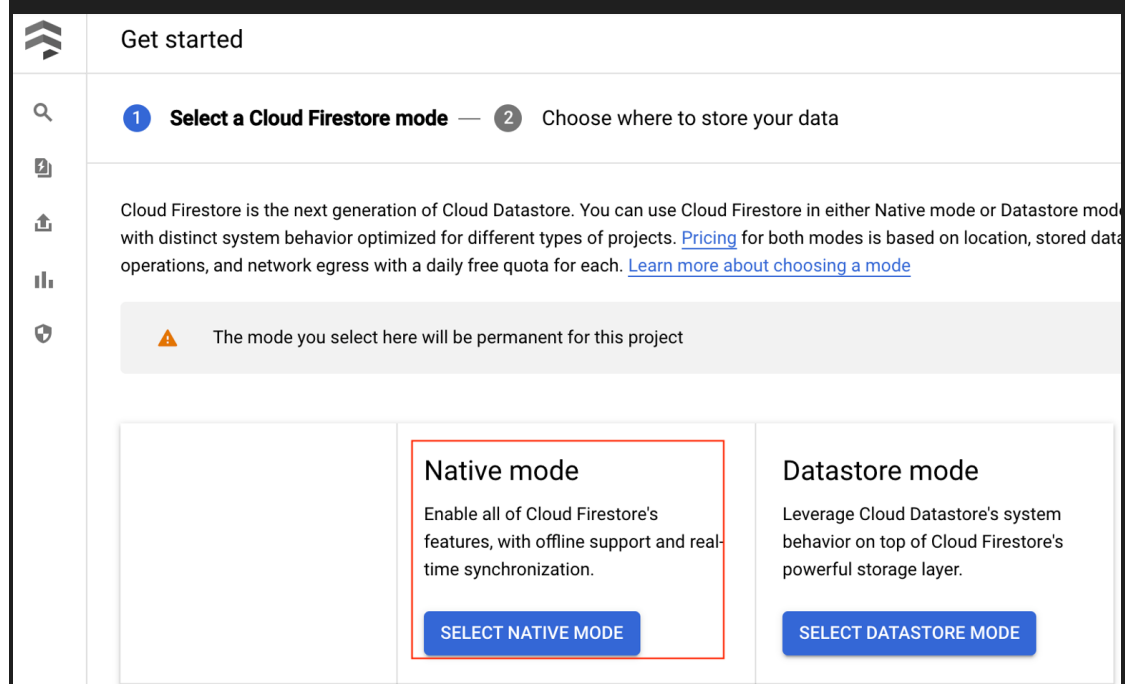


This document provides guidance for connecting to Google's Cloud Console, Firestore. The content and images are primarily derived from Dimitris Trihinas' notes for the Big Data Management and Processing course in the MSc in Data Science, with some modifications made by me.

Getting Started – preliminary.

1. To get started, navigate to the Google Cloud console and select an existing project or create a new one from the <https://console.cloud.google.com/projectselector2/home/dashboard>) view.
2. After selecting a project, select from the navigation menu Firestore under the Databases segment.
3. From the Select a database service screen, choose Firestore in **Native** mode.



4. Select a location for your database (i.e., europe `eur3`) and click, Create Database.

Get started

✓ Select a Cloud Firestore mode — ✓ Choose where to store your data

You selected Cloud Firestore in Native mode. Now choose a database location.

The location of your database affects its cost, availability, and durability. Choose a regional location (lower write latency, lower cost) or a multi-region location (higher availability, higher cost). [Learn more](#)

⚠ Choose carefully: your location selection is permanent and will also apply to this project's App Engine app

Select a location
eur3 (Europe)

To improve performance, store your data close to the users and services that need it

CREATE DATABASE BACK

Setting up the Authentication

1. In Google Cloud navigate to the 'Create Service Account' with this <https://console.cloud.google.com/projectselector/iam-admin/serviceaccounts/create?supportedpurview=project>).
2. Select the project you have associated with the created database.
3. In the `Service account name` field, enter a name. The Cloud Console fills in the Service account ID field based on this name. When done, click **Create and Continue**.

Google Cloud Platform

comp548-bigdata

Search products

Create service account

1 Service account details

Service account name

dttrihinas

Display name for this service account

Service account ID

dttrihinas

@comp548-bigdata.iam.gserviceaccount.com

X

↺

Service account description

Describe what this service account will do

CREATE AND CONTINUE

4. Click the *Select a role* field. Under Quick access, click **Basic**, then click **Owner**.

Google Cloud Platform comp548-bigdata Search products and resources

Create service account

✓ Service account details

2 Grant this service account access to project (optional)

Grant this service account access to comp548-bigdata so that it has permission to complete specific actions on the resources in your project. [Learn more](#)

Select a role Condition

Type to filter

Quick access	Browser
Currently used	Editor
Basic	Owner
All roles	Viewer
Access Approval	
Access Context Manager	
Actions	

Owner
Full access to all resources.

3

DONE

MANAGE ROLES

5. After, click **Done** to finish creating the service account.

Below, we create our key - Important for the connection

6. From the **service account view** click on the service account you have just created.

Google Cloud Platform comp548-bigdata Search products and resources

Service accounts + CREATE SERVICE ACCOUNT DELETE + MANAGE ACCESS

Service accounts for project "comp548-bigdata"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. [Learn more about service accounts.](#)

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. [Learn more service account organization policies.](#)

Filter Enter property name or value

Email	Status	Name	Description	Key ID	Key creation date	Actions
comp548-bigdata@appspot.gserviceaccount.com	✓	App Engine default service account		No keys		
comp548-bigdata@comp548-bigdata.iam.gserviceaccount.com	✓	comp548-bigdata		No keys		
dtrihinas@comp548-bigdata.iam.gserviceaccount.com	✓	dtrihinas		b6c9c00380b073a7c19681050e2905e9e97e62c2	Jun 30, 2021	

7. Go to the **Keys** tab and click **ADD KEY** and then **Create Key** (if given an option, use as key type ``json``). Download your key to your local environment.

Google Cloud Platform comp548-bigdata Search products and resources

← dtrihinas

DETAILS PERMISSIONS **KEYS** METRICS LOGS

Keys

Service account keys could pose a security risk if compromised. We recommend you avoid downloading service account keys and learn more about the best way to authenticate service accounts on Google Cloud [here](#).

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using [organization policies](#).
[Learn more about setting organization policies for service accounts](#)

ADD KEY ▾

- Create new key
- Upload existing key

Key	Key creation date	Key expiration date	
b6c9c00380b073a7c19681050e2905e9e97e62c2	Jun 30, 2021	Jan 1, 10000	

Note: the below only needs to be performed once and then you can use it for as many databases as you need to associate with your google cloud project.

8. Now, that you have downloaded your key, it is time to use it in your python application/notebook through the firestore SDK. To do so, we must first set an environment variable for the current session. This is OS dependent. To avoid having to write instructions for linux, macOS, windows, android, etc in a notebook environment you may use the following:

```
%env GOOGLE_APPLICATION_CREDENTIALS=TYPE_HERE_THE_PATH_TO_YOUR_KEY
```

The line above is the first line in my Jupyter Notebook Project file. Hence, that concludes the explanation and setting-up stage, and we proceed with the ipynb file.