IMPLEMENT A MAPREDUCE PROGRAM TO PROCESS A WEATHER DATASET

AIM:

To implement the python mapper and reducer programs using MapReduce to process the weather dataset file using Hadoop.

PROCEDURE:

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

```
C:\Windows\System32>cd C:/hadoop/sbin

C:\hadoop\sbin>start-all
This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

C:\hadoop\sbin>jps
20756 DataNode
20008 ResourceManager
21944 NodeManager
23996 Jps
6476 NameNode
```

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

   Upload the weather text file into the weather directory using the

   command:

```
C:\hadoop\sbin>hadoop fs -mkdir /weather

C:\hadoop\sbin>hadoop fs -put D:\Data_Analytics\weather\sample_weather.txt /weather
```

3. Create the mapper and reducer files.

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

```
C:\hadoop\sbin>hadoop jar C:\hadoop\share\hadoop\tools\lib\hadoop-streaming-3.3.6.jar ^
More?     -files "file:///D:/Data_Analytics/weather/mapper.py,file:///D:/Data_Analytics/weather/reducer.py" ^
More?     -input /weather/sample_weather.txt ^
More?     -output /weather/output ^
More?     -mapper "python D:/Data_Analytics/weather/mapper.py" ^
More?     -reducer "python D:/Data_Analytics/weather/reducer.py"
packageJobJar: [/C:/Users/OVIYA/AppData/Local/Temp/hadoop-unjar3512853263035869467/] [] C:\Users\OVIYA\AppData\Local\T
p\streamjob480328786206735836.jar tmpDir=null
2024-08-30 17:25:01,242 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-08-30 17:25:01,423 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-08-30 17:25:06,738 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/stagin
OVIYA/.staging/job_1725018253317_0002
2024-08-30 17:25:07,084 INFO mapred.FileInputFormat: Total input files to process : 1
2024-08-30 17:25:07,142 INFO mapreduce.JobSubmitter: number of splits:2
2024-08-30 17:25:07,254 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1725018253317_0002
```

MAPPER.PY

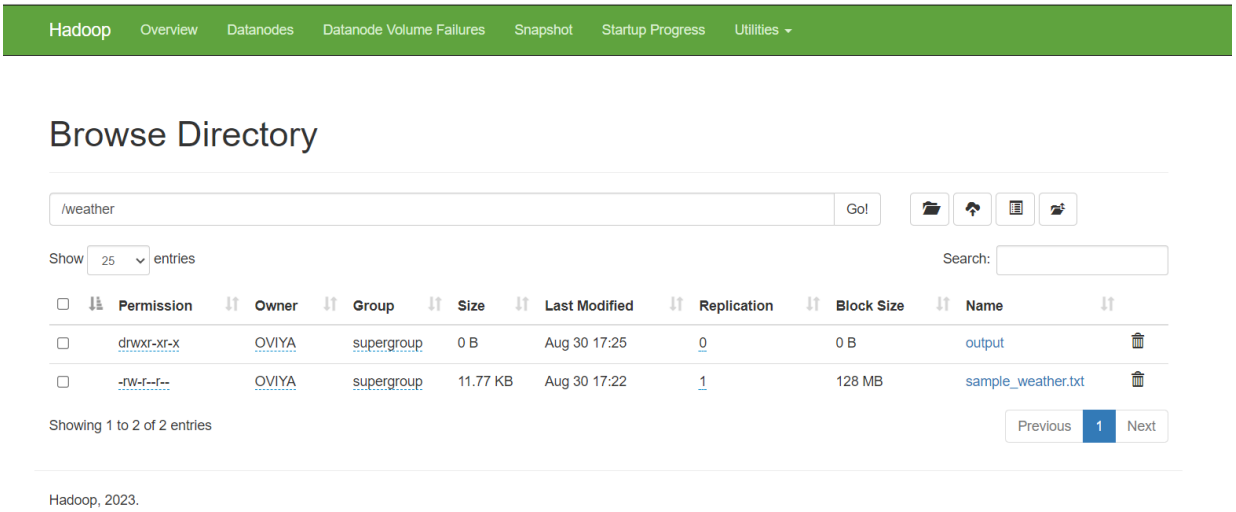
```
#!/usr/bin/python3
import sys

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()
```

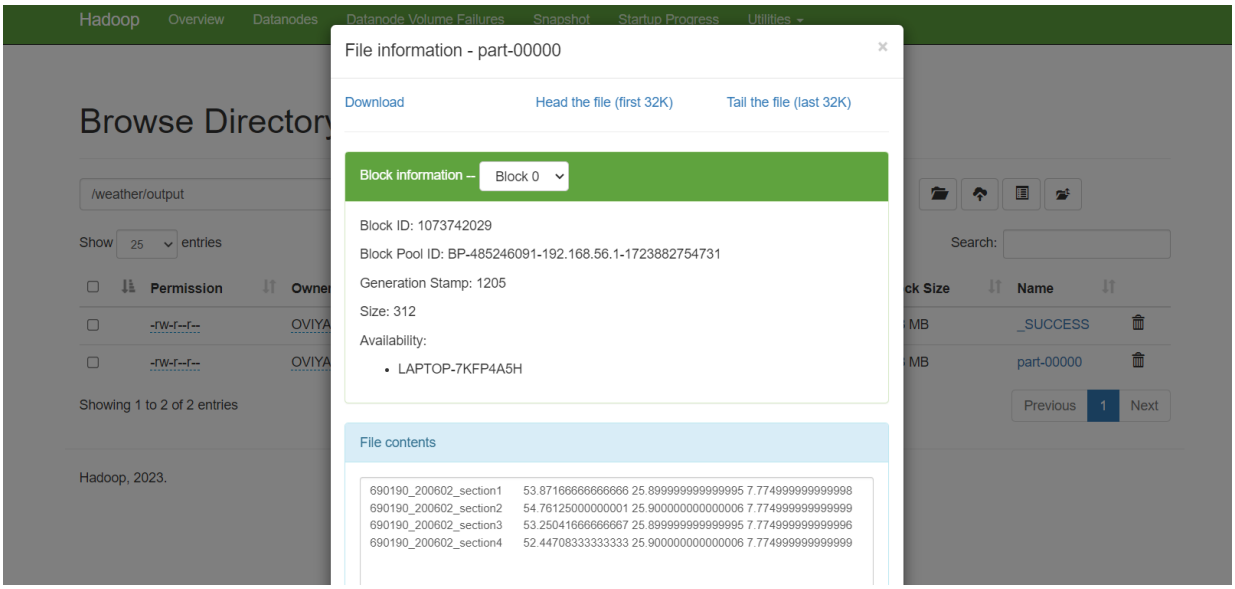
REDUCER.PY

```
#!/usr/bin/python3
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:





RESULT:

Thus the implementation of the python mapper and reducer programs using MapReduce to process weather dataset file using Hadoop is executed successfully.