**EXP3** 210701249

**Map Reduce program to process a weather dataset.**

**Aim:**

ToimplementMapReduceprogram toprocessa weatherdataset

**Procedure:**

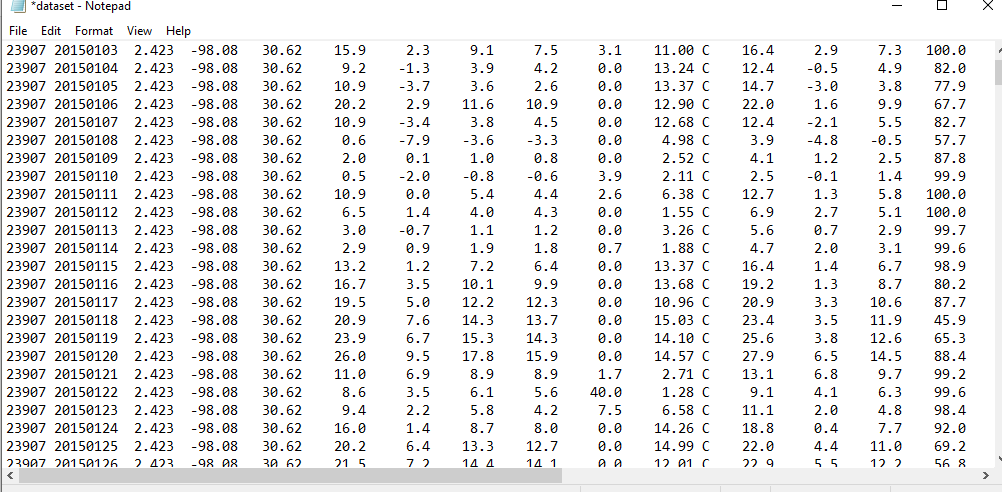
**Step1:CreateDataFile:**

Create a file named "word\_count\_data.txt" and populate it with text data that you wish toanalyse.

Loginwith yourhadoop user.

# Download thedataset(weatherdata)

Output:



# Step2:MapperLogic- mapper.py:

Create a file named "mapper.py" to implement the logic for the mapper. The mapperwillreadinputdatafrom STDIN,splitlinesintowords,andoutputeachwordwithitscount.

nanomapper.py

#Copyand pastethemapper.pycode

#!/usr/bin/env pythonimportsys

#input comesfrom STDIN(standard input)

# the mapper will get daily max temperature and group it by month. so output will be(month,dailymax\_temperature)

forlinein sys.stdin:

# remove leading and trailing whitespaceline=line.strip()

# split the line into wordswords= line.split()

#See the README hosted on the weather websitewhich help us understand how eachpositionrepresents a column

month = line[10:12]daily\_max = line[38:45]daily\_max = daily\_max.strip()#increasecounters

forwordinwords:

#writetheresultstoSTDOUT(standardoutput);

# what we output here will be go through the shuffle proess and then# bethe input fortheReducestep, i.e. the input forreducer.py

#

# tab-delimited; month and daily max temperature as outputprint('%s\t%s' %(month ,daily\_max))

.

# Step3:ReducerLogic-reducer.py:

Create a file named "reducer.py" to implement the logic for the reducer. The reducerwillaggregate theoccurrences of each word and generatethe final output.

nanoreducer.py

#Copyand pastethereducer.pycode

# reducer.py

#!/usr/bin/envpython

from operator import itemgetterimport sys

#reducer will get the input from stdid which will be a collection of key, value(Key=month ,value=daily max temperature)

#reducer logic: will get all the daily max temperature for a month and find max temperatureforthe month

#shuffle will ensure that key are sorted(month)current\_month= None

current\_max = 0month =None

# input comes from STDINforlinein sys.stdin:

# remove leading and trailing whitespaceline=line.strip()

# parse the input we got from mapper.pymonth,daily\_max =line.split('\t', 1)

# convert daily\_max (currently a string) to floattry:

daily\_max = float(daily\_max)exceptValueError:

# daily\_max was not a number, so silently#ignore/discard thisline

continue

# this IF-switch only works because Hadoop shuffle process sorts map output#by key (here: month)beforeit is passed tothereducer

ifcurrent\_month==month:

if daily\_max > current\_max:current\_max =daily\_max

else:

ifcurrent\_month:

#writeresulttoSTDOUT

print ('%s\t%s' % (current\_month, current\_max))current\_max=daily\_max

current\_month=month

# outputof thelast month

ifcurrent\_month==month:

print('%s\t%s'%(current\_month,current\_max))

# Step4:PrepareHadoopEnvironment:

StarttheHadoop daemonsand createadirectory inHDFS tostoreyourdata.

start-all.sh

# Step6:MakePythonFilesExecutable:

Giveexecutablepermissionstoyourmapper.py andreducer.pyfiles.

chmod777mapper.pyreducer.py

# Step7:RuntheprogramusingHadoopStreaming:

Downloadthelatest hadoop-streamingjarfileandplaceit inalocation youcaneasily

access.

