains sessions of different types, for example **PAGE** (for page views) or **EVENT** (for triggered events like "viewed product categories" or "added to cart"). To avoid double counting session pageviews, add a filter to only include page view related hits.

1. In the histogram below the **type** column, click the bar for **PAGE**. All rows with the type **PAGE** are now highlighted in green.
2. In the **Suggestions** panel, in **Keep rows**, and click **Add**.

Task 6. Enriching the data

Search your schema documentation for **visitId** and read the description to determine if it is unique across all user sessions or just the user.

- **visitId**: an identifier for this session. This is part of the value usually stored as the **utmb** cookie. This is only unique to the user. For a completely unique ID, you should use a combination of **fullVisitorId** and **visitId**.

As we see, **visitId** is not unique across all users. We will need to create a unique identifier.

Creating a new column for a unique session ID

As you discovered, the dataset has no single column for a unique visitor session. Create a unique ID for each session by concatenating the **fullVisitorId** and **visitId** fields.

1. Click on the **Merge columns** icon in the toolbar.



2. For **Columns**, select **fullVisitorId** and **visitId**.
3. For **Separator** type a single hyphen character: **-**.
4. For the **New column name**, type **unique_session_id**.

Preview

ABC unique_session_id

97 Categories

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

4459051096138771020-1501639940

6439711993201552616-1501609990

6439711993201552616-1501609990

Merge columns

Columns required

Multiple

ABC fullVisitorId x

visitId x X v

Separator

-

New column name

unique_session_id

Cancel Add

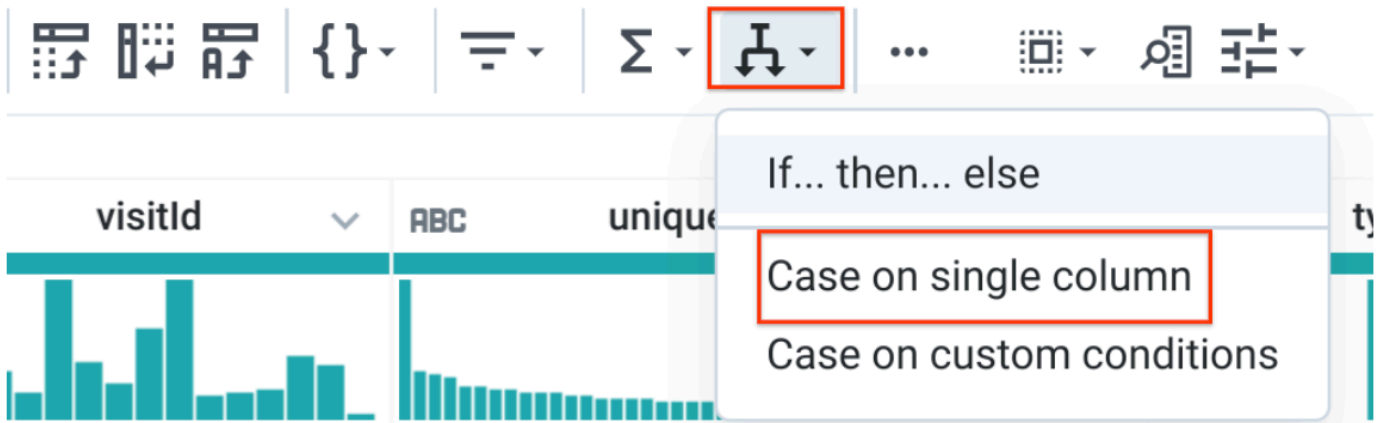
5. Click **Add**.

The `unique_session_id` is now a combination of the `fullVisitorId` and `visitId`. We will explore in a later lab whether each row in this dataset is at the unique session level (one row per user session) or something even more granular.

Creating a case statement for the ecommerce action type

As you saw earlier, values in the `eCommerceAction_type` column are integers that map to actual ecommerce actions performed in that session. For example, 3 = "Add to Cart" or 5 = "Check out". This mapping will not be immediately apparent to our end users so let's create a calculated field that brings in the value name.

