Building and Executing a Pipeline Graph with Data Fusion | Qwiklabs

Thursday, November 12, 2020 2:06 PM

Clipped from: https://googlecourses.qwiklabs.com/course_sessions/71782/labs/11630

Overview

This tutorial shows you how to use the Wrangler and Data Pipeline features in Cloud Data Fusion to clean, transform, and process taxi trip data for further analysis.

What you learn

In this lab, you will:

- Connect Cloud Data Fusion to a couple of data sources
- Apply basic transformations
- Join two data sources
- Write data to a sink

Introduction

Often times, data needs go through a number of pre-processing steps before analysts can leverage the data to glean insights. For example, data types may need to be adjusted, anomalies removed, and vague identifiers may need to be converted to more meaningful entries. Cloud Data Fusion is a service for efficiently building ETL/ELT data pipelines. Cloud Data Fusion uses Cloud Dataproc cluster to perform all transforms in the pipeline.

The use of Cloud Data Fusion will be exemplified in this tutorial by using a subset of the NYC TLC Taxi Trips dataset on BigQuery.

Setup and requirements

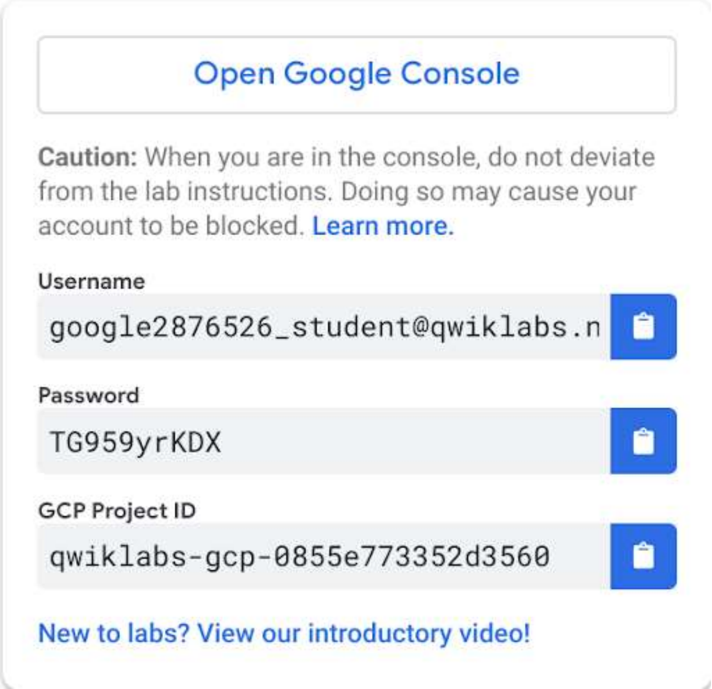
For each lab, you get a new Google Cloud project and set of resources for a fixed time at no cost.

1. Make sure you signed into Qwiklabs using an **incognito window**.

2. Note the lab's access time (for example,

and make sure you can finish in that time block.
3. When ready, click

.
4. Note your lab credentials. You will use them to sign in to the Google Cloud Console.



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.

Username

google2876526_student@qwiklabs.n

Password

TG959yrKDX

GCP Project ID

qwiklabs-gcp-0855e773352d3560

New to labs? View our introductory video!

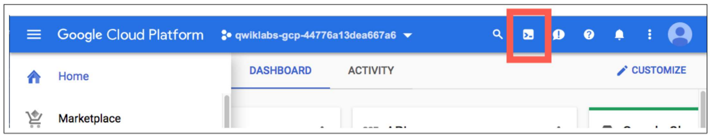
5. Click **Open Google Console**.
6. Click **Use another account** and copy/paste credentials for **this** lab into the prompts.

1. Accept the terms and skip the recovery resource page.

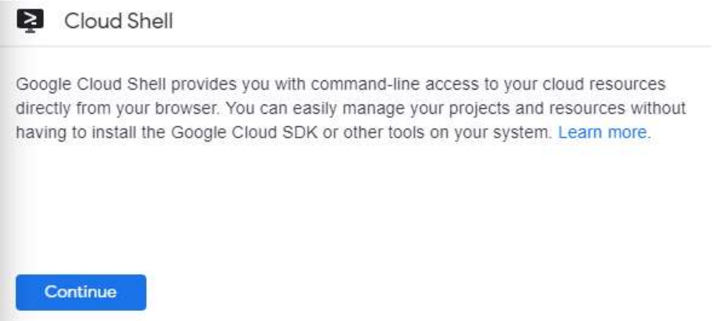
Activate Cloud Shell

Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Cloud Shell provides command-line access to your Google Cloud resources.

In the Cloud Console, in the top right toolbar, click the **Activate Cloud Shell** button.

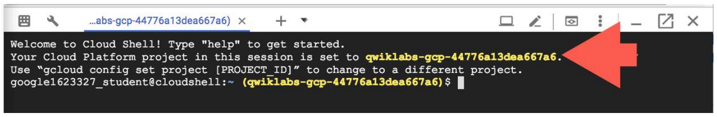


Click **Continue**.



Google Cloud Shell provides you with command-line access to your cloud resources directly from your browser. You can easily manage your projects and resources without having to install the Google Cloud SDK or other tools on your system. Learn more.

It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your *PROJECT_ID*. For example:



```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to qwiklabs-gcp-44776a13dea667a6.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
google1623327_student@cloudshell:~ (qwiklabs-gcp-44776a13dea667a6)$
```

`gcloud` is the command-line tool for Google Cloud. It comes pre-installed on Cloud Shell and supports tab-completion.

