

EXP 2: Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

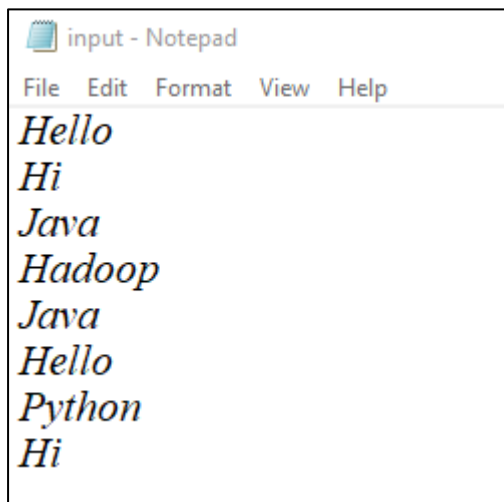
AIM:

To run a basic Word Count MapReduce program using Hadoop.

PROCEDURE:

Step 1: Create Data File:

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



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"

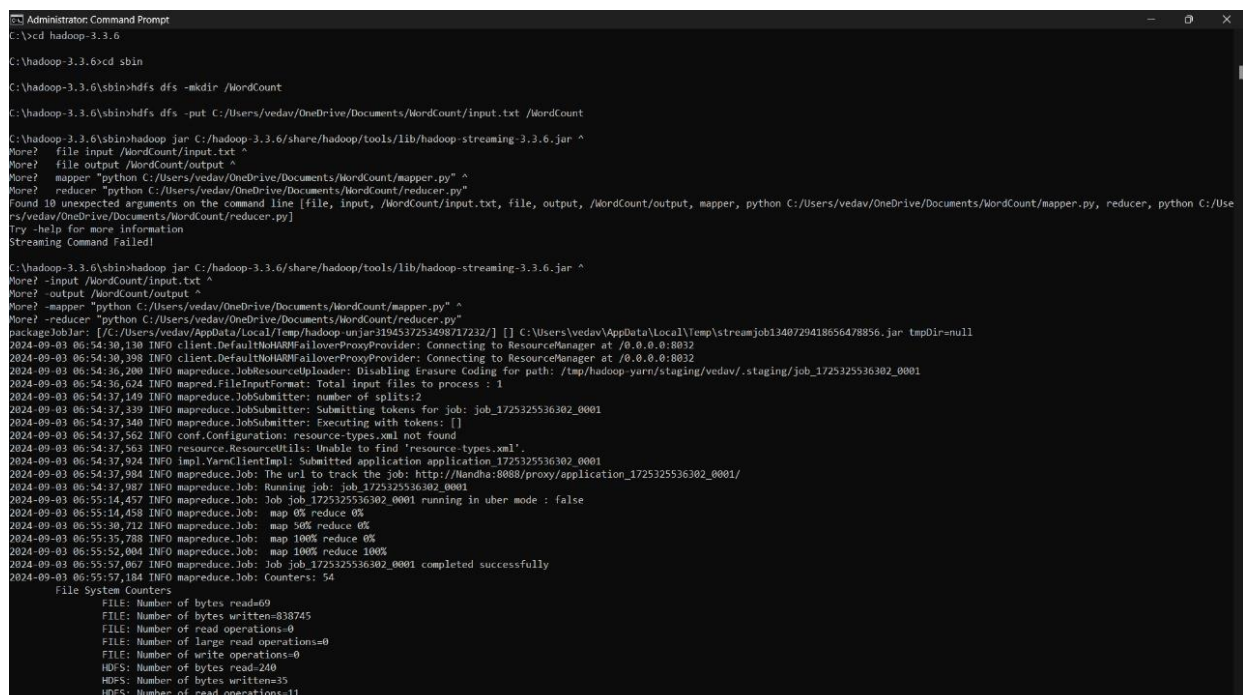
```

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:



