

Creating a Data Transformation Pipeline with Cloud Dataprep

GSP430



Google Cloud Self-Paced Labs

Overview

[Cloud 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 Cloud Dataprep UI to build a data transformation pipeline that runs at a scheduled interval and outputs results into BigQuery.

The dataset you'll use is an [ecommerce dataset](#) that has millions of Google Analytics session records for the [Google Merchandise Store](#) loaded into BigQuery. You have a copy of that dataset for this lab and will explore the available fields and row for insights.

Objectives

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

- Connect BigQuery datasets to Cloud Dataprep.
- Explore dataset quality with Cloud Dataprep.
- Create a data transformation pipeline with Cloud Dataprep.
- Schedule transformation jobs 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 Cloud Dataprep.

It is recommended that you take the [Working with Google Cloud Dataprep](#) 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 Qwiklabs 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.

What you need

To complete this lab, you need:

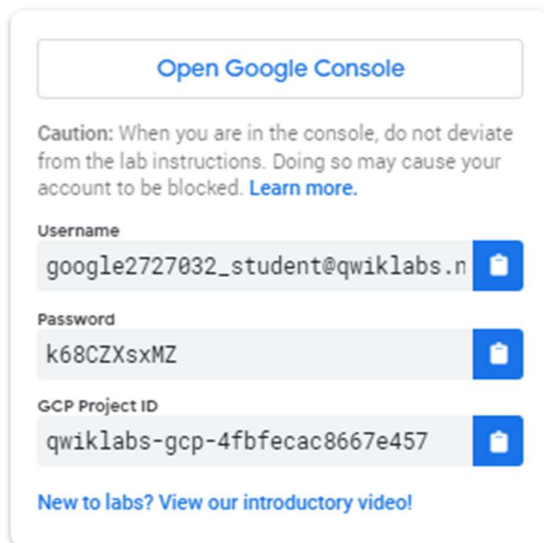
- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.

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

Note: If you are using a Pixelbook, open an Incognito window to run this lab.

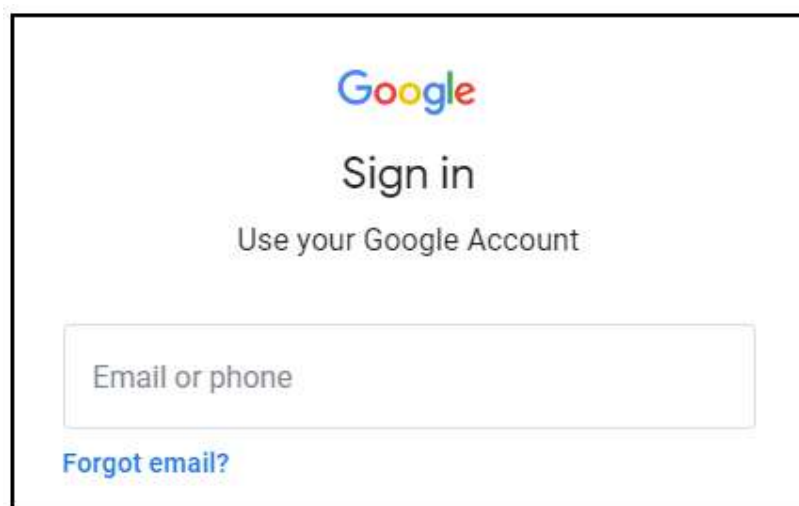
How to start your lab and sign in to the Google Cloud Console

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 a panel populated with the temporary credentials that you must use for this lab.



The screenshot shows a panel with a button at the top that says "Open Google Console". Below the button is a caution message: "Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)". Underneath the caution are three fields, each with a label and a value, and a blue copy icon to the right of each value. The first field is labeled "Username" and contains "google2727032_student@qwiklabs.n". The second field is labeled "Password" and contains "k68CZxsxMZ". The third field is labeled "GCP Project ID" and contains "qwiklabs-gcp-4fbfecac8667e457". At the bottom of the panel is a link that says "New to labs? View our introductory video!".

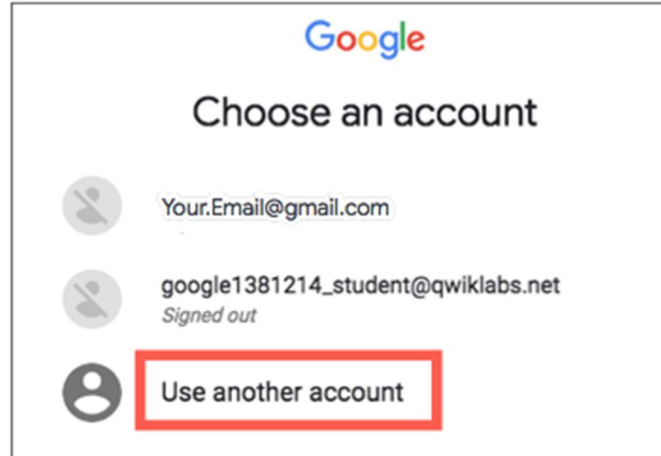
2. Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Sign in** page.



The screenshot shows the Google Sign in page. At the top is the Google logo. Below the logo is the text "Sign in" and "Use your Google Account". There is a large input field for "Email or phone". Below the input field is a link that says "Forgot email?".

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

If you see the **Choose an account** page, click **Use Another**



Account.

3. In the **Sign in** page, paste the username that you copied from the Connection Details panel. Then copy and paste the password.

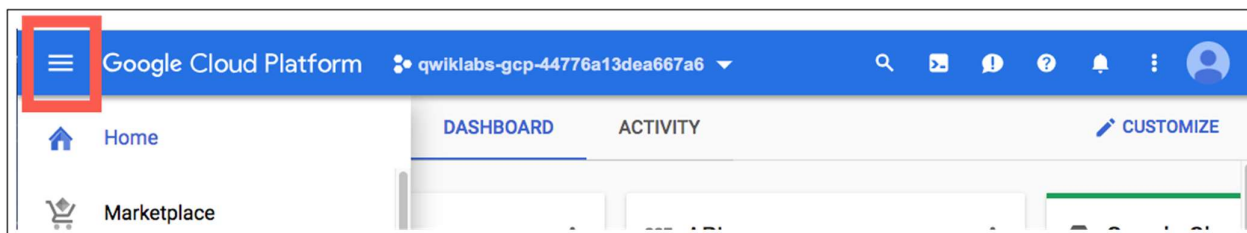
Important: You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

4. 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.

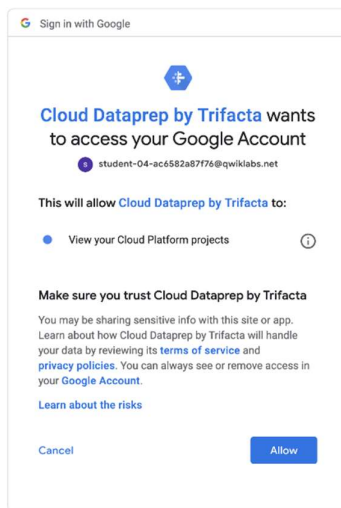
After a few moments, the Cloud Console opens in this tab.

Note: You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.

