

Assessing Data Quality with Dataplex

experiment Lab schedule 1 hour 30 minutes universal_currency_alt No cost

show_chart Introductory



GSP1158



Google Cloud Self-Paced Labs

Overview

Dataplex is an intelligent data fabric that enables organizations to centrally discover, manage, monitor, and govern their data across data lakes, data warehouses, and data marts to power analytics at scale.

A valuable feature of Dataplex is the ability to define and run data quality checks on Dataplex assets such as BigQuery tables and Cloud Storage files. Using Dataplex data quality tasks, you can integrate data quality checks into everyday workflows by validating data that is part of a data production pipeline, regularly monitoring the quality of your data against a set of criteria, and building data quality reports for regulatory requirements.

In this lab, you learn how to assess data quality using Dataplex by creating a custom data quality specification file and using it to define and run a data quality job on BigQuery data.

What you'll do

- Create a Dataplex lake, zone, and asset
- Query a BigQuery table to review data quality
- Create and upload a data quality specification file
- Define and run a data quality job
- Review the results of a data quality job

Setup and requirements

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-02-71e205c512c5@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.

2sxtztXR4ggq

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.

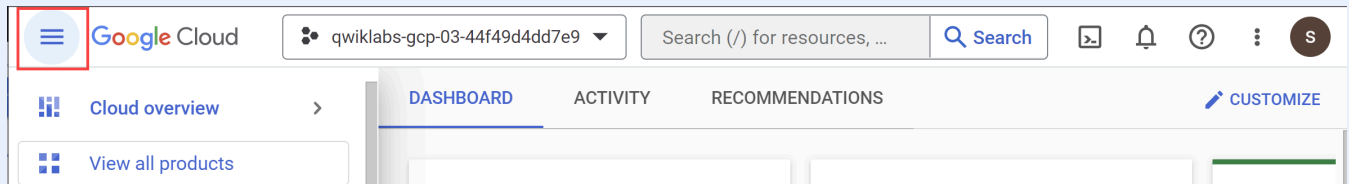
Note: Using your own Google Cloud account for this lab may incur extra charges.

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.



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.

1. Click **Activate Cloud Shell**  at the top of the Google Cloud console.

When you are connected, you are already authenticated, and the project is set to your **Project_ID**, `qwiklabs-gcp-02-62755358c608`. The output contains a line that declares the **Project_ID** for this session:

```
Your Cloud Platform project in this session is set to qwiklabs-gcp-02-62755358c60
```

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

2. (Optional) You can list the active account name with this command:

```
gcloud auth list
```

content_co

3. Click **Authorize**.

Output:

```
ACTIVE: *  
ACCOUNT: student-02-71e205c512c5@qwiklabs.net  
  
To set the active account, run:  
$ gcloud config set account `ACCOUNT`
```

4. (Optional) You can list the project ID with this command:

```
gcloud config list project
```

content_co

Output:

```
[core]  
project = qwiklabs-gcp-02-62755358c608
```

Note: For full documentation of `gcloud`, in Google Cloud, refer to the `gcloud` CLI overview guide.

Enable Dataproc API

1. In the Google Cloud Console, enter **Cloud Dataproc API** in the top search bar.
2. Click on the result for **Cloud Dataproc API** under Marketplace.
3. Click **Enable**.

Task 1. Create a lake, zone, and asset in Dataplex

To define and run data quality tasks, you first need to create some Dataplex resources.

In this task, you create a new Dataplex lake to store ecommerce customer information, add a raw zone to the lake, and then attach a pre-created BigQuery dataset as a new asset in the zone.

Create a lake

1. In the Google Cloud Console, in the **Navigation menu** () , navigate to **Analytics > Dataplex**.

If prompted **Welcome to the new Dataplex experience**, click **Close**.

2. Under **Manage lakes**, click **Manage**.
3. Click **Create lake**.
4. Enter the required information to create a new lake:

Property	Value
Display Name	Ecommerce Lake
ID	Leave the default value.
Region	us-central1

Leave the other default values.

5. Click **Create**.

It can take up to 3 minutes for the lake to be created.

Add a zone to the lake

1. On the **Manage** tab, click on the name of your lake.
2. Click **Add zone**.
3. Enter the required information to create a new zone:

Property	Value
Display Name	Customer Contact Raw Zone
ID	Leave the default value.
Type	Raw zone
Data locations	Regional

Leave the other default values.

For example, the option for **Enable metadata discovery** under **Discovery settings** is enabled by default and allows authorized users to discover the data in the zone.

4. Click **Create**.

It can take up to 2 minutes for the zone to be created.

You can perform the next task once the status of the zone is **Active**.

