

Setting up a Virtual Machine with Linux as the Operating System

Overview

Prerequisites:

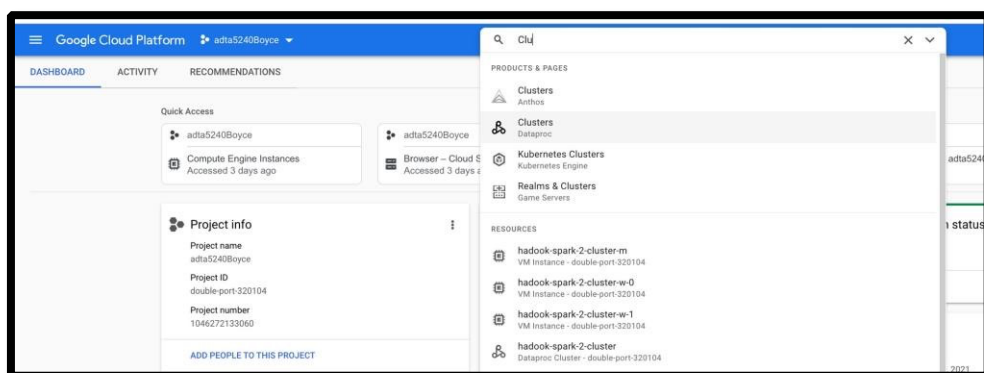
1. Google account OR Google Gmail account Before proceeding:
 - The user should have a Google account or Google Gmail account at hand.
 - If not, the user should create a new one first
2. Access Google Cloud Platform (GCP) console
 - The user should be able to access his/her Google Cloud Platform (GCP) console
3. An existing project to host the Hadoop-Spark cluster
 - The user has an existing project under this account to host the to-be-created cluster
4. GCP storage bucket ready for use
 - The user has created a GCP storage bucket and have it ready for use.
5. Exploring Hadoop Ecosystem - You can do this assignment before this assignment. The order does not matter.

NOTES: Please see the following documents, if you need a refresher.

- How to Setup a GCP Account with Free Credit
- How to Create Projects in GCP
- How to Create New Storage Bucket, 3 Folders, and Load Data into a Folder in GCP
- How to Start and Stop Cluster Nodes in GCP
- Exploring Hadoop Ecosystem- You can do this assignment before this assignment. The order does not matter.

Step 1: Enter the GCP Console

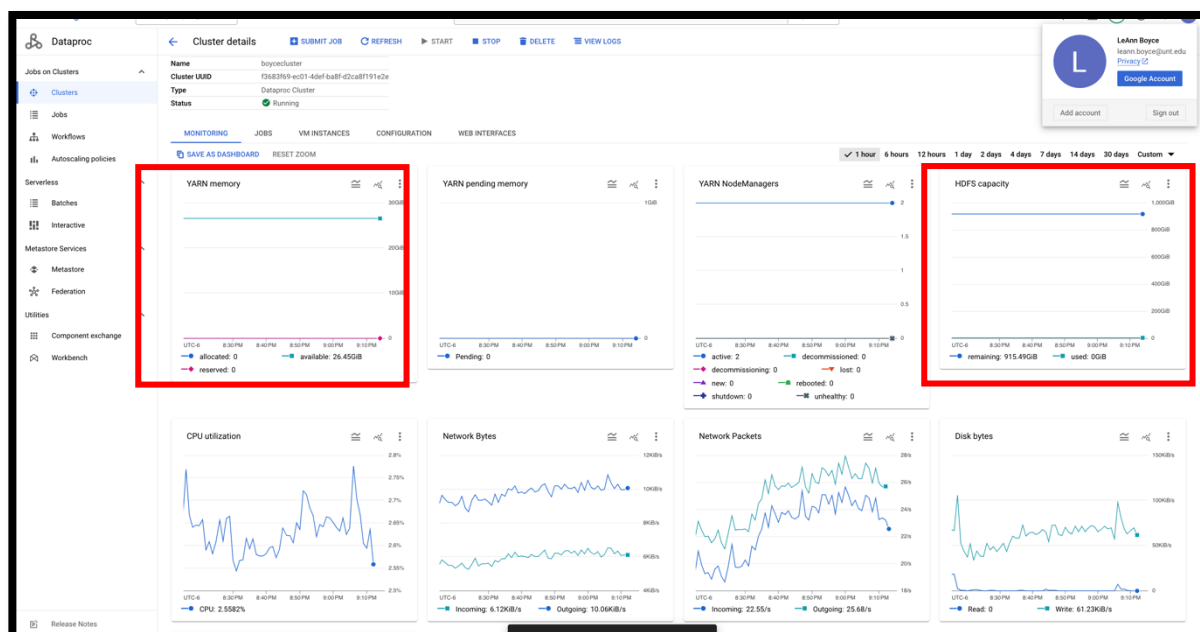
- You can start with a Google Search and Type “Google GCP Console”
- Click on “Google Cloud Platform Sign in”
- This should take you to your Dashboard, if not, click on the 3 horizontal lines in the upper left hand side.
- In the search bar, type in “ Clusters”. Click on “Clusters Dataproc”



