Ex.No – 2 Roll No – 210701149

Implement word count/frequency programs using MapReduce

AIM:

To implement word count / frequency programs using MapReduce with Python in Hadoop.

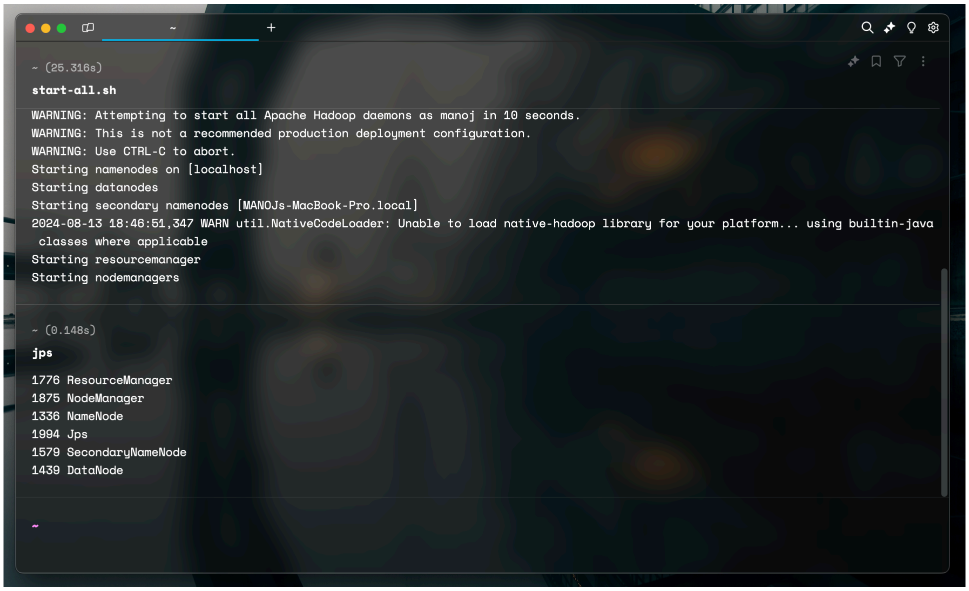
PROCEDURES:

1. Open the terminal and start Hadoop using start-all.sh command
2. Open the browser and go to the URL localhost:9870.
3. In the terminal using the command hadoop fs -mkdir /user create a directory called user.
4. Upload the input.txt file to hdfs using the command hadoop fs -put input.txt /user.

   Then perform the mapreduce operation using the command

   ```
   hadoop jar /path/to/hadoop-streaming.jar \
   -files /path/to/mapper.py, /path/to/reducer.py \
   -input /path/to/input \
   -output /path/to/output \
   -mapper mapper.py \
   -reducer reducer.py
   ```
5. Check the output using the command hadoop fs -cat /user/output/part-00000.

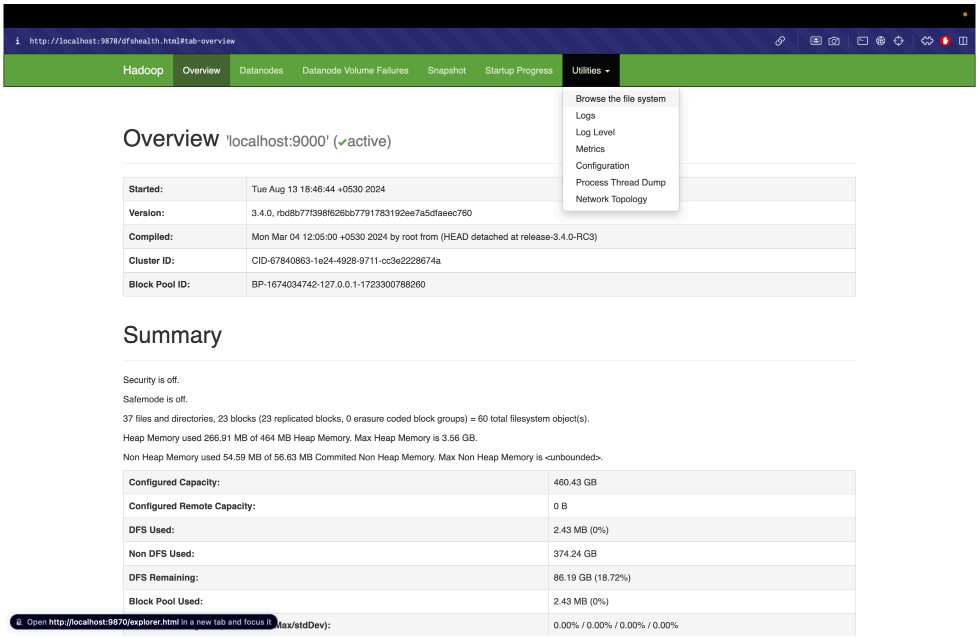
OUTPUT:



```
~ (25.316s)
start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as manoj in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [MANOJs-MacBook-Pro.local]
2024-08-13 18:46:51,347 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java
 classes where applicable
Starting resourcemanager
Starting nodemanagers

~ (0.148s)
jps
1776 ResourceManager
1875 NodeManager
1336 NameNode
1994 Jps
1579 SecondaryNameNode
1439 DataNode

~
```