1. Click on **Conditions** in the toolbar, then click **Case on single column**.



2. For **Column to evaluate**, specify `eCommerceAction_type`.

3. Next to **Cases (1)**, click **Add** 8 times for a total of 9 cases.

Conditions

×

Condition type

required

Case on single column

▼

Specify multiple conditions on a single value or formula, using the case statement

Column to evaluate

required

Select a column

▼

Cases (1)

+ Add

Comparison

Enter a value or formula

New value

Enter a value or formula

Default value

Edit formula

New column name

Insert new name

Cancel

Add

4. For each **Case**, specify the following mapping values (including the single quote characters):

Comparison	New value
------------	-----------

0	'Unknown'
1	'Click through of product lists'
2	'Product detail views'
3	'Add product(s) to cart'
4	'Remove product(s) from cart'
5	'Check out'
6	'Completed purchase'
7	'Refund of purchase'
8	'Checkout options'

Source

Preview

#	eCommerceAction_type	RBC	eCommerceAction_label
0 - 6			4 Categories
			0 Unknown
			0 Unknown
			0 Unknown
			0 Unknown
			2 Product detail views
			0 Unknown
			0 Unknown
			0 Unknown
			0 Unknown
			0 Unknown
			2 Product detail views
			0 Unknown
			0 Unknown
			0 Unknown
			2 Product detail views
			0 Unknown
			0 Unknown
			0 Unknown
			0 Unknown
			0 Unknown
			0 Unknown
			2 Product detail views
			0 Unknown
			0 Unknown

☐ Show only affected
☐ Columns

Conditions

Condition type

Case on single column

Specify multiple conditions on a single value or formula, using the case statement

Column to evaluate

eCommerceAction_type

Cases (9)

Comparison

0

New value

'Unknown'

Comparison

1

New value

'Click through of product lists'

Comparison

2

New value

'Product detail views'

Comparison

3

Cancel


Add

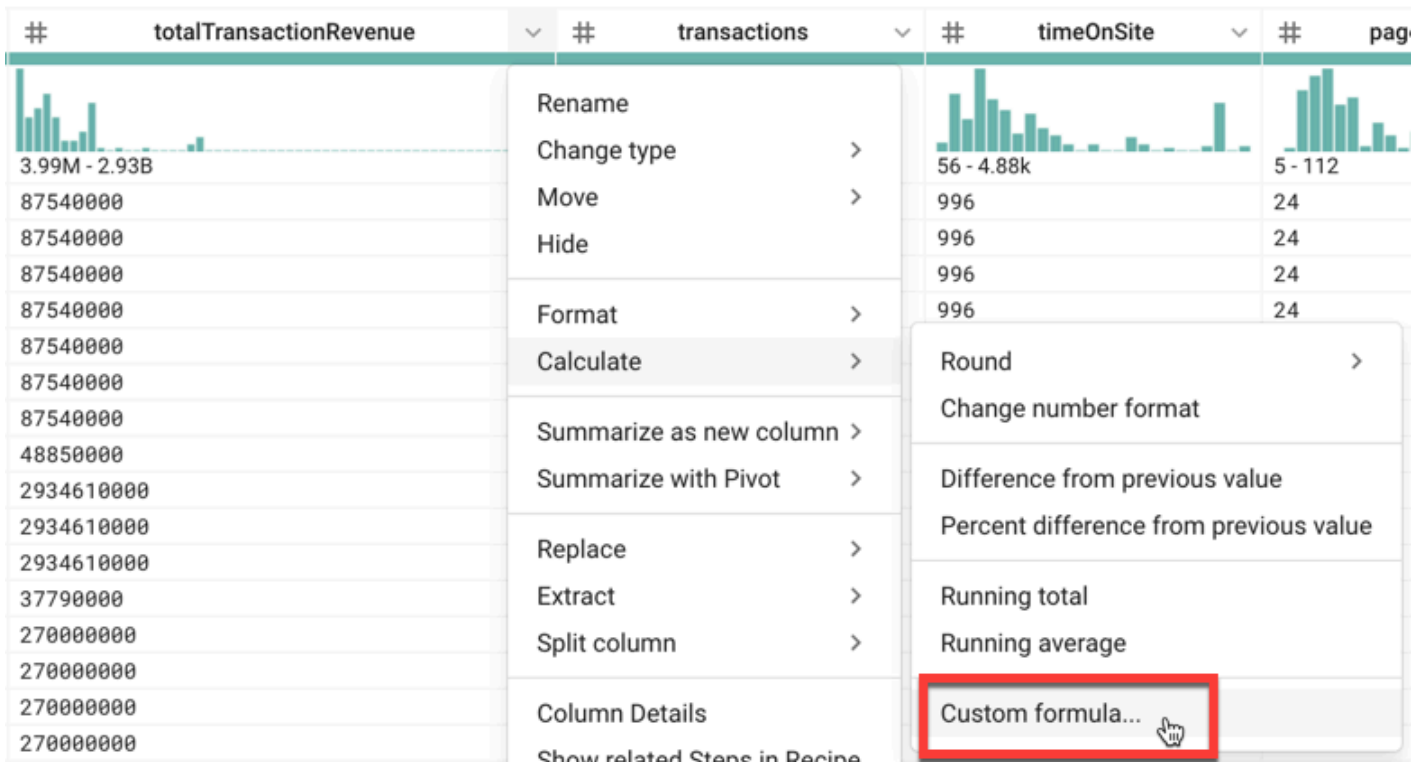
5. For **New column name**, type `eCommerceAction_label`. Leave the other fields at their default values.