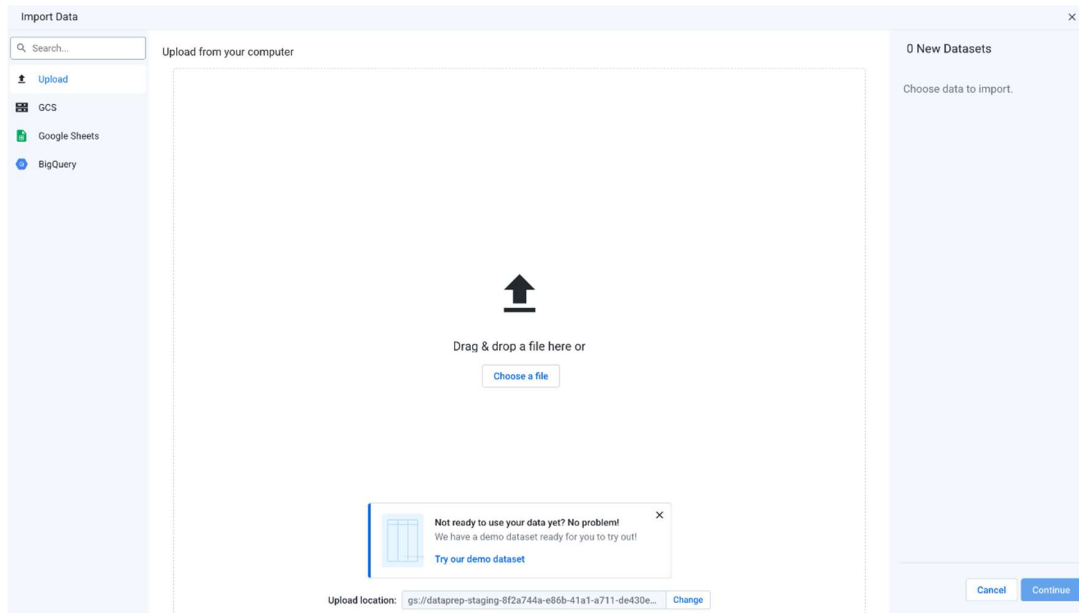


Open Google Cloud Dataprep

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



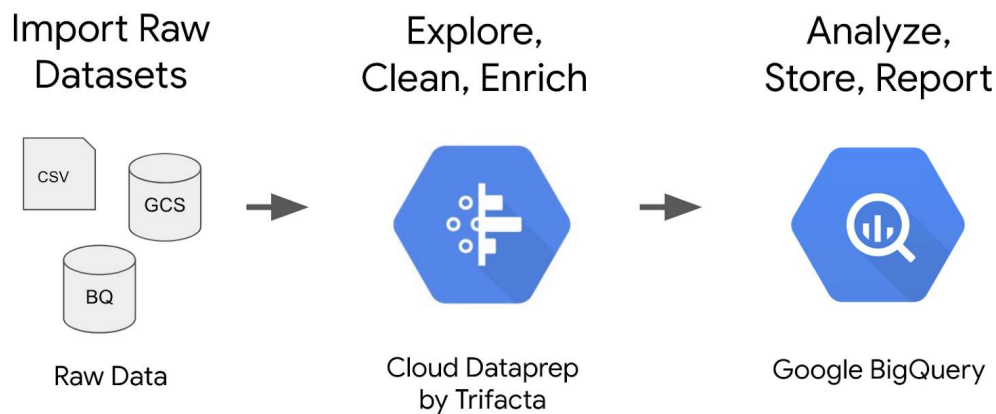
6. Check the box and click **Accept** to agree to Trifacta Terms of Service.
7. If prompted to use the default location for the storage bucket, click **Continue**.
8. For new users, a tutorial will launch, asking you to select datasets. Quit out of this screen by clicking **Cancel** or exiting out.



9. Click on the Dataprep icon on the top left corner to go to the home screen.

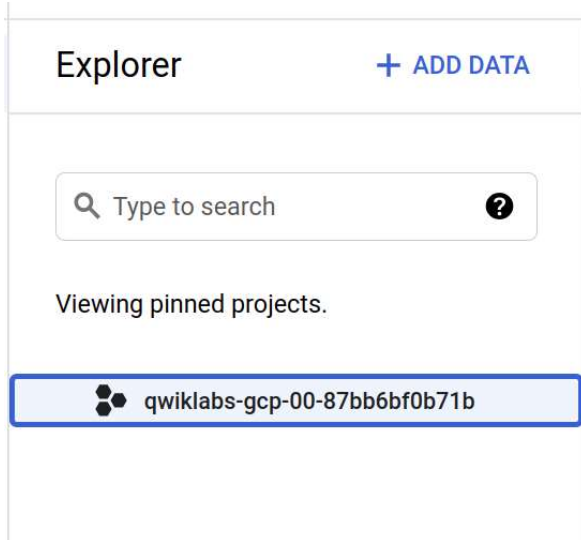
Creating a BigQuery Dataset

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.

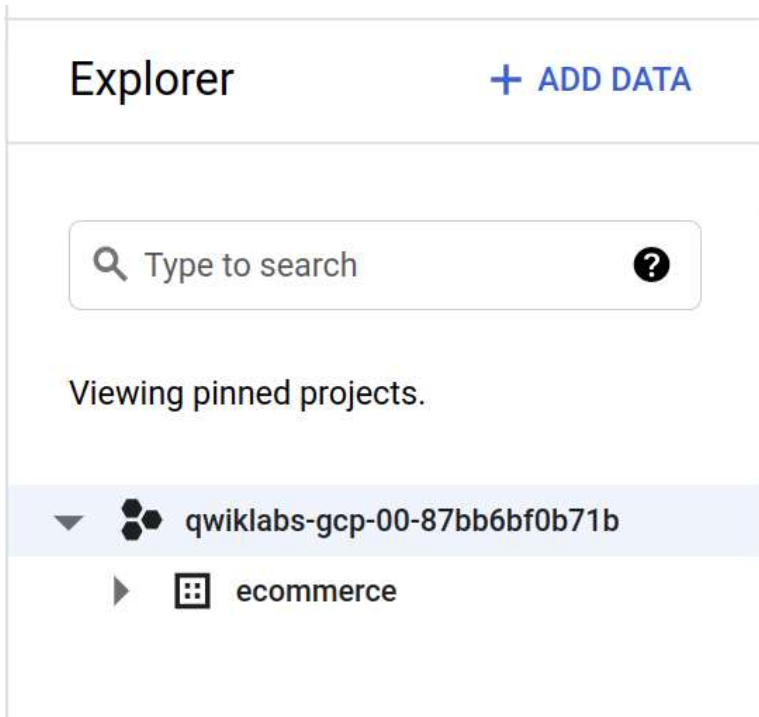


1. In the Cloud Console, select **Navigation menu > BigQuery**.

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. Then from the right-hand side of the Console, 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-hand menu:

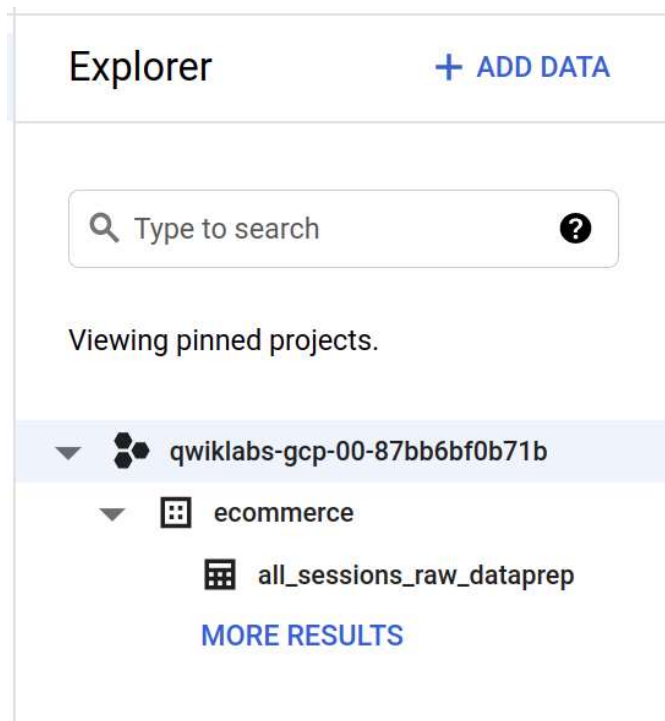


7. Navigate to the query **EDITOR** and copy and paste the following SQL query into it:

```
#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
```