You can list the active account name with this command:

```
gcloud auth list
```

(Output)

```
Credentialed accounts:
- <myaccount>@<mydomain>.com (active)
```

(Example output)

```
Credentialed accounts:
- google1623327_student@qwiklabs.net
```

You can list the project ID with this command:

```
gcloud config list project
```

(Output)

```
[core]
project = <project_ID>
```

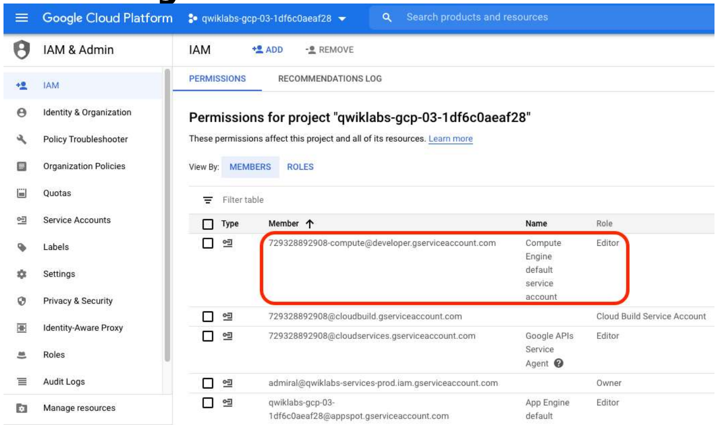
(Example output)

```
[core]
project = qwiklabs-gcp-
44776a13dea667a6
```

Check project permissions

Before you begin your work on Google Cloud, you need to ensure that your project has the correct permissions within Identity and Access Management (IAM).

1. In the Google Cloud console, on the **Navigation menu** (☰), click **IAM & Admin** > **IAM**.
2. Confirm that the default compute Service Account [{project-number}-compute@developer.gserviceaccount.com]() is present and has the `editor` role assigned. The account prefix is the project number, which you can find on **Navigation menu** > **Home**.



Permissions for project "qwiklabs-gcp-03-1df6c0aeaf28"

These permissions affect this project and all of its resources. Learn more

□ Type	Member ↑	Name	Role
□	729328892908-compute@developer.gserviceaccount.com	Compute Engine default service account	Editor
□	729328892908@cloudbuild.gserviceaccount.com		Cloud Build Service Account
□	729328892908@cloudservices.gserviceaccount.com	Google APIs Service Agent	Editor
□	admiral@qwiklabs-services-prod.iam.gserviceaccount.com		Owner
□	qwiklabs-gcp-03-1df6c0aeaf28@appspot.gserviceaccount.com	App Engine default	Editor

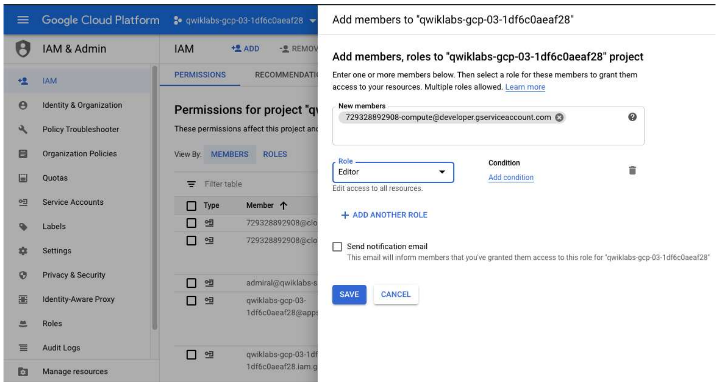
If the account is not present in IAM or does not have the `editor` role, follow the steps below to assign the required role.

- In the Google Cloud console, on the **Navigation menu**, click **Home**.
- Copy the project number (e.g. `729328892908`).
- On the **Navigation menu**, click **IAM & Admin** > **IAM**.
- At the top of the **IAM** page, click **Add**.
- For **New members**, type:

{project-number}-compute@developer.gserviceaccount.com

Replace `{project-number}` with your project number.

- For **Role**, select **Project** (or Basic) > **Editor**. Click **Save**.



Add members to "qwiklabs-gcp-03-1df6c0aeaf28"

Add members, roles to "qwiklabs-gcp-03-1df6c0aeaf28" project

Enter one or more members below. Then select a role for these members to grant them access to your resources. Multiple roles allowed. Learn more

New members: 729328892908-compute@developer.gserviceaccount.com

Role: Editor

Edit access to all resources.

Condition: Add condition

+ ADD ANOTHER ROLE

□ Send notification email

This email will inform members that you've granted them access to this role for "qwiklabs-gcp-03-1df6c0aeaf28"

SAVE CANCEL

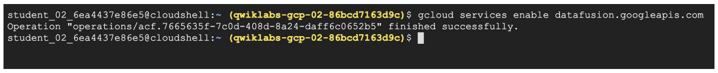
Task 1: Creating a Cloud Data Fusion instance

Thorough directions for creating a Cloud Data Fusion instance can be found here. The essential steps are as follows:

1. Run the below command in the Cloud Shell. It will take a few minutes to complete. A message will be returned that the API enablement operation successfully completed (your operation identifier will vary).

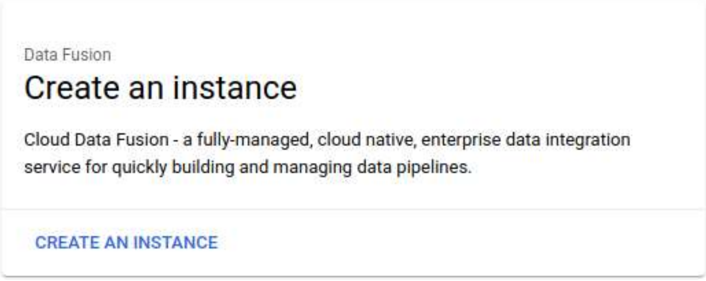
```
gcloud services enable
datafusion.googleapis.com
```

Your output will appear similar to the screenshot:



```
student_02_6ea4437e86e5@cloudshell:~ (qwiklabs-gcp-02-86bcd7163d9c)$ gcloud services enable datafusion.googleapis.com
Operation "operations/acf.7665635f-7c0d-408d-8a24-daff6c0652b5" finished successfully.
student_02_6ea4437e86e5@cloudshell:~ (qwiklabs-gcp-02-86bcd7163d9c)$ 
```

1. On the **Navigation menu**, select **Data Fusion**.
2. To create a Cloud Data Fusion instance, click **Create an Instance**.



