

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:/Users/gjega/OneDrive/Documents/hadoop_weather/WeatherPrediction/sample_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 ^ -file  
C:/Users/gjega/Documents/ hadoop_weather /WeatherPrediction/mapper.py ^ -file  
C:/Users/gjega/Documents/ hadoop_weather /WeatherPrediciton/reducer.py ^ -input  
/weather/sample_weather.txt ^ -output /weather/output ^ -mapper "python mapper.py" ^ -  
reducer "python reducer.py"
```

MAPPER.PY:

```
import sys  
  
for line in sys.stdin:  
    # Strip whitespace and skip empty lines  
    line = line.strip()  
  
    if not line:  
        continue  
  
    fields = line.split(',')  
  
    if len(fields) < 2:  
        continue # Skip lines that don't have enough fields
```

```
date = fields[0]
year = date[:4] # Extract the year (first 4 characters of date)
temperature = fields[1]

# Print the year and temperature
print(f"{year}\t{temperature}")
```

REDUCER.PY:

```
import sys

current_year = None
current_sum = 0.0
current_count = 0

for line in sys.stdin:
    line = line.strip()
    year, temperature = line.split('\t')

    # Skip non-numeric temperatures
    try:
        temperature = float(temperature)
    except ValueError:
        continue

    if current_year == year:
        current_sum += temperature
        current_count += 1
    else:
        if current_year:
            # Output the average temperature for the previous year
            print(f"{current_year}\t{current_sum / current_count:.2f}")

        current_year = year
        current_sum = temperature
        current_count = 1

    # Output the average temperature for the last year
    if current_year == year:
        print(f"{current_year}\t{current_sum / current_count:.2f}")
```

OUTPUT:

/

Go!

Show

25

entries

Search:

<input type="checkbox"/>	<div><div></div><div></div></div> Permission	<div><div></div><div></div></div> Owner	<div><div></div><div></div></div> Group	<div><div></div><div></div></div> Size	<div><div></div><div></div></div> Last Modified	<div><div></div><div></div></div> Replication	<div><div></div><div></div></div> Block Size	<div><div></div><div></div></div> Name	<div><div></div><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

Browse Directory

/wordcount_output

Go!

Show

25

entries

Search:

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

Showing 1 to 2 of 2 entries

Previous

1

Next

The screenshot displays the Hadoop Distributed File System (HDFS) interface. On the left, a sidebar shows the 'Overview' and 'Datanodes' tabs. The main content area is titled 'Block information' and shows details for 'Block 0'. The block information includes: Block ID: 1073741859, Block Pool ID: BP-55145513-172.28.96.1-1724741555419, Generation Stamp: 1035, Size: 24, and Availability: Jegan. Below this, the 'File contents' section shows a list of file contents: 16 A 229.00 and 17 A 230.00. On the right, a sidebar shows a table with columns 'Block Size' and 'Name'. The table contains two rows: 'MB' and 'part-00000'. The 'part-00000' row is highlighted. Below the table, there is a 'Previous' button and a '1' button. A 'Close' button is located at the bottom right of the main content area.

Overview Datanodes

Download Head the file (first 32K) Tail the file (last 32K)

Block information — Block 0 ▾

Block ID: 1073741859
Block Pool ID: BP-55145513-172.28.96.1-1724741555419
Generation Stamp: 1035
Size: 24
Availability:
• Jegan

File contents

16 A 229.00
17 A 230.00

Close

Block Size ▴ ▾ Name ▴ ▾

MB _SUCCESS
MB part-00000

Previous 1

RESULT:

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