/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*commands\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

hduser@divya-HP-Notebook:~$ hdfs dfs -mkdir /assignment3

18/04/18 15:17:44 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

hduser@divya-HP-Notebook:~$ hdfs dfs -put logdata.txt /assignment3

18/04/18 15:18:22 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

hduser@divya-HP-Notebook:~$ hadoop jar demo1.jar /assignment3/sample.txt /assignment3/output

18/04/18 15:18:58 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

18/04/18 15:18:59 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032

18/04/18 15:18:59 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032

18/04/18 15:18:59 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.

18/04/18 15:18:59 INFO mapreduce.JobSubmitter: Cleaning up the staging area /tmp/hadoop-yarn/staging/hduser/.staging/job\_1524037315409\_0002

org.apache.hadoop.mapred.InvalidInputException: Input path does not exist: hdfs://localhost:54310/assignment3/sample.txt

at org.apache.hadoop.mapred.FileInputFormat.singleThreadedListStatus(FileInputFormat.java:285)

at org.apache.hadoop.mapred.FileInputFormat.listStatus(FileInputFormat.java:228)

at org.apache.hadoop.mapred.FileInputFormat.getSplits(FileInputFormat.java:313)

at org.apache.hadoop.mapreduce.JobSubmitter.writeOldSplits(JobSubmitter.java:329)

at org.apache.hadoop.mapreduce.JobSubmitter.writeSplits(JobSubmitter.java:321)

at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:197)

at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1297)

at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1294)

at java.security.AccessController.doPrivileged(Native Method)

at javax.security.auth.Subject.doAs(Subject.java:422)

at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1692)

at org.apache.hadoop.mapreduce.Job.submit(Job.java:1294)

at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:562)

at org.apache.hadoop.mapred.JobClient$1.run(JobClient.java:557)

at java.security.AccessController.doPrivileged(Native Method)

at javax.security.auth.Subject.doAs(Subject.java:422)

at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1692)

at org.apache.hadoop.mapred.JobClient.submitJobInternal(JobClient.java:557)

at org.apache.hadoop.mapred.JobClient.submitJob(JobClient.java:548)

at org.apache.hadoop.mapred.JobClient.runJob(JobClient.java:841)

at MyDriver.main(MyDriver.java:37)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)

at java.lang.reflect.Method.invoke(Method.java:498)

at org.apache.hadoop.util.RunJar.run(RunJar.java:221)

at org.apache.hadoop.util.RunJar.main(RunJar.java:136)

hduser@divya-HP-Notebook:~$ hadoop jar demo1.jar /assignment3/logdata.txt /assignment3/output

18/04/18 15:19:30 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

18/04/18 15:19:31 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032

18/04/18 15:19:31 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032

18/04/18 15:19:31 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.

18/04/18 15:19:32 INFO mapred.FileInputFormat: Total input paths to process : 1

18/04/18 15:19:32 INFO mapreduce.JobSubmitter: number of splits:2

18/04/18 15:19:32 INFO mapreduce.JobSubmitter: Submitting tokens for job: job\_1524037315409\_0003

18/04/18 15:19:32 INFO impl.YarnClientImpl: Submitted application application\_1524037315409\_0003

18/04/18 15:19:32 INFO mapreduce.Job: The url to track the job: http://divya-HP-Notebook:8088/proxy/application\_1524037315409\_0003/

18/04/18 15:19:32 INFO mapreduce.Job: Running job: job\_1524037315409\_0003

18/04/18 15:19:37 INFO mapreduce.Job: Job job\_1524037315409\_0003 running in uber mode : false

18/04/18 15:19:37 INFO mapreduce.Job: map 0% reduce 0%

18/04/18 15:19:42 INFO mapreduce.Job: map 100% reduce 0%

18/04/18 15:19:47 INFO mapreduce.Job: map 100% reduce 100%

18/04/18 15:19:48 INFO mapreduce.Job: Job job\_1524037315409\_0003 completed successfully