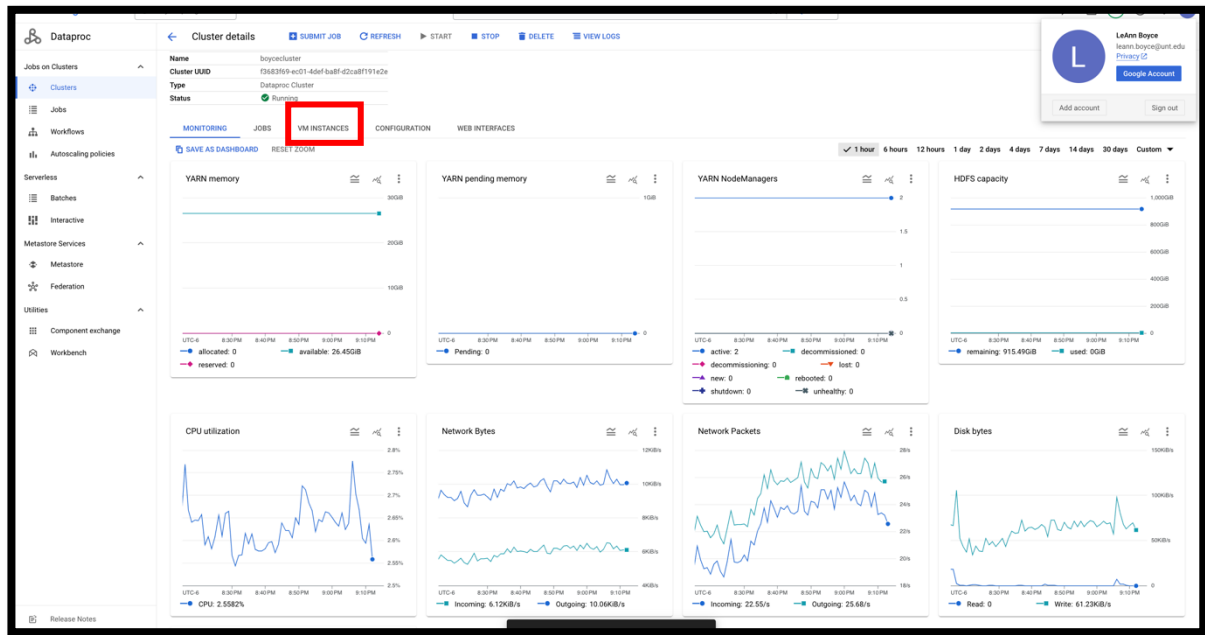
- This will bring up the cluster that you previously created, and you will see that it is running (green check mark)

Name	Status	Region	Zone	Total worker nodes	Flexible VMs?	Scheduled deletion	Cloud Storage staging bucket	Created
boycecluster	Running	us-central1	us-central1-c	2	No	Off	boycebucketone	Feb 11, 2024, 8:04:57 PM

- **IMPORTANT:** You will need to start your nodes by clicking on the navigation panel and click on “Compute Engine”. Remember, you click on the 3 vertical dots next to SSH and click “Start/Resume”. If you need a reminder please refer to “How to Start and Stop Cluster Nodes in GCP”. Once you have started all the nodes, type in “ Clusters”. Click on “Clusters Dataproc”. This will bring up your cluster that you previously created and you will see that it is running (green check mark). See above.
- Now click on the cluster. You can see a dashboard that monitors the cluster (you might need to scroll down). Note: Your cluster needs to be running for about 10 minutes or so to see the capacity used and remaining. You see there is 915.49 GiB remaining (0 GiB used) in HDFS capacity and 26.45GiB of YARN memory.