Overview 'localhost:9000' (✓active)

Started:	Tue Aug 13 18:46:44 +0530 2024
Version:	3.4.0, rbd8b77f398f626bb7791783192ee7a5dfaeec760
Compiled:	Mon Mar 04 12:05:00 +0530 2024 by root from (HEAD detached at release-3.4.0-RC3)
Cluster ID:	CID-67840863-1e24-4928-9711-cc3e2228674a
Block Pool ID:	BP-1674034742-127.0.0.1-1723300788260

Summary

Security is off.

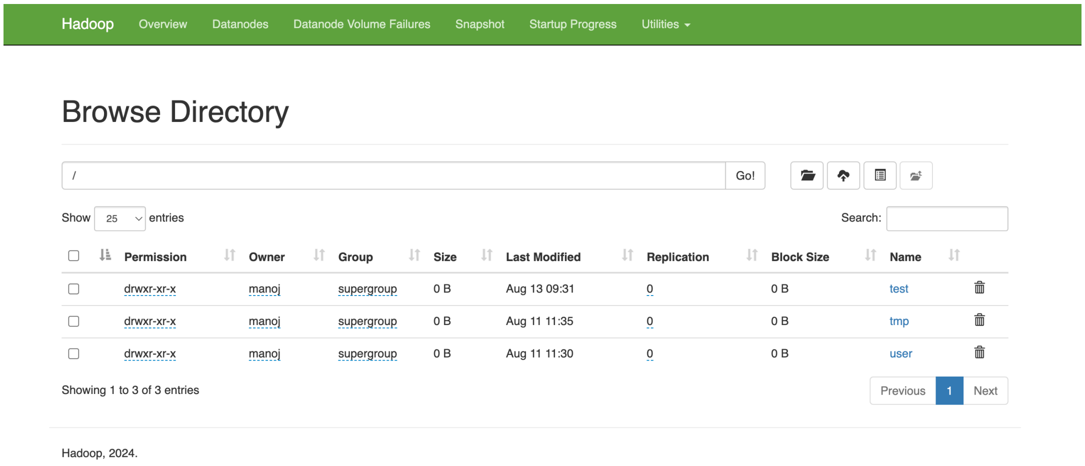
Safemode is off.

37 files and directories, 23 blocks (23 replicated blocks, 0 erasure coded block groups) = 60 total filesystem object(s).

Heap Memory used 266.91 MB of 464 MB Heap Memory. Max Heap Memory is 3.56 GB.

Non Heap Memory used 54.59 MB of 56.63 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	460.43 GB
Configured Remote Capacity:	0 B
DFS Used:	2.43 MB (0%)
Non DFS Used:	374.24 GB
DFS Remaining:	86.19 GB (18.72%)
Block Pool Used:	2.43 MB (0%)
Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%



