Use Google Dataproc to Run Hadoop Wordcount Program

Dataproc is a Google Cloud Platform managed service for Spark and Hadoop which helps you with Big Data Processing, ETL, and Machine Learning. It provides a Hadoop cluster and supports Hadoop ecosystems tools like Flink, Hive, Presto, Pig, and Spark.

Dataproc is an auto-scaling cluster which manages logging, monitoring, cluster creation of your choice and job orchestration. You'll need to manually provision the cluster, but once the cluster is provisioned you can submit jobs to Spark, Flink, Presto, and Hadoop.

How to Create a Dataproc Cluster

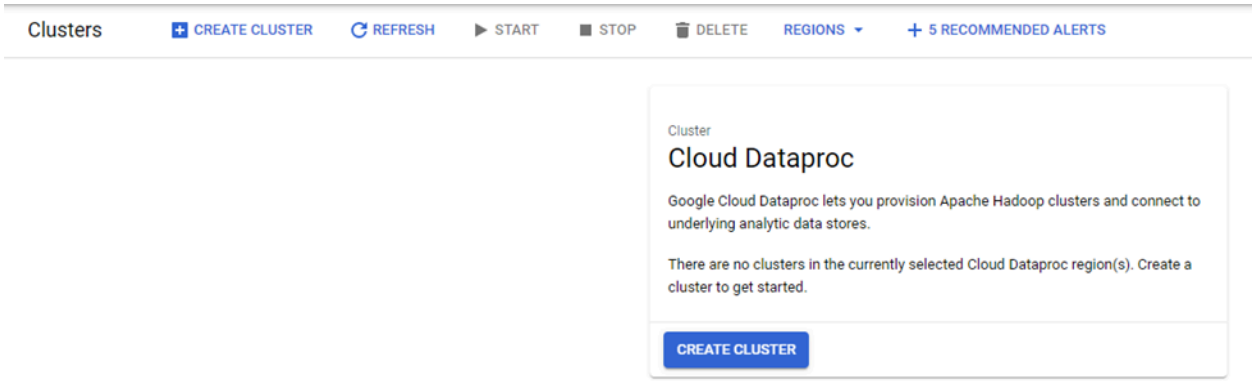
Dataproc has three cluster types:

1. Standard
2. Single Node
3. High Availability

The Standard cluster can consist of 1 master and N worker nodes. The Single Node has only 1 master and 0 worker nodes. For production purposes, you should use the High Availability cluster which has 3 master and N worker nodes.