- Scroll back up, if you scrolled down and click on “VM Instances” (Virtual Machine Instances). Notice how the monitoring is changing according to usage.

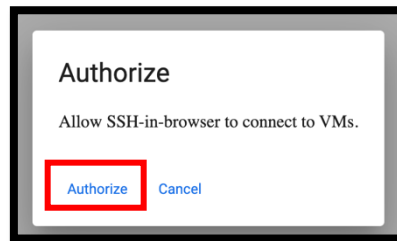


- You now see your cluster details. You see the one master node and two worker nodes. Remember, once you finish, you need to turn off all 3 nodes to stop the charges.
- Now we will access the master node through “SSH”
- Click on “Open in browser window”

The screenshot shows the Databroc Cluster details page for a cluster named 'boycecluster'. The 'VM INSTANCES' tab is selected. The page displays a table of VM instances with columns for Name, Role, and Status. The 'SSH' button is highlighted with a red box, and the 'Open in browser window' option is also highlighted with a red box.

Name	Role	Status
boycecluster-m	Master	Running
boycecluster-w-0	Worker	Running
boycecluster-w-1	Worker	Running

- You may need to Authorize. If so, click “Authorize.” This may take a little while, be patient.



Step 2: This opens the SSH terminal. This connects our local computer to the remote server (the master node of the cluster). We will use this SSH terminal to work with the cluster. I recommend working in a text editor as it makes it easier if you make a mistake.

- We will first type in “clear” so we have more room to work. Hit enter

```
Linux boycecluster-m 5.10.0-27-cloud-amd64 #1 SMP Debian 5.10.205-2 (2023-12-31) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Feb 12 03:34:41 2024 from 35.235.244.33
leann_boyce@boycecluster-m:~$ clear
```

- We are going to access the Hadoop Distributed File System (HDFS) of our cluster.
- First, let’s go over some of the basic Linux commands. You will see I have a different account name but this will not change the code you will use since you use your account name.
 - whoami (This shows you the user – you)
 - pwd (This is the command to see which directory you are currently in the master node of the cluster in the virtual machine in GCP)
 - hdfs dfs -ls /

This code has to be exactly as it is above. Please look at the code in the screenshot to verify spaces. This is very important.

```
SSH-in-browser

leann_boyce@boycecluster-m:~$ whoami
leann_boyce
leann_boyce@boycecluster-m:~$ pwd
/home/leann_boyce
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /
Found 3 items
drwxrwxrwt - hdfs hadoop 0 2024-02-12 02:06 /tmp
drwxrwxrwt - hdfs hadoop 0 2024-02-12 02:06 /user
drwxrwxrwt - hdfs hadoop 0 2024-02-12 02:06 /var
leann_boyce@boycecluster-m:~$
```

You see 3 folders

- drwxrwxrwt – hdfs hadoop 0 2024-02-12 02:06 /tmp. This shows the tmp folder belongs to Hadoop.
- drwxrwxrwt - hdfs hadoop 0 2024-02-12 /var ○ The var folder belongs to Hadoop. We will not do anything with this folder.
 - You see drwxrwxrwt. Do you remember what this means?
 - Remember file, owner, group, other?
 - So, the file is directory, the owner can read, write, and execute. The group can read, write, and execute. The other (like us) may also read, write, and t. T means that a file that lives in the file, only the owner can delete the folder.
- drwxrwxrwt - hdfs hadoop 0 2024-02-12 02:06 /user ○ The user folder belongs to Hadoop. This is the folder we will work with in this class.

Step 3: Let's see if there is anything in the user folder in hdfs.

