Deploy Spark Cluster on Google Cloud Dataproc:

Cloud Dataproc is a fast, easy-to-use, fully-managed cloud service for running Apache Spark and Apache Hadoop clusters in a simpler, more cost-efficient way.

Create a bucket (if you haven't already done so) – The following steps need to be taken only once:

1. Login to your Google Cloud account (https://console.cloud.google.com) and go to Cloud Storage.
2. Create a new bucket. Call it odsc2023-xx (xx you initial – or any other name)
3. Select **Multi-region** and for location select **us**, and leave the rest as default and click on **Create**
4. To test if everything is working fine from Cloud Shell run the following query.
    a. gsutil ls
    b. This should list all of the buckets in your project including the one you just made.

Create a cluster – The following steps need to be done every time we create a new cluster:

1. Login to your Google Cloud (https://console.cloud.google.com) and open a Google Cloud Shell. Make sure you are in the intended project.

2. Run the following script in Cloud Shell to launch a new cluster named *myspark33*. Copy this command to an editor of your choice and make sure to replace <PROJECT-ID> with your project-id (it can be found by clicking on the project name, under ID) and <BUCKET-NAME> with the bucket you created in the previous section:

```
gcloud dataproc clusters create myspark33 \
  --project <PROJECT-ID> \
  --bucket <BUCKET-NAME> \
  --region us-east1 \
  --zone us-east1-b \
  --single-node \
  --master-machine-type e2-highmem-4 \
  --master-boot-disk-size 50 \
  --optional-components=JUPYTER \
  --image-version 2.1-debian11 \
  --initialization-actions gs://dataproc-initialization-actions/python/pip-install.sh \
  --metadata 'PIP_PACKAGES=google-cloud-storage' \
  --enable-component-gateway
```

(This script is also accessible in 05-A-Tour-of-Spark/Deploy-Dataproc-Cluster-Single-Node.txt)

- This will give you a single-node cluster with 50 GB HDD, 4 CPU cores, and 32GB Memory (e2-highmem-4). This cluster will cost about $0.22/hour.

3. From Dataproc page (located under Big Data under the Google Cloud menu) confirm that your cluster is up and running.

Connecting to the Cluster

- From the Dataproc tab, select the cluster and under web interfaces select Jupyter or JupyterLab.
- OR - From Cloud SDK shell execute the following command to establish a secure SSH tunnel:

```
gcloud compute ssh --zone us-east1-b myspark33-m -- -L 2222:localhost:8123 -L 8088:localhost:8088
```

Note: You could get disconnected time to time, if that happened you can simply repeat the step above and refresh your browser. Your work will stay saved in your cluster and in your bucket under *notebook* folder.

Stopping your Cluster

To make sure you don't run out of money please stop your instance once you are not using it. You can do this by going to the Compute Engine page and stop the instance that are being used with your cluster. The name of the instance should be an indicative of its cluster. You can then start them before connecting to your cluster.

Clean up

To make sure we won't be charged for any of the resources we are not using, delete the cluster after each use:

- From the Dataproc page select the cluster and click on the DELETE button.
    - Alternatively, you can also use the following command from **a new** Cloud SDK terminal (please check from the UI to make sure the cluster is being deleted):

```
gcloud dataproc clusters delete myspark33 --region us-east1
```

Note1: Notice that even after deleting the cluster your notebooks will persist in the bucket and when you create a new cluster that points to the same bucket you can simply reuse those notebooks.

Note2: You can't terminate a cluster from within itself. Make sure that you open a new Cloud SDK terminal and you are not using the one that is tunneled to the cluster.

Deploying a Large Cluster

To create a large cluster (CAUTION) with multiple nodes, use the following command:

```
gcloud dataproc clusters create bigspark33 \
 --project <PROJECT-ID> \
 --bucket <BUCKET-NAME> \
 --region us-east1 \
 --zone us-east1-b \
 --master-machine-type e2-highmem-4 \
 --master-boot-disk-size 50 \
 --num-workers 2 \
 --worker-machine-type e2-highmem-2 \
 --worker-boot-disk-size 50 \
 --optional-components=JUPYTER \
 --image-version 2.1-debian11 \
 --initialization-actions gs://dataproc-initialization-actions/python/pip-install.sh \
 --metadata 'PIP_PACKAGES=google-cloud-storage' \
 --enable-component-gateway
```

(This script is also accessible in 05-A-Tour-of-Spark/Deploy-Dataproc-Cluster-Multi-Node.txt)

Important note: Please be advised of its cost. Cost can be calculated using Dataproc cost calculator (make sure to include GCE is selected): https://cloud.google.com/products/calculator/

The cluster above has 1 master node and 2 workers (8 CPUs & 64 GB memory). We could have used some preemptible instances to reduce cost, however, a cluster with preemptible instances cannot be stopped, so it's not ideal for our use-case. Preemptible instances are offered with a big discount (~70%) but will not last for more than 24 hours. Add the following line to include 2 preemptible workers:

```
--num-secondary-workers 2
```

This cluster will cost about $0.44/hour.

Note about preemptible worker pool: One can easily add/remove preemptible workers from the UI (or in the command line). Go to the cluster of interest in Dataproc (make sure the cluster is running) > Configuration > Edit > add/remove secondary workers > save. After ~30 seconds your cluster will reflect the change.

Connecting to the Cluster

Same as the single node cluster mentioned above.

Clean up

```
gcloud dataproc clusters delete bigspark33 --region us-east1
```