Attach an asset to a zone

1. On the **Zones** tab, click on the name of your zone.
2. On the **Assets** tab, click **Add assets**.
3. Click **Add an asset**.
4. Enter the required information to attach a new asset:

Property	Value
Type	BigQuery dataset
Display Name	Contact Info
ID	Leave the default value.
Dataset	qwiklabs-gcp-02-62755358c608.customers

Leave the other default values.

5. Click **Done**.

6. Click **Continue**.

7. For **Discovery settings**, select **Inherit** to inherit the Discovery settings from the zone level, and then click **Continue**.

8. Click **Submit**.

Click *Check my progress* to verify the objective.



Create a lake, zone, and asset in Dataplex

Check my progress


Assessment Completed!

Task 2. Query a BigQuery table to review data quality

In the previous task, you created a new Dataplex asset from a BigQuery dataset named **customers** that has been pre-created for this lab. This dataset contains a table named **contact_info** which contains raw contact

information for customers of a fictional ecommerce company.

In this task, you query this table to start identifying some potential data quality issues that you can include as checks in a data quality job. You also identify another precreated dataset that you can use to store data quality job results in a later task.

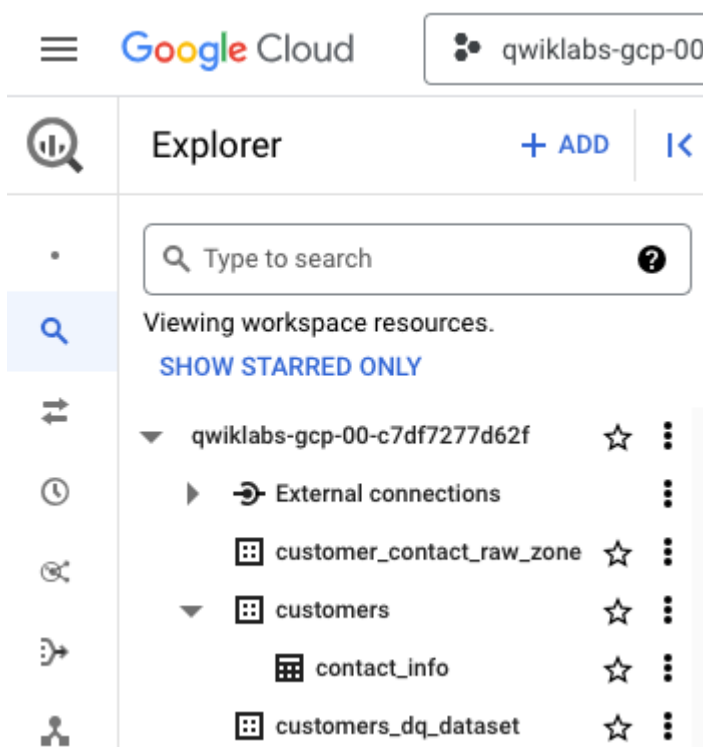
1. In the Google Cloud Console, in the **Navigation menu** () , navigate to **BigQuery > SQL Workspace**.
2. In the Explorer pane, expand the arrow next to your project ID to list the contents: **qwiklabs-gcp-02-62755358c608**

In addition to the **customer_contact_raw_zone** dataset created by Dataplex to manage that zone, there are two BigQuery datasets that were precreated for this lab:

- customers
- customers_dq_dataset

The dataset named **customers** contains one table named **contact_info**, which contains contact information for customers such as a customer ID, name, email, and more. This is the table that you explore and check for data quality issues throughout this lab.

The dataset named **customers_dq_dataset** does not contain any tables. When you define a data quality job in a later task, you use this dataset as the destination for a new table containing the data quality job results.



3. In the SQL Editor, click on **Compose a new query**. Paste the following query, and then click **Run**:

```
SELECT * FROM `qwiklabs-gcp-02-62755358c608.customers.contact_info`
ORDER BY id
LIMIT 50
```

content_c

This query selects 50 records from the original table and orders the records by the customer id in the results.

4. Scroll through the results in the **Results** pane.

Notice that some records are missing customer IDs or have incorrect emails, which can make it difficult to manage customer orders.

Query results [SAVE RESULTS](#) [EXPLORE DATA](#)