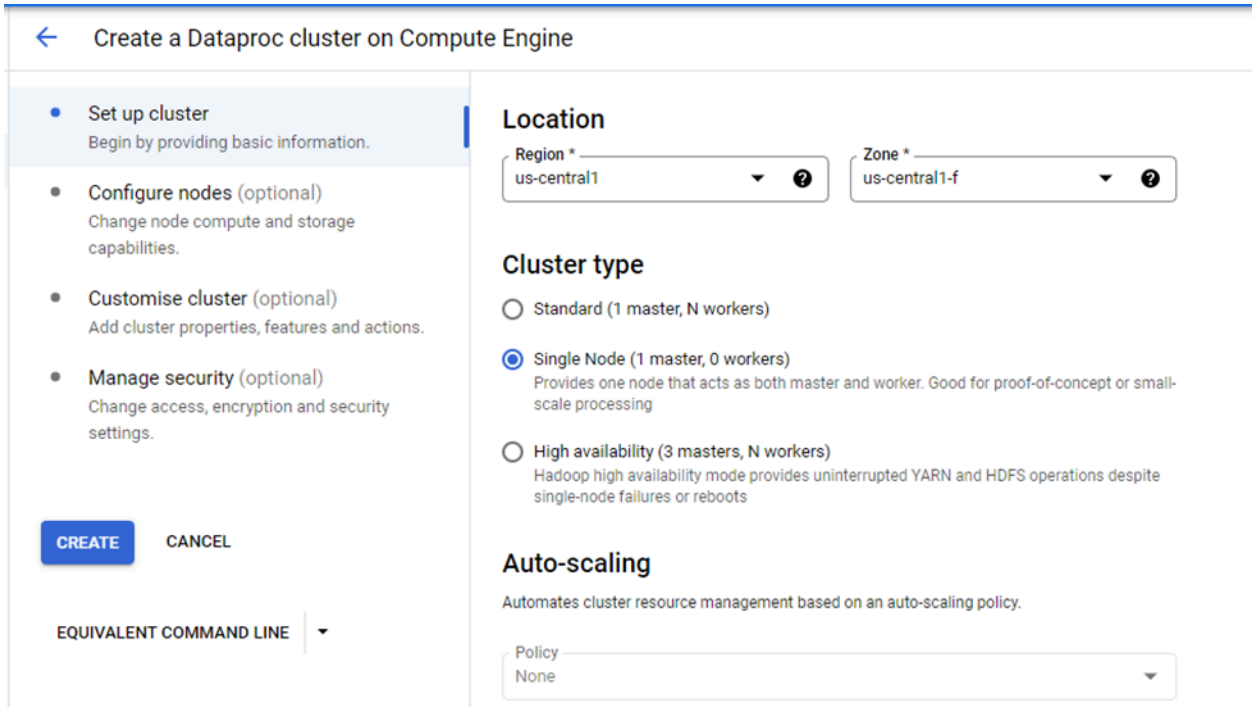
For our learning purposes, a single node cluster is sufficient which has only 1 master Node.

Creating Dataproc clusters in GCP is straightforward. First, we'll need to enable Dataproc, and then we'll be able to create the cluster.



Start Dataproc cluster creation

When you click "Create Cluster", GCP gives you the option to select Cluster Type, Name of Cluster, Location, Auto-Scaling Options, and more.



