Exp .No : 9 **Date :**

DEMONSTRATE THE MAP REDUCE PROGRAMMING MODEL BY COUNTING THE NUMBER OF WORDS IN A FILE

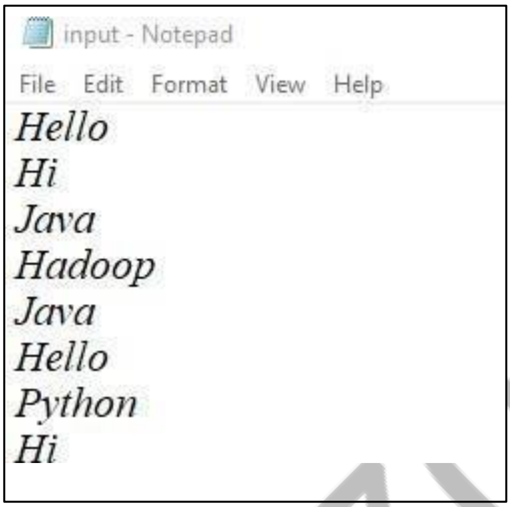
AIM:

To demonstrate the MAP REDUCE programming model for counting the number of words in a file.

PROCEDURE:

Step 1: Create Data File:

Create a file named "input.txt" and populate it with text data that you wish to analyse.



```
Hello
Hi
Java
Hadoop
Java
Hello
Python
Hi
```

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.

mapper.py:

```
#!C:/Users/user/AppData/Local/Microsoft/WindowsApps/python.exe
import sys for line in sys.stdin: line = line.strip() words = line.split()
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.

reducer.py:

```
#!C:/Users/user/AppData/Local/Microsoft/WindowsApps/python.exe
import sys prev_word = None prev_count = 0 for line in sys.stdin:
    line = line.strip()
    word, count = line.split('\t')
    count = int(count) if
    prev_word == word:
        prev_count += count else:
```