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		EXECUTION GRAPH	
Row	created_at	id	first_name	last_name	age	email	
1	2021-10-26 10:23:59 UTC	null	William	Long	29	william.long@example.com	
2	2020-11-12 16:26:57 UTC	null	Anthony	Walker	70	anthony.walker@example.com	
3	2020-03-26 06:54:33 UTC	null	Chris	Smith	37	chris.smith@example.com	
4	2022-07-05 09:50:01 UTC	null	James	Peden	69	james.peden@example.com	
5	2020-08-05 08:29:54 UTC	null	Daniel	Carroll	33	daniel.carroll@example.com	
6	2020-08-30 20:11:12 UTC	null	Ricardo	Jordan	14	ricardo.jordan@example.com	
7	2022-11-25 19:13:05 UTC	null	Robert	Todd	52	robert.todd@example.com	
8	2022-08-06 11:58:25 UTC	null	Adrian	Tolbert	33	null	
9	2022-02-11 16:48:25 UTC	null	David	Williams	47	david.williams@example.com	
10	2022-11-21 19:36:13 UTC	null	Max	Leffler	47	max.leffler@example.com	
11	2020-01-01 19:24:12 UTC	137	Joe	Jones	48	joe.jones@example.com	
12	2020-01-27 10:18:16 UTC	1333	Lee	Hodson	18	lee.hodson@example.com	
13	2020-02-09 03:53:14 UTC	2031	Johnnie	Rosado	48	johnnie.rosado@example.com	
14	2020-02-13 11:47:24 UTC	2242	Steven	White	39	steven.white@example.com	
15	2020-04-09 14:44:31 UTC	5006	Emil	Flemings	59	emil.flemings@example.com	

Results per page: 50 1 – 50 of 50

Click *Check my progress* to verify the objective.



Query BigQuery table to review data quality

Check my progress

Assessment completed!

Task 3. Create and upload a data quality specification file

Dataplex data quality check requirements are defined using CloudDQ YAML specification files. Once created, the YAML specification file is uploaded to a Cloud Storage bucket that is made accessible to the data quality job.

The YAML file has four keys sections:

- a list of rules to run (either pre-defined or customized rules)
- row filters to select a subset of data for validation
- rule bindings to apply the defined rules to the table(s)
- optional rule dimensions to specify the types of the rules that the YAML file can contain

In this task, you define a new YAML specification file for data quality checks that identify null customer IDs and emails in the specified BigQuery table. After you define the file, you upload it to a pre-created Cloud Storage bucket for use in a later task to run the data quality job.

Create the data quality specification file

1. In Cloud Shell, run the following command to create a new empty file for the data quality specification:

```
nano dq-customer-raw-data.yaml
```

content_copy

2. Paste the following code:

```
metadata_registry_defaults:
  dataplex:
    projects: qwiklabs-gcp-02-62755358c608
    locations: us-central1
    lakes: ecommerce-lake
    zones: customer-contact-raw-zone
row_filters:
  NONE:
    filter_sql_expr: |-
      True
  INTERNATIONAL_ITEMS:
    filter_sql_expr: |-
      REGEXP_CONTAINS(item_id, 'INTNL')
rule_dimensions:
```

content_copy

```

- consistency
- correctness
- duplication
- completeness
- conformance
- integrity
- timeliness
- accuracy
rules:
  NOT_NULL:
    rule_type: NOT_NULL
    dimension: completeness
  VALID_EMAIL:
    rule_type: REGEX
    dimension: conformance
    params:
      pattern: |-
        ^[^\@]+\@[0-9]{1}[\^\@]+\$
rule_bindings:
  VALID_CUSTOMER:
    entity_uri: bigquery://projects/qwiklabs-gcp-02-62755358c608/datasets
    column_id: id
    row_filter_id: NONE
    rule_ids:
      - NOT_NULL
  VALID_EMAIL_ID:
    entity_uri: bigquery://projects/qwiklabs-gcp-02-62755358c608/datasets
    column_id: email
    row_filter_id: NONE
    rule_ids:
      - VALID_EMAIL

```

3. Review the code to identify the two primary data quality rules that are defined in this file.

The `dq-customer-raw-data.yaml` file begins with key parameters to identify the Dataplex resources including the project ID, region, and names of the Dataplex lake and zone.

Next, it specifies the allowed rule dimensions and two primary rules:

- The rule for **NOT_NULL** values refers to the completeness dimension such as null values.
- The rule for **VALID_EMAIL** values refers to the conformance dimension such as invalid values.

Last, the rules are bound to entities (tables) and columns using rule bindings for data quality validation:

- The first rule binding named **VALID_CUSTOMER** binds the **NOT_NULL** rule to the **id** column of the **contact_info** table, which will validate if the ID column has any NULL values.
- The second rule binding named **VALID_EMAIL_ID** binds the **VALID_EMAIL** rule to the **email** column of the **contact_info** table, which will check for valid emails.

4. Enter `Ctrl+X`, then `Y`, to save and close the file.

Upload the file to Cloud Storage

- In Cloud Shell, run the following command to upload the file to a Cloud Storage bucket that has been created for this lab:

```
gsutil cp dq-customer-raw-data.yaml gs://qwiklabs-gcp-02-62755358c608-buc content_co
```

Click *Check my progress* to verify the objective.



Create and upload a data quality specification file

Check my progress

Assessment Completed!

Task 4. Define and run a data quality job in Dataplex

The data quality process uses a data quality specification YAML file to run a data quality job and generates data quality metrics that are written to a BigQuery dataset.