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:



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 Flow** in the top-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`

Rename

×

Flow Name

Ecommerce Analytics Pipeline

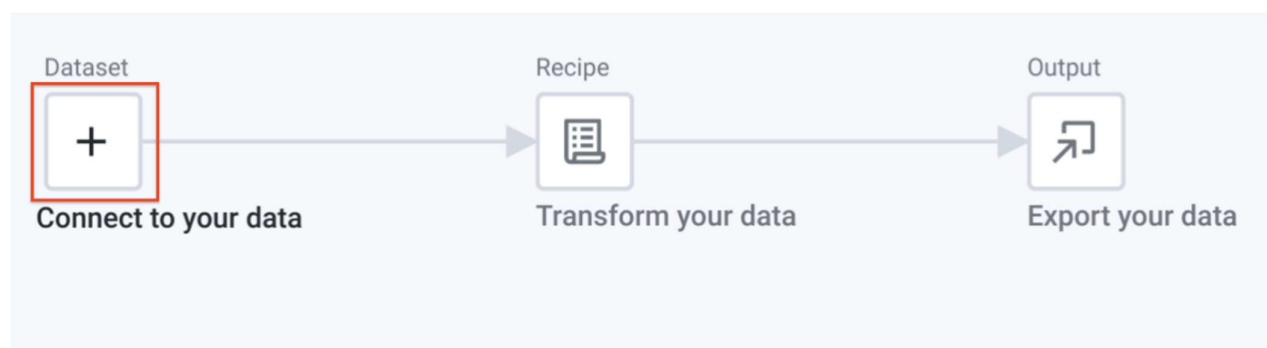
Flow Description

Revenue reporting table

Cancel

OK

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**.

Add Datasets to Flow


×

Search...

All (1)

Imported (1)

Reference (0)


	NAME	SOURCE	LAST UPDATED
<input type="checkbox"/> 	all_sessions_raw_dataprep	BIGQUERY	Today at 6:04 PM



Import Datasets

Cancel

Add

- In the left pane, click **BigQuery**.
- When your **ecommerce** dataset is loaded, click on it.

NAME 

 **ecommerce** 

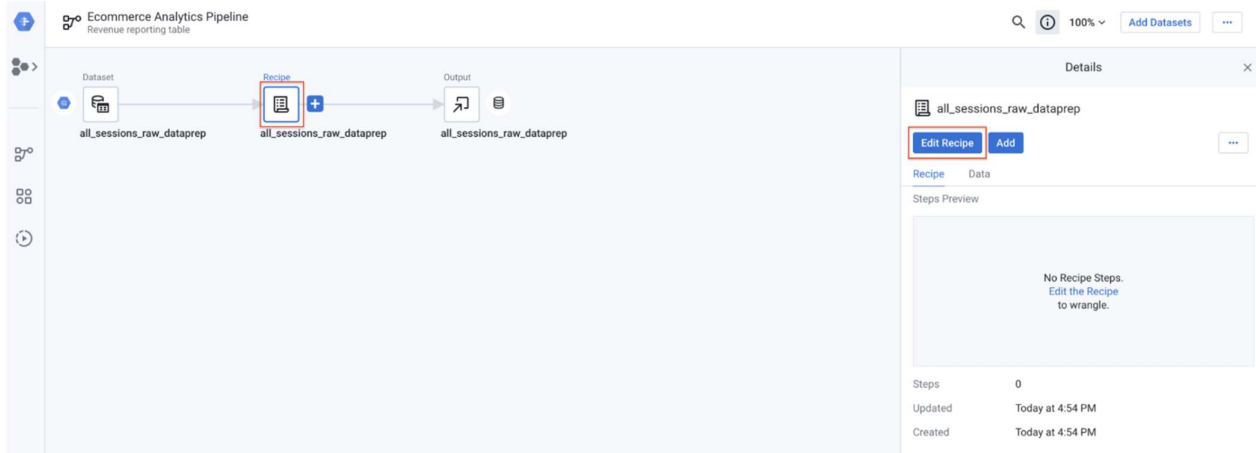
- Click on the **Create dataset** icon (+ sign) on the left of the `all_sessions_raw_dataprep` table.
- Click **Import & Add to Flow** in the bottom right corner.

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

Exploring ecommerce data fields with a UI

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

1. Click on 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:

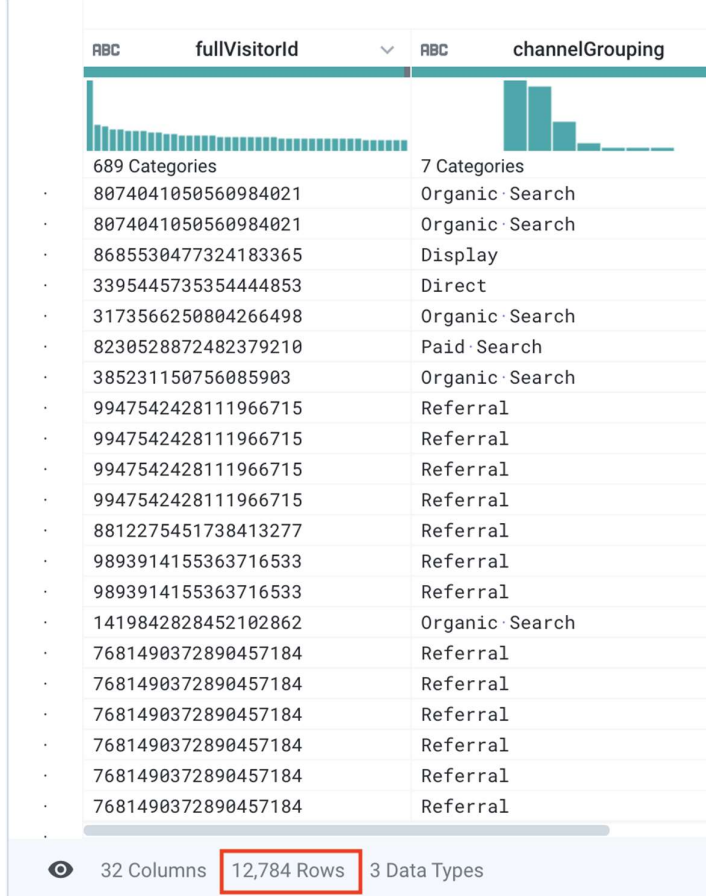
- How many columns are there in the dataset?

