LABORATORY PROGRAM - 5 WordCount Program in HADOOP

Driver Code

```
// Importing libraries
import java.io.IOException;
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.FileInputFormat;
import org.apache.hadoop.mapred.FileOutputFormat;
import org.apache.hadoop.mapred.JobClient;
import org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;
public class WCDriver extends Configured implements Tool {
public int run(String[] args) throws IOException {
if (args.length < 2) {
System.out.println("Please give valid inputs");
return -1;
}
JobConf conf = new JobConf(WCDriver.class);
conf.setJobName("WordCount");
FileInputFormat.setInputPaths(conf, new Path(args[0]));
FileOutputFormat.setOutputPath(conf, new Path(args[1]));
conf.setMapperClass(WCMapper.class);
conf.setReducerClass(WCReducer.class);
conf.setMapOutputKeyClass(Text.class);
conf.setMapOutputValueClass(IntWritable.class);
conf.setOutputKeyClass(Text.class);
conf.setOutputValueClass(IntWritable.class);
```

```
JobClient.runJob(conf);
return 0;
}
// Main Method
public static void main(String[] args) throws Exception {
int exitCode = ToolRunner.run(new WCDriver(), args);
System.out.println("Job Exit Code: " + exitCode);
}
}
                                           Mapper Code
// Importing libraries
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.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
14 | Page
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;
public class WCMapper extends MapReduceBase implements Mapper<LongWritable, Text, Text,
IntWritable> {
// Map function
public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter
reporter)
throws IOException {
String line = value.toString();
// Splitting the line on whitespace
for (String word : line.split("\\s+")) {
if (word.length() > 0) {
```

```
output.collect(new Text(word), new IntWritable(1));
}
}
}
                                           Reducer Code
// Importing libraries
import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;
public class WCReducer extends MapReduceBase implements Reducer<Text, IntWritable, Text,
IntWritable> {
// Reduce function
public void reduce(Text key, Iterator<IntWritable> values,
OutputCollector<Text, IntWritable> output,
Reporter reporter) throws IOException {
int count = 0;
// Counting the frequency of each word
while (values.hasNext()) {
count += values.next().get();
}
output.collect(key, new IntWritable(count));
}
```

OBSERVATION

```
hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-$ cd ./Desktop/
 hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~/Desktop$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [bmscecse-HP-Elite-Tower-800-G9-Desktop-PC]
Starting resourcemanager
Starting nodemanagers
hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-/Desktop$ hdfs dfs -mkdir /Lab06
hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-/Desktop$ hadoop fs -ls /Lab06
 hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-/Desktop$ jps
 7360 DataNode
 7928 ResourceManager
 8681 Jps
 7178 NameNode
 8091 NodeManager
 7644 SecondaryNameNode
 hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~/Desktop$ cd ...
 hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-$ cd ./Desktop/
 hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~/Desktop$ nano file1.txt
hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-/Desktop$ hadoop fs -copyFromLocal -f /home/hadoop/Desktop/file1.txt /gs/test.txt hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-/Desktop$ hadoop jar /home/hadoop/Desktop/WordCount.jar wordcount.WordCount /rgs/test.txt /output JAR does not exist or is not a normal file: /home/hadoop/Desktop/WordCount.jar wordcount.WordCount /rgs/test.txt /output Exception in thread "main" java.lang.ClassNotFoundException: wordcount.WordCount.WordCount at java.base/java.net.URLClassLoader.findClass(URLClassLoader.java:476) at java.base/java.lang.ClassLoader.loadClass(ClassLoader.java:594) at java.base/java.lang.ClassLoader.loadClass(ClassLoader.java:597) at java.base/java.lang.Class.forNameO(Native Method) at java.base/java.lang.Class.forNameO(Native Method) at java.base/java.lang.Class.forName(Native Method) at java.base/java.lang.Class.forName(Native Method) at java.base/java.lang.Class.forName(Native Method) hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:-/Desktos S hadoop fs -cat /output/part-00000
                                                                  p$ hadoop fs -cat /output/part-00000
are
brother
family
hi
job
sister
 Addopgbmscecse-m
Found 2 items
-rw-r--r-- 1 hadoop supergroup
            scecse-HP-Elite-Tower-800-G9-Desktop-PC:~/Desktop$ hadoop fs -ls /output
                                               0 2024-05-21 15:21 /output/_SUCCESS
69 2024-05-21 15:21 /output/part-00000
```