In this task, you define and run a data quality job using the data quality specification YAML file uploaded to Cloud Storage in the previous task. When you define the job, you also specify a pre-created BigQuery dataset named **customer_dq_dataset** to store the data quality results.

1. In the Google Cloud Console, in the **Navigation menu** (≡), navigate to **Analytics > Dataplex**.
2. Under **Manage lakes**, click **Process**.

3. Click **Create task**.

4. Under Check Data Quality, click **Create task**.

5. Enter the required information to create a new data quality job:

Property	Value
Dataplex lake	ecommerce-lake
Display name	Customer Data Quality Job
ID	Leave the default value.
Select GCS file	qwiklabs-gcp-02-62755358c608 -bucket/dq-customer-raw-data.yaml
Select BigQuery dataset	qwiklabs-gcp-02-62755358c608 .customers_dq_dataset
BigQuery table	dq_results
User service account	Compute Engine default service account

Leave the other default values.

Note that the Compute Engine default service account has been preconfigured for this lab to have the appropriate IAM roles and permissions. For more information, review the Dataplex documentation titled [Create a service account](#).

6. Click **Continue**.

7. For **Start**, select **Immediately**.

8. Click **Create**.

It can take several minutes for the job to run. You may need to refresh the page to see that the job has run successfully.

Lakes

All Lakes

▼

RESET

TASKS

SCHEDULED QUERIES

SCHEDULED NOTEBOOKS

Filter by: [DATAFLOW PIPELINES](#) [DATA QUALITY](#) [CUSTOM SPARK](#)

 Filter Filter tasks

Name	Lake	Task template	Last run
Customer Data Quality Job	ecommerce-lake	Check Data Quality	✓ Succeeded , June 30, 2023 at 3:14:13 PM UTC-5

Click *Check my progress* to verify the objective.




Define and run a data quality job in Dataplex

[Check my progress](#)

Assessment Completed!

Task 5. Review data quality results in BigQuery

In this task, you review the tables in the **customers_dq_dataset** to identify records that are missing customer ID values or have an invalid values for emails.

1. In the Google Cloud Console, in the **Navigation menu** () , navigate to **BigQuery > SQL Workspace**.
2. In the Explorer pane, expand the arrow next to your project ID to list the contents: **qwiklabs-gcp-02-62755358c608**

3. Expand the arrow next to the **customer_dq_dataset** dataset.

4. Click on the **dq_summary** table.

5. Click on the **Preview** tab to see the results.

The **dq_summary** table provides useful information about the overall data quality including the number of records that were identified to not adhere to the two rules in the data quality specification file.

6. Scroll to the last column named **failed_records_query**.

7. Click on the down arrow in the first row to expand the text and view the entire query for the **VALID_EMAIL** rule results.

Note that the query is quite long and ends with `ORDER BY _dq_validation_rule_id`.

8. Click on **Compose new query**. Copy and paste the query into SQL Editor, and click **Run**.

The results of the query provide the email values in the **contact_info** table that are not valid.

Query results

SAVE RESULTS

EXPLORE DATA

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		EXECUTION GRAPH	
Row	_dq_validation_invocation_id	_dq_validation_rule_binding_id	_dq_validation_rule_id	_dq_validation_column_id	_dq_validation_column_value	_dq_validation_dimension	
1	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	hicks	CONFORMANCE	
2	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	arthur.diaz	CONFORMANCE	
3	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	anthony.morton	CONFORMANCE	
4	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	example.com	CONFORMANCE	
5	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	sydney	CONFORMANCE	
6	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	@	CONFORMANCE	
7	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	example.com	CONFORMANCE	
8	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	adam.neuman	CONFORMANCE	
9	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	example.com	CONFORMANCE	
10	fce82f48-1a38-4d88-8309-58a...	VALID_EMAIL_ID	VALID_EMAIL	email	duane	CONFORMANCE	

9. Repeat steps 7-8 for the second cell that contains the query for the **VALID_CUSTOMER** rule results.

The results of the query identify that there are 10 records in the **contact_info** table that are missing ID values.

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		EXECUTION GRAPH		
Row	_dq_validation_invocation_id	_dq_validation_rule_binding_id	_dq_validation_rule_id	_dq_validation_column_id	_dq_validation_column	_dq_validation_dimension	_dq_validation_result	
1	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
2	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
3	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
4	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
5	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
6	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
7	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
8	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
9	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	
10	fce82f48-1a38-4d88-8309-58a...	VALID_CUSTOMER	NOT_NULL	id	null	COMPLETENESS	false	

Click *Check my progress* to verify the objective.



Review data quality results in BigQuery table

Check my progress

Assessment completed!

Congratulations!

You assessed data quality using Dataplex by creating a custom data quality specification file and using it to run a data quality job on a BigQuery table.

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 July 04, 2023

Lab Last Tested July 04, 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.