

[Start Lab](#)

02:15:00

Creating and Populating a Bigtable Instance

 Lab  1 hour 30 minutes  No cost  Introductory**GSP1054**

Google Cloud Self-Paced Labs

Lab instructions and tasks

GSP1054

Overview

Setup and requirements

Task 1. Create a Bigtable instance

Task 2. Create a new Bigtable table

Task 3. Load data files from Cloud Storage using a Dataflow template

Task 4. Verify data loaded into Bigtable

Task 5. Delete a Bigtable table and instance

Congratulations!

Overview

[Bigtable](#) is Google's fully managed, scalable NoSQL database service. Bigtable is ideal for storing large amounts of data in a key-value store and for use cases such as personalization, ad tech, financial tech, digital media, and Internet of Things (IoT). Bigtable supports high read and write throughput at low latency for fast access to large amounts of data for processing and analytics.

For personalization use cases, Bigtable can handle both high writes to store users' interactions with online products and high reads to provide these interactions to models that produce personalized recommendations.

In this lab, you use the Google Cloud Console to create a Bigtable instance with a table to store user interactions with products. Then, you use a Dataflow template to populate the table from pre-generated data files on Cloud Storage. After the Dataflow job has finished, you verify that the table has been successfully populated with data and then complete the lab by deleting the Bigtable data.

What you'll do

In this lab, you learn how to create a Bigtable instance and load data from Cloud Storage without writing any code.

- Create a Bigtable instance and a Bigtable table with column families.
- Use a Dataflow template to load SequenceFile files from Cloud Storage into Bigtable.
- Verify the data loaded into Bigtable.
- Delete the Bigtable table and instance.

Prerequisites

- Basic understanding of database concepts and terms such as instances, schemas, and keys
- Completion of the lab titled [Designing and Querying Bigtable Schemas](#)

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.

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 have a 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.

Log in to Google Cloud Console

1. Using the browser tab or window you are using for this lab session, copy the **Username** from the **Connection Details** panel and click the **Open Google Console** button.

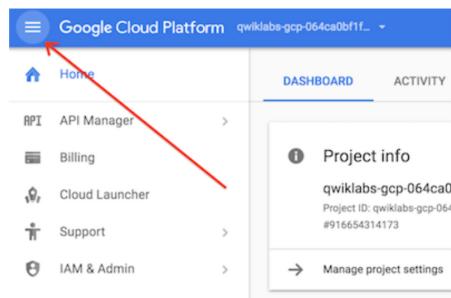
Note: If you are asked to choose an account, click **Use another account**.

2. Paste in the **Username**, and then the **Password** as prompted.
3. Click **Next**.
4. Accept the terms and conditions.

Since this is a temporary account, which will last only as long as this lab:

- Do not add recovery options
- Do not sign up for free trials

5. Once the console opens, view the list of services by clicking the **Navigation menu** (\equiv) at the top-left.



Activate Cloud Shell

Cloud Shell is a virtual machine that contains development tools. It offers a persistent 5-GB home directory and runs on Google Cloud. Cloud Shell provides command-line access to your Google Cloud resources. `gcloud` is the command-line tool for Google Cloud. It comes pre-installed on Cloud Shell and supports tab completion.

1. Click the **Activate Cloud Shell** button (💡) at the top right of the console.

2. Click **Continue**.

It takes a few moments to provision and connect to the environment. When you are connected, you are also authenticated, and the project is set to your `PROJECT_ID`.

Sample commands

- List the active account name:

```
gcloud auth list
```



(Output)

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

(Example output)

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

- List the project ID:

```
gcloud config list project
```



(Output)

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

(Example output)

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

Note: Full documentation of `gcloud` is available in the [gcloud CLI overview guide](#).

Check project permissions

Before you begin working on Google Cloud, you must 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** > **Cloud overview**.

The screenshot shows the Google Cloud Platform interface with the navigation bar at the top. Below it, the 'IAM & Admin' section is selected. The main area displays the 'IAM' tab, which includes a table with columns for 'PERMISSIONS' and 'RECOMMENDATIONS HISTORY'. A single row is visible in the table, representing the default compute service account with the role 'editor' assigned.

The screenshot shows the Google Cloud IAM & Admin > IAM page. On the left, there is a sidebar with various navigation options: Identity & Organization, Policy Troubleshooter, Policy Analyzer, Organization Policies, Service Accounts, Workload Identity Federated, Labels, Tags, Settings, Privacy & Security, Identity-Aware Proxy, Roles, and Audit Logs. The main area displays a table titled "Permissions for project 'qwiklabs-gcp-03-e30ac90a32e4'". The table has columns for Type, Principal, Name, and Role. A red box highlights the first row where the Principal is "407543585891-compute@developer.gserviceaccount.com" and the Role is "Editor". Other rows include "Cloud Build Service Account" (Google APIs Service Agent, Editor), "Owner" (admiral@qwiklabs-services-prod.iam.gserviceaccount.com), and "App Engine Admin" (Qwiklabs User Service Account, BigQuery Admin).