6. Click **Add**.

Adjusting values in the totalTransactionRevenue column

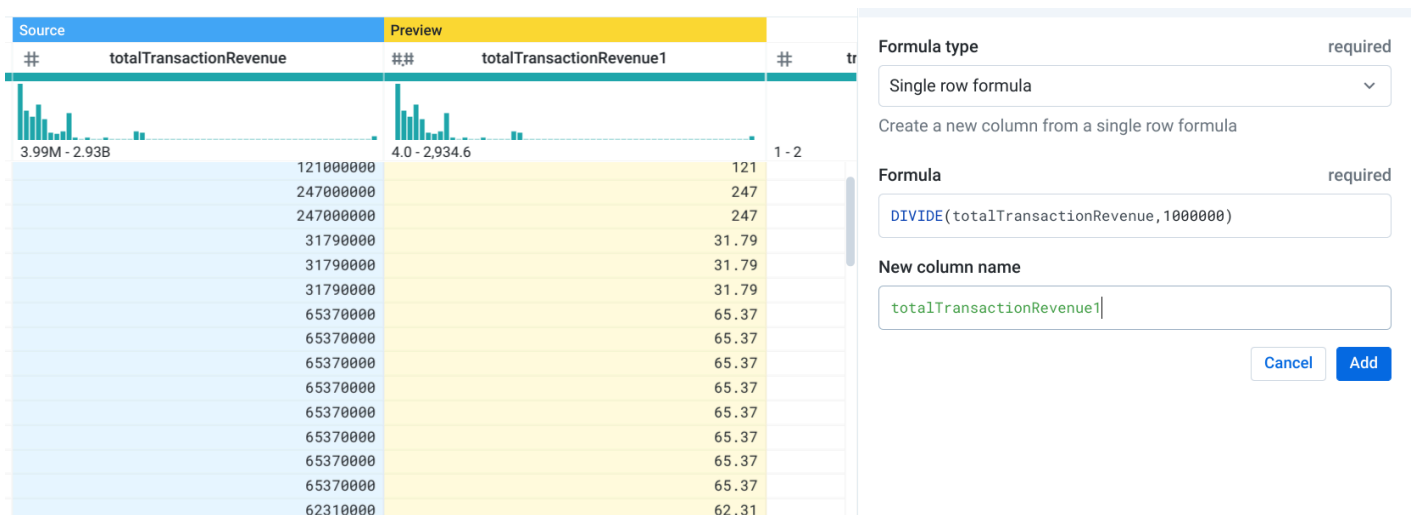
As mentioned in the schema, the **totalTransactionRevenue** column contains values passed to Analytics multiplied by 10^6 (e.g., 2.40 would be given as 2400000). You now divide the contents of that column by 10^6 to get the original values.

1. Open the menu to the right of the **totalTransactionRevenue** column by clicking , then select **Calculate > Custom formula**.