```

Administrator: Command Prompt
C:\>cd hadoop-3.3.6
C:\hadoop-3.3.6>cd sbin
C:\hadoop-3.3.6\sbin>hdfs dfs -mkdir /WordCount
C:\hadoop-3.3.6\sbin>hdfs dfs -put C:/Users/vedav/OneDrive/Documents/WordCount/input.txt /WordCount
C:\hadoop-3.3.6\sbin>hadoop jar C:/hadoop-3.3.6/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar ^
More? file input /WordCount/input.txt ^
More? file output /WordCount/output ^
More? mapper "python C:/Users/vedav/OneDrive/Documents/WordCount/mapper.py" ^
More? reducer "python C:/Users/vedav/OneDrive/Documents/WordCount/reducer.py"
Found 10 unexpected arguments on the command line [file, input, /WordCount/input.txt, file, output, /WordCount/output, mapper, python C:/Users/vedav/OneDrive/Documents/WordCount/mapper.py, reducer, python C:/Users/vedav/OneDrive/Documents/WordCount/reducer.py]
Try -help for more information
Streaming Command Failed!
C:\hadoop-3.3.6\sbin>hadoop jar C:/hadoop-3.3.6/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar ^
More? -input /WordCount/input.txt ^
More? -output /WordCount/output ^
More? -mapper "python C:/Users/vedav/OneDrive/Documents/WordCount/mapper.py" ^
More? -reducer "python C:/Users/vedav/OneDrive/Documents/WordCount/reducer.py"
packageJobJar: [C:/Users/vedav/AppData/Local/Temp/hadoop-unjar319453723408/17232/] [C:/Users/vedav/AppData/Local/Temp/streamjob1340729418656478856.jar tmpDir=null]
2024-09-03 06:54:30,130 INFO client.DefaultHadoopFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-09-03 06:54:30,398 INFO client.DefaultHadoopFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-09-03 06:54:36,200 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/vedav/.staging/job_1725325536302_0001
2024-09-03 06:54:36,624 INFO mapred.FileInputFormat: Total input files to process : 1
2024-09-03 06:54:37,140 INFO mapreduce.JobSubmitter: number of splits:2
2024-09-03 06:54:37,339 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1725325536302_0001
2024-09-03 06:54:37,340 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-09-03 06:54:37,562 INFO conf.Configuration: resource-types.xml not found
2024-09-03 06:54:37,563 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2024-09-03 06:54:37,924 INFO impl.YarnClientImpl: Submitted application application_1725325536302_0001
2024-09-03 06:54:37,984 INFO mapreduce.Job: The url to track the job: http://Nandha:8088/proxy/application_1725325536302_0001/
2024-09-03 06:54:37,987 INFO mapreduce.Job: Running job: job_1725325536302_0001
2024-09-03 06:55:14,457 INFO mapreduce.Job: Job job_1725325536302_0001 running in uber mode : false
2024-09-03 06:55:14,458 INFO mapreduce.Job: map 0% reduce 0%
2024-09-03 06:55:30,712 INFO mapreduce.Job: map 50% reduce 0%
2024-09-03 06:55:35,788 INFO mapreduce.Job: map 100% reduce 0%
2024-09-03 06:55:52,004 INFO mapreduce.Job: map 100% reduce 100%
2024-09-03 06:55:57,067 INFO mapreduce.Job: Job job_1725325536302_0001 completed successfully
2024-09-03 06:55:57,184 INFO mapreduce.Job: Counters: 54
File System Counters
  FILE: Number of bytes read=0
  FILE: Number of bytes written=838745
  FILE: Number of read operations=0
  FILE: Number of large read operations=0
  FILE: Number of write operations=0
  HDFS: Number of bytes read=240
  HDFS: Number of bytes written=35
  HDFS: Number of read operations=11

```

```

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)=27822
  Total time spent by all reduces in occupied slots (ms)=14530
  Total time spent by all map tasks (ms)=27822
  Total time spent by all reduce tasks (ms)=14530
  Total vcore-milliseconds taken by all map tasks=27822
  Total vcore-milliseconds taken by all reduce tasks=14530
  Total megabyte-milliseconds taken by all map tasks=28489728
  Total megabyte-milliseconds taken by all reduce tasks=14878720