Then run the program using Hadoop Streaming.hadoopfs -mkdir-p /weatherdata

hadoop fs -copyFromLocal /home/sx/Downloads/dataset.txt /weatherdatahdfsdfs -ls /weatherdata

hadoopjar/home/sx/hadoop-3.2.3/share/hadoop/tools/lib/hadoop-streaming-3.2.3.jar\

-input/weatherdata/dataset.txt\

-output/weatherdata/output \

-file"/home/sx/Downloads/mapper.py"\

-mapper"python3mapper.py"\

-file"/home/sx/Downloads/reducer.py"\

-reducer"python3reducer.py"

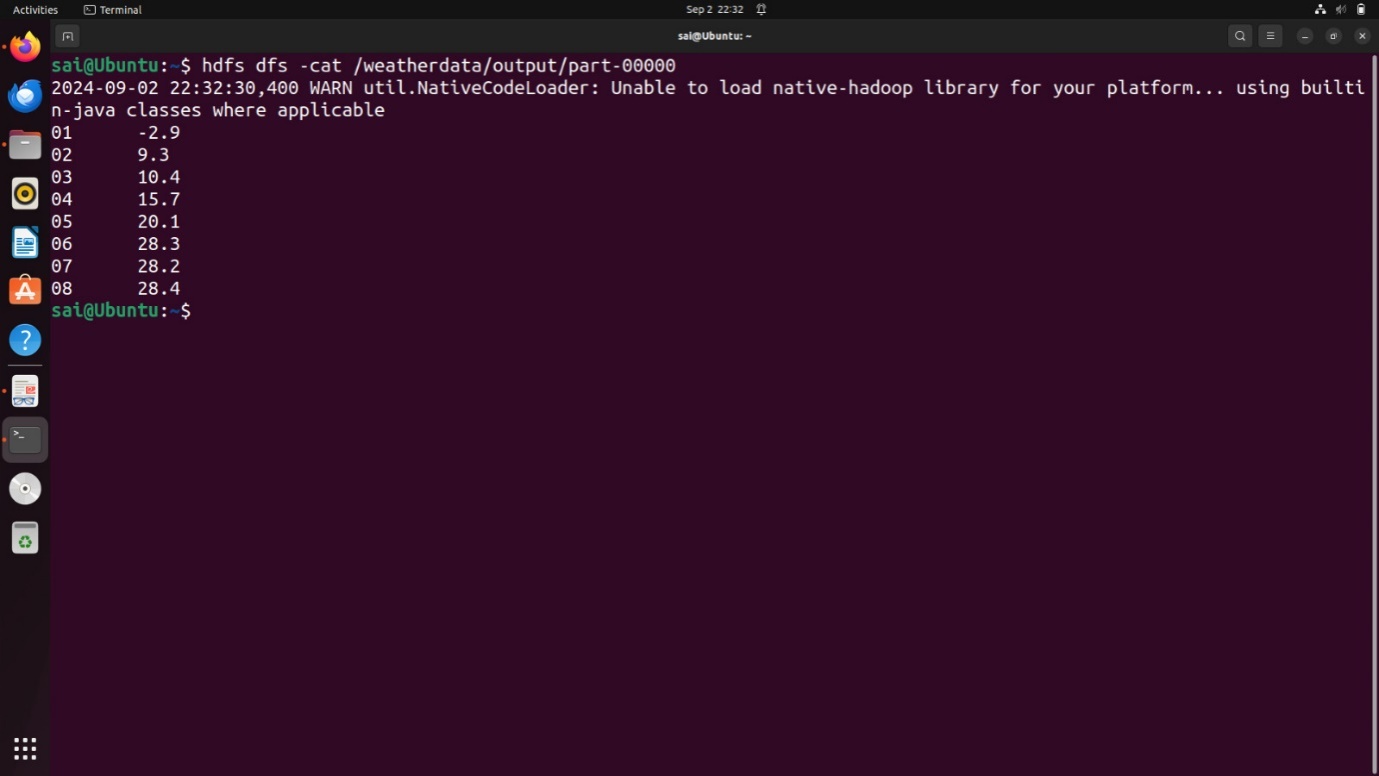
hdfsdfs-text/weatherdata/output/\*>/home/sx/Downloads/outputfile.txt

|  |
| --- |
|  |
| jbdkb |
| ouOutput: |

# Step8:CheckOutput:

ChecktheoutputoftheprograminthespecifiedHDFSoutput directory.

**OUTPUT:**



**Result:**

Thus,theprogramforweatherdatasetusing MapReducehasbeenexecutedsuccessfully.