

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/gjega/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/gjega/Documents/Hadoop_wordcount/mapper.py ^ -file  
C:/Users/gjega/Documents/Hadoop_wordcount/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

```
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:

Hadoop

Overview

Datanodes

Datanode Volume Failures

Snapshot

Startup Progress

Utilities

Browse Directory

/

Go!

Show

25

entries

Search:

<input type="checkbox"/>	<div>Permission</div>	<div>Owner</div>	<div>Group</div>	<div>Size</div>	<div>Last Modified</div>	<div>Replication</div>	<div>Block Size</div>	<div>Name</div>	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 27 12:31	0	0 B	plg	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 29 14:15	0	0 B	tmp	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 29 14:27	0	0 B	user	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 27 14:11	0	0 B	weather_input	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 27 14:13	0	0 B	weather_output	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 27 14:04	0	0 B	wordcount_input	<div></div>
<input type="checkbox"/>	drwxr-xr-x	glega	supergroup	0 B	Aug 27 14:08	0	0 B	wordcount_output	<div></div>

Showing 1 to 7 of 7 entries

Previous

1

Next

[Hadoop](#) [Overview](#) [Datanodes](#) [Datanode Volume Failures](#) [Snapshot](#) [Startup Progress](#) [Utilities](#)

Browse Directory

Show entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rW-r--r--	glega	supergroup	0 B	Aug 27 14:08	1	128 MB	_SUCCESS	
<input type="checkbox"/>	-rW-r--r--	glega	supergroup	21 B	Aug 27 14:08	1	128 MB	part-00000	

Showing 1 to 2 of 2 entries

[Overview](#) [Datanodes](#) [Datanode Volume Failures](#) [Snapshot](#) [Startup Progress](#) [Utilities](#)

Browse Directory

Show entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rW-r--r--	glega	supergroup	0 B	Aug 27 14:08	1	128 MB	_SUCCESS	
<input type="checkbox"/>	-rW-r--r--	glega	supergroup	21 B	Aug 27 14:08	1	128 MB	part-00000	

Showing 1 to 2 of 2 entries

File information - part-00000

[Download](#) [Head the file \(first 32K\)](#) [Tail the file \(last 32K\)](#)

Block information --

Block ID: 1073741848

Block Pool ID: BP-55145513-172.28.96.1-1724741555419

Generation Stamp: 1024

Size: 21

Availability:

- Jegan

File contents

```
hello 2
hi 1
world 2
```

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.