Map-Reduce Framework
  Map input records=3
  Map output records=7
  Map output bytes=49
  Map output materialized bytes=75
  Input split bytes=186
  Combine input records=0
  Combine output records=0
  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)=1201
  Physical memory (bytes) snapshot=935358464
  Virtual memory (bytes) snapshot=1490304256
  Total committed heap usage (bytes)=793772832
  Peak Map Physical memory (bytes)=342470656
  Peak Map Virtual memory (bytes)=498909184
  Peak Reduce Physical memory (bytes)=255193088
  Peak Reduce Virtual memory (bytes)=473677824

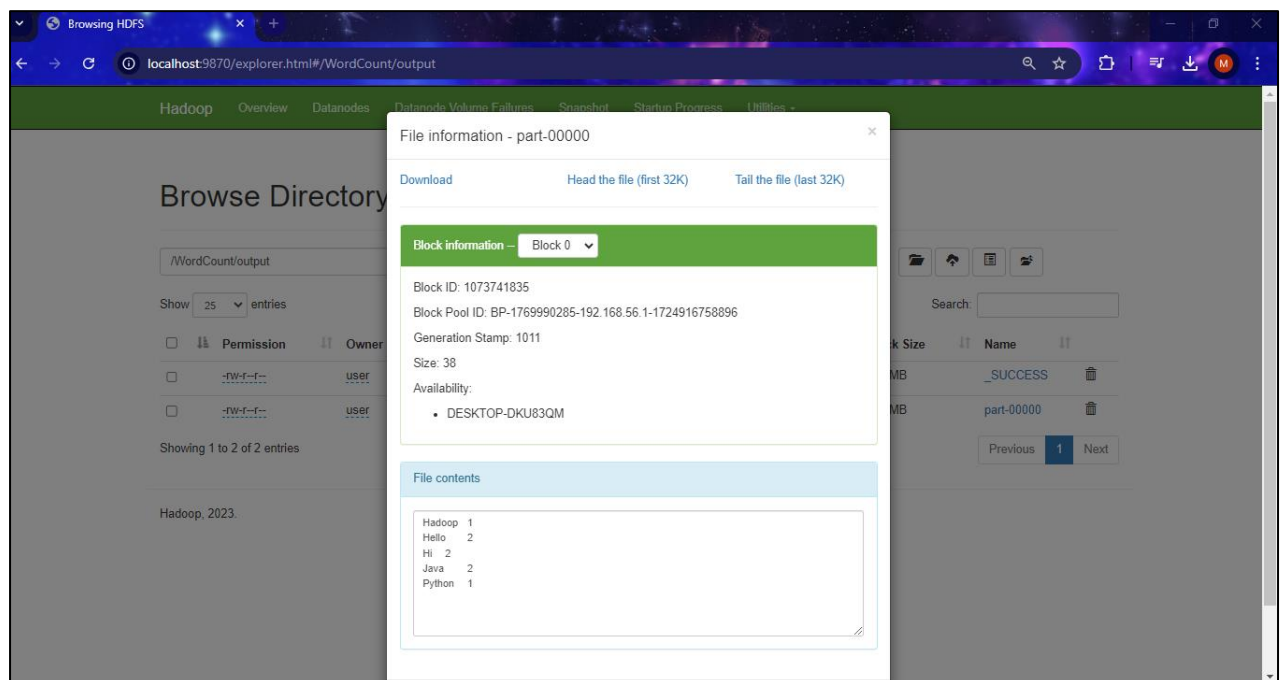
Shuffle Errors
  BAD_ID=0
  CONNECTION=0
  IO_ERROR=0
  WRONG_LENGTH=0
  WRONG_MAP=0
  WRONG_REDUCE=0

File Input Format Counters
  Bytes Read=54

File Output Format Counters
  Bytes Written=35

2024-09-03 06:55:57,184 INFO streaming.StreamJob: Output directory: /WordCount/output
C:\hadoop-3.3.6\sbin>

```



RESULT:

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