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

1. In the Google Cloud console, on the **Navigation menu**, click **Cloud overview**.
2. From the **Project info** card, copy the **Project number**.
3. On the **Navigation menu**, click **IAM & Admin > IAM**.
4. At the top of the **IAM** page, click **Add**.
5. For **New principals**, type:

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

6. For **Select a role**, select **Basic (or Project) > Editor**.
7. Click **Save**.

Dataflow API

Ensure that the Dataflow API is successfully enabled

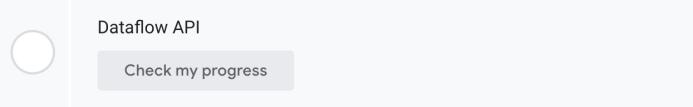
To ensure access to the necessary API, restart the connection to the Dataflow API.

1. In the Google Cloud Console, enter **Dataflow API** in the top search bar.
2. Click on the result for **Dataflow API**.
3. Click **Manage**.
4. Click **Disable API**.

If you are asked to confirm, click **Disable**.

5. Click **Enable**.

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



Task 1. Create a Bigtable instance

To create a new table in Bigtable, you first need to create a Bigtable instance to store your table.

1. In the Google Cloud Console, on the **Navigation menu** (≡), under **Databases**, click **Bigtable**.

2. Click **Create instance**.

3. Enter the required information to create a Bigtable instance:

Property	Value
Instance name	Personalized Sales
Instance ID	Leave the default value
Storage Type	SSD

Task 1. Create a Bigtable instance

To create a new table in Bigtable, you first need to create a Bigtable instance to store your table.

1. In the Google Cloud Console, on the **Navigation menu** (≡), under **Databases**, click **Bigtable**.

2. Click **Create instance**.

3. Enter the required information to create a Bigtable instance:

Property	Value
Instance name	Personalized Sales
Instance ID	Leave the default value
Storage Type	SSD

Task 1. Create a Bigtable instance

To create a new table in Bigtable, you first need to create a Bigtable instance to store your table.

1. In the Google Cloud Console, on the **Navigation menu** (≡), under **Databases**, click **Bigtable**.

2. Click **Create instance**.

3. Enter the required information to create a Bigtable instance:

Property	Value
Instance name	Personalized Sales
Instance ID	Leave the default value
Storage Type	SSD

Task 1. Create a Bigtable instance

To create a new table in Bigtable, you first need to create a Bigtable instance to store your table.

1. In the Google Cloud Console, on the **Navigation menu** (≡), under **Databases**, click **Bigtable**.

2. Click **Create instance**.

3. Enter the required information to create a Bigtable instance:

Property	Value
Instance name	Personalized Sales
Instance ID	Leave the default value
Storage Type	SSD

Task 3. Load data files from Cloud Storage using a Dataflow template

In this task, you run a Dataflow job to load data from Cloud Storage to Bigtable. In order to run the Dataflow job successfully, you first have to create a Cloud Storage bucket for Dataflow to write temporary files as needed. Then you can successfully create and run a new Dataflow job from a template.

Create a Cloud Storage bucket

1. In the Google Cloud Console, on the **Navigation menu** (≡), click **Cloud overview > Dashboard**.

2. Under **Project info**, copy the value for **Project ID** (such as **Project ID**).

You will use your Qwiklabs Project ID, which is already globally unique, as the Cloud Storage bucket name.

3. In the Google Cloud Console, on the **Navigation menu** (≡), click **Cloud Storage > Buckets**.

4. Click **Create Bucket**.

5. Enter the required information to create a Cloud Storage bucket, replacing *project-id* with the project ID you copied in step 2.

Property	Value
Name	<i>project-id</i>
Location type	Multi-region
Location	us (multiple regions in United States)

Leave the default values for the remaining parameters.

6. Click **Create**.

7. If prompted **Public access will be prevented**, click **Confirm**.

Create a Dataflow job using a template

1. In the Google Cloud Console, on the **Navigation menu** (≡), under **Analytics**, click **Dataflow > Jobs**.

2. Click **Create job from template**.

3. Enter the required information to create a Dataflow job from a template, replacing *project-id* with the project ID you previously copied.

Property	Value

Job name	import-usersessions
Regional endpoint	-----
Dataflow template	SequenceFile Files on Cloud Storage to Cloud Bigtable
Project ID	project-id
Instance ID	personalized-sales
Table ID	UserSessions
Source path pattern	gs://cloud-training/OCBL377/retail-interactions-sales-00000-of-00001
Temporary location	gs://project-id/temp

Leave the default values for the remaining parameters.

