Hands-on lab on Hadoop Map-Reduce (20 mins)

Objectives

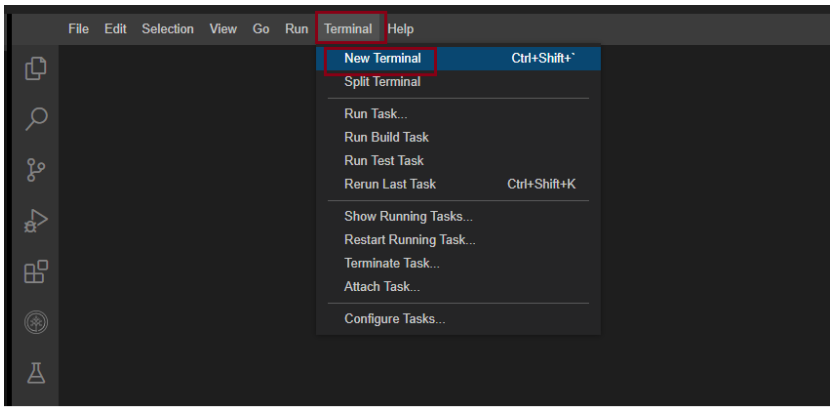
- Run a single-node Hadoop instance
- Perform a word count using Hadoop **Map Reduce**.

Set up Single-Node Hadoop

The steps outlined in this lab use the single-node Hadoop Version 3.2.3. **Hadoop** is most useful when deployed in a fully distributed mode on a large cluster of networked servers sharing a large volume of data. However, for basic understanding, we will configure Hadoop on a single node.

In this lab, we will run the WordCount example with an input text and see how the content of the input file is processed by WordCount.

1. Start a new terminal



2. Download hadoop-3.2.3.tar.gz to your theia environment by running the following command.

```
1
curl https://dlcdn.apache.org/hadoop/common/hadoop-3.2.3/hadoop-3.2.3.tar.gz --output hadoop-3.2.3.tar.gz
```

Copied!

3. Extract the tar file in the currently directory.

```
1
tar -xvf hadoop-3.2.3.tar.gz
```

Copied!

4. Navigate to the hadoop-3.2.3 directory.

```
1
cd hadoop-3.2.3
```

Copied!

5. Check the hadoop command to see if it is setup. This will display the usage documentation for the hadoop script.

```
1
bin/hadoop
```

Copied!

6. Run the following command to download data.txt to your current directory.

```
1
curl https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-BD0225EN-SkillsNetwork/labs/data/data.txt --output data.txt
```

Copied!

7. Run the Map reduce application for wordcount on data.txt and store the output in **/user/root/output**

```
1
bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.3.jar wordcount data.txt output
```

Copied!

   This may take some time.

8. Once the word count runs successfully, you can run the following command to see the output file it has generated.

```
1
ls output
```

Copied!

You should see **part-r-00000** with **_SUCCESS** indicating that the wordcount has been done.

   While it is still processing, you may only see *'_temporary'* listed in the output directory. Wait for a couple of minutes and run the command again till you see output as shown above.

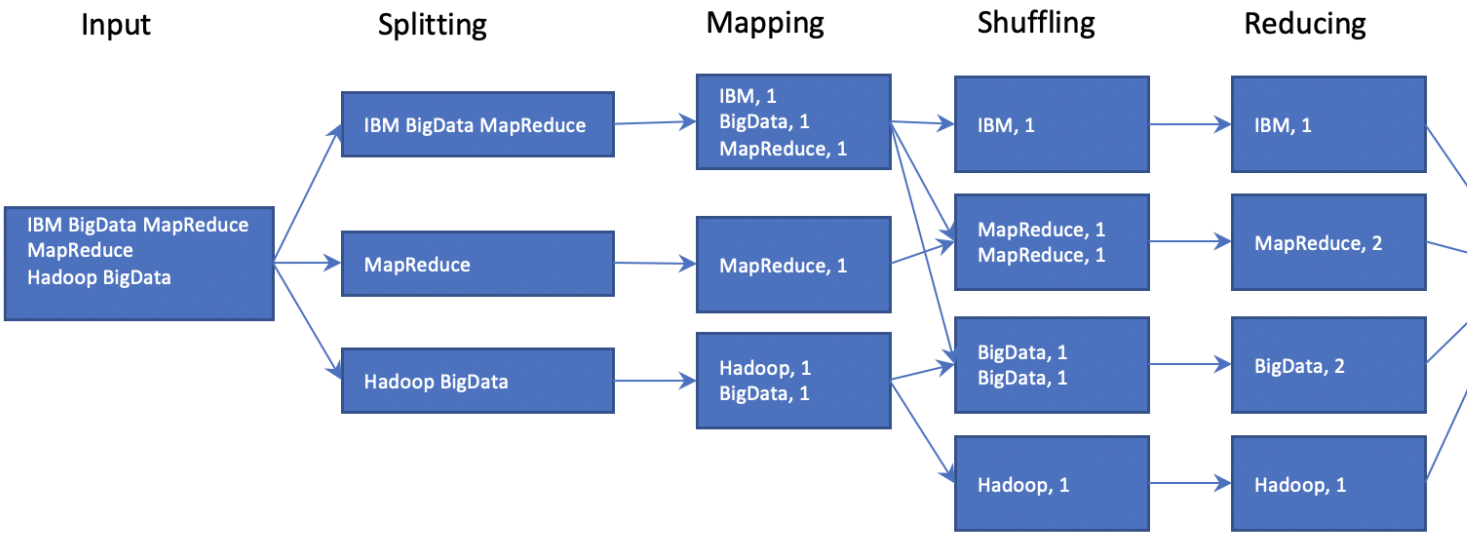
9. Run the following command to see the word count output.

```
1
cat  output/part-r-00000
```

Copied!

```
theia@theiadocker-lavanyas:/home/project/hadoop-3.2.2$ cat output/part-r-00000
BigData 2
Hadoop  1
IBM     1
MapReduce       2
```

The image below shows how the MapReduce wordcount happens.



Practice Lab

1. Do a word count on a file with the following content.

```
1. 1
2. 2
3. 3
```

```
1. Italy Venice
2. Italy Pizza
3. Pizza Pasta Gelato
```

Copied!

▼ Click here for a hint on how to get started
- Delete the data.txt file and output folder

```
1. 1
```

```
1. rm data.txt
```

Copied!

```
1. 1
```

```
1. rm -rf output
```

Copied!

▼ Click here for hint on how to create a file to wordcount
Create data.txt with the required content. You may either use the file editor.
▼ Click here for solution on how to do word count on the file
Run the following command

```
1. 1
```

```
1. bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.2.3.jar wordcount data.txt output
```

Copied!

▼ Click here for sample output

The output will be as below.

```
root@e4d298bfe26c:/# hdfs dfs -cat  /user/root/output/part-r-00000
2021-07-13 05:21:45,467 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHo
sted = false
Gelato  1
Italy   2
Pasta   1
Pizza   2
Venice  1
```

Congratulations! You have:

- Deployed Hadoop using Docker
- Copied data into HDFS
- Used MapReduce to do a word count

Tweet and share your achievement!

Author(s)

Lavanya T S

Contributor(s)

Aije Egwaikhide

Changelog

Date	Version	Changed by	Change Description
05-04-2022	1.3	Sourabh	Updated Hadoop version
18-01-2022	1.2	Lavanya	Changed to single node hadoop
16-07-2021	1.1	Aije	Modified multiple areas
11-07-2021	1.0	Lavanya	Created lab instructions for Word count using MapReduce