Data Fusion

Create an instance

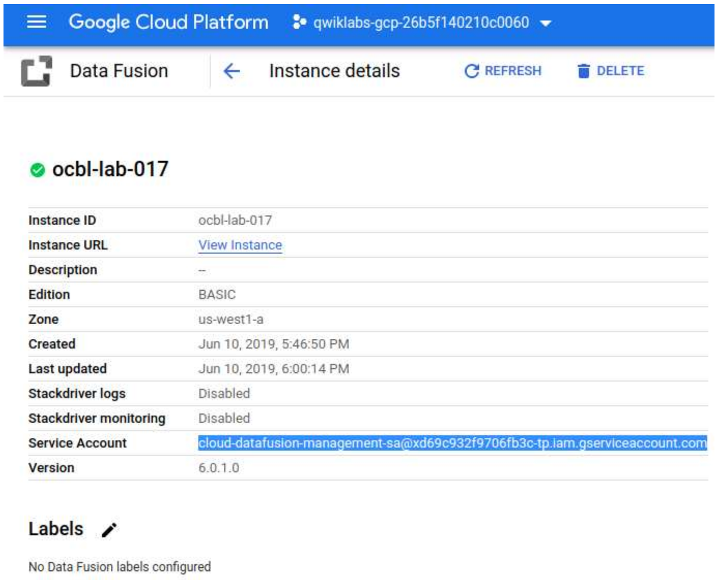
Cloud Data Fusion - a fully-managed, cloud native, enterprise data integration service for quickly building and managing data pipelines.

CREATE AN INSTANCE

1. Enter a name for your instance.
2. Select **Basic** for the Edition type.
3. Leave all other fields as their defaults and click **Create**.

Note: Creation of the instance can take around 15 minutes.

1. Once the instance is created, you need one additional step to grant the service account associated with the instance permissions on your project. Navigate to the instance detail page by clicking the instance name.



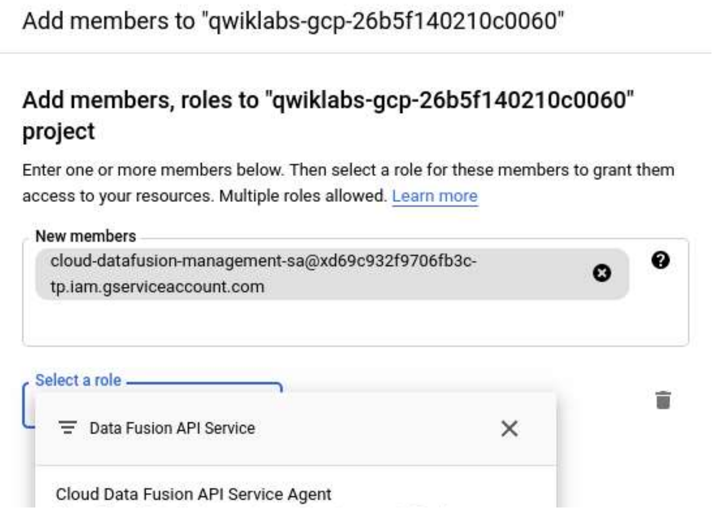
ocbl-lab-017

Instance ID	ocbl-lab-017
Instance URL	View Instance
Description	–
Edition	BASIC
Zone	us-west1-a
Created	Jun 10, 2019, 5:46:50 PM
Last updated	Jun 10, 2019, 6:00:14 PM
Stackdriver logs	Disabled
Stackdriver monitoring	Disabled
Service Account	cloud-datafusion-management-sa@xd69c932f9706fb3c-tp.iam.gserviceaccount.com
Version	6.0.1.0

Labels

No Data Fusion labels configured

1. Copy the service account to your clipboard.
2. In the GCP Console navigate to the **IAM & Admin > IAM**.
3. On the IAM Permissions page, add the service account you copied earlier as a new member and grant the **Cloud Data Fusion API Service Agent** role, by clicking the **Add** button.



Add members to "qwiklabs-gcp-26b5f140210c0060"

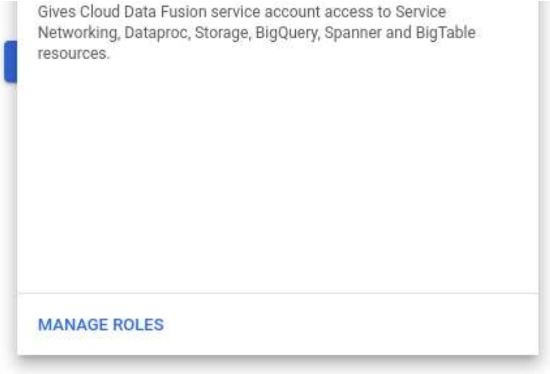
Add members, roles to "qwiklabs-gcp-26b5f140210c0060" project

Enter one or more members below. Then select a role for these members to grant them access to your resources. Multiple roles allowed. Learn more

New members: cloud-datafusion-management-sa@xd69c932f9706fb3c-tp.iam.gserviceaccount.com

Select a role: Data Fusion API Service

Cloud Data Fusion API Service Agent



Gives Cloud Data Fusion service account access to Service Networking, Dataproc, Storage, BigQuery, Spanner and BigTable resources.

1. Click **Save**.

Task 2: Loading the data

Once the Cloud Data Fusion instance is up and running, you can start using Cloud Data Fusion. However, before Cloud Data Fusion can start ingesting data you have to take some preliminary steps.

1. In this example, Cloud Data Fusion will read data out of a storage bucket. Open a cloud shell console and execute the following commands to create a new bucket and copy the relevant data into it:

```
export BUCKET=$GOOGLE_CLOUD_PROJECT
gsutil mb gs://$BUCKET
gsutil cp gs://cloud-
training/OCBL017/ny-taxi-2018-sample.csv
gs://$BUCKET
```

Note: The created bucket name is your project id.

1. In the command line, execute the following command to create a bucket for temporary storage items that Cloud data Fusion will create.

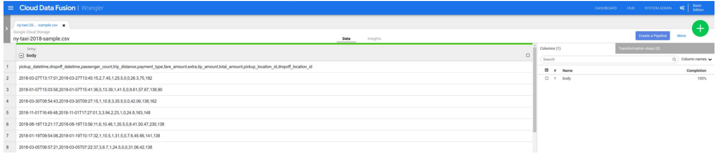
```
gsutil mb gs://$BUCKET-temp
```

Note: The created bucket name is your project id followed by "-temp".

1. Click the **View Instance** link on the Cloud Data Fusion instances page, or the details page of an instance. If prompted to take a tour of the service click on **No, Thanks**. You should now be in the Cloud Data Fusion UI.

Note: You may need to reload or refresh the Cloud Fusion UI pages to allow prompt loading of the page.

