Big Data

“MapReduce II”

Oleh :

Rajendra Rakha Arya

1941720080

TI 3H

