

Program 2: Run a basic word count Map Reduce program to understand Map Reduce Paradigm:

Login into Hadoop user you used while installing Hadoop , here we use hdoop user

```
hdoop@NuvobookV1:~$ |
```

```
$ su - hdoop
```

-start the Hadoop server

```
$ cd hadoop-3.4.0/sbin/
```

```
$ ./start-dfs.sh
```

```
$ ./start-yarn
```

```
$ jps
```

```
hdoop@hdoop-VirtualBox:~$ cd hadoop-3.2.1/sbin
hdoop@hdoop-VirtualBox:~/hadoop-3.2.1/sbin$ ./start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [hdoop-VirtualBox]
hdoop@hdoop-VirtualBox:~/hadoop-3.2.1/sbin$ ./start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hdoop@hdoop-VirtualBox:~/hadoop-3.2.1/sbin$ jps
4817 DataNode
5298 ResourceManager
5000 SecondaryNameNode
5450 NodeManager
4683 NameNode
5982 Jps
hdoop@hdoop-VirtualBox:~/hadoop-3.2.1/sbin$
```

```
$ sudo mkdir p2/
```

```
$ cd p2/
```

```
$ start-all.sh
```

```
$sudo nano mapper.py
```

-inside it paste this program:

```
#!/usr/bin/env python3
```

```
# import sys because we need to read and write data to STDIN and STDOUT
```

```
import sys
```

```
# reading entire line from STDIN (standard input)
```

```
for line in sys.stdin:
```

```
    # to remove leading and trailing whitespace
```

```
    line = line.strip()
```

```
    # split the line into words
```

```
    words = line.split()
```

```
    # we are looping over the words array and printing the word
```

```
    # with the count of 1 to the STDOUT
```

```
    for word in words:
```

```
        # write the results to STDOUT (standard output);
```

```
        # what we output here will be the input for the
```

```
        # Reduce step, i.e. the input for reducer.py
```

```
        print('%s\t%s' % (word, 1))
```

-click Ctrl+s and Ctrl+x to close it

```
$ nano reducer.py
```

-insert this program inside it:

```
#!/usr/bin/env python3
```

```
from operator import itemgetter
```

```
import sys
```

```
current_word = None
```

```
current_count = 0
```

```
word = None
```

```
# read the entire line from STDIN
```

```
for line in sys.stdin:
```

```
    # remove leading and trailing whitespace
```

```
    line = line.strip()
```

```
    # splitting the data on the basis of tab we have provided in mapper.py
```

```
    word, count = line.split('\t', 1)
```

```
    # convert count (currently a string) to int
```

```
    try:
```

```
        count = int(count)
```

```
    except ValueError:
```

```
        # count was not a number, so silently
```

```
        # ignore/discard this line
```

```
        continue
```

```
# this IF-switch only works because Hadoop sorts map output
```

```
# by key (here: word) before it is passed to the reducer
```

```
if current_word == word:
```

```
    current_count += count
```

```
else:
```

```
    if current_word:
```

```
        # write result to STDOUT
```

```
        print ('%s\t%s' % (current_word, current_count))
```

```
    current_count = count
```

```
    current_word = word
```

```
# do not forget to output the last word if needed!
```

```
if current_word == word:
```

```
    print ('%s\t%s' % (current_word, current_count))
```

-click Ctrl+s and Ctrl+x to close it

```
$ sudo sample.txt
```

-insert this program inside it:

The server processes data, repeating tasks to repeat actions efficiently. Users rely on it, repeating operations to repeat access to critical data

-click Ctrl+s and Ctrl+x to close it

```
$ sudo chmod 777 reducer.py
```

```
$ sudo chmod 777 mapper.py
```

```
$ hdfs dfs -mkdir /p2/
```

```
$ hdfs dfs -copyFromLocal sample.txt /p2/
```

\$ cd

\$ wget <https://repo1.maven.org/maven2/org/apache/hadoop/hadoop-streaming/2.7.3/hadoop-streaming-2.7.3.jar>

\$ hadoop jar /home/hdoop/hadoop-streaming-2.7.3.jar -input /p2/sample.txt -output /p2/output -mapper /home/hdoop/p2/mapper.py -reducer /home/hdoop/p2/reducer.py