- hdfs dfs -ls /user
- We have to use this entire command when we are in hdfs. In Hadoop framework we have to write the whole path.
- We see there are many folders in the user directory: dataproc, hbase, hdfs, hive, etc.
 - All of these folders have been created for us by the system.
 - You again see the permission types discussed above.
- You may have different folders but they should generally be the same.

```
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user
Found 11 items
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/dataproc
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/hbase
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/hdfs
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/hive
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/mapred
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/pig
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/solr
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/spark
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/yarn
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/zeppelin
drwxrwxrwt   - hdfs hadoop          0 2024-02-12 02:06 /user/zookeeper
leann_boyce@boycecluster-m:~$
```

- Let's create a new directory (subfolder) in this folder. This will be owned by the user (you). I recommend that you use the same name as your user name. Your name will be different than mine. If you forget, you can always type "whoami".

Step 4: Create directory in hdfs

- `hdfs dfs -mkdir /user/leannboyce`
- You will not see anything after this command.
- Type `hdfs dfs -ls /user`
- You will now see that there are now 12 items instead of 11 items because we created a new folder with the command `hdfs dfs -mkdir /user/leannboyce`

```
leann_boyce@boycecluster-m:~$ hdfs dfs -mkdir /user/leannboyce
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user
Found 12 items
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/dataproc
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/hbase
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/hdfs
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/hive
drwxr-xr-x - leann_boyce hadoop      0 2024-02-12 04:09 /user/leannboyce
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/mapred
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/pig
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/solr
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/spark
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/yarn
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/zeppelin
drwxrwxrwt - hdfs      hadoop      0 2024-02-12 02:06 /user/zookeeper
```

- Notice who is the owner of the new subfolder (you). This is added into the Hadoop group automatically. See the permissions on this new folder. Owner (you) can read, write, and execute. The group and others can read and execute only. They cannot make new or modify any content.
- Now we will write a command to access the content of the new subfolder. You can copy the command line into a notepad file (pure text file). This is much faster and convenient than writing the whole line each time.
 - `hdfs dfs -ls /user/leannboyce19`
 - Nothing will come back because there is no content in this newly created subfolder.

```
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce
leann_boyce@boycecluster-m:~$
```

Step 5: Create a subfolder “data” to hold new subfolders which will eventually hold the data files we created in GCP (userdata & weblog) ○

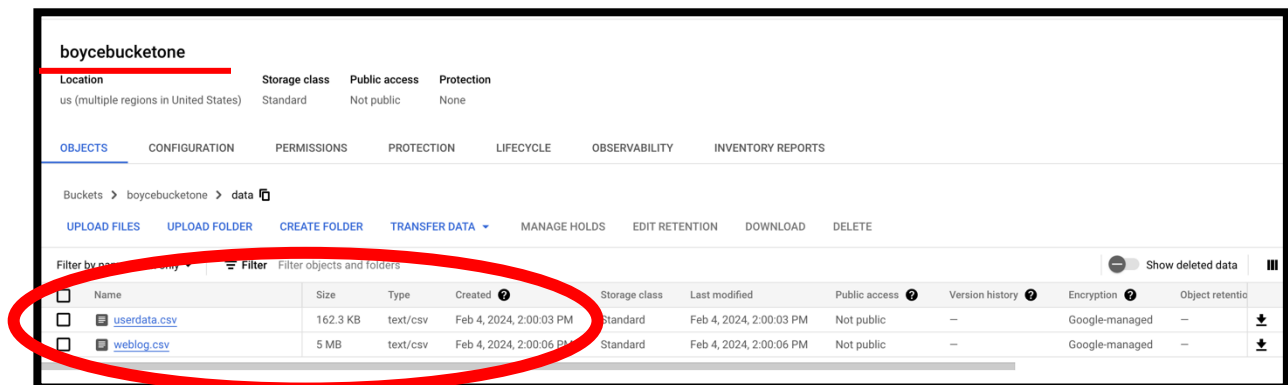
- We will now create another subfolder in this folder and name it “data”.

Remember in hdfs you have to include the full path.