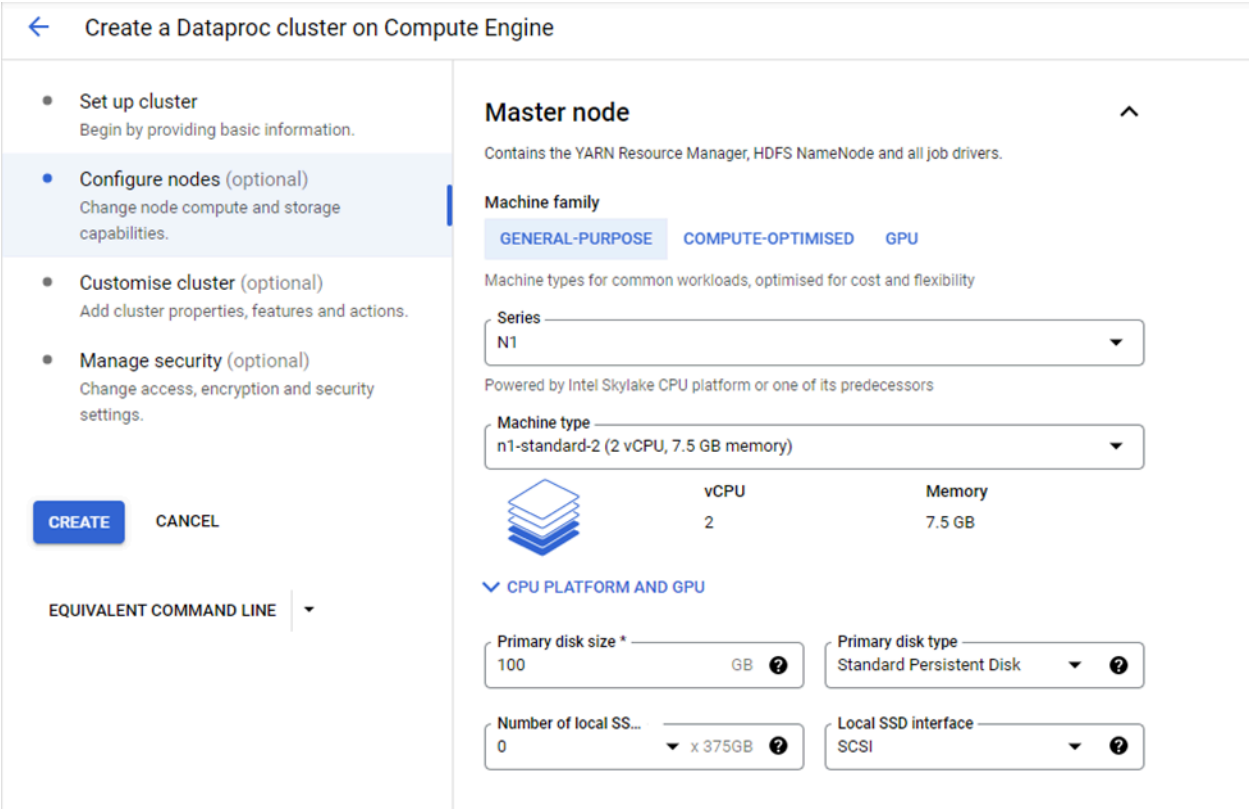
Parameters required for Cluster

Since we've selected the Single Node Cluster option, this means that auto-scaling is disabled as the cluster consists of only 1 master node.

The Configure Nodes option allows us to select the type of machine family like Compute Optimized, GPU and General-Purpose.

In this tutorial, we'll be using the General-Purpose machine option. Through this, you can select Machine Type, Primary Disk Size, and Disk-Type options.

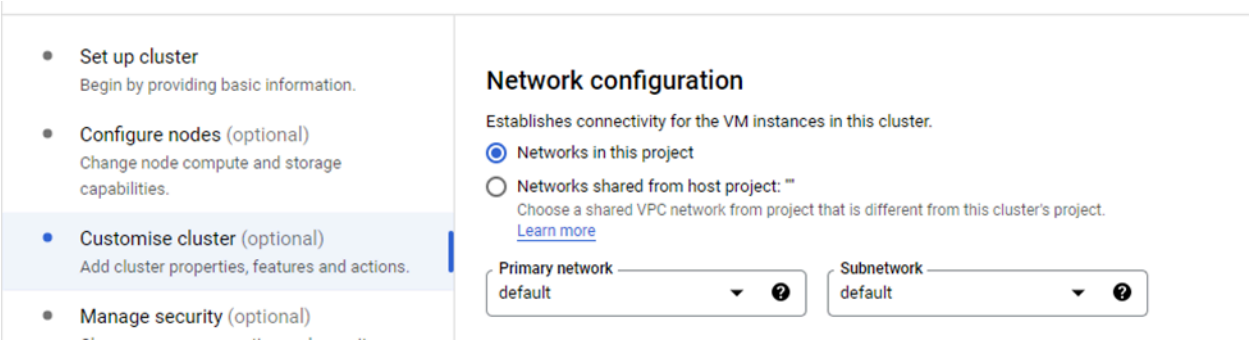
The Machine Type we're going to select is n1-standard-2 which has 2 CPU's and 7.5 GB of memory. The Primary Disk size is 100GB which is sufficient for our demo purposes here.



