Starting Hadoop

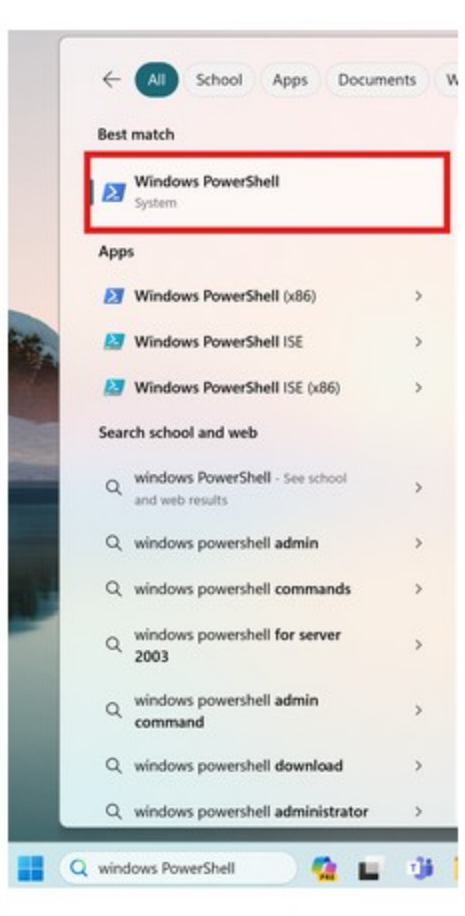
Learning Goals

By the end of this activity, you will be able to:

- Start Hadoop using Docker containers
- Copy files into and out of the Hadoop Distributed File System (HDFS).

Instructions

1. **Open a terminal shell.** Open your local system terminal shell. In Windows, you can use Windows PowerShell, while in Mac, you can use Terminal.



2. Go to your Hadoop directory. In the terminal shell, change to your big-data-1/hadoop directory

Type *ls* to see the files within the directory.

```
ls
```

```
PS C:\Users\[redacted]\Desktop\coursera\big-data-1\hadoop> ls

    Directory: C:\Users\pramonettivega\Desktop\coursera\big-data-1\hadoop

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        11/27/2023   9:46 AM           1789 config
-a----          4/1/2024   1:47 PM            783 docker-compose.yaml
-a----          4/1/2024   1:25 PM            803 mapred-site.xml
-a----        11/27/2023   9:57 AM        5458199 words.txt
```

3. Start Docker. In order to start working, we need to start the Docker Engine. The easiest way to do this is to simply open Docker Desktop. Wait for it to initiate.

4. Build your Hadoop cluster. To run a Hadoop cluster locally, we are going to run multiple Docker containers, each of them running a different service. The docker-compose.yaml file within the directory is the recipe to achieve this.

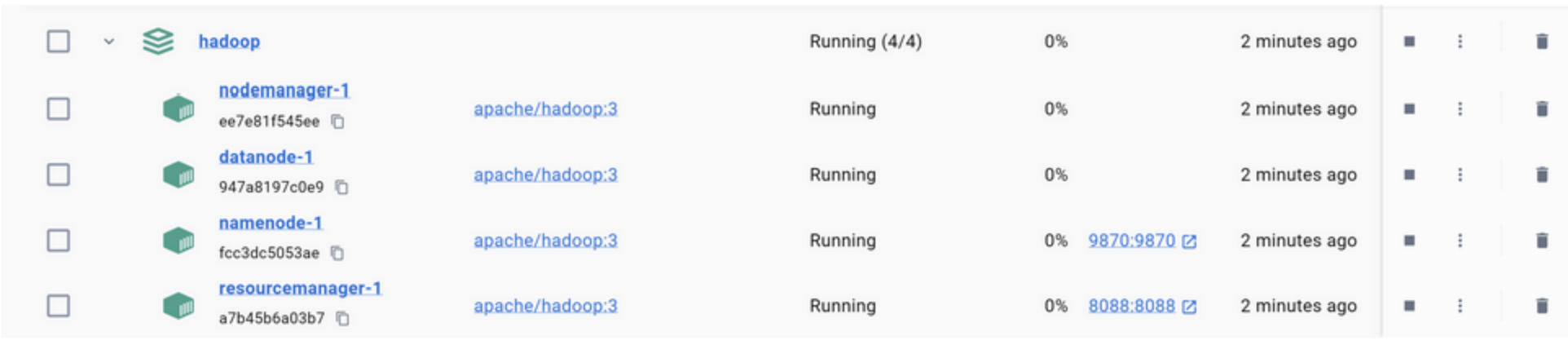
Go back to your terminal shell and run the following command:

```
docker-compose up -d
```

```
PS C:\Users\[redacted]\Desktop\coursera\big-data-1\hadoop> docker-compose up -d
[+] Building 0.0s (0/0)                                                  docker:default
[+] Running 5/5
 ✓Network hadoop_default                   Created                              0.0s
 ✓Container hadoop-resourcemanager-1       Started                              0.1s
 ✓Container hadoop-datanode-1              Started                              0.1s
 ✓Container hadoop-namenode-1              Started                              0.1s
 ✓Container hadoop-nodemanager-1           Started                              0.1s
```

This will start the creation of the multiple containers needed for this activity. This process might take a few minutes.

Go to Docker Desktop, click on *Containers* in the top left, and make sure the containers are running.



