

IMPLEMENT WORD COUNT/FREQUENCY PROGRAMS USING MAPREDUCE

AIM:

To implement the python mapper and reducer programs using MapReduce to count the words in a text file using Hadoop.

PROCEDURE:

1. Open command prompt as administrator and start the Hadoop by using the command:

```
start-all.cmd
```

2. Create a new directory in the Hadoop file systems using the command:

```
hadoop fs -mkdir /wordCount
```

3. Upload the input text file into the wordCount directory using the command:

```
hadoop fs -put C:/Users/mercy/OneDrive/Documents/DataAnalytics/input.txt /wordcount
```

4. Create the mapper and reducer files.

5. To execute the files with Hadoop streaming run the following command:

```
hadoop jar C:/hadoop-3.3.6/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar ^ -file  
C:/Users/mercy/Documents/DataAnalytics/mapper.py ^ -file  
C:/Users/mercy/Documents/DataAnalytics/reducer.py ^ -input /wordCount/input.txt ^ -output  
/user/output ^ -mapper "python mapper.py" ^ -reducer "python reducer.py"
```

MAPPER.PY

```
#!/C:/ProgramData/chocolatey/bin/python3.exe
```

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

```
line.strip() words = line.split() for word
```

```
in words:
```

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

REDUCER.PY

```
#!/C:/ProgramData/chocolatey/bin/python3.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)

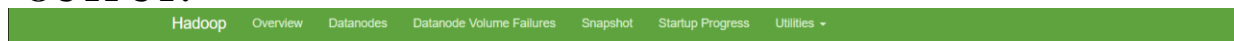
if(prev_word == word):    prev_count += count

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))

```

OUTPUT:



Browse Directory

/

Go!

Show

25

entries

Search:

| <input type="checkbox"/> | Permission | Owner | Group | Size | Last Modified | Replication | Block Size | Name | |
|--------------------------|----------------------------|---------------------------|----------------------------|------|---------------|-------------------|------------|--------------------------------|--|
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Aug 12 22:41 | 0 | 0 B | input | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Aug 13 08:27 | 0 | 0 B | output1 | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Sep 08 18:54 | 0 | 0 B | pig | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Sep 08 18:52 | 0 | 0 B | tmp | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Sep 08 13:01 | 0 | 0 B | user | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Aug 20 19:37 | 0 | 0 B | user_output | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Aug 23 08:48 | 0 | 0 B | weather | |
| <input type="checkbox"/> | drwxr-xr-x | MONASHREE | supergroup | 0 B | Aug 23 08:54 | 0 | 0 B | weather_output | |

Showing 1 to 8 of 8 entries

Previous

1

Next

Browse Directory

/user_output

Go!

Show

25

entries

Search:

| <input type="checkbox"/> | Permission | Owner | Group | Size | Last Modified | Replication | Block Size | Name | |
|--------------------------|------------|-----------|------------|------|---------------|-------------------|------------|----------------------------|--|
| <input type="checkbox"/> | -rw-r--r-- | MONASHREE | supergroup | 0 B | Aug 20 19:37 | 1 | 128 MB | _SUCCESS | |
| <input type="checkbox"/> | -rw-r--r-- | MONASHREE | supergroup | 40 B | Aug 20 19:37 | 1 | 128 MB | part-00000 | |

Showing 1 to 2 of 2 entries

Previous

1

Next

Block information —

Block 0 ▾

Block ID: 1073741879

Block Pool ID: BP-468074218-172.16.11.91-1722913037250

Generation Stamp: 1055

Size: 40

Availability:

- LAPTOP-VG848917

File contents

```
evening 1
good 2
hello 3
hi 1
morning 1
```

Close

RESULT:

Thus the implementation of the python mapper and reducer programs using MapReduce to count the words in a text file using Hadoop is executed successfully.