18/04/18 15:19:49 INFO mapreduce.Job: Counters: 49

File System Counters

FILE: Number of bytes read=26793

FILE: Number of bytes written=375440

FILE: Number of read operations=0

FILE: Number of large read operations=0

FILE: Number of write operations=0

HDFS: Number of bytes read=147376

HDFS: Number of bytes written=3838

HDFS: Number of read operations=9

HDFS: Number of large read operations=0

HDFS: Number of write operations=2

Job Counters

Launched map tasks=2

Launched reduce tasks=1

Data-local map tasks=2

Total time spent by all maps in occupied slots (ms)=5914

Total time spent by all reduces in occupied slots (ms)=2338

Total time spent by all map tasks (ms)=5914

Total time spent by all reduce tasks (ms)=2338

Total vcore-milliseconds taken by all map tasks=5914

Total vcore-milliseconds taken by all reduce tasks=2338

Total megabyte-milliseconds taken by all map tasks=6055936

Total megabyte-milliseconds taken by all reduce tasks=2394112

Map-Reduce Framework

Map input records=1295

Map output records=1295

Map output bytes=24197

Map output materialized bytes=26799

Input split bytes=196

Combine input records=0

Combine output records=0

Reduce input groups=227

Reduce shuffle bytes=26799

Reduce input records=1295

Reduce output records=227

Spilled Records=2590

Shuffled Maps =2

Failed Shuffles=0

Merged Map outputs=2

GC time elapsed (ms)=207

CPU time spent (ms)=2040

Physical memory (bytes) snapshot=681951232

Virtual memory (bytes) snapshot=5891522560

Total committed heap usage (bytes)=529006592

Shuffle Errors

BAD\_ID=0

CONNECTION=0

IO\_ERROR=0

WRONG\_LENGTH=0

WRONG\_MAP=0

WRONG\_REDUCE=0

File Input Format Counters

Bytes Read=147180

File Output Format Counters

Bytes Written=3838

hduser@divya-HP-Notebook:~$ hadoop dfs -get /assignment3/output

DEPRECATED: Use of this script to execute hdfs command is deprecated.

Instead use the hdfs command for it.

18/04/18 15:20:14 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Mapper\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.io.IOException;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.\*;

public class MyMapper extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {

private final static IntWritable one = new IntWritable(1);

public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {

String valueString = value.toString();

String[] SingleCountryData = valueString.split("-");

output.collect(new Text(SingleCountryData[0]), one);

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Reducer\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

package SalesCountry;

import java.io.IOException;

import java.util.\*;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.\*;

public class MyReducer extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {

public void reduce(Text t\_key, Iterator<IntWritable> values, OutputCollector<Text,IntWritable> output, Reporter reporter) throws IOException {

Text key = t\_key;

int frequencyForUser = 0;

while (values.hasNext()) {

// replace type of value with the actual type of our value

IntWritable value = (IntWritable) values.next();

frequencyForUser += value.get();

}

output.collect(key, new IntWritable(frequencyForUser));

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Driver\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.\*;

import org.apache.hadoop.mapred.\*;

public class MyDriver{

public static void main(String[] args) {

JobClient my\_client = new JobClient();

// Create a configuration object for the job

JobConf job\_conf = new JobConf(MyDriver.class);

// Set a name of the Job-optional

job\_conf.setJobName("LOG ANALYSIS WITH MAP REDUCE");

// Specify data type of output key and value

job\_conf.setOutputKeyClass(Text.class);

job\_conf.setOutputValueClass(IntWritable.class);

// Specify names of Mapper and Reducer Class

job\_conf.setMapperClass(MyMapper.class);

job\_conf.setReducerClass(MyReducer.class);

// Specify formats of the data type of Input and output

job\_conf.setInputFormat(TextInputFormat.class);

job\_conf.setOutputFormat(TextOutputFormat.class);

// Set input and output directories using command line arguments,

//arg[0] = name of input directory on HDFS, and arg[1] = name of output directory to be created to store the output file.

FileInputFormat.setInputPaths(job\_conf, new Path(args[0]));

FileOutputFormat.setOutputPath(job\_conf, new Path(args[1]));

my\_client.setConf(job\_conf);

try {

// Run the job

JobClient.runJob(job\_conf);

} catch (Exception e) {

e.printStackTrace();

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Output\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