1. **Wrangler** is an interactive, visual tool that lets you see the effects of transformations on a small subset of your data before dispatching large, parallel-processing jobs on the entire dataset. On the Cloud Data Fusion UI, choose **Wrangler**. On the left side, there is a panel with the pre-configured connections to your data, including the Cloud Storage connection.
2. Under **Google Cloud Storage**, select **Cloud Storage Default**.
3. Click on the bucket corresponding to your project name.
4. Select **ny-taxi-2018-sample.csv**. The data is loaded into the Wrangler screen in row/column form.



	body
1	pickup_datetime,dropoff_datetime,passenger_count,trip_distance,payment_type,fare_amount,extra,tip_amount,total_amount,pickup_location_id,dropoff_location_id
2	2018-03-27T13:17:01,2018-03-27T13:45:15,2,7.45,1,25.5,0,0,26.3,75,182
3	2018-01-07T15:03:56,2018-01-07T15:41:36,5,13.39,1,41.5,0,9.61,57.67,138,90
4	2018-03-30T08:54:43,2018-03-30T09:27:15,1,10.8,3,35.5,0,0,42.06,138,162
5	2018-11-01T16:49:48,2018-11-01T17:27:01,3,3.94,2,23,1,0,24.8,163,148
6	2018-08-18T13:21:17,2018-08-18T13:56:11,6,10.46,1,35.5,0,8.41,50.47,230,138
7	2018-01-19T09:54:06,2018-01-19T10:17:32,1,10.5,1,31.5,0,7.6,45.66,141,138
8	2018-03-05T06:57:21,2018-03-05T07:22:37,3,6.7,1,24.5,0,0,31.06,42,138

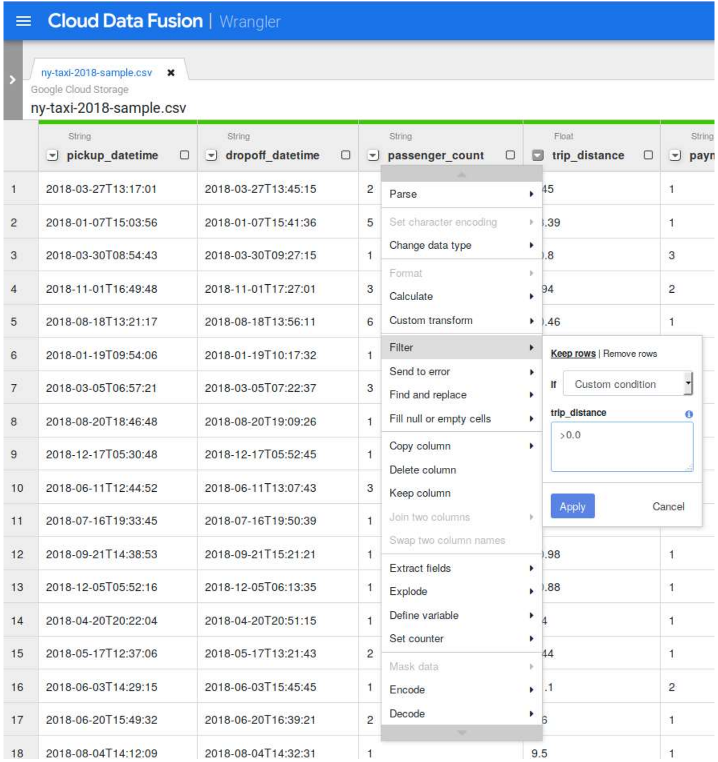
#	Name	Completion
1	body	100%

Task 3: Cleaning the data

Now, you will perform some transformations to parse and clean the taxi data.

1. To the left of the body column, click the **Down** arrow.
2. Click **Parse > CSV**, select **Set first row as header** and then click **Apply.** The data splits into multiple columns.
3. Because the body column isn't needed anymore, click the **Down** arrow next to the body column and choose **Delete column**.
4. You'll notice that all of the column types have been loaded in as `String`. Click the **Down** arrow next to the `trip_distance` column, select **Change data type** and then click on **Float**. Repeat for the `total_amount` column.
5. If you look at the data closely, you may find some anomalies, such as negative trip distances. You can avoid those negative values by filtering out in **Wrangler**. Click the **Down** arrow next to the `trip_distance` column and

select **Filter**. Click if **Custom condition** and input `>0.0`



	String pickup_datetime	String dropoff_datetime	String passenger_count	Float trip_distance	String payn
1	2018-03-27T13:17:01	2018-03-27T13:45:15	2	45	1
2	2018-01-07T15:03:56	2018-01-07T15:41:36	5	.39	1
3	2018-03-30T08:54:43	2018-03-30T09:27:15	1	.8	3
4	2018-11-01T16:49:48	2018-11-01T17:27:01	3	94	2
5	2018-08-18T13:21:17	2018-08-18T13:56:11	6	.46	1
6	2018-01-19T09:54:06	2018-01-19T10:17:32	1		
7	2018-03-05T06:57:21	2018-03-05T07:22:37	3		
8	2018-08-20T18:46:48	2018-08-20T19:09:26	1		
9	2018-12-17T05:30:48	2018-12-17T05:52:45	1		
10	2018-06-11T12:44:52	2018-06-11T13:07:43	3		
11	2018-07-16T19:33:45	2018-07-16T19:50:39	1		
12	2018-09-21T14:38:53	2018-09-21T15:21:21	1	.98	1
13	2018-12-05T05:52:16	2018-12-05T06:13:35	1	.88	1
14	2018-04-20T20:22:04	2018-04-20T20:51:15	1	4	1
15	2018-05-17T12:37:06	2018-05-17T13:21:43	2	44	1
16	2018-06-03T14:29:15	2018-06-03T15:45:45	1	.1	2
17	2018-06-20T15:49:32	2018-06-20T16:39:21	2	6	1
18	2018-08-04T14:12:09	2018-08-04T14:32:31	1	9.5	1

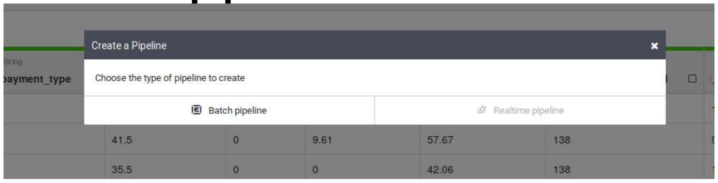
1. Click on **Apply**.