Output:

```
hadoop@NuvoBookV1:~/lab/p2$ hadoop jar /home/hdoop/hadoop-streaming-2.7.3.jar -input /p2/python/sample.txt -output /p2/python/output -mapper /home/hdoop/lab/
p2/mapper.py -reducer /home/hdoop/lab/p2/reducer.py
packageJobJar: [/tmp/hadoop-unjar4769045384438272080/] [] /tmp/streamjob724948752243046205.jar tmpDir=null
2024-10-16 17:08:52,535 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /172.18.132.51:8032
2024-10-16 17:08:52,799 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /172.18.132.51:8032
2024-10-16 17:08:53,145 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hdoop/.staging/job_1729063021124_001
1
2024-10-16 17:08:53,409 INFO mapred.FileInputFormat: Total input files to process : 1
2024-10-16 17:08:53,489 INFO mapreduce.JobSubmitter: number of splits:2
2024-10-16 17:08:53,674 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1729063021124_0011
2024-10-16 17:08:53,674 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-10-16 17:08:53,890 INFO conf.Configuration: resource-types.xml not found
2024-10-16 17:08:53,890 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'
2024-10-16 17:08:54,019 INFO impl.YarnClientImpl: Submitted application application_1729063021124_0011
2024-10-16 17:08:54,062 INFO mapreduce.Job: The url to track the job: http://172.18.132.51:8088/proxy/application_1729063021124_0011/
2024-10-16 17:08:54,064 INFO mapreduce.Job: Running job: job_1729063021124_0011
2024-10-16 17:09:01,400 INFO mapreduce.Job: Job job_1729063021124_0011 running in uber mode : false
2024-10-16 17:09:01,402 INFO mapreduce.Job: map 0% reduce 0%
2024-10-16 17:09:06,470 INFO mapreduce.Job: map 100% reduce 0%
2024-10-16 17:09:11,508 INFO mapreduce.Job: map 100% reduce 100%
2024-10-16 17:09:13,541 INFO mapreduce.Job: Job job_1729063021124_0011 completed successfully
2024-10-16 17:09:13,648 INFO mapreduce.Job: Counters: 54
File System Counters
  FILE: Number of bytes read=241
  FILE: Number of bytes written=934444
  FILE: Number of read operations=0
  FILE: Number of large read operations=0
  FILE: Number of write operations=0
  HDFS: Number of bytes read=417
  HDFS: Number of bytes written=160
  HDFS: Number of read operations=11
  HDFS: Number of large read operations=0
  HDFS: Number of write operations=2
  HDFS: Number of bytes read erasure-coded=0
Job Counters
  Launched map tasks=2
  Launched reduce tasks=1
  Data-local map tasks=2
  Total time spent by all maps in occupied slots (ms)=5124
  Total time spent by all reduces in occupied slots (ms)=2764
  Total time spent by all map tasks (ms)=5124
  Total time spent by all reduce tasks (ms)=2764
  Total vcore-milliseconds taken by all map tasks=5124
```

```
Total time spent by all reduce tasks (ms)=2764
Total vcore-milliseconds taken by all map tasks=5124
Total vcore-milliseconds taken by all reduce tasks=2764
Total megabyte-milliseconds taken by all map tasks=5246976
Total megabyte-milliseconds taken by all reduce tasks=2830336
Map-Reduce Framework
  Map input records=1
  Map output records=22
  Map output bytes=191
  Map output materialized bytes=247
  Input split bytes=196
  Combine input records=0
  Combine output records=0
  Reduce input groups=18
  Reduce shuffle bytes=247
  Reduce input records=22
  Reduce output records=18
  Spilled Records=44
  Shuffled Maps =2
  Failed Shuffles=0
  Merged Map outputs=2
  GC time elapsed (ms)=162
  CPU time spent (ms)=2180
  Physical memory (bytes) snapshot=885735424
  Virtual memory (bytes) snapshot=7685988352
  Total committed heap usage (bytes)=678952960
  Peak Map Physical memory (bytes)=323989504
  Peak Map Virtual memory (bytes)=2561187840
  Peak Reduce Physical memory (bytes)=240586752
  Peak Reduce Virtual memory (bytes)=2564431872
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=221
File Output Format Counters
  Bytes Written=160
2024-10-16 17:09:13,649 INFO streaming.StreamJob: Output directory: /p2/python/output
hadoop@NuvebookV1:~/lab/p2$ |
```