ABC	fullVisitorId	ABC	channelGrouping
689 Categories		7 Categories	
8074041050560984021		Organic Search	
8074041050560984021		Organic Search	
8685530477324183365		Display	
3395445735354444853		Direct	
3173566250804266498		Organic Search	
8230528872482379210		Paid Search	
385231150756085903		Organic Search	
9947542428111966715		Referral	
9947542428111966715		Referral	
9947542428111966715		Referral	
9947542428111966715		Referral	
8812275451738413277		Referral	
9893914155363716533		Referral	
9893914155363716533		Referral	
1419842828452102862		Organic Search	
7681490372890457184		Referral	
7681490372890457184		Referral	
7681490372890457184		Referral	
7681490372890457184		Referral	
7681490372890457184		Referral	
7681490372890457184		Referral	

32 Columns 12,784 Rows 3 Data Types

Answer: 32 columns.

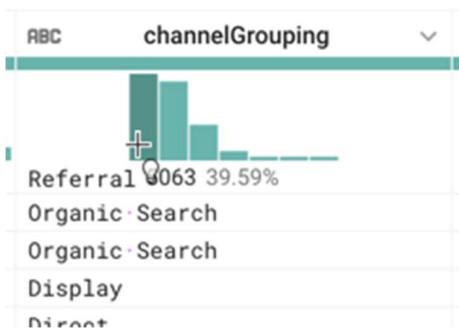
- How many rows does the sample contain?



Answer: About 12 thousand rows.

- What is the most common value in the channelGrouping column?

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.

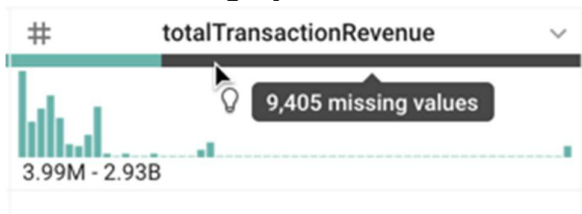
Tip: When looking for a specific column, click the **Find column** icon (🔍) in the top right corner, then start typing the column's name in the **Find column** textfield, then click on the column's name. This will automatically scroll the grid to bring the column on the screen.

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

ABC	country	
	United	10226 79.97%
	United States	
	United States	
	Singapore	
	United States	

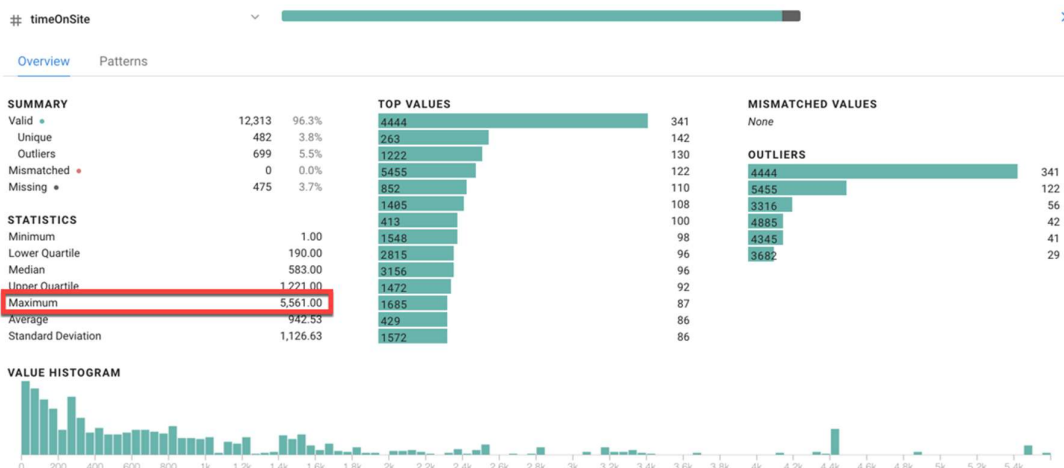
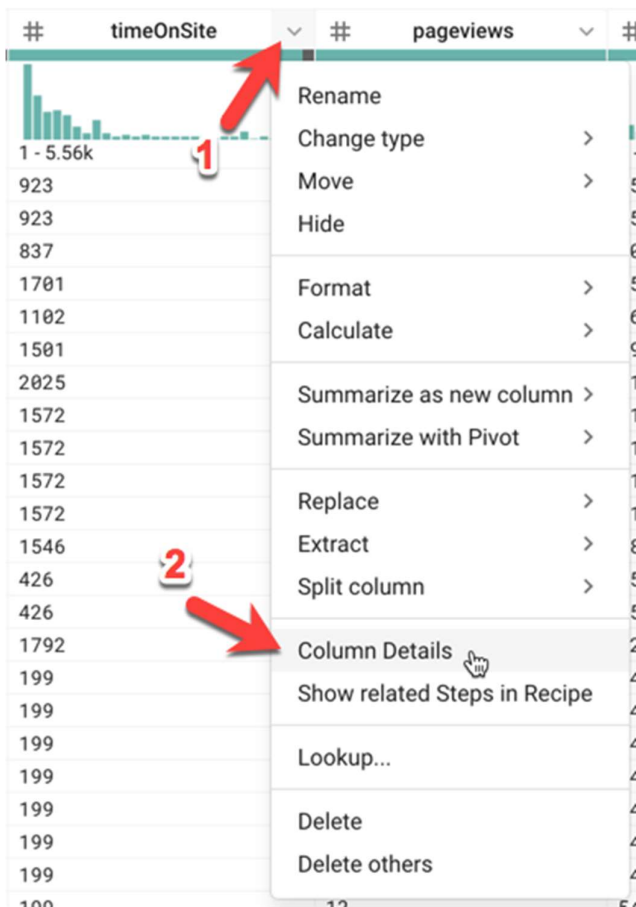
Answer: United States, India, United Kingdom

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

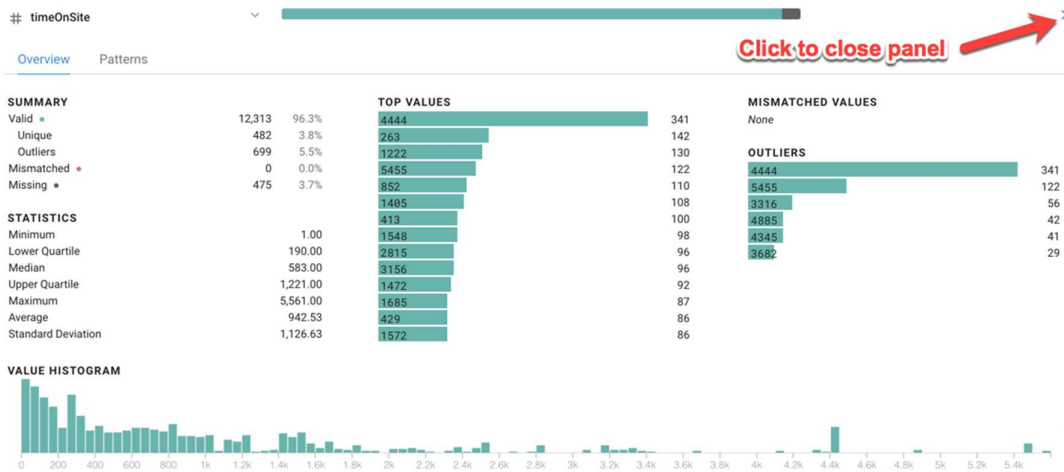


Answer: Missing values for the `totalTransactionRevenue` field. This means that a lot of sessions in this sample did not generate revenue. Later, we will filter out these values so our final table only has customer transactions and associated revenue.