- `hdfs dfs -mkdir /user/leannboyce/data`
- There is nothing in this subfolder but let’s check it to be sure.
- `hdfs dfs -ls /user/leannboyce/data`

```
leann_boyce@boycecluster-m:~$  
leann_boyce@boycecluster-m:~$ hdfs dfs -mkdir /user/leannboyce/data  
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce/data  
leann_boyce@boycecluster-m:~$
```

- Now we will fill the “data” subfolder. If you remember, you created a bucket in GCP. This subfolder will associate with that bucket. Remember we loaded two datasets into the GCP Bucket: userdata and weblog. If you need to refresh your memory, go to GCP and look at Cloud Storage.



boycebucketone										
Location		Storage class	Public access	Protection						
us (multiple regions in United States)		Standard	Not public	None						
OBJECTS										
Buckets > boycebucketone > data										
UPLOAD FILES UPLOAD FOLDER CREATE FOLDER TRANSFER DATA MANAGE HOLDS EDIT RETENTION DOWNLOAD DELETE										
Filter by name Filter Filter objects and folders Show deleted data										
Name	Size	Type	Created	Storage class	Last modified	Public access	Version history	Encryption	Object retention	
userdata.csv	162.3 KB	text/csv	Feb 4, 2024, 2:00:03 PM	Standard	Feb 4, 2024, 2:00:03 PM	Not public	—	Google-managed	—	Download
weblog.csv	5 MB	text/csv	Feb 4, 2024, 2:00:06 PM	Standard	Feb 4, 2024, 2:00:06 PM	Not public	—	Google-managed	—	Download

Step 6: We will copy the data files from the bucket into our newly created subfolder in hdfs. This will allow us to run MapReduce, Spark, or Hive.



- If you went out to look at the bucket, go back to the SSH terminal.
- We will create a subfolder for each dataset (userdata & weblog) in the folder “data”
- Let’s first create “userdata”
 - `hdfs dfs -mkdir /user/leannboyce/data/userdata`
- Let’s see that the subfolder “userdata” was created
 - `hdfs dfs -ls /user/ leannboyce/data`
 - We see there is 1 item found in the data folder

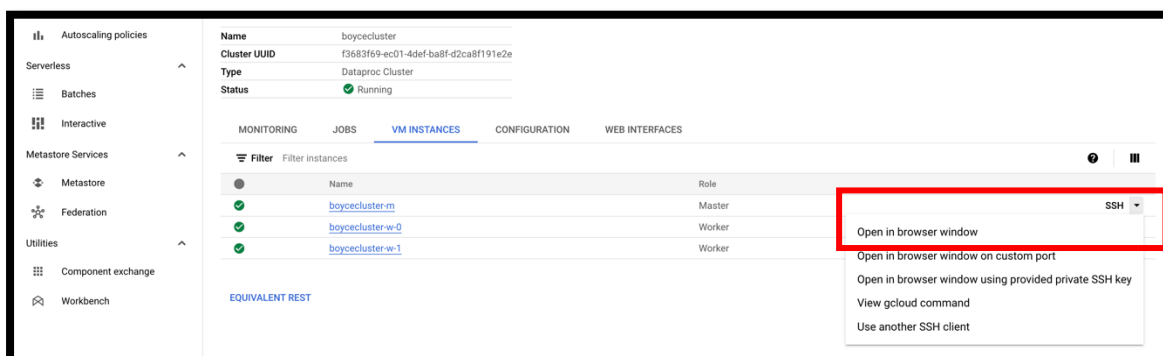
```
leann_boyce@boycecluster-m:~$ hdfs dfs -mkdir /user/leannboyce/data/userdata
leann_boyce@boycecluster-m:~$
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce/data
Found 1 items
drwxr-xr-x - leann_boyce hadoop 0 2024-02-12 04:34 /user/leannboyce/data/userdata
leann_boyce@boycecluster-m:~$
```

