Ex No: 2

Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm

AIM:

To run a basic Word Count MapReduce program.

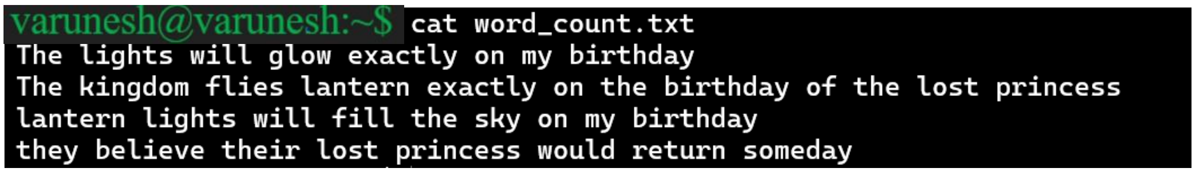
Procedure:

Step 1: Create Data File:

Create a file named "word_count_data.txt" and populate it with text data that you wish to analyse. Login with your hadoop user.

```
nano word_count.txt
```

Output: Type the below content in word_count.txt



```
varunesh@varunesh:~$ cat word_count.txt
The lights will glow exactly on my birthday
The kingdom flies lantern exactly on the birthday of the lost princess
lantern lights will fill the sky on my birthday
they believe their lost princess would return someday
```

Step 2: Mapper Logic - mapper.py:

Create a file named "mapper.py" to implement the logic for the mapper. The mapper will read input data from STDIN, split lines into words, and output each word with its count.

```
nano mapper.py
# Copy and paste the mapper.py code
```

```
#!/usr/bin/env python3
# import sys because we need to read and write data to STDIN and STDOUT
#!/usr/bin/python3
import sys
for line in sys.stdin:
    line = line.strip()    # remove leading and trailing whitespace
    words = line.split()  # split the line into words
    for word in words:

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

            .
```

Step 3: Reducer Logic - reducer.py:

Create a file named "reducer.py" to implement the logic for the reducer. The reducer will aggregate the occurrences of each word and generate the final output.

```
nano reducer.py
# Copy and paste the reducer.py code
```

reducer.py

```
#!/usr/bin/python3 from operator
import itemgetter import sys
current_word = None current_count
= 0 word = None for line in
sys.stdin:        line = line.strip()
word, count = line.split('\t', 1)
try:
        count = int(count)
except ValueError:
continue          if current_word
== word:           current_count
+= count     else:
        if current_word:
            print( '%s\t%s' % (current_word, current_count))
current_count = count        current_word = word if
current_word == word:        print( '%s\t%s' %
(current_word, current_count))
```

Step 4: Prepare Hadoop Environment:

Start the Hadoop daemons and create a directory in HDFS to store your data.

```
start-all.sh hdfsdfs -mkdir /word_count_in_python hdfsdfs -copyFromLocal
/path/to/word_count.txt/word_count_in_python
```

Step 6: Make Python Files Executable:

Give executable permissions to your mapper.py and reducer.py files.

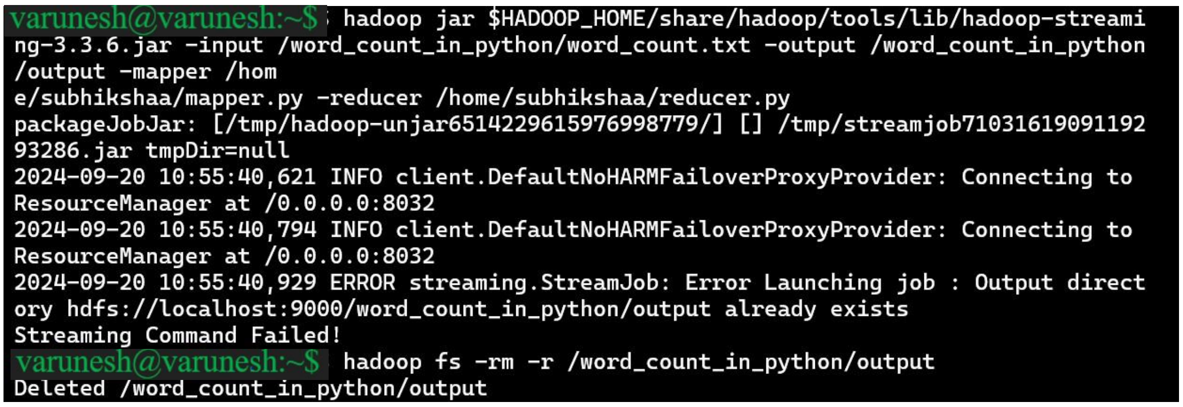
```
chmod 777 mapper.py reducer.py
```

Step 7: Run Word Count using Hadoop Streaming:

Download the latest hadoop-streaming jar file and place it in a location you can easily access.