- 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)

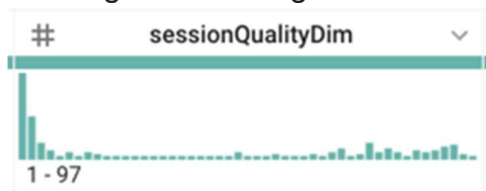


To close the details window, click the **Close Column Details** button in the top right corner. Then repeat the process to view details for the `pageviews` and `sessionQualityDim` columns.



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)

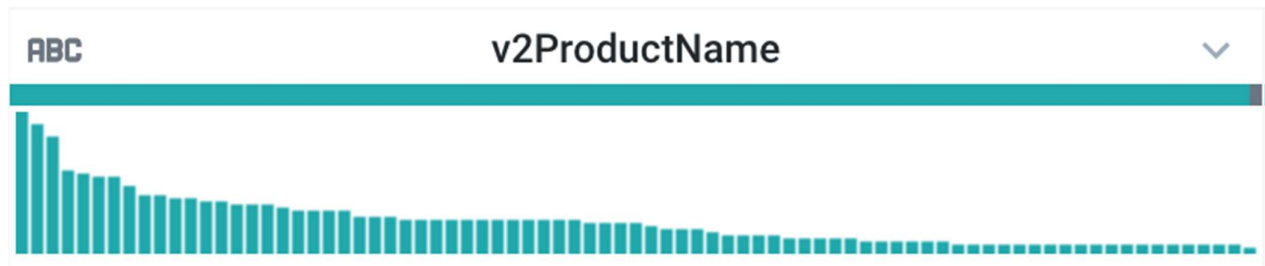
- You might see a red bar under the `productSKU` column. If so, what might that mean?



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,

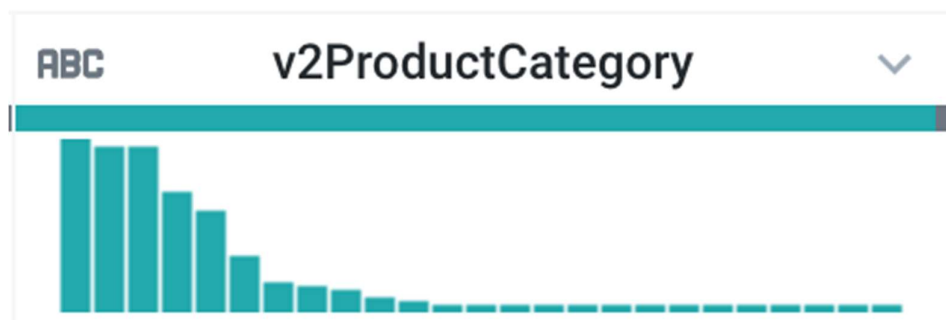
Cloud Dataprep incorrectly identified the column type: it should be a string, not an integer. You will fix that later in this lab.

- 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?

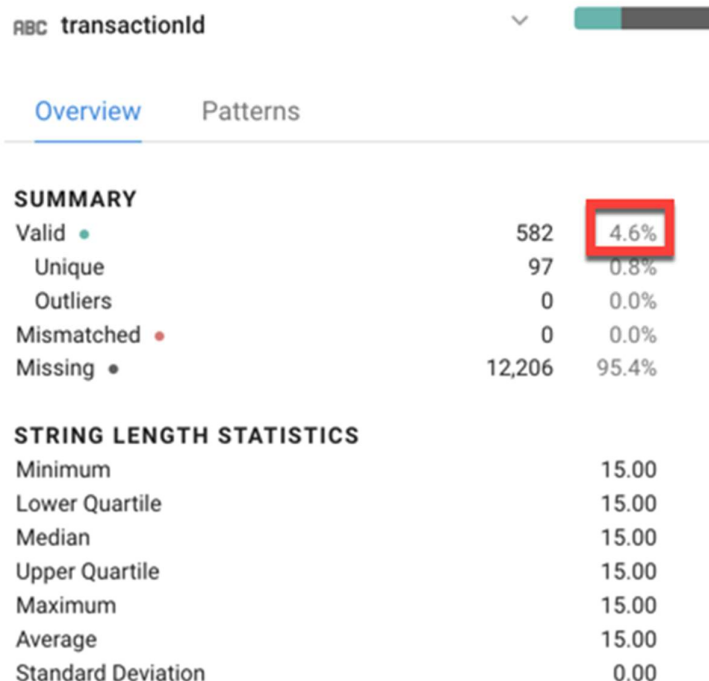
Answer: USD (United States Dollar)

- 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?



- 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.

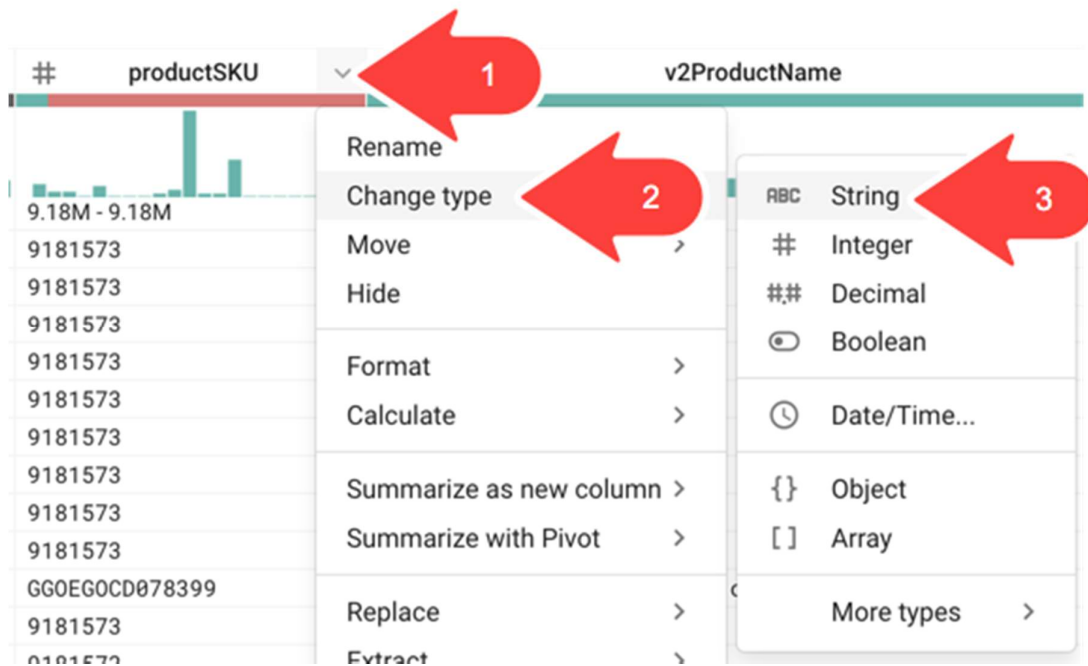
<code>commerceAction.action_type</code>	STRING	The action type. Click through of product lists = 1, Product detail views = 2, Add product(s) to cart = 3, Remove product(s) from cart = 4, Check out = 5, Completed purchase = 6, Refund of purchase = 7, Checkout options = 8, Unknown = 0.
---	--------	---

Cleaning the data

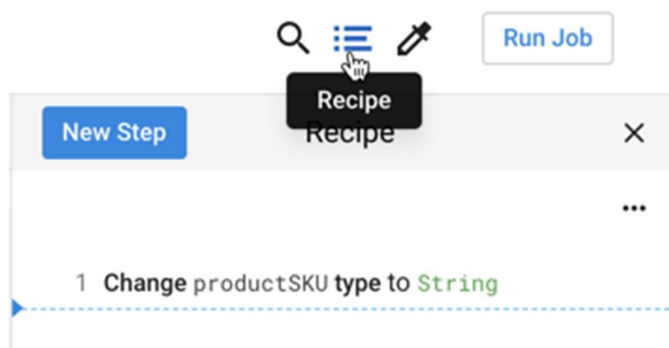
In this task, you will clean the data by deleting unused columns, eliminating duplicates, creating calculated fields, and filtering out unwanted rows.