- Now we will create a subfolder titled “weblog” following the same steps above.
 - `hdfs dfs -mkdir /user/leannboyce/data/weblog`
- Let’s see that the subfolder “weblog” was created
 - `hdfs dfs -ls /user/ leannboyce/data`
- We see there are 2 items found in the data folder

```
leann_boyce@boycecluster-m:~$ hdfs dfs -mkdir /user/leannboyce/data/weblog
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce/data
Found 2 items
drwxr-xr-x - leann_boyce hadoop 0 2024-02-12 04:34 /user/leannboyce/data/userdata
drwxr-xr-x - leann_boyce hadoop 0 2024-02-12 04:44 /user/leannboyce/data/weblog
leann_boyce@boycecluster-m:~$
```

Step 7: Now you will have to open another new SSH terminal. Go back to the master node to access another SSH terminal. Go back to GCP and type “Compute Engine” in the search bar - OR- use the Navigation Pane and scroll down to “Compute Engine”.

- You will then click on “SSH” and click on “Open in browser” I



Step 8: In this new SSH terminal, we will see what is in the directory (yellow line). Remember, you can always use “whoami” and “pwd” if you forget your user name. Note in Linux, we do NOT have to type long command line like we did in hdfs.

- Let's start by making a folder named DATA in your home directory to copy files into from the bucket we created in a previous assignment
 - `mkdir DATA`
- And let's check to see if the folder is created
 - `ls -l`
- Using the above command line, you see the DATA folder is in the home directory.

```
leann_boyce@boycecluster-m:~$ mkdir DATA
leann_boyce@boycecluster-m:~$ ls -l
total 4
drwxr-xr-x 2 leann_boyce leann_boyce 4096 Feb 12 04:58 DATA
leann_boyce@boycecluster-m:~$
```

- Now we will move to the directory “DATA”
 - `cd DATA`
- And we will check to see what is in the “DATA” folder
 - `ls -l`
- We see there is no content in the “DATA” folder

```
leann_boyce@boycecluster-m:~$ cd DATA
leann_boyce@boycecluster-m:~/DATA$
leann_boyce@boycecluster-m:~/DATA$ ls -l
total 0
leann_boyce@boycecluster-m:~/DATA$
```

Step 9: We want to copy the data files (userdata & weblog) from the GCP Bucket to HDFS

- Copy GCP Bucket into the Master Node of the Cluster
- It is best to type command into a text editor first, check and double check the command, and then paste it into the SSH terminal.
 - `cp=copy`
 - `bucket` = your bucket name --- my bucket name is boycebucketone
 - `data` = the folder where we have the subfolders
 - `userdata.csv` = the file
- Enter this code in your terminal
 - `gsutil cp gs://boycebucketone/data/userdata.csv userdata.csv`

```
leann_boyce@boycecluster-m:~/DATA$ gsutil cp gs://boycebucketone/data/userdata.csv userdata.csv
Copying gs://boycebucketone/data/userdata.csv...
/ [0 files][ 0.0 B/166.2 KiB] / [1 files][166.2 KiB/166.2 KiB]
Operation completed over 1 objects/166.2 KiB.
leann_boyce@boycecluster-m:~/DATA$
```

- In the above screen, you see that the file “userdata” has been copied from the bucket to the master node. The file is 166.2 KiB. Below you see that the file was copied into the master node “DATA” folder.
- Look at the permissions below. You see that owner is me (you). The owner can read and write but I cannot execute because this is not an executable file. All others can only read the file.

```
leann_boyce@boycecluster-m:~/DATA$ ls -l
total 168
-rw-r--r-- 1 leann boyce leann boyce 170207 Feb 12 05:19 userdata.csv
leann_boyce@boycecluster-m:~/DATA$
```

- We will do the same for the weblog file that we did for the userdata file.
- Enter this code in your terminal using the name of your storage bucket.
 - `gsutil cp gs://boycebucketone/data/weblog.csv weblog.csv`
- Below, we see this was a successful copy, and we now have two data files in the “DATA” folder.
- We also see that I am the owner of this also. (Of course, you will see your name.)

```
leann_boyce@boycecluster-m:~/DATA$ gsutil cp gs://boycebucketone/data/weblog.csv weblog.csv
Copying gs://boycebucketone/data/weblog.csv...
/ [1 files][ 5.0 MiB/ 5.0 MiB]
Operation completed over 1 objects/5.0 MiB.
leann_boyce@boycecluster-m:~/DATA$ ls -l
total 5240
-rw-r--r-- 1 leann boyce leann boyce 170207 Feb 12 05:19 userdata.csv
-rw-r--r-- 1 leann boyce leann boyce 5192992 Feb 12 05:27 weblog.csv
leann_boyce@boycecluster-m:~/DATA$
```

***Notice the spelling error in my file. I will correct this in the next step. If you make an error, you can either delete or continue (just remember what you did).