Browse Directory

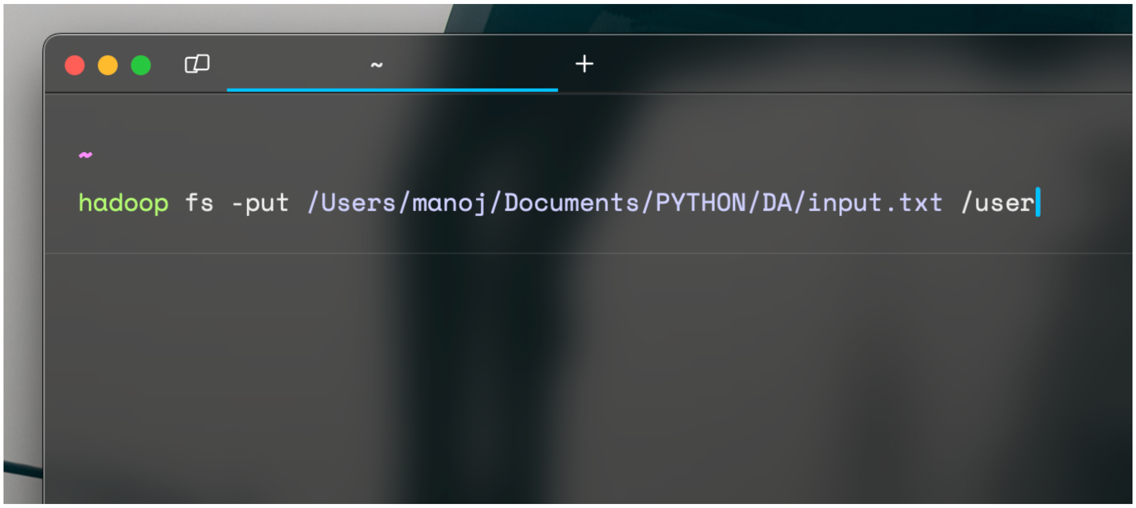
/ Go!

Show 25 entries Search:

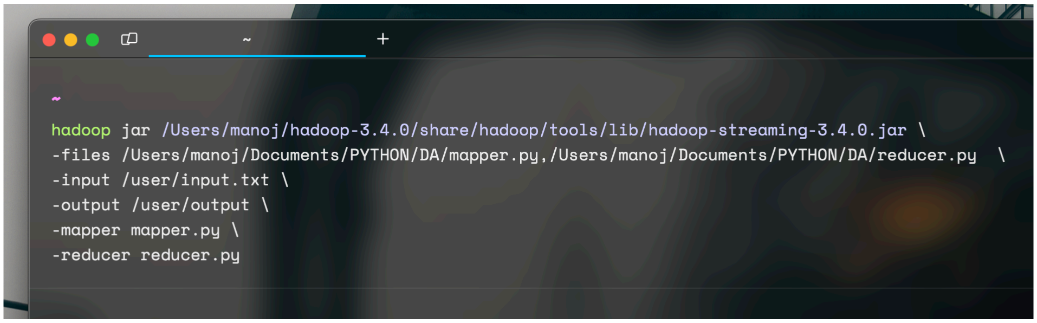
□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
□	drwxr-xr-x	manoj	supergroup	0 B	Aug 13 09:31	0	0 B	test	🗑
□	drwxr-xr-x	manoj	supergroup	0 B	Aug 11 11:35	0	0 B	tmp	🗑
□	drwxr-xr-x	manoj	supergroup	0 B	Aug 11 11:30	0	0 B	user	🗑

Showing 1 to 3 of 3 entries

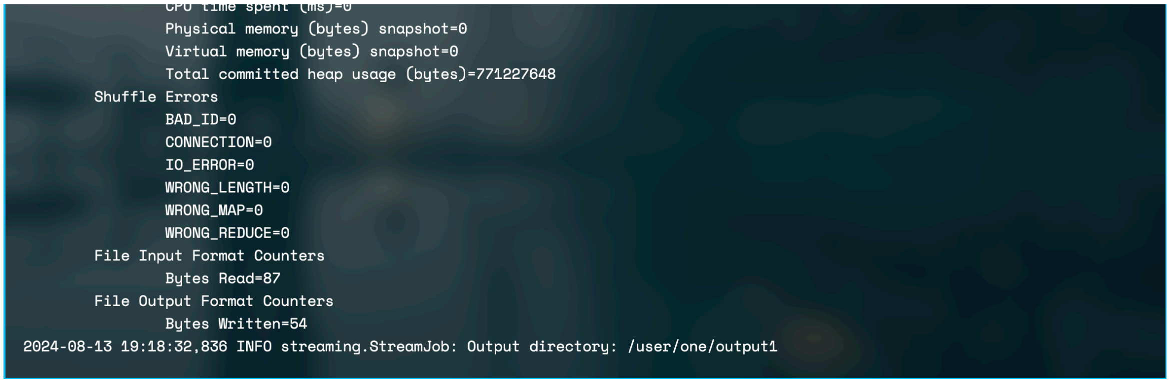
Hadoop, 2024.