```
        if prev_word: print('%s\t%s' %(prev_word,
            prev_count))
        prev_count = count prev_word
        = word
if prev_word == word:
    print('%s\t%s' %(prev_word, prev_count))
```

Step 4: Prepare Hadoop Environment:

Start the Hadoop daemons and create a directory in HDFS to store your data. Run the following commands to store the data in the WordCount Directory.

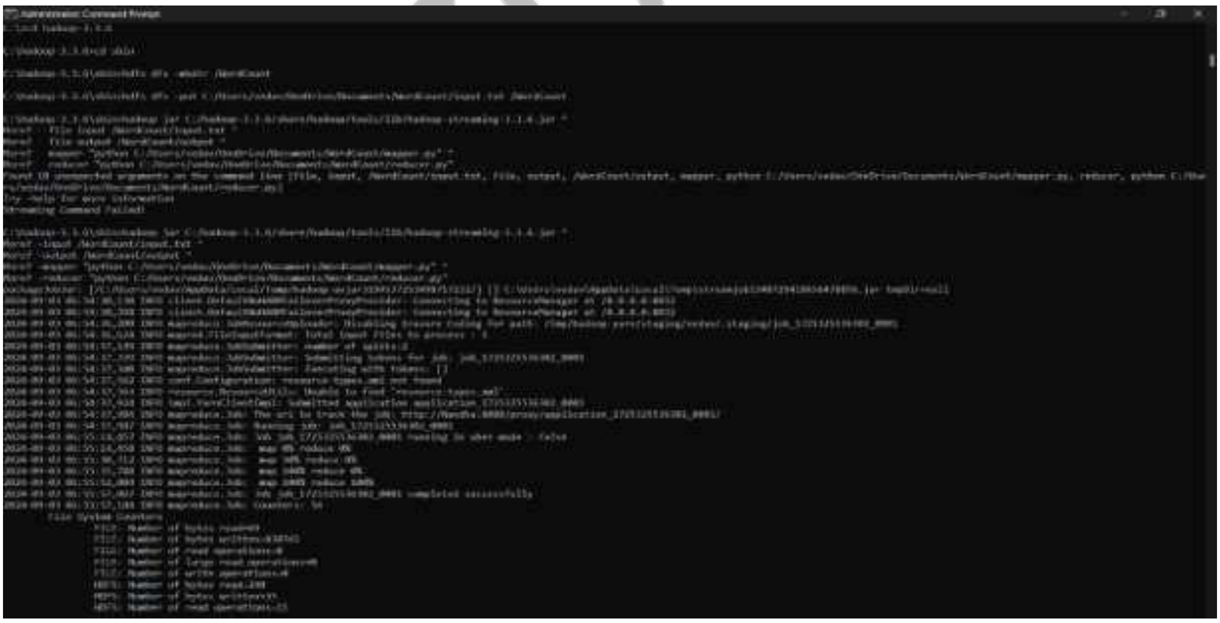
```
start-all.cmd cd C:/Hadoop/sbin hdfs dfs -mkdir /WordCount hdfs dfs -put
C:/Users/user/Documents/DataAnalytics/input.txt /WordCount hadoop jar
C:\hadoop\share\hadoop\tools\lib\hadoop-streaming-3.3.6.jar ^
-input /WordCount/input.txt ^
-output /WordCount/output ^
-mapper "python C:/ Users/user/Documents/DataAnalytics/mapper.py" ^ -reducer
"python C:/ Users/user/Documents/DataAnalytics/reducer.py"
```

Step 5: Check Output:

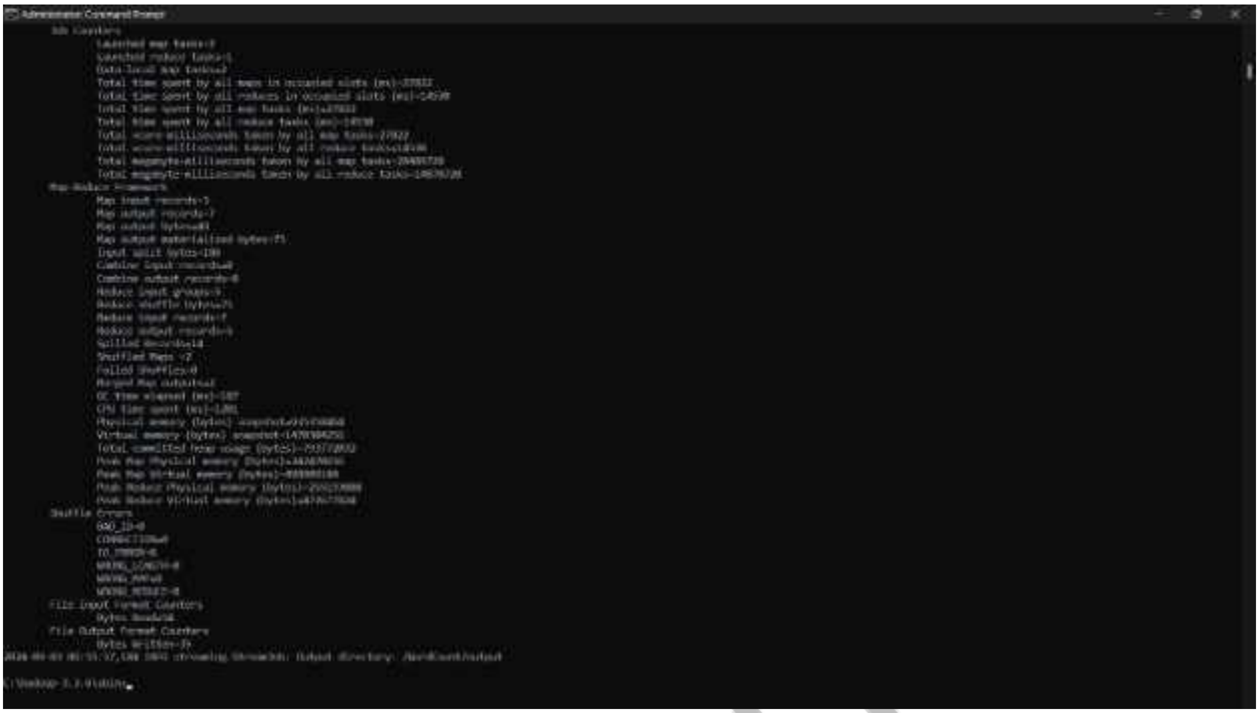
Check the output of the Word Count program in the specified HDFS output directory.