10.1.1.236 7

10.1.181.142 14

10.1.232.31 5

10.10.55.142 14

10.102.101.66 1

10.103.184.104 1

10.103.190.81 53

10.103.63.29 1

10.104.73.51 1

10.105.160.183 1

10.108.91.151 1

10.109.21.76 1

10.11.131.40 1

10.111.71.20 8

10.112.227.184 6

10.114.74.30 1

10.115.118.78 1

10.117.224.230 1

10.117.76.22 12

10.118.19.97 1

10.118.250.30 7

10.119.117.132 23

10.119.33.245 1

10.119.74.120 1

10.12.113.198 2

10.12.219.30 1

10.120.165.113 1

10.120.207.127 4

10.123.124.47 1

10.123.35.235 1

10.124.148.99 1

10.124.155.234 1

10.126.161.13 7

10.127.162.239 1

10.128.11.75 10

10.13.42.232 1

10.130.195.163 8

10.130.70.80 1

10.131.163.73 1

10.131.209.116 5

10.132.19.125 2

10.133.222.184 12

10.134.110.196 13

10.134.242.87 1

10.136.84.60 5

10.14.2.86 8

10.14.4.151 2

10.140.139.116 1

10.140.141.1 9

10.140.67.116 1

10.141.221.57 5

10.142.203.173 7

10.143.126.177 32

10.144.147.8 1

10.15.208.56 1

10.15.23.44 13

10.150.212.239 14

10.150.227.16 1

10.150.24.40 13

10.152.195.138 8

10.153.23.63 2

10.153.239.5 25

10.155.95.124 9

10.156.152.9 1

10.157.176.158 1

10.164.130.155 1

10.164.49.105 8

10.164.95.122 10

10.165.106.173 14

10.167.1.145 19

10.169.158.88 1

10.170.178.53 1

10.171.104.4 1

10.172.169.53 18

10.174.246.84 3

10.175.149.65 1

10.175.204.125 15

10.177.216.164 6

10.179.107.170 2

10.181.38.207 13

10.181.87.221 1

10.185.152.140 1

10.186.56.126 16

10.186.56.183 1

10.187.129.140 6

10.187.177.220 1

10.187.212.83 1

10.187.28.68 1

10.19.226.186 2

10.190.174.142 10

10.190.41.42 5

10.191.172.11 1

10.193.116.91 1

10.194.174.4 7

10.198.138.192 1

10.199.103.248 2

10.199.189.15 1

10.2.202.135 1

10.200.184.212 1

10.200.237.222 1

10.200.9.128 2

10.203.194.139 10

10.205.72.238 2

10.206.108.96 2

10.206.175.236 1

10.206.73.206 7

10.207.190.45 17

10.208.38.46 1

10.208.49.216 4

10.209.18.39 9

10.209.54.187 3

10.211.47.159 10

10.212.122.173 1

10.213.181.38 7

10.214.35.48 1

10.215.222.114 1

10.216.113.172 48

10.216.134.214 1

10.216.227.195 16

10.217.151.145 10

10.217.32.16 1

10.218.16.176 8

10.22.108.103 4

10.220.112.1 34

10.221.40.89 5

10.221.62.23 13

10.222.246.34 1

10.223.157.186 11

10.225.137.152 1

10.225.234.46 1

10.226.130.133 1

10.229.60.23 1

10.230.191.135 6

10.231.55.231 1

10.234.15.156 1

10.236.231.63 1

10.238.230.235 1

10.239.100.52 1

10.239.52.68 4

10.24.150.4 5

10.24.67.131 13

10.240.144.183 15

10.240.170.50 1

10.241.107.75 1

10.241.9.187 1

10.243.51.109 5

10.244.166.195 5

10.245.208.15 20

10.246.151.162 3

10.247.111.104 9

10.247.175.65 1

10.247.229.13 1

10.248.24.219 1

10.248.36.117 3

10.249.130.132 3

10.25.132.238 2

10.25.44.247 6

10.250.166.232 1

10.27.134.23 1

10.30.164.32 1

10.30.47.170 8

10.31.225.14 7

10.32.138.48 11

10.32.247.175 4

10.32.55.216 12

10.33.181.9 8

10.34.233.107 1

10.36.200.176 1

10.39.45.70 2

10.39.94.109 4

10.4.59.153 1

10.4.79.47 15

10.41.170.233 9

10.41.40.17 1

10.42.208.60 1

10.43.81.13 1

10.46.190.95 10

10.48.81.158 5

10.5.132.217 1

10.5.148.29 1

10.50.226.223 9

10.50.41.216 3

10.52.161.126 1

10.53.58.58 1

10.54.242.54 10

10.54.49.229 1

10.56.48.40 16

10.59.42.194 11

10.6.238.124 6

10.61.147.24 1

10.61.161.218 1

10.61.23.77 8

10.61.232.147 3

10.62.78.165 2

10.63.233.249 7

10.64.224.191 13

10.66.208.82 2

10.69.20.85 26

10.70.105.238 1

10.70.238.46 6

10.72.137.86 6

10.72.208.27 1

10.73.134.9 4

10.73.238.200 1

10.73.60.200 1

10.73.64.91 1

10.74.218.123 1

10.75.116.199 1

10.76.143.30 1

10.76.68.178 16

10.78.95.24 8

10.80.10.131 10

10.80.215.116 17

10.81.134.180 1

10.82.30.199 63

10.82.64.235 1

10.84.236.242 1

10.87.209.46 1

10.87.88.214 1

10.88.204.177 1

10.89.178.62 1

10.89.244.42 1

10.94.196.42 1

10.95.136.211 4

10.95.232.88 1

10.98.156.141 1

10.99.228.224 1