Converting the productSKU column data type

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**.



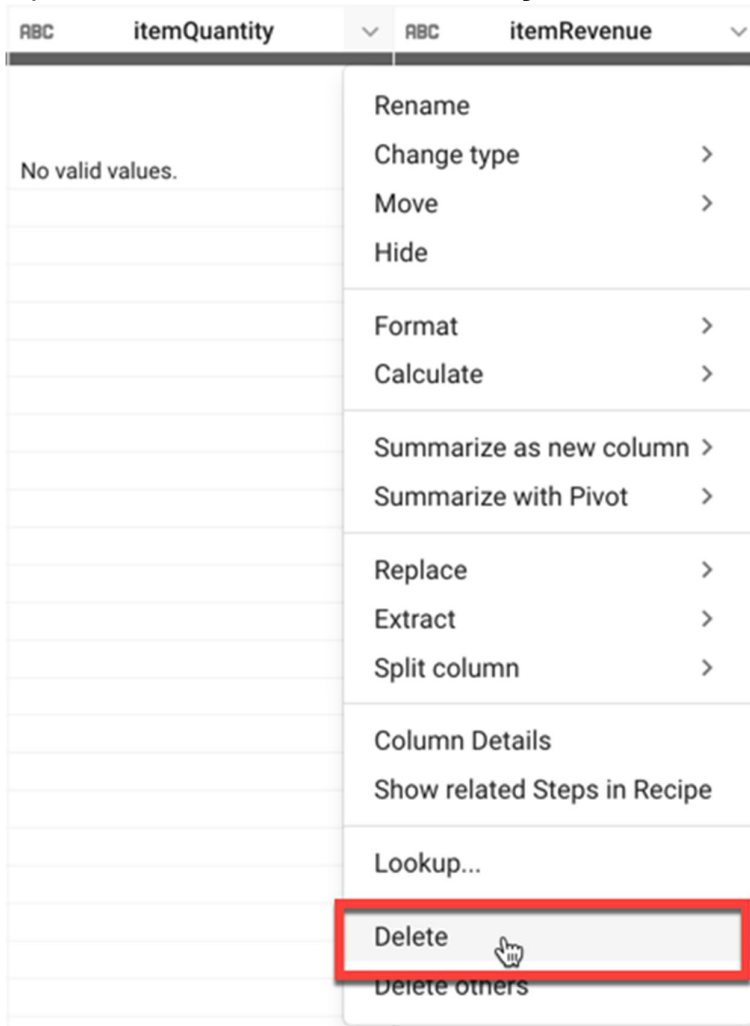
Verify that the first step in your data transformation pipeline was created by clicking on the **Recipe** icon:



Deleting unused columns

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

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

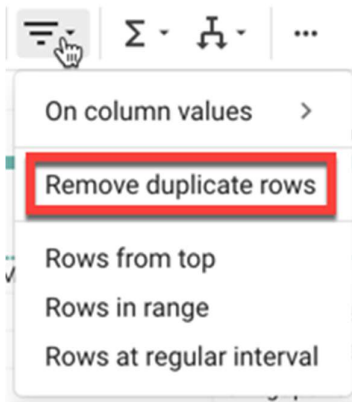


- Repeat the process to delete the **itemRevenue** column.

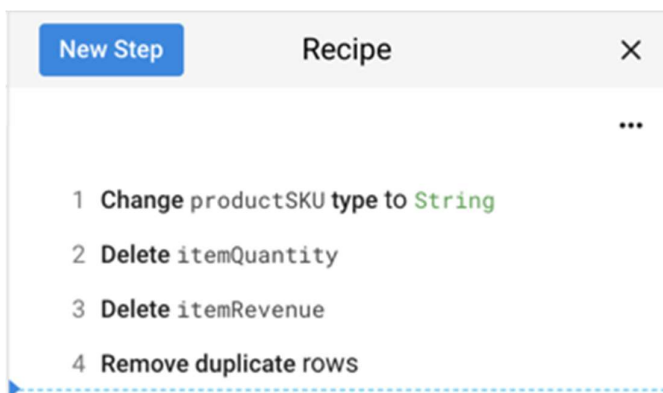
Deduplicating rows

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**.



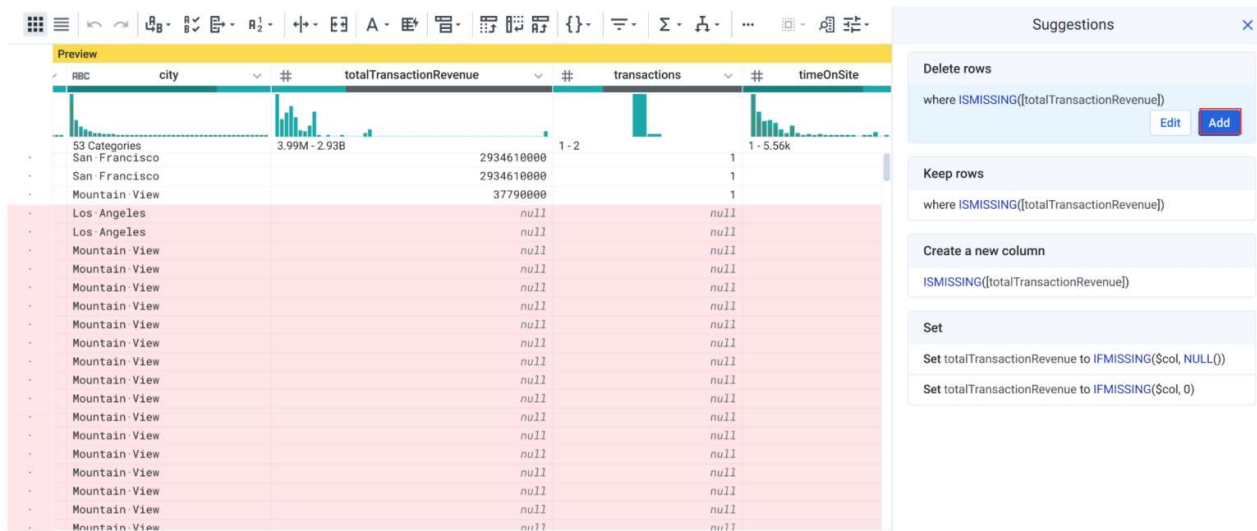
2. Click **Add** in the right-hand panel.
3. Review the recipe that you created so far, it should resemble the following:



Filtering out sessions without revenue

Your team has asked you to create a table of all user sessions that bought at least one item from the website. Filter out user sessions with NULL revenue.

1. Under the **totalTransactionRevenue** column, click the grey **Missing values** bar. All rows with a missing value for **totalTransactionRevenue** are now highlighted in red.
2. In the **Suggestions** panel, in **Delete rows** , click **Add**.

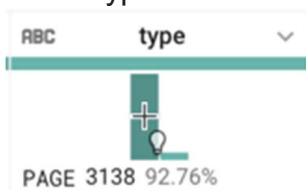


This step filters your dataset to only include transactions with revenue (where **totalTransactionRevenue** is not NULL).