Then run the Word Count program using Hadoop Streaming.

```
hadoop jar /path/to/hadoop-streaming-3.3.6.jar \     -input
/word_count_in_python/word_count_data.txt \
    -output /word_count_in_python/new_output \
    -mapper /path/to/mapper.py \
    -reducer /path/to/reducer.py
```

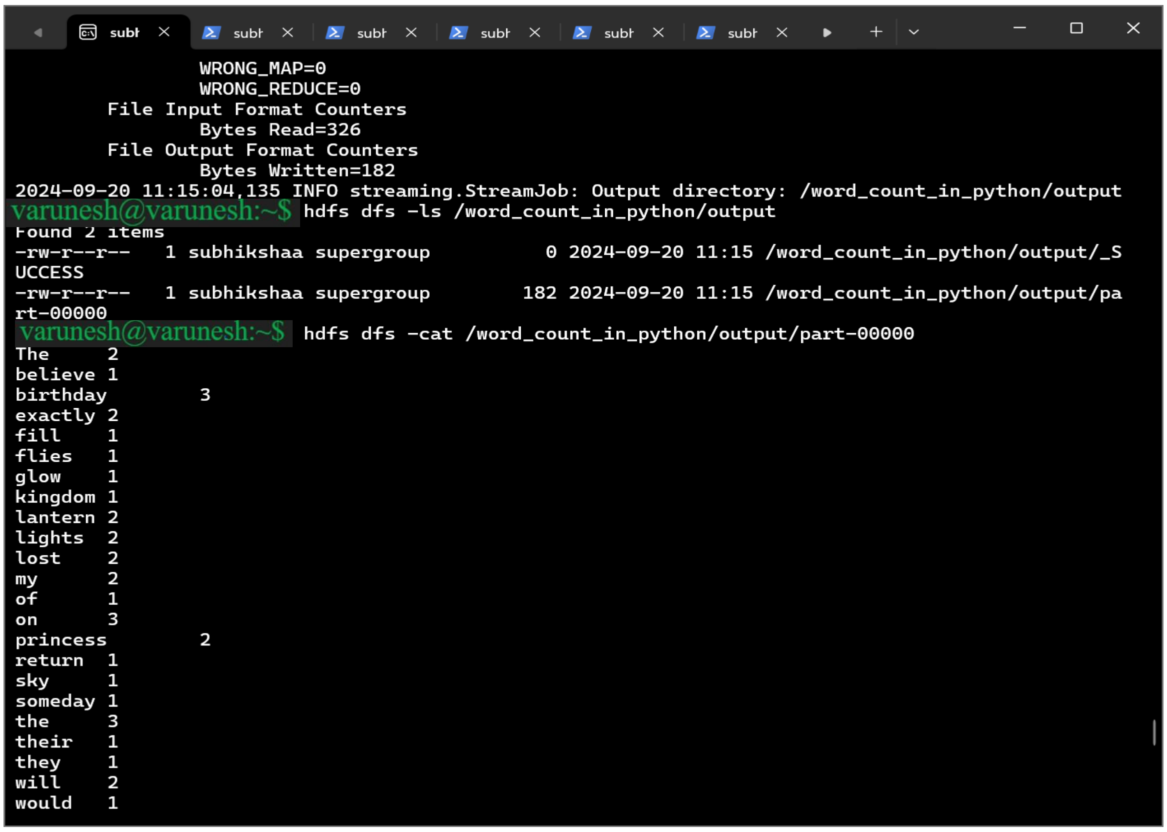


```
varunesh@varunesh:~$ hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streami
ng-3.3.6.jar -input /word_count_in_python/word_count.txt -output /word_count_in_python
/output -mapper /hom
e/subhikshaa/mapper.py -reducer /home/subhikshaa/reducer.py
packageJobJar: [/tmp/hadoop-unjar6514229615976998779/] [] /tmp/streamjob71031619091192
93286.jar tmpDir=null
2024-09-20 10:55:40,621 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to
ResourceManager at /0.0.0.0:8032
2024-09-20 10:55:40,794 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to
ResourceManager at /0.0.0.0:8032
2024-09-20 10:55:40,929 ERROR streaming.StreamJob: Error Launching job : Output direct
ory hdfs://localhost:9000/word_count_in_python/output already exists
Streaming Command Failed!
varunesh@varunesh:~$ hadoop fs -rm -r /word_count_in_python/output
Deleted /word_count_in_python/output
```

Step 8: Check Output:

Check the output of the Word Count program in the specified HDFS output directory.

hdfs dfs -cat /word_count_in_python/new_output/part-00000



```
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=326
        File Output Format Counters
                Bytes Written=182
2024-09-20 11:15:04,135 INFO streaming.StreamJob: Output directory: /word_count_in_python/output
varunesh@varunesh:~$ hdfs dfs -ls /word_count_in_python/output
Found 2 items
-rw-r--r--   1 subhikshaa supergroup          0 2024-09-20 11:15 /word_count_in_python/output/_S
UCCESS
-rw-r--r--   1 subhikshaa supergroup        182 2024-09-20 11:15 /word_count_in_python/output/pa
rt-00000
varunesh@varunesh:~$ hdfs dfs -cat /word_count_in_python/output/part-00000
The     2
believe 1
birthday        3
exactly 2
fill    1
flies   1
glow    1
kingdom 1
lantern 2
lights  2
lost    2
my      2
of      1
on      3
princess        2
return  1
sky     1
someday 1
the     3
their   1
they    1
will    2
would   1
```

Result:

Thus, the program for basic Word Count Map Reduce has been executed successfully.