

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

Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	mercy	supergroup	0 B	Aug 19 09:01	0	0 B	tmp
drwxr-xr-x	mercy	supergroup	0 B	Aug 18 21:18	0	0 B	weather
drwxr-xr-x	mercy	supergroup	0 B	Aug 13 19:41	0	0 B	wordCount

Showing 1 to 3 of 3 entries

Previous **1** Next

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities -

Browse Directory

/user/jayas

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	-rw-r--r--	jayas	supergroup	28 B	Aug 13 09:39	1	128 MB	data.txt	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 26 11:27	0	0 B	input	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 26 18:50	0	0 B	output	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 23 09:35	0	0 B	output2	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 28 14:05	0	0 B	outputpig	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 26 18:49	0	0 B	scripts	<input type="checkbox"/>

Showing 1 to 6 of 6 entries 1

Hadoop, 2023.

Hadoop Overview Datanodes

Browse Directory

/user/jayas/output

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	-rw-r--r--	jayas	supergroup	28 B	Aug 13 09:39	1	128 MB	data.txt	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 26 11:27	0	0 B	input	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 26 18:50	0	0 B	output	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 23 09:35	0	0 B	output2	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 28 14:05	0	0 B	outputpig	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	jayas	supergroup	0 B	Aug 26 18:49	0	0 B	scripts	<input type="checkbox"/>

Showing 1 to 3 of 3 entries 1

Hadoop, 2023.

File information - part-00000

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

Block information -- Block 0

Block ID: 1073741853
Block Pool ID: BP-182655827-192.168.130.1-1723180668634
Generation Stamp: 1029
Size: 65
Availability:
• LAPTOP-5AEVBO11

File contents

```
a 2
app 2
bus 2
djsa 1
fdtghf 1
hai 2
hello 1
jehan 1
```

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.