4. Click **Run Job**.

5. On the **Job Graph** page, under **Job steps view**, select **Graph view**.

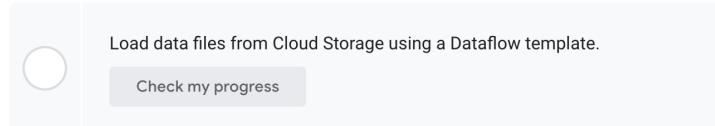
The **Graph view** displays a graph of how the job progresses to complete the following steps:

- Read the SequenceFiles on Cloud Storage.
- Mutate the data for loading into Bigtable.
- Write the data to Bigtable.

6. To see a table view of the same steps, select **Table view**.

When the job has successfully completed, a green check mark for a *Succeeded* status is displayed next to each task in the Job Graph. This job will take approximately 3 to 5 minutes to run successfully.

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



Task 4. Verify data loaded into Bigtable

After your Dataflow job has successfully completed, you can use `cbt` (Cloud Bigtable command-line tool) [commands](#) to connect to your Bigtable instance and verify that the table has been populated with data.

Configure the Bigtable CLI

To connect to Bigtable using `cbt` CLI commands, you first need to update the `.cbtrc` configuration file with your project ID and your Bigtable instance ID using Cloud Shell.

For a review of how to access Cloud Shell, click **Setup and Requirements** on the right-side menu of this page.

1. To modify the `.cbtrc` file with the project ID and instance ID, run the following commands in Cloud Shell:

```
echo project = `gcloud config get-value project` \
>> ~/.cbtrc
```



```
echo instance = personalized-sales \
```

After your Dataflow job has successfully completed, you can use cbt (Cloud Bigtable command-line tool) [commands](#) to connect to your Bigtable instance and verify that the table has been populated with data.

Configure the Bigtable CLI

To connect to Bigtable using cbt CLI commands, you first need to update the `.cbtrc` configuration file with your project ID and your Bigtable instance ID using Cloud Shell.

For a review of how to access Cloud Shell, click **Setup and Requirements** on the right-side menu of this page.

1. To modify the `.cbtrc` file with the project ID and instance ID, run the following commands in Cloud Shell:

```
echo project = `gcloud config get-value project` \  
>> ~/.cbtrc
```

```
echo instance = personalized-sales \
```

After your Dataflow job has successfully completed, you can use cbt (Cloud Bigtable command-line tool) [commands](#) to connect to your Bigtable instance and verify that the table has been populated with data.

Configure the Bigtable CLI

To connect to Bigtable using cbt CLI commands, you first need to update the `.cbtrc` configuration file with your project ID and your Bigtable instance ID using Cloud Shell.

For a review of how to access Cloud Shell, click **Setup and Requirements** on the right-side menu of this page.

1. To modify the `.cbtrc` file with the project ID and instance ID, run the following commands in Cloud Shell:

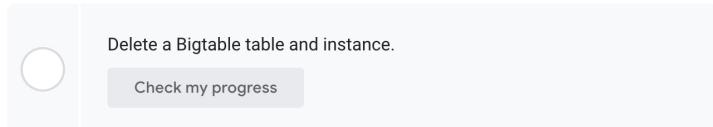
```
echo project = `gcloud config get-value project` \  
>> ~/.cbtrc
```

```
echo instance = personalized-sales \  
Interactions:blue_shoes @ 2022/06/07-18:34:28.858000  
    "viewed details"  
Interactions:green_skirt @ 2022/06/07-18:34:28.858000  
    "seen"  
Interactions:orange_dress @ 2022/06/07-18:34:28.858000  
    "seen"  
Interactions:purple_dress @ 2022/06/07-18:34:28.858000  
    "seen"  
Interactions:purple_skirt @ 2022/06/07-18:34:28.858000  
    "seen"  
Interactions:user_id @ 2022/06/07-18:34:28.858000  
    "blue1"  
Interactions:yellow_blouse @ 2022/06/07-18:34:28.858000  
    "seen"  
Sales:sale @ 2022/06/07-18:34:28.858000  
    "blue_blouse"
```

Task 5. Delete a Bigtable table and instance

1. In the Google Cloud Console, on the **Navigation menu** (≡), under **Databases**, click **Bigtable**.
2. From the list of Bigtable Instances, click on the Instance ID named **personalized-sales**.
3. In the navigation menu, under **Instance**, click **Tables**.
4. Click on the table named **UserSessions**.
5. Click **Delete table**.
6. In the confirmation dialog, type **UserSessions**
7. Click **Delete**.
8. In the navigation menu, under **Instance**, click **Overview**.
9. Click **Delete instance**.
10. In the confirmation dialog, type **personalized-sales**
11. Click **Delete**.

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



Congratulations!

In this lab, you used Bigtable to create a new instance and table, loaded data into the table using a Dataflow template, and confirmed that the data was successfully loaded by running simple `cbt` CLI commands. Then, you completed the lab by deleting the Bigtable table and instance.

Next steps / Learn more

- Check out the lab titled [Streaming Data to Bigtable](#).

Manual Last Updated April 25, 2024

Lab Last Tested February 17, 2023

Copyright 2022 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.