DataProc – Qwik Start – Console

Confirm Cloud Dataproc API is enabled

To create a Dataproc cluster in Google Cloud, the Cloud Dataproc API must be enabled. To confirm the API is enabled:

1. Click **Navigation menu** > **APIs & Services** > **Library**:
2. Type **Cloud Dataproc** in the **Search for APIs & Services** dialog. The console will display the Cloud Dataproc API in the search results.
3. Click on **Cloud Dataproc API** to display the status of the API. If the API is not already enabled, click the **Enable** button.

If the API's enabled, you're good to go:

**Task 1. Create a cluster**

1. In the Cloud Platform Console, select **Navigation menu** > **Dataproc** > **Clusters**, then click **Create cluster**.
2. Click **Create** for **Cluster on Compute Engine**.
3. Set the following fields for your cluster and accept the default values for all other fields:

|  |  |
| --- | --- |
| **Field** | **Value** |
| Name | example-cluster |
| Region | us-central1 |
| Zone | us-central1-a |

**Note:** A Zone is a special multi-region namespace that is capable of deploying instances into all Google Compute zones globally. You can also specify distinct regions, such as us-east1 or europe-west1, to isolate resources (including VM instances and Cloud Storage) and metadata storage locations utilized by Cloud Dataproc within the user-specified region.

1. Click **Create** to create the cluster.

Your new cluster will appear in the Clusters list. It may take a few minutes to create, the cluster Status shows as **Provisioning** until the cluster is ready to use, then changes to **Running**.

**Task 2. Submit a job**

To run a sample Spark job:

1. Click **Jobs** in the left pane to switch to Dataproc's jobs view, then click **Submit job**.
2. Set the following fields to update Job. Accept the default values for all other fields:

|  |  |
| --- | --- |
| **Field** | **Value** |
| Cluster | example-cluster |
| Job type | Spark |
| Main class or jar | org.apache.spark.examples.SparkPi |
| Jar files | file:///usr/lib/spark/examples/jars/spark-examples.jar |
| Arguments | 1. is sets the number of tasks.) |

1. Click **Submit**.

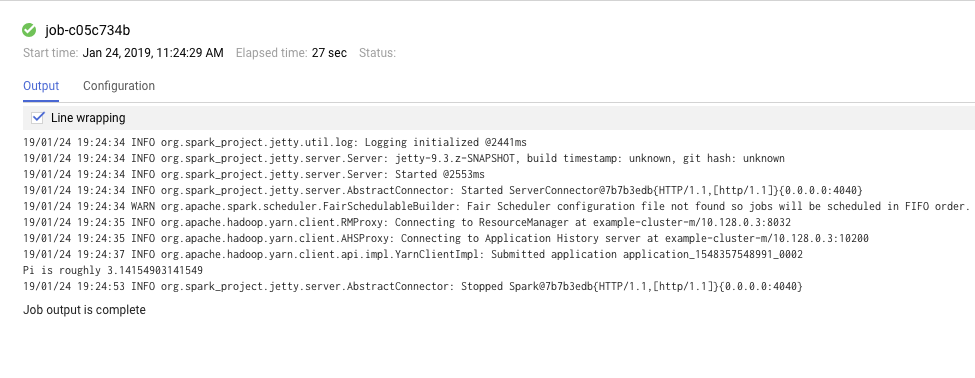
**Note: How the job calculates Pi:** The Spark job estimates a value of Pi using the [Monte Carlo method](https://en.wikipedia.org/wiki/Monte_Carlo_method). It generates x,y points on a coordinate plane that models a circle enclosed by a unit square. The input argument (1000) determines the number of x,y pairs to generate; the more pairs generated, the greater the accuracy of the estimation. This estimation leverages Cloud Dataproc worker nodes to parallelize the computation. For more information, see [Estimating Pi using the Monte Carlo Method](https://academo.org/demos/estimating-pi-monte-carlo/) and see [JavaSparkPi.java on GitHub](https://github.com/Apache/spark/blob/master/examples/src/main/java/org/apache/spark/examples/JavaSparkPi.java).

Your job should appear in the **Jobs** list, which shows your project's jobs with its cluster, type, and current status. Job status displays as **Running**, and then **Succeeded** after it completes.

**Task 3. View the job output**

To see your completed job's output:

1. Click the job ID in the **Jobs** list.
2. Check **Line wrapping** or scroll all the way to the right to see the calculated value of Pi. Your output, with **Line wrapping** checked, should look something like this:



Your job has successfully calculated a rough value for pi!

**Task 4. Update a cluster**

To change the number of worker instances in your cluster:

1. Select **Clusters** in the left navigation pane to return to the Dataproc Clusters view.
2. Click **example-cluster** in the **Clusters** list. By default, the page displays an overview of your cluster's CPU usage.
3. Click **Configuration** to display your cluster's current settings.
4. Click **Edit**. The number of worker nodes is now editable.
5. Enter **4** in the **Worker nodes** field.
6. Click **Save**.

Your cluster is now updated. Check out the number of VM instances in the cluster.

1. To rerun the job with the updated cluster, you would click **Jobs** in the left pane, then click **SUBMIT JOB**.
2. Set the same fields you set in the **Submit a job** section:

|  |  |
| --- | --- |
| **Field** | **Value** |
| Cluster | example-cluster |
| Job type | Spark |
| Main class or jar | org.apache.spark.examples.SparkPi |
| Jar files | file:///usr/lib/spark/examples/jars/spark-examples.jar |
| Arguments | 1000 (This sets the number of tasks.) |

1. Click **Submit**.