Task 4: Creating the pipeline

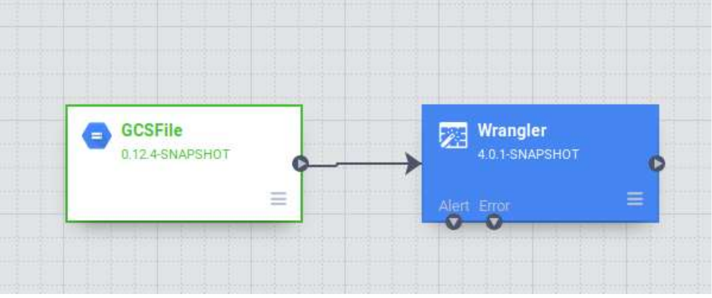
Basic data cleansing is now complete and you've run transformations on a subset of your data. You can now create a batch pipeline to run transformations on all your data.

Cloud Data Fusion translates your visually built pipeline into an Apache Spark or MapReduce program that executes transformations on an ephemeral Cloud Dataproc cluster in parallel. This enables you to easily execute complex transformations over vast quantities of data in a scalable, reliable manner, without having to wrestle with infrastructure and technology.

1. On the upper-right side of the Google Cloud Fusion UI, click **Create a Pipeline**.
2. In the dialog that appears, select **Batch pipeline**.



1. In the Data Pipelines UI, you will see a GCSFile source node connected to a Wrangler node. The Wrangler node contains all the transformations you applied in the Wrangler view captured as directive grammar. Hover over the Wrangler node and select **Properties**.



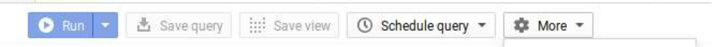
.

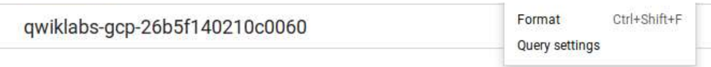
1. At this stage, you can apply more transformations by clicking the **Wrangle** button. Delete the `extra` column by pressing the red trashcan icon beside its name. To close the Wrangler tool click the **X** button in the top right corner.

Task 5: Adding a data source

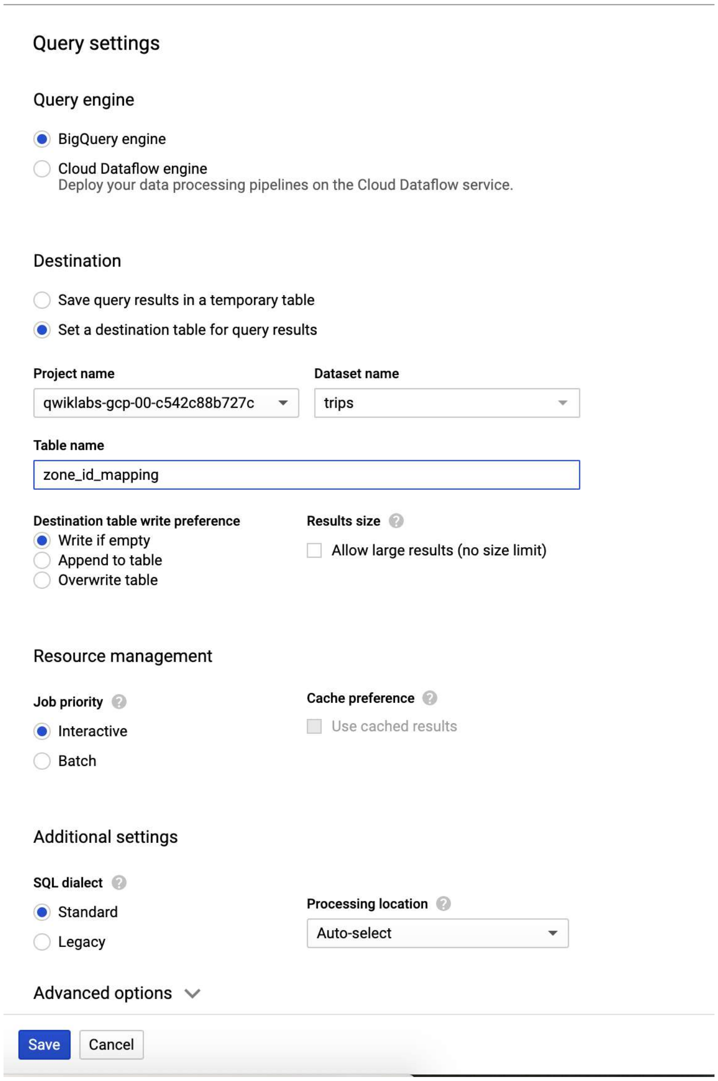
The taxi data contains several cryptic columns such as `pickup_location_id`, that aren't immediately transparent to an analyst. You are going to add a data source to the pipeline that maps the `pickup_location_id` column to a relevant location name. The mapping information will be stored in a BigQuery table.

1. In a separate tab, [open the BigQuery UI in the GCP Console](). Click **Done** on the 'Welcome to BigQuery in the Cloud Console' launch page.
2. In the left pane, in the **Resources** section, click your GCP Project ID (it will start with qwiklabs).
3. Select **Create dataset**.
4. In the **Dataset ID** field type in `trips`.
5. Click on **Create dataset**.
6. To create the desired table in the newly created dataset, navigate to **More > Query Settings**. This process will ensure you can access your table from Cloud Data Fusion.




qwiklabs-gcp-26b5f140210c0060

1. Select the item for **Set a destination table for query results**. Also, under **Table name** input `zone_id_mapping`. Click **Save**.


Query settings

Query engine

● BigQuery engine
○ Cloud Dataflow engine
Deploy your data processing pipelines on the Cloud Dataflow service.