As part of the setup, we need to update a file within the namenode-1 container. Run *docker cp ./mapred-site.xml hadoop-namenode-1:/opt/hadoop/etc/hadoop/mapred-site.xml* to replace the mapred-site.xml file within the container.

```
docker cp ./mapred-site.xml hadoop-namenode-1:/opt/hadoop/etc/hadoop/mapred-site.xml
```

```
PS C:\Users\[redacted]\Desktop\coursera\big-data-1\hadoop> docker cp ./mapred-site.xml hadoop-namenode-1:/opt/hadoop/etc/hadoop/mapred-site.xml
Successfully copied 2.56kB to hadoop-namenode-1:/opt/hadoop/etc/hadoop/mapred-site.xml
```

We also need to transfer the words.txt file (which is a collection of Williams Shakespeare's works) to the container. Run *docker cp ./words.txt hadoop-namenode-1:/opt/hadoop/words.txt* to copy the file from your local directory to the container.

```
docker cp ./words.txt hadoop-namenode-1:/opt/hadoop/words.txt
```

```
PS C:\Users\[redacted]\Desktop\coursera\big-data-1\hadoop> docker cp ./words.txt hadoop-namenode-1:/opt/hadoop/words.txt
Successfully copied 5.46MB to hadoop-namenode-1:/opt/hadoop/words.txt
```

5. Access containers terminal shell. Run *docker exec -it hadoop-namenode-1 /bin/bash* to access namenode-1 shell.

```
docker exec -it hadoop-namenode-1 /bin/bash
```

```
PS C:\Users\[redacted]\Desktop\coursera\big-data-1\hadoop> docker exec -it hadoop-namenode-1 /bin/bash
bash-4.2$
```

Now we are inside our local Hadoop cluster.

6. Creating the HDFS root folder. To start interacting with the HDFS, we first need to create a user folder to be able to place our files in it. Run *hdfs dfs -mkdir -p /user/hadoop*

```
hdfs dfs -mkdir -p /user/hadoop
```

```
bash-4.2$ hdfs dfs -mkdir -p /user/hadoop
bash-4.2$
```

7. Copy file to HDFS. Run *hdfs dfs -ls* to see the files within the file system. Nothing will come out, as there are currently no files in it.

```
hdfs dfs -ls
```

```
bash-4.2$ hdfs dfs -ls
bash-4.2$
```

Run *hdfs dfs -copyFromLocal words.txt* to copy the text file to HDFS

```
hdfs dfs -copyFromLocal words.txt
```

```
bash-4.2$ hdfs dfs -copyFromLocal words.txt
bash-4.2$
```

8. Verify file was copied to HDFS. Run *hdfs dfs –ls* to verify the file was copied to HDFS.

```
hdfs dfs -ls
```

```
bash-4.2$ hdfs dfs -ls
Found 1 items
-rw-r--r--   1 hadoop supergroup    5458199 2024-04-02 21:39 words.txt
```

9. Copy a file within HDFS. You can make a copy of a file in HDFS. Run *hdfs dfs -cp words.txt words2.txt* to make a copy of words.txt called words2.txt

```
hdfs dfs -cp words.txt words2.txt
```

```
bash-4.2$ hdfs dfs -cp words.txt words2.txt
bash-4.2$
```

We can see the both files by running *hdfs dfs -ls*

```
hdfs dfs -ls
```

```
bash-4.2$ hdfs dfs -ls
Found 2 items
-rw-r--r--   1 hadoop supergroup    5458199 2024-04-02 21:39 words.txt
-rw-r--r--   1 hadoop supergroup    5458199 2024-04-02 21:43 words2.txt
```

10. Copy a file from HDFS. We can also copy a file from HDFS to the local file system. Run *hdfs dfs -copyToLocal words2.txt .* to copy words2.txt to the local directory.

```
hdfs dfs -copyToLocal words2.txt .
```

```
bash-4.2$ hdfs dfs -copyToLocal words2.txt .
bash-4.2$
```

Let's run *ls* to see that words2.txt is there.

```
ls
```

```
bash-4.2$ ls
LICENSE-binary  LICENSE.txt  NOTICE-binary  NOTICE.txt  README.txt  bin  etc  include  lib  libexec  licenses-binary  sbin  share  words.txt  words2.txt
```

11. Delete a file in HDFS. Let's the delete words2.txt in HDFS. Run *hdfs dfs -rm words2.txt*

```
hdfs dfs -rm words2.txt
```

```
bash-4.2$ hdfs dfs -rm words2.txt
Deleted words2.txt
```

Run *hdfs dfs -ls* to see that the file is gone.

```
hdfs dfs -ls
```

```
bash-4.2$ hdfs dfs -ls
Found 1 items
-rw-r--r--   1 hadoop supergroup    5458199 2024-04-02 21:39 words.txt
```

12. Deleting cluster. From here, you can continue to the next activity. If you're are planning to stop working and come back to the course later, make sure to delete the cluster first and repeat steps 3-5 next time you come back. Given the current configuration, if you simply stop and restart the cluster from Docker Desktop, the cluster won't work properly.

Run *exit* to exit the container's shell. You will be send back to your local terminal shell.

```
exit
```

```
bash-4.2$ exit
exit
```

To delete the containers, run *docker compose down*.

```
docker compose down
```

```
PS C:\Users\[redacted]\Desktop\Docker Containers for Coursera\coursera\big-data-1\hadoop> docker compose down
[+] Running 5/5
 ✓Container hadoop-nodemanager-1       Removed                              0.7s
 ✓Container hadoop-datanode-1          Removed                              1.0s
 ✓Container hadoop-namenode-1          Removed                              1.1s
 ✓Container hadoop-resourcemanager-1   Removed                              0.8s
 ✓Network hadoop_default               Removed                              0.3s
```

Mark as completed

Like   Dislike   Report an issue