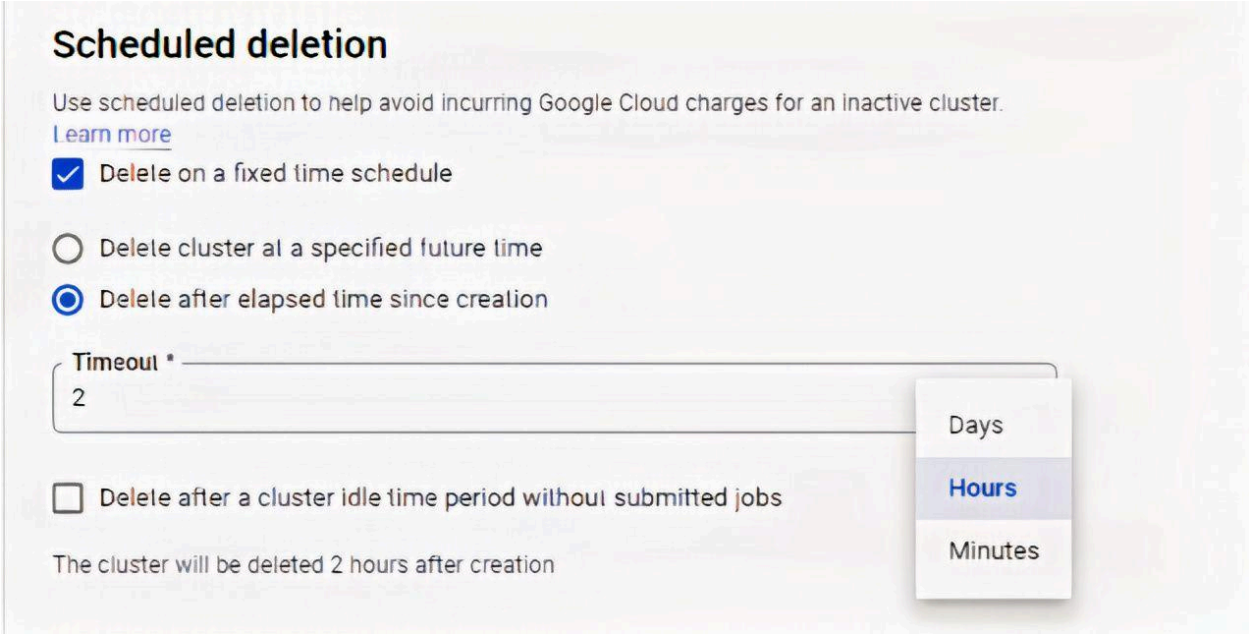
Master Node Configuration

We've selected the cluster type of Single Node, which is why the configuration consists only of a master node. If you select any other Cluster Type, then you'll also need to configure the master node and worker nodes.

From the Customise Cluster option, select the default network configuration:



Use the option "Scheduled Deletion" in case no cluster is required at a specified future time (or say after a few hours, days, or minutes).



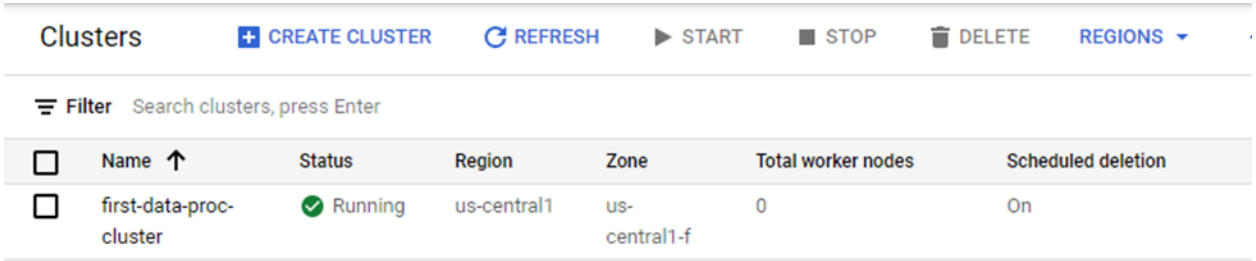
Schedule Deleting Setting

Here, we've set "Timeout" to be 2 hours, so the cluster will be automatically deleted after 2 hours.

We'll use the default security option which is a Google-managed encryption key. When you click "Create", it'll start creating the cluster.

You can also create the cluster using the ‘gcloud’ command which you'll find on the ‘EQUIVALENT COMMAND LINE’ option as shown in image below.

After a few minutes the cluster with 1 master node will be ready for use.