Destination

○ Save query results in a temporary table
● Set a destination table for query results

Project name: qwiklabs-gcp-00-c542c88b727c
Dataset name: trips

Table name: zone_id_mapping

Destination table write preference
● Write if empty
○ Append to table
○ Overwrite table

Results size
□ Allow large results (no size limit)

Resource management

Job priority
● Interactive
○ Batch

Cache preference
□ Use cached results

Additional settings

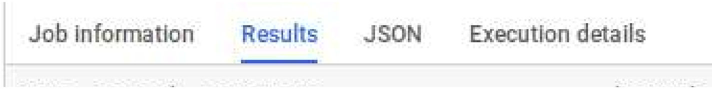
SQL dialect
● Standard
○ Legacy

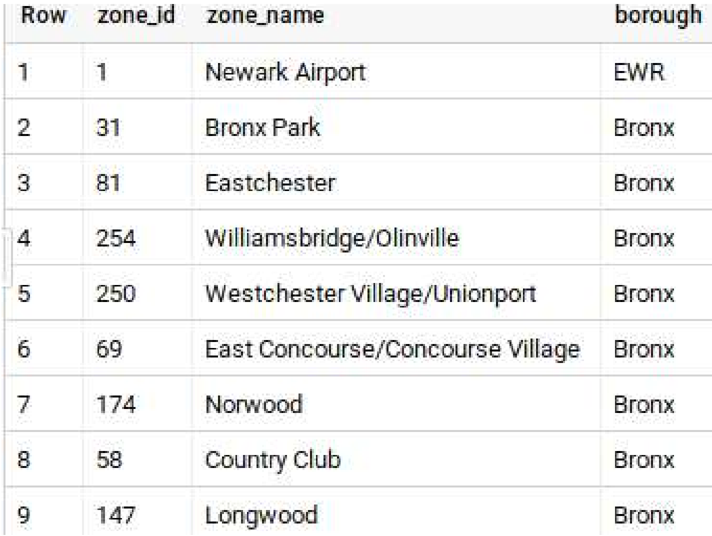
Processing location: Auto-select

1. Enter the following query in the Query Editor and then click **Run**:

```
SELECT
  zone_id,
  zone_name,
  borough
FROM
  `bigquery-public-
data.new_york_taxi_trips.taxi_zone_geom`
```

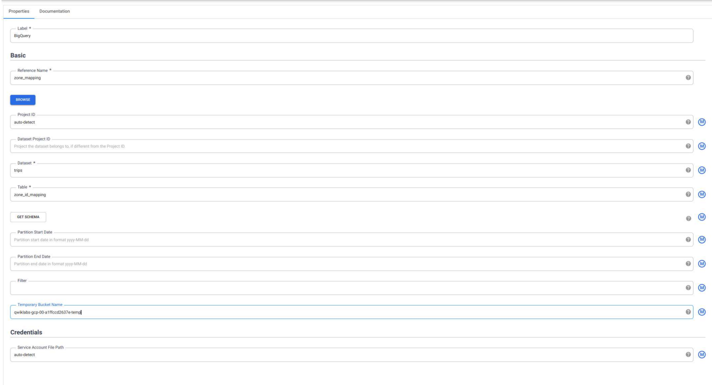
You can see that this table contains the mapping from `zone_id` to its name and borough.





Row	zone_id	zone_name	borough
1	1	Newark Airport	EWR
2	31	Bronx Park	Bronx
3	81	Eastchester	Bronx
4	254	Williamsbridge/Olinville	Bronx
5	250	Westchester Village/Unionport	Bronx
6	69	East Concourse/Concourse Village	Bronx
7	174	Norwood	Bronx
8	58	Country Club	Bronx
9	147	Longwood	Bronx

1. Now, you will add a source in your pipeline to access this BigQuery table. Return to tab where you have Cloud Data Fusion open, from the Plugin palette on the left, select **BigQuery** from the **Source** section. A BigQuery source node appears on the canvas with the two other nodes.
2. Hover over the new BigQuery source node and click **Properties**.
3. To configure the **Reference Name**, enter `zone_mapping`, which is used to identify this data source for lineage purposes. The BigQuery **Dataset** and **Table** configurations are the Dataset and Table you setup in BigQuery a few steps earlier: `trips` and `zone_id_mapping`. For **Temporary Bucket Name** input the name of your project followed by "-temp", which corresponds to the bucket you created in Task 2.



Label *
BigQuery

Basic

Reference Name *
zone_mapping

Project ID
auto-detect

Dataset Project ID
Project the dataset belongs to, if different from the Project ID.

Dataset *
trips

Table *
zone_id_mapping

Partition Start Date
Partition start date in format yyyy-MM-dd

Partition End Date
Partition end date in format yyyy-MM-dd

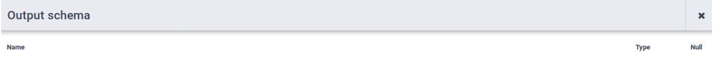
Filter

Temporary Bucket Name
qwiklabs-gcp-00-a1ffccd2637e-temp

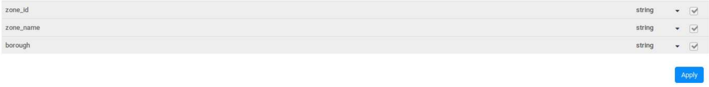
Credentials

Service Account File Path
auto-detect

1. To populate the schema of this table from BigQuery, click **Get Schema**. The fields will appear on the right side of the wizard.


Output schema

Name	Type	Null



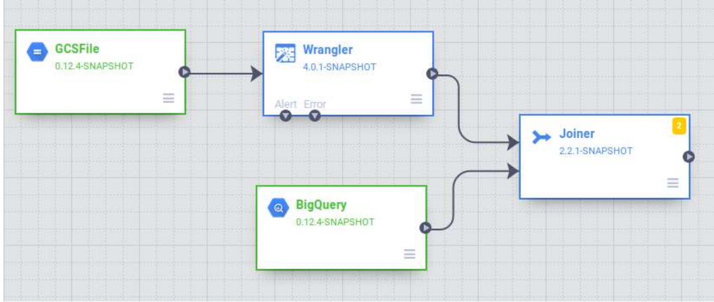
zone_id	string	☑
zone_name	string	☑
borough	string	☑

1. To close the BigQuery Properties window click the **X** button in the top right corner.

Task 6: Joining two sources

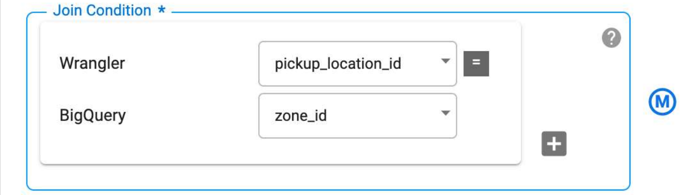
Now you can join the two data sources—taxi trip data and zone names—to generate more meaningful output.