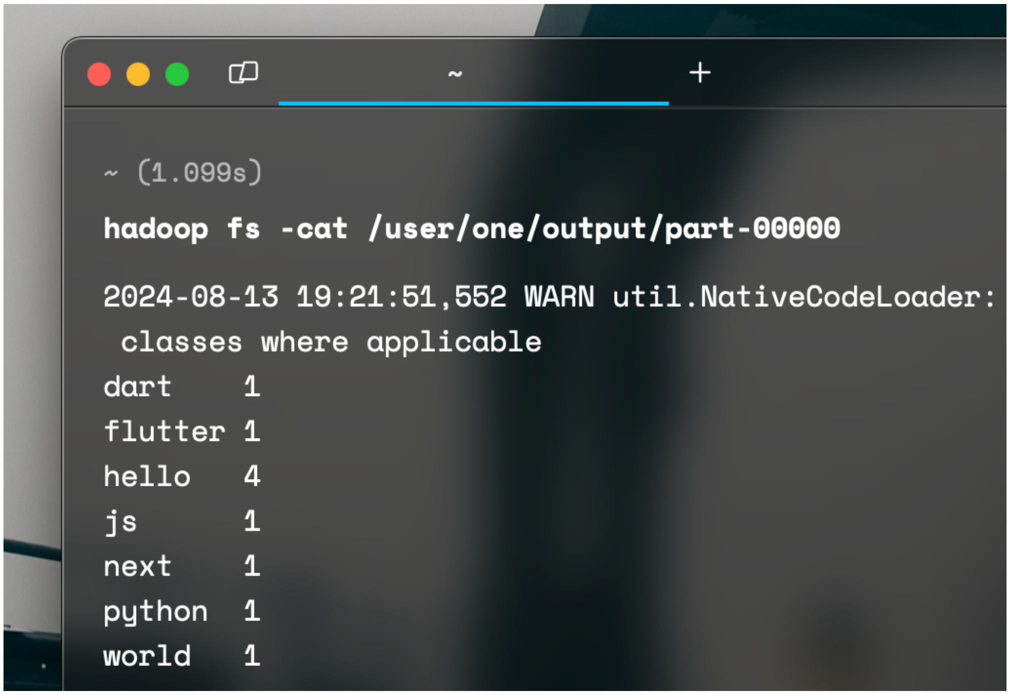
```
~
hadoop fs -put /Users/manoj/Documents/PYTHON/DA/input.txt /user
```



```
~
hadoop jar /Users/manoj/hadoop-3.4.0/share/hadoop/tools/lib/hadoop-streaming-3.4.0.jar \
-files /Users/manoj/Documents/PYTHON/DA/mapper.py,/Users/manoj/Documents/PYTHON/DA/reducer.py  \
-input /user/input.txt \
-output /user/output \
-mapper mapper.py \
-reducer reducer.py
```



```
                CPU time spent (ms)=0
                Physical memory (bytes) snapshot=0
                Virtual memory (bytes) snapshot=0
                Total committed heap usage (bytes)=771227648
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=87
        File Output Format Counters
                Bytes Written=54
2024-08-13 19:18:32,836 INFO streaming.StreamJob: Output directory: /user/one/output1
```



```
~ (1.099s)
hadoop fs -cat /user/one/output/part-00000
2024-08-13 19:21:51,552 WARN util.NativeCodeLoader:
 classes where applicable
dart    1
flutter 1
hello   4
js      1
next    1
python  1
world   1
```

RESULT:

Thus, to implement the word count program using MapReduce in hadoop has been completed successfully.