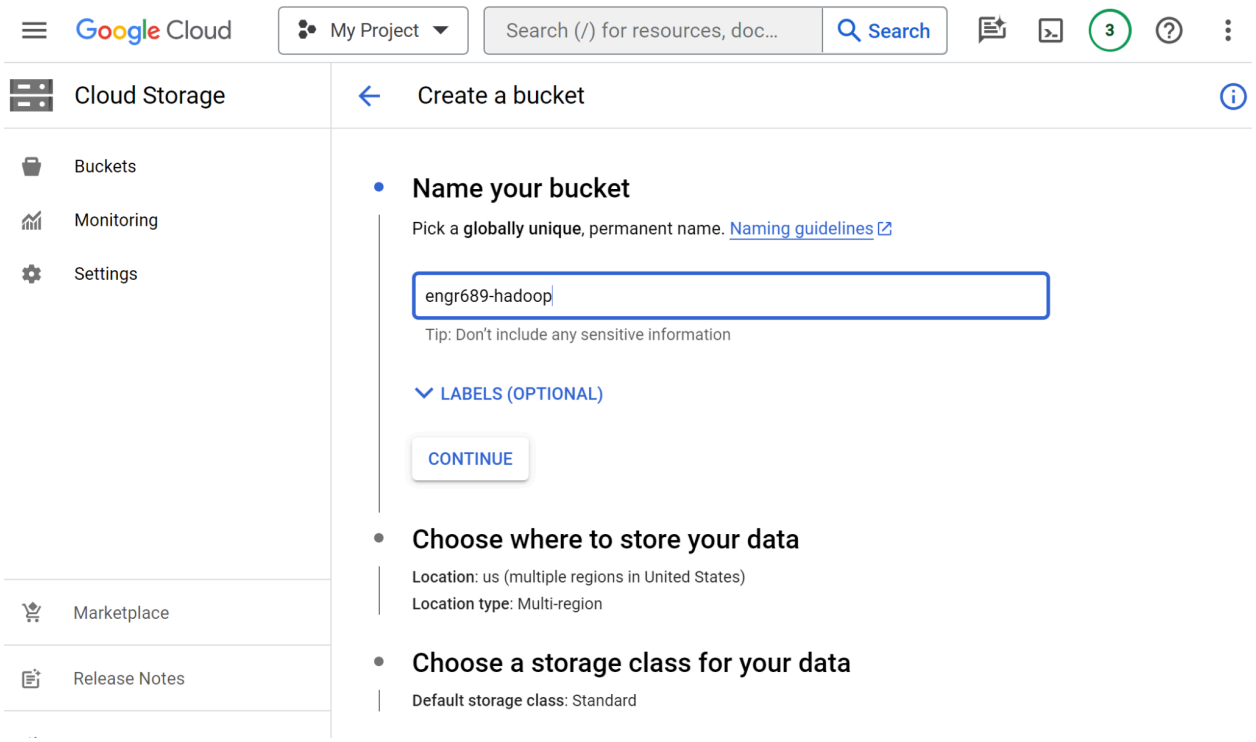
Cluster Up and Running

Create a Bucket in the Google Cloud Storage

Before running the Hadoop job, you need to create a bucket in Google Cloud Storage for storing the output of Hadoop. Click the following link to access Google Cloud Storage, or search “Google Cloud Storage” in the search bar on the top of Google Cloud Platform.

Go to: [Google Cloud Storage Console]

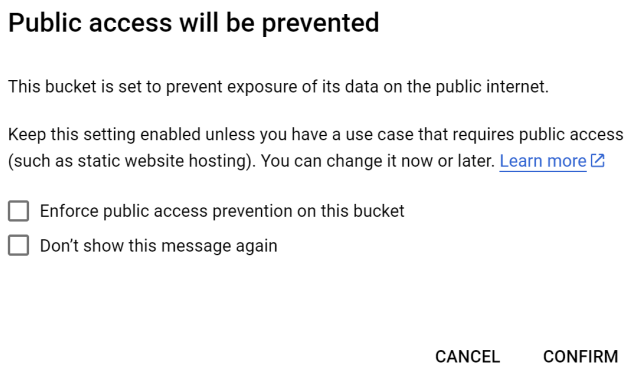
Be sure to choose the same project. In the Buckets tab, click CREATE to create a new bucket:



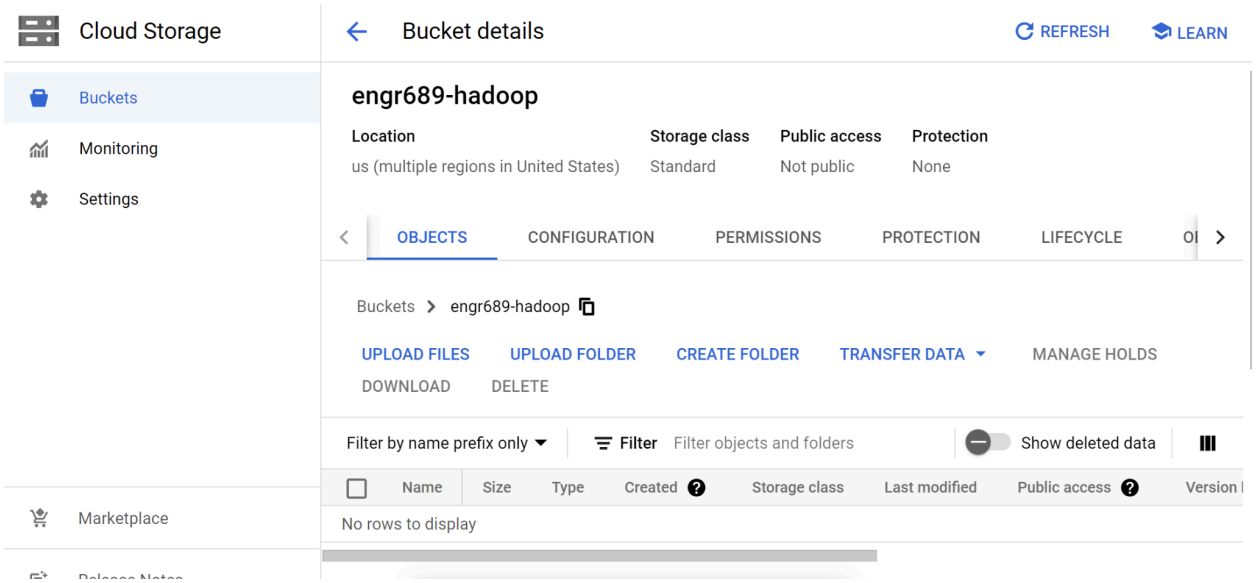
You need to give a **<u>globally unique name</u>** to the bucket, such as "engr689-hadoop-<your name>", and click "CONTINUE". If the name is already used by someone else, find a new name for the bucket.

For the rest of the options, you all leave the default options and click "CONTINUE" until the last step.

Finally, click "CREATE" to create the bucket. You may be asked "Whether you want to prevent Public Access", be sure to unclick the button of "Enforce public access prevention on this bucket":



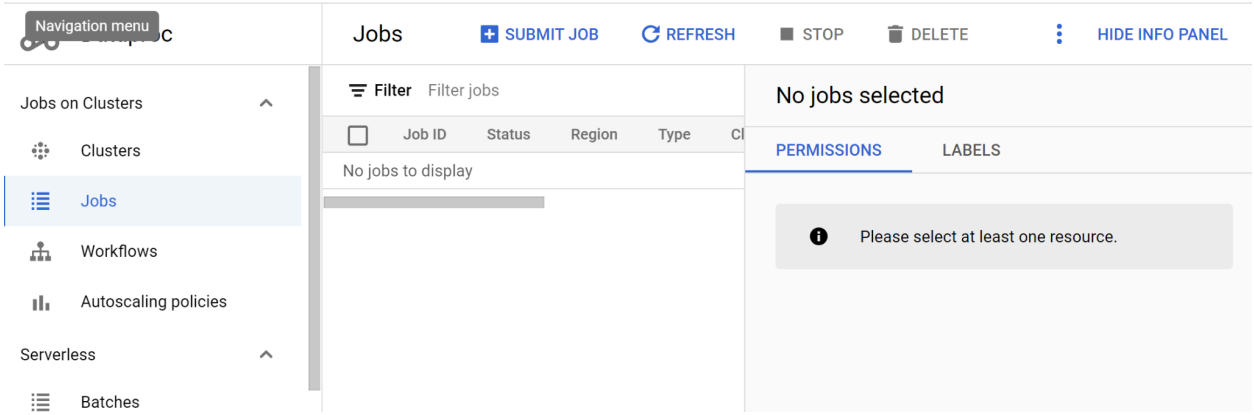
You shall see your bucket in the list. You can click the bucket to further see the objects inside (which is nothing right now).