```
hdfs dfs -cat /WordCount/output/part-00000
```

OUTPUT:

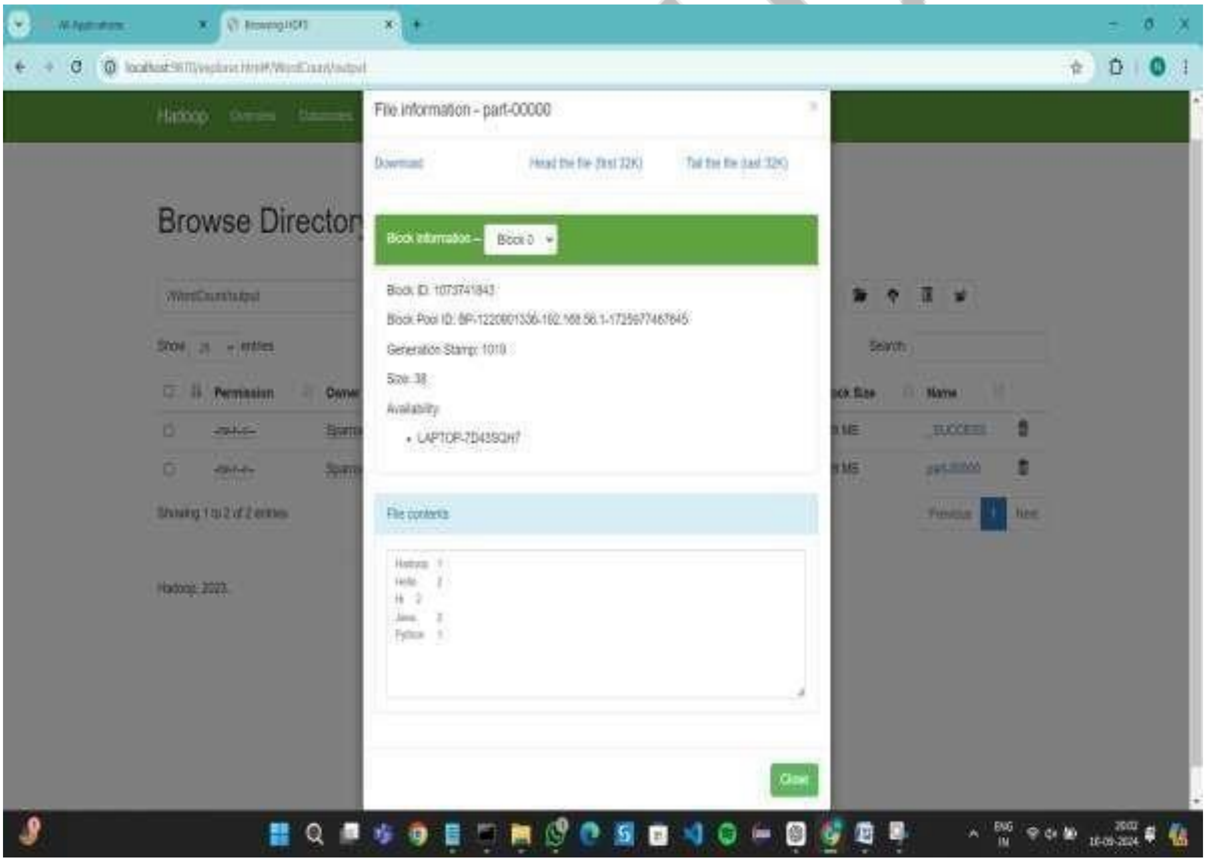


```
[illegible]
Try -help for more information
Streaming Command Failed!
[illegible]
INFO mapreduce.Job:  map 0% reduce 0%
[illegible]
INFO mapreduce.Job:  map 100% reduce 100%
[illegible] completed successfully
[illegible]
        File System Counters
                [illegible]
```



```
Administrator: Command Prompt
        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)=27022
                Total time spent by all reduces in occupied slots (ms)=14598
                Total time spent by all map tasks (ms)=27022
                Total time spent by all reduce tasks (ms)=14598
                Total vcore-milliseconds taken by all map tasks=27022
                Total vcore-milliseconds taken by all reduce tasks=14598
                Total megabyte-milliseconds taken by all map tasks=[illegible]
                Total megabyte-milliseconds taken by all reduce tasks=[illegible]
        Map-Reduce Framework
                Map input records=5
                Map output records=7
                Map output bytes=[illegible]
                Map output materialized bytes=75
                Input split bytes=198
                Combine input records=[illegible]
                Combine output records=[illegible]
                Reduce input groups=5
                Reduce shuffle bytes=75
                Reduce input records=7
                Reduce output records=5
                Spilled Records=14
                Shuffled Maps =2
                Failed Shuffles=0
                Merged Map outputs=2
                GC time elapsed (ms)=187
                CPU time spent (ms)=[illegible]
                Physical memory (bytes) snapshot=[illegible]
                Virtual memory (bytes) snapshot=[illegible]
                Total committed heap usage (bytes)=[illegible]
                Peak Map Physical memory (bytes)=[illegible]
                Peak Map Virtual memory (bytes)=[illegible]
                Peak Reduce Physical memory (bytes)=[illegible]
                Peak Reduce Virtual memory (bytes)=[illegible]
        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=[illegible]
        File Output Format Counters
                Bytes Written=[illegible]
2024-09-[illegible] [illegible] INFO streaming.StreamJob: Output directory: /WordCount/output

C:\hadoop-3.3.6\sbin>
```



Browse Directory

/WordCount/output

Show 25 entries

Permission	Owner	…	Block Size	Name
-rw-r--r--	Sparro…	…	… MB	_SUCCESS
-rw-r--r--	Sparro…	…	… MB	part-00000

Showing 1 to 2 of 2 entries

Hadoop, 2023.

File information - part-00000

Download   Head the file (first 32K)   Tail the file (last 32K)

Block information -- Block 0

Block ID: 1073741843

Block Pool ID: BP-1220801336-192.168.56.1-1725977467845

Generation Stamp: 1019

Size: 38

Availability:

- LAPTOP-7D43SQH7

File contents

```
Hadoop  1
Hello   2
Hi  2
Java    2
Python  1
```

RESULT:

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