1. Under the **Analytics** section in the Plugin Palette, choose **Joiner**. A **Joiner** node appears on the canvas.
2. To connect the Wrangler node and the BigQuery node to the Joiner node: Drag a connection arrow > on the right edge of the source node and drop on the destination node.



1. To configure the Joiner node, which is similar to a SQL JOIN syntax:

- Click **Properties** of **Joiner**.
- Leave the label as **Joiner**.
- Change the **Join Type** to **Inner**
- Set the **Join Condition** to join the `pickup_location_id` column in the Wrangler node to the `zone_id` column in the BigQuery node.

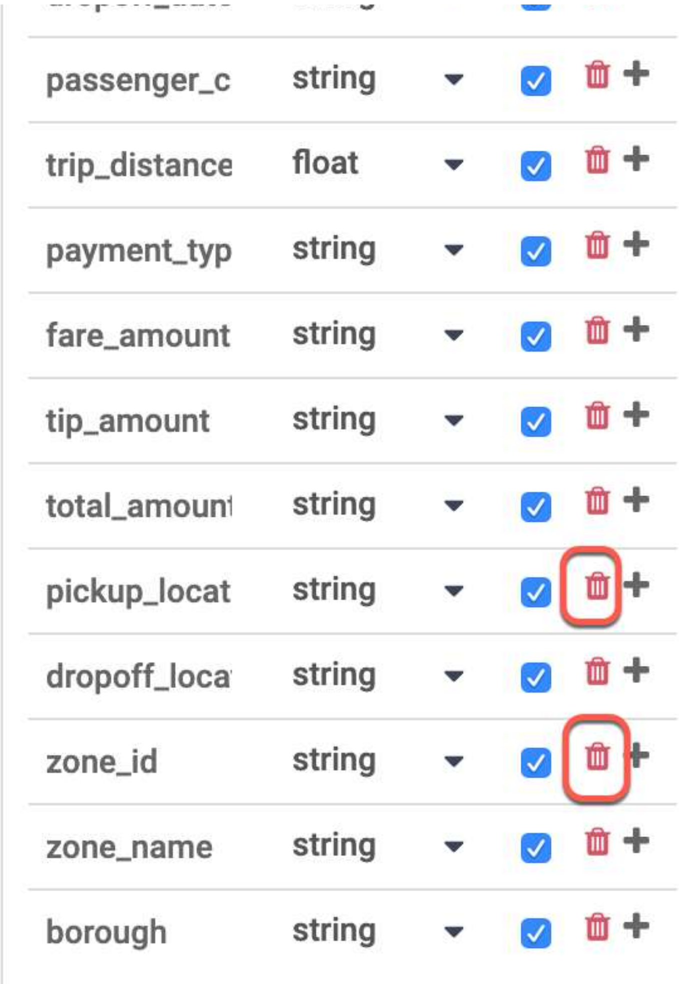


Join Condition *

Wrangler pickup_location_id =
BigQuery zone_id

- To generate the schema of the resultant join, click **Get Schema**.
- In the **Output Schema** table on the right, remove the `zone_id` and `pickup_location_id` fields by hitting the red garbage can icon.

dropoff_date string ☑



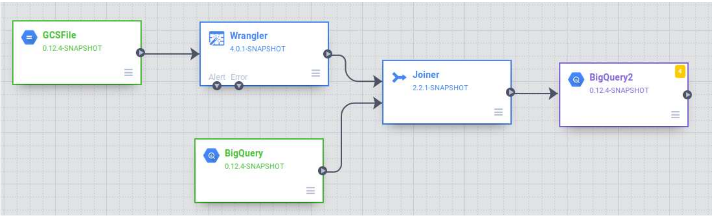
passenger_c	string	☑
trip_distance	float	☑
payment_typ	string	☑
fare_amount	string	☑
tip_amount	string	☑
total_amount	string	☑
pickup_locat	string	☑
dropoff_loca	string	☑
zone_id	string	☑
zone_name	string	☑
borough	string	☑

- Close the window by clicking the **X** button in the top right corner.

Task 7: Storing the output to BigQuery

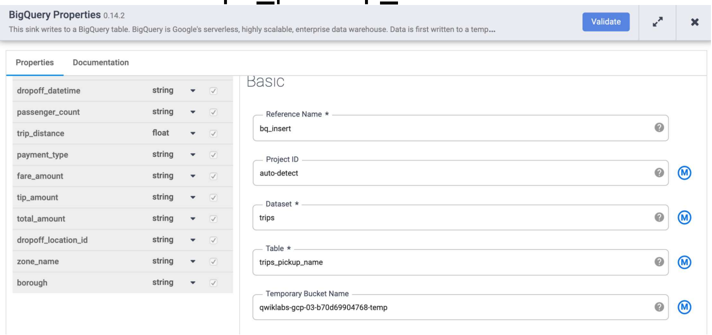
You will store the result of the pipeline into a BigQuery table. Where you store your data is called a sink.

1. In the **Sink** section of the Plugin Palette, choose **BigQuery**.
2. Connect the **Joiner** node to the **BigQuery** node. Drag a connection arrow > on the right edge of the source node and drop on the destination node.



1. Open the BigQuery node by hovering on it and then clicking **Properties**. You will next configure the node as shown below. You will use a

configuration that's similar to the existing BigQuery source. Provide `bq_insert` for the **Reference Name** field and then use `trips` for the **Dataset** and the name of your project followed by "-temp" as **Temporary Bucket Name**. You will write to a new table that will be created for this pipeline execution. In **Table** field, enter `trips_pickup_name`.



BigQuery Properties 0.14.2

This sink writes to a BigQuery table. BigQuery is Google's serverless, highly scalable, enterprise data warehouse. Data is first written to a temp...

dropoff_datetime	string	☑
passenger_count	string	☑
trip_distance	float	☑
payment_type	string	☑
fare_amount	string	☑
tip_amount	string	☑
total_amount	string	☑
dropoff_location_id	string	☑
zone_name	string	☑
borough	string	☑

Basic

Reference Name *: bq_insert

Project ID: auto-detect

Dataset *: trips

Table *: trips_pickup_name

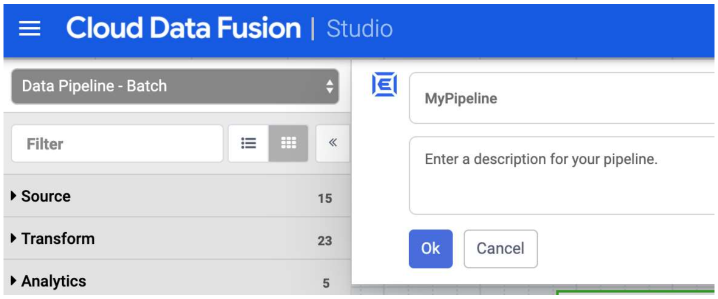
Temporary Bucket Name: qwiklabs-gcp-03-b70d69904768-temp

1. Close the window by clicking the **X** button in the top right corner.

Task 8: Deploying and running the pipeline

At this point you have created your first pipeline and can deploy and run the pipeline.