Submit a Hadoop Job

Now, we will submit a Hadoop Job to run the Wordcount example, to show that how you can use Hadoop to process a lot of data. You do not have to write Java code in this exercise, since there is already a Wordcount example in the existing Java libraries located on the cluster Mater node.

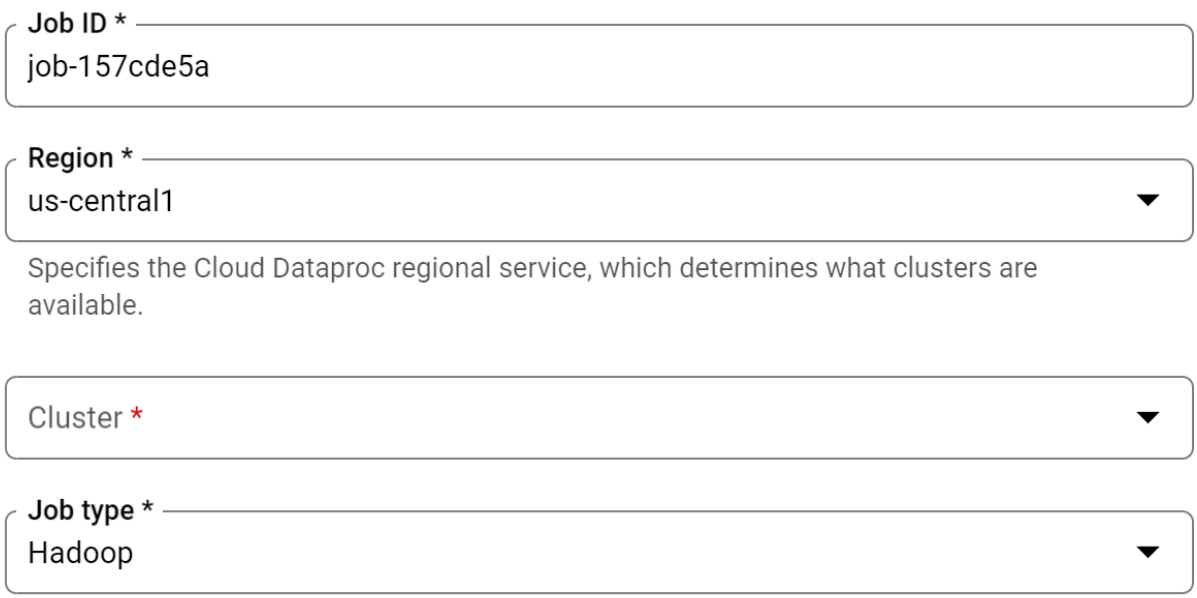
To submit a new job, click the "Jobs" task in Google Dataproc:



Click 

to create a new Hadoop job:

On the configuration page, first choose the cluster you just created, and make sure the Job type is “Hadoop”:

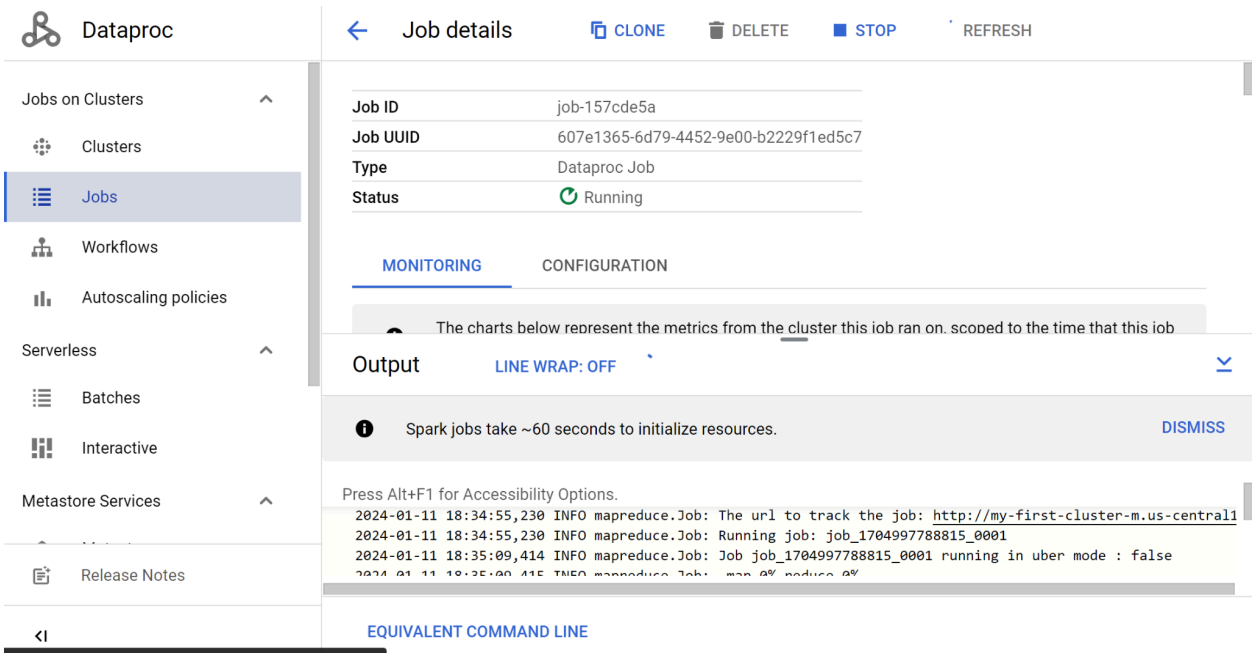


For the rest of the options, enter as follows:

- MAIN class
  - needs fully-qualified name

    - case-sensitive main function
    - `org.apache.hadoop.examples.WordCount`
- JAR file with examples
    - examples included with base image
    - verify version of jar file
    - `file:///usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar`
- ARGS
    - are positional `<in>`, `<out>`
    - `gs://tamu-engr689/alice_in_wonderland.txt (press ENTER)`
    - `gs://<Your bucket name>/output` (press ENTER again) - this CREATES the output directory, the directory should NOT exist in advance

Then press “SUBMIT”. The Job will be submitted and start running.



Finally, wait for the job to finish and check the log:

Job ID	job-157cde5a
Job UUID	607e1365-6d79-4452-9e00-b2229f1ed5c7
Type	Dataproc Job
Status	Succeeded

Check the Output on Google Cloud Storage

Upon the completion of the Hadoop job, you can head over to Google Cloud Storage to see the results. You shall find multiple files in the “output” directory of your bucket:

Filter by name prefix only | Filter | Filter objects and folders | Show deleted data

Name	Size	Type	Created
_SUCCESS	0 B	application/octet-stream	Jan 11, 2024, 12:
part-r-00000	16.8 KB	application/octet-stream	Jan 11, 2024, 12:
part-r-00001	16.6 KB	application/octet-stream	Jan 11, 2024, 12:
part-r-00002	16.3 KB	application/octet-stream	Jan 11, 2024, 12:

Now, download one of the output files (except the _SUCCESS file) and upload to Canvas.

Delete the Dataproc Cluster and Google Cloud Storage

Finally, you can delete the dataproc cluster and the bucket you created in the Google Cloud Storage.