The screenshot shows a data table with columns: #, totalTransactionRevenue, #, transactions, #, timeOnSite, #, and pag. The 'totalTransactionRevenue' column is selected, and a context menu is open. The menu options are: Rename, Change type, Move, Hide, Format, Calculate, Summarize as new column, Summarize with Pivot, Replace, Extract, Split column, Column Details, and Show related Steps in Recipe. The 'Calculate' option is selected, and a sub-menu is open with options: Round, Change number format, Difference from previous value, Percent difference from previous value, Running total, Running average, and Custom formula... (highlighted with a red box and a hand cursor).

2. For **Formula**, type: `DIVIDE(totalTransactionRevenue,1000000)` and for **New column name**, type: `totalTransactionRevenue1`. Notice the preview for the transformation:



The screenshot shows the data table with the 'totalTransactionRevenue' column and a new column 'totalTransactionRevenue1'. The 'totalTransactionRevenue' column has values like 3.99M - 2.93B, 87540000, 87540000, 87540000, 87540000, 87540000, 87540000, 48850000, 2934610000, 2934610000, 2934610000, 37790000, 270000000, 270000000, 270000000, 270000000. The 'totalTransactionRevenue1' column has values like 121, 247, 247, 31.79, 31.79, 31.79, 65.37, 65.37, 65.37, 65.37, 65.37, 65.37, 65.37, 65.37, 65.37, 62.31. The formula configuration panel on the right shows the formula type set to 'Single row formula', the formula `DIVIDE(totalTransactionRevenue,1000000)`, and the new column name `totalTransactionRevenue1`. The 'Add' button is highlighted.

3. Click **Add**.

Note: You might see a red bar under the `totalTransactionRevenue1` column. Open the menu to the right of the `totalTransactionRevenue1` column by clicking , then click **Change type > Decimal**.

4. Review the full list of steps in your recipe:

New Step

Recipe

×

☐ ...

⚙️

1

Change productSKU type to String

2

Delete itemQuantity

3

Delete itemRevenue

4

Remove duplicate rows

5

Delete rows where
`ISMISSING([totalTransactionRevenue])`

6

Keep rows where type == `'PAGE'`

7

Concatenate fullVisitorId, visitId separated by `' '`

8

Create eCommerceAction_label from 9 case
conditions on eCommerceAction_type

9

Create totalTransactionRevenue1 from
`DIVIDE`(totalTransactionRevenue, 1000000)

5. You can now click **Run**.

Task 7. Running Cloud Dataprep jobs to BigQuery

1. In the **Run Job** page, select **Dataflow + Bigquery** for your **Running Environment**.

2. Under **Publishing Actions**, click on **Edit** on the right of **Create-CSV**.

3. In the following page, select **BigQuery** from the left hand menu.

4. Select your **ecommerce** dataset.

5. Click **Create a New Table** from the panel on the right.

6. Name your table **revenue_reporting**.

7. Select **Drop the Table every run**.

8. Click on **Update**.

9. Click **RUN**.

Once your Cloud Dataprep job is completed, refresh your BigQuery page and confirm that the output table **revenue_reporting** exists.

Note: If your job fails, try waiting a minute, pressing the back button on your browser, and running the job again with the same settings.

Click **Check my progress** to verify the objective.



Verify if the Cloud Dataprep jobs output the data to BigQuery

Check my progress

Assessment Completed!

Congratulations!

You've successfully explored your ecommerce dataset and created a data transformation pipeline with Cloud Dataprep.

Next steps / Learn more

- Check out Alteryx on the Google Cloud Marketplace.
- Have a Google Analytics account and want to query your own datasets in BigQuery? Follow this export guide.
- Check out these labs:
 - ETL Processing on Google Cloud Using Dataflow and BigQuery
 - Predict Visitor Purchases with a Classification Model in BigQuery ML

Google Cloud training and certification

...helps you make the most of Google Cloud technologies. Our classes include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. Certifications help you validate and prove your skill and expertise in Google Cloud technologies.

Manual Last Updated February 11, 2024

Lab Last Tested September 20, 2023

Copyright 2024 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.