Step 10: Now we will take the files from the master node to the HDFS ecosystem. Again, it is good to type into a text editor and make sure you have no errors first and then copy and paste into HDFS. This is the full path to the HDFS file.

- `hdfs dfs -put userdata.csv /user/leannboyce/data/userdata`
- I will do the same with the weblog data, but first notice that I entered the correction for weblog.

```
leann_boyce@boycecluster-m:~/DATA$ gsutil cp gs://boycebucketone/data/weblog.csv weblog.csv
Copying gs://boycebucketone/data/weblog.csv...
/ [1 files][ 5.0 MiB/ 5.0 MiB]
Operation completed over 1 objects/5.0 MiB.
leann_boyce@boycecluster-m:~/DATA$ ls -l
total 5240
-rw-r--r-- 1 leann_boyce leann_boyce 170207 Feb 12 05:19 userdata.csv
-rw-r--r-- 1 leann_boyce leann_boyce 5192992 Feb 12 05:27 weblog.csv
leann_boyce@boycecluster-m:~/DATA$ hdfs dfs -put userdata.csv /user/leannboyce/data/userdata
leann_boyce@boycecluster-m:~/DATA$ gsutil cp gs://boycebucketone/data/weblog.csv weblog.csv
Copying gs://boycebucketone/data/weblog.csv...
/ [0 files][ 0.0 B/ 5.0 MiB] / [1 files][ 5.0 MiB/ 5.0 MiB]
Operation completed over 1 objects/5.0 MiB.
leann_boyce@boycecluster-m:~/DATA$ ls -l
total 10312
-rw-r--r-- 1 leann_boyce leann_boyce 170207 Feb 12 05:19 userdata.csv
-rw-r--r-- 1 leann_boyce leann_boyce 5192992 Feb 12 05:27 weblog.csv
-rw-r--r-- 1 leann_boyce leann_boyce 5192992 Feb 12 05:42 weblog.csv
leann_boyce@boycecluster-m:~/DATA$ hdfs dfs -put weblog.csv /user/leannboyce/data/weblog
leann_boyce@boycecluster-m:~/DATA$
```

• In order to check if the file was moved, you need to go back to the first terminal. If you lost your connect to the first terminal, don't worry, you can restart an SSH terminal (see above) and type in the following command:

- hdfs dfs -ls /user/leannboyce/data
- We will also see if the file "userdata" was copied into hdfs using the following command:
 - hdfs dfs -ls /user/leannboyce/data/userdata
- We will also see if the file "weblog" was copied into hdfs using the following command
 - hdfs dfs -ls /user/leannboyce/data/weblog

Note: There is not blue "/DATA" in the screenshot below. This is how you know you are in the correct terminal.

```
SSH-in-browser
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce/data
Found 2 items
drwxr-xr-x - leann_boyce hadoop 0 2024-02-12 05:40 /user/leannboyce/data/userdata
drwxr-xr-x - leann_boyce hadoop 0 2024-02-12 05:43 /user/leannboyce/data/weblog
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce/data/userdata
Found 1 items
-rw-r--r-- 2 leann_boyce hadoop 170207 2024-02-12 05:40 /user/leannboyce/data/userdata/userdata.csv
leann_boyce@boycecluster-m:~$ hdfs dfs -ls /user/leannboyce/data/weblog
Found 1 items
-rw-r--r-- 2 leann_boyce hadoop 5192992 2024-02-12 05:43 /user/leannboyce/data/weblog/weblog.csv
leann_boyce@boycecluster-m:~$
```

We have wrapped up the topic of HDFS for now. We will use this again when we work with queries.

DON'T FORGET TO TURN OFF YOUR CLUSTERS!!!

