

IMPLEMENT A MAPREDUCE PROGRAM TO PROCESS A WEATHER DATASET

AIM:

To implement a MapReduce python program to process a weather dataset in 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 /weather
```

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

```
hadoop fs -put C:/DA/weather/weather.txt /weather
```

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" ^-input  
/weather/weather.txt ^-output /weather/output/ ^-mapper "python C:/weather/mapper.py" ^-  
reducer "python C:/weather/reducer.py"
```

MAPPER.PY:

```
#!C:/Users/md_aa/AppData/Local/Microsoft/  
WindowsApps/python.exe
```

```
import sys
```

```
def map1():
```

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

```
        tokens = line.strip().split()
```

```
        if len(tokens) < 13:
```

```
            continue
```

```
        station = tokens[0]
```

```
if "STN" in station:
    continue

date_hour = tokens[2]
temp = tokens[3]
dew = tokens[4]
wind = tokens[12]

if temp == "9999.9" or dew == "9999.9" or wind == "999.9":
    continue

hour = int(date_hour.split("_")[-1])
date = date_hour[:date_hour.rfind("_")-2]

if 4 < hour <= 10:
    section = "section1"
elif 10 < hour <= 16:
    section = "section2"
elif 16 < hour <= 22:
    section = "section3"
else:
    section = "section4"

key_out = f"{station}_{date}_{section}"
value_out = f"{temp} {dew} {wind}"
print(f"{key_out}\t{value_out}")

if __name__ == "__main__":
    map1()
```

REDUCER.PY:

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

import sys

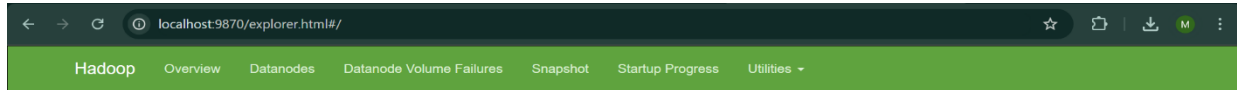
def reduce1():
    current_key = None

    sum_temp, sum_dew, sum_wind = 0, 0, 0
```

```
count = 0

for line in sys.stdin:
    key, value = line.strip().split("\t")
    temp, dew, wind = map(float, value.split())
    if current_key is None:
        current_key = key
    if key == current_key:
        sum_temp += temp
        sum_dew += dew
        sum_wind += wind
        count += 1
    else:
        avg_temp = sum_temp / count
        avg_dew = sum_dew / count
        avg_wind = sum_wind / count
        print(f"{current_key}\t{avg_temp} {avg_dew} {avg_wind}")
        current_key = key
        sum_temp, sum_dew, sum_wind = temp, dew, wind
        count = 1
    if current_key is not None:
        avg_temp = sum_temp / count
        avg_dew = sum_dew / count
        avg_wind = sum_wind / count
        print(f"{current_key}\t{avg_temp} {avg_dew} {avg_wind}")
if __name__ == "__main__":
    reduce1()
```

OUTPUT:



Browse Directory

/ Go!

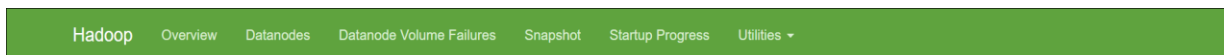
Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	drwxr-xr-x	md_aa	supergroup	0 B	Sep 17 23:03	0	0 B	tmp	
<input type="checkbox"/>	drwxr-xr-x	md_aa	supergroup	0 B	Sep 17 23:29	0	0 B	weather	
<input type="checkbox"/>	drwxr-xr-x	md_aa	supergroup	0 B	Sep 17 23:15	0	0 B	wordCount	

Showing 1 to 3 of 3 entries

Previous 1 Next

Hadoop, 2023.



Browse Directory

/weather Go!

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	drwxr-xr-x	md_aa	supergroup	0 B	Sep 17 23:29	0	0 B	output	
<input type="checkbox"/>	-rw-r--r--	md_aa	supergroup	144 B	Sep 17 23:27	1	128 MB	weather.txt	

Showing 1 to 2 of 2 entries

Previous 1 Next

Hadoop, 2023.

The screenshot displays the Hadoop web interface. On the left, the 'Browse Directory' view shows the path '/weather/output' with a search bar and a table of files. The table has columns for 'Permission', 'Owner', and 'Size'. Two entries are visible, both with permissions '-rw-r--r--' and owner 'md_ae'. The right pane shows 'File information - part-000000'. It includes tabs for 'Download', 'Head the file (first 32K)', and 'Tail the file (last 32K)'. The 'Block information' section shows 'Block 0' with details: Block ID: 1073741968, Block Pool ID: BP-551977252-172.17.48.1-1724733081844, Generation Stamp: 1144, Size: 98, and Availability: mohamed-aadhil.mshome.net. The 'File contents' section shows a table with four rows of data: 2000, 2001, 2002, and 2003, each followed by a long hexadecimal string.

Block information --	Block 0
Block ID:	1073741968
Block Pool ID:	BP-551977252-172.17.48.1-1724733081844
Generation Stamp:	1144
Size:	98
Availability:	mohamed-aadhil.mshome.net

File contents	
2000	15.533333333333333
2001	15.433333333333332
2002	16.433333333333334
2003	15.600000000000001

RESULT:

Thus, the implementation of the MapReduce python program to process a weather dataset inHadoop is executed successfully.