Filtering sessions for PAGE views

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**.

#	sessionQualityDim	date	#	visitId	ABC	type	ABC	productRefundAu
2 - 97		Aug 1 - Aug 1	1.5B - 1.5B	1501639940	2 Categories	No valid values.		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	EVENT	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	EVENT	null		
-	88	20170801		1501639940	EVENT	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	EVENT	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		
-	88	20170801		1501639940	PAGE	null		

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.

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



- For **Columns**, select `fullVisitorId` and `visitId`.
- For **Separator** type a single hyphen character: `-`.

4. For the **New column name**, type `unique_session_id`.

The screenshot shows the 'Merge columns' dialog box. On the left, a preview of the 'unique_session_id' column is shown, displaying a list of concatenated IDs. The dialog box has the following fields:

- Columns:** A dropdown menu set to 'Multiple'. Below it, two columns are selected: 'fullVisitorId' and 'visitId', each with a red 'x' icon.
- Separator:** A text input field containing a hyphen '-'.
- New column name:** A text input field containing 'unique_session_id'.
- Buttons:** 'Cancel' and 'Add' buttons at the bottom right. The 'Add' button is highlighted with a red box.

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**.

The screenshot shows the data tool interface. The 'Conditions' menu is open, and the 'Case on single column' option is highlighted with a red box. The background shows a data table with columns 'visitId', 'ABC', and 'unique'.

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

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

< Recipe Conditions X

Condition type required
Case on single column

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

Column to evaluate required
Choose column

Cases (9) Add

Value to compare
New value Remove

Value to compare
New value Remove

Value to compare
New value Remove

Cancel Add

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

Value to compare	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	eCommerceAction_label
0-6		4 Categories
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
0		Unknown
2		Product detail views
0		Unknown
2		Product detail views
0		Unknown
0		Unknown
0		Unknown
2		Product detail views
0		Unknown
0		Unknown
0		Unknown
0		Unknown

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)

- 0: 'Unknown'
- 1: 'Click through of product lists'
- 2: 'Product detail views'

Show only affected ☐ Columns

Cancel Add

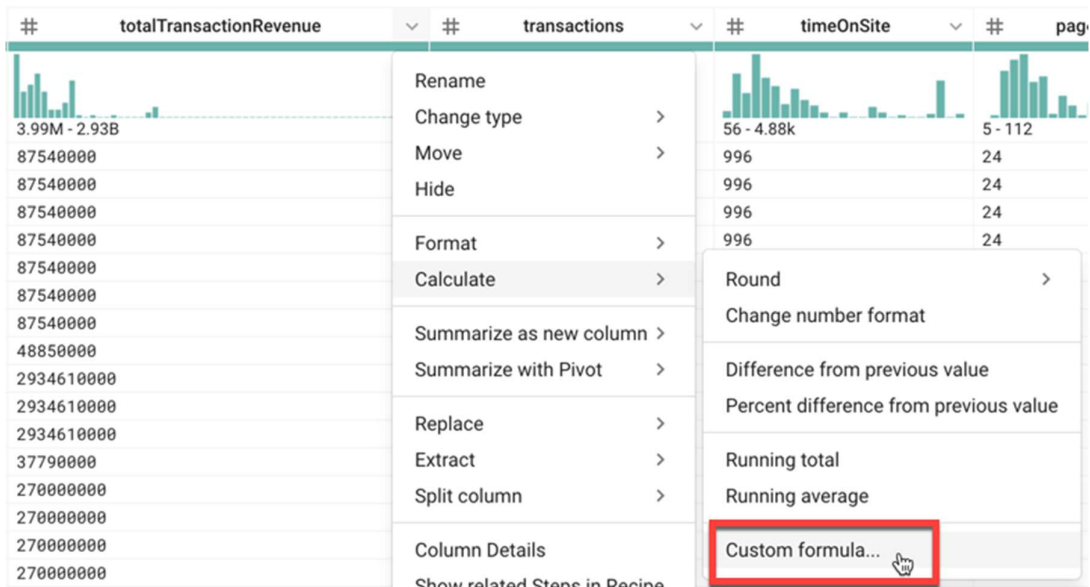
5. For **New column name**, type `eCommerceAction_label`. Leave the other fields at their default values.

6. Click **Add**.

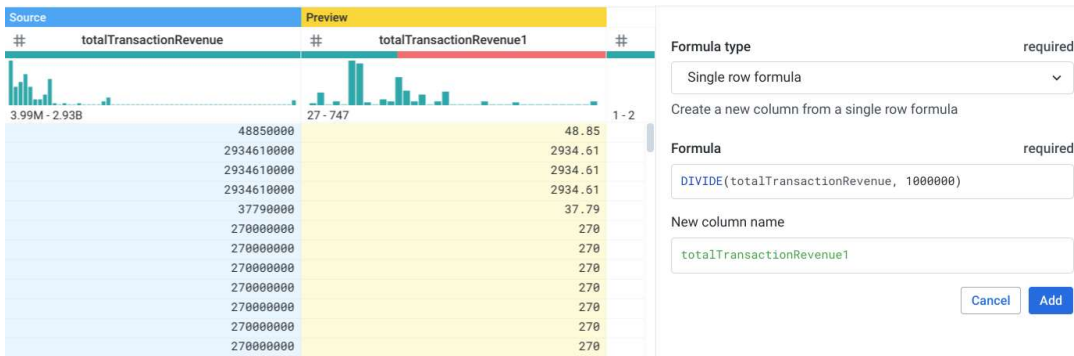
Adjusting values in the totalTransactionRevenue column


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 contents of that column by 10^6 to get the original values.

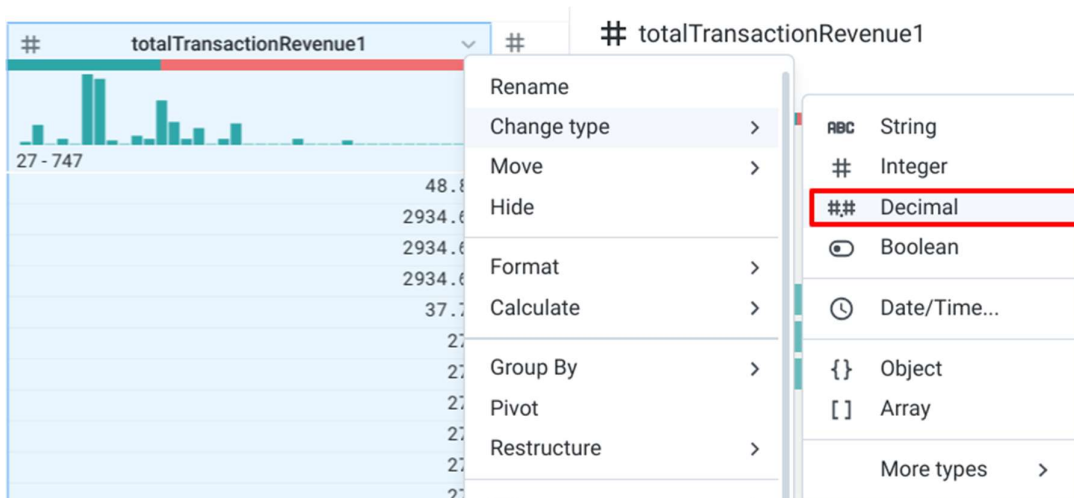
1. Open the menu to the right of the **totalTransactionRevenue** column by clicking , then select **Calculate > Custom formula**.