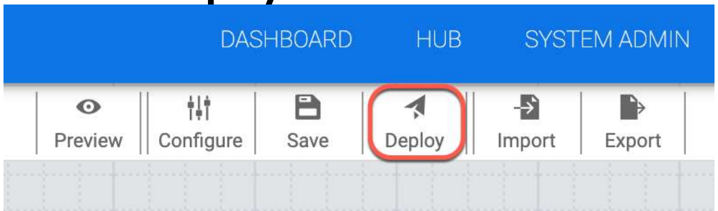
1. Name your pipeline in the upper left corner of the Data Fusion UI and click **OK**.



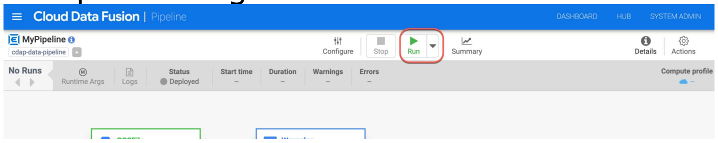
MyPipeline

Enter a description for your pipeline.

1. Now you will deploy the pipeline. In the upper-right corner of the page, click **Deploy**.



1. On the next screen click **Run** to start processing data.

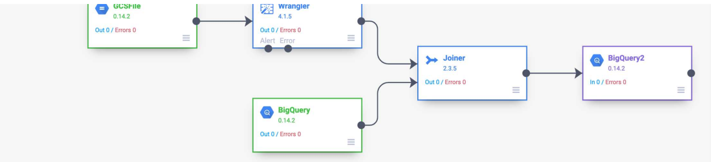


MyPipeline

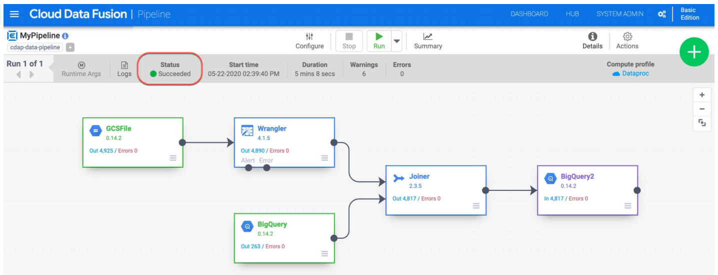
cdap-data-pipeline

No Runs

Runtime Args	Logs	Status	Start time	Duration	Warnings	Errors	Compute profile
		Deployed	–	–	–	–	–



When you run a pipeline, Cloud Data Fusion provisions an ephemeral Cloud Dataproc cluster, runs the pipeline, and then tears down the cluster. This could take a few minutes. You can observe the status of the pipeline transition from *Provisioning* to *Starting* and from *Starting* to *Running* to *Succeeded* during this time.



Status: Succeeded
Start time: 05-22-2020 02:39:40 PM
Duration: 5 mins 8 secs
Warnings: 6
Errors: 0
Compute profile: Dataproc

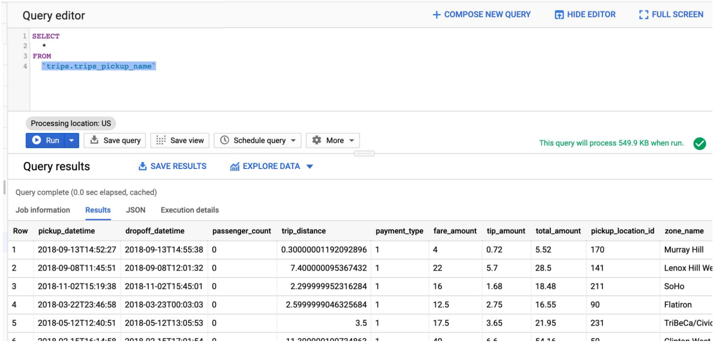
Task 9: Viewing the results

To view the results after the pipeline runs:

1. Return to the tab where you have BigQuery open. Run the query below to see the values in the `trips_pickup_name` table.

```
SELECT
  *
FROM
  `trips.trips_pickup_name`
```

BQ RESULTS



```
SELECT
  *
FROM
  `trips.trips_pickup_name`
```

Processing location: US

This query will process 549.9 KB when run.

Query complete (0.0 sec elapsed, cached)

Row	pickup_datetime	dropoff_datetime	passenger_count	trip_distance	payment_type	fare_amount	tip_amount	total_amount	pickup_location_id	zone_name
1	2018-09-13T14:52:27	2018-09-13T14:55:38	0	0.30000001192092896	1	4	0.72	5.52	170	Murray Hill
2	2018-09-08T11:45:51	2018-09-08T12:01:32	0	7.400000095367432	1	22	5.7	28.5	141	Lenox Hill We
3	2018-11-02T15:19:38	2018-11-02T15:45:01	0	2.299999952316284	1	16	1.68	18.48	211	SoHo
4	2018-03-22T23:46:58	2018-03-23T00:03:03	0	2.5999999046325684	1	12.5	2.75	16.55	90	Flatiron
5	2018-05-12T12:40:51	2018-05-12T13:05:53	0	3.5	1	17.5	3.65	21.95	231	TriBeCa/Civic

End your lab

When you have completed your lab, click **End Lab**. Qwiklabs removes the resources you've used and cleans the account for you.

You will be given an opportunity to rate the lab experience. Select the applicable number of stars, type a comment, and then click **Submit**.

The number of stars indicates the following:

- 1 star = Very dissatisfied
- 2 stars = Dissatisfied
- 3 stars = Neutral
- 4 stars = Satisfied
- 5 stars = Very satisfied

You can close the dialog box if you don't want to provide feedback.

For feedback, suggestions, or corrections, please use the **Support** tab.