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



- Click **Add**.
- To convert the new `totalTransactionRevenue1` column's type to a decimal data type, open the menu to the right of the `totalTransactionRevenue1` column by clicking , then click **Change type > Decimal**.



5. 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)

10

Change totalTransactionRevenue1 type to
Decimal

6. You can now click **Run**.

Running and scheduling Cloud Dataprep jobs to BigQuery

Challenge: Now that you are satisfied with the flow, it's time to execute the transformation recipe against your source dataset. The challenge for you is to load the output of the job into the BigQuery dataset that you created earlier. Make sure you load the output into a separate table and name it `revenue_reporting`. Once your Cloud Dataprep job is completed, refresh your BigQuery page and confirm that the output table **revenue_reporting** exists.

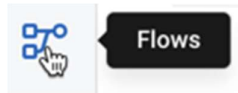
Click *Check my progress* to verify the objective.


Verify if the Cloud Dataprep jobs output the data to BigQuery

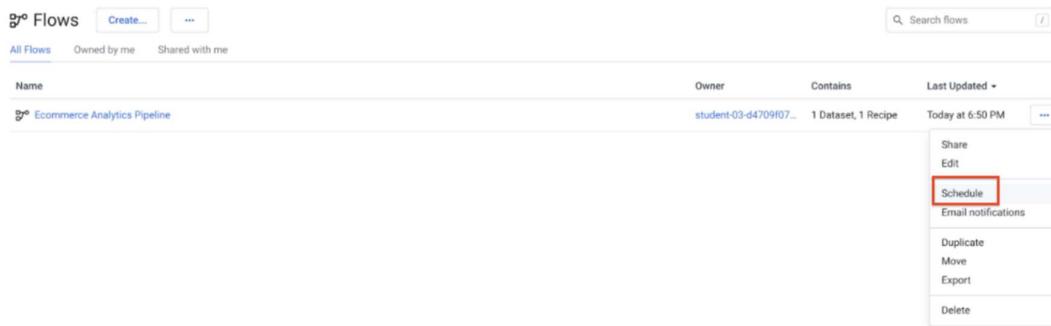
Check my progress

While it's running, you can also schedule the execution of pipeline in the next step so the job can be re-run automatically on a regular basis to account for newer data. Note: You can navigate and perform other operations while jobs are running.

1. You will now schedule a recurrent job execution. Click the **Flows** icon on the left of the screen.



2. On the right of your **Ecommerce Analytics Pipeline** flow click the **More** icon (), then click **Schedule**.



3. In the **Add schedule** dialog:

- For **Frequency**, select **Weekly**.
- For day of week, select **Saturday** and unselect **Sunday**.
- For time, enter **3:00** and select **AM**.

4. Click **Save**.

Add Schedule

Scheduling Options

Timezone

America/Los_Angeles

Frequency

Weekly

on

Saturday

at

3:00

AM

Cancel

Save

The job is now scheduled to trigger every Saturday at 3AM. You can see the schedule on the top panel.

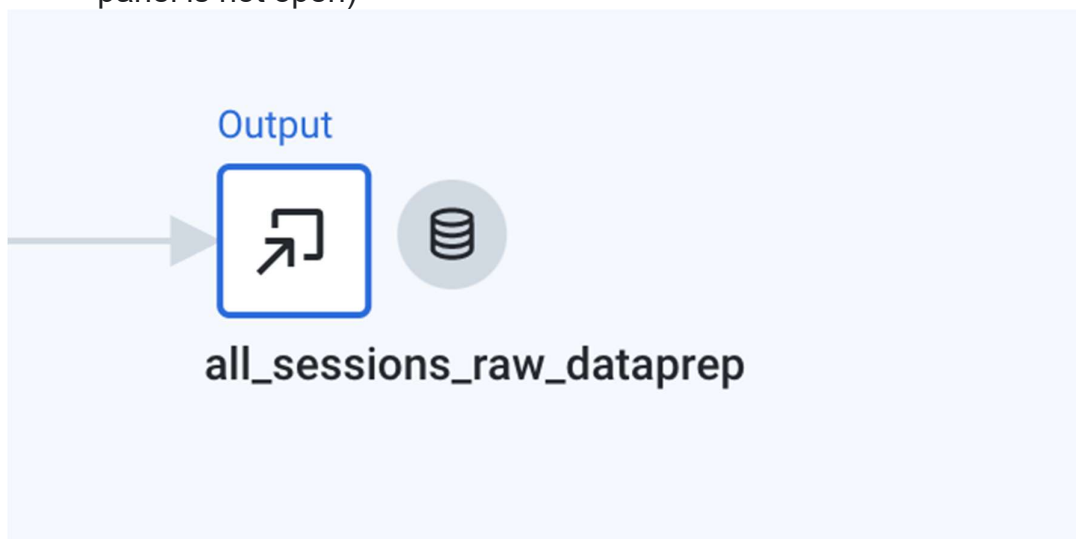
You will see the following notification at the top of your flow:

 Ecommerce Analytics Pipeline
Revenue reporting table

No scheduled destinations set. Create an output to set a destination.

Cloud Dataprep allows you to set different publishing destinations for manual job runs versus scheduled jobs.

- Click on the **Output** node to see the Details. (**Right click** and **View Details** if the panel is not open)



- In the **Destinations** section, click on **Add** next to Scheduled Destinations to add a new publishing destination for scheduled jobs.

Details

all_sessions_raw_dataprep

Run Job

Destinations Jobs

Manual Destinations [Edit](#)

Create-CSV

gs://dataprep-staging-a15e460d-3331-46b6-ad16-600a51101c8e/student-04-0f6d6218c022@qwiklabs.net/jobrun/all_sessions_raw_dataprep.csv

Environment Dataflow

Profiling yes

Scheduled Destinations [Add](#)

The dataset has no scheduled destinations set

Add a scheduled destination to automatically run the Output when the flow is executed by a schedule.

Challenge: Create another table to load the scheduled jobs into. Make sure you load the output into a separate table than the manual job and name it `revenue_reporting_recurring`. Since this table would be run weekly with the full data, make sure that the results overwrite the full table with each run.

7. Click the **Jobs** icon on the left of the screen.



8. You see the list of jobs, and wait until your job is marked as **Completed**.

Job	Status	Flow	User	Started
all_sessions_raw_dataprep - 3 Job ID: 7423896	Completed	Ecommerce Analytics Pipeline	student-03-d4709f579bd1@qwiklabs.net (...)	Today at 6:56 PM Run for 7 minutes

Congratulations!

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



Finish Your Quest

This self-paced lab is part of the Qwiklabs [Data Engineering](#) Quest. A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, to recognize your achievement. You can make your badge (or badges) public and link to them in your online resume or social media account. Enroll in this Quest and get immediate completion credit if you've taken this lab. [See other available Qwiklabs Quests](#).

Take Your Next Lab

Continue your Quest with [Building an IoT Analytics Pipeline on Google Cloud](#), or check out these suggestions:

- [ETL Processing on Google Cloud Using Dataflow and BigQuery](#)
- [Predict Visitor Purchases with a Classification Model in BQML](#)

Next steps / learn more

Do you already have a Google Analytics account and want to query your own datasets in BigQuery? Follow this [export guide](#).

Google Cloud Training & 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.

Last Tested Date: March 5, 2021

Last Updated Date: March 5, 2021

Copyright 2021 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.

solution: <https://www.youtube.com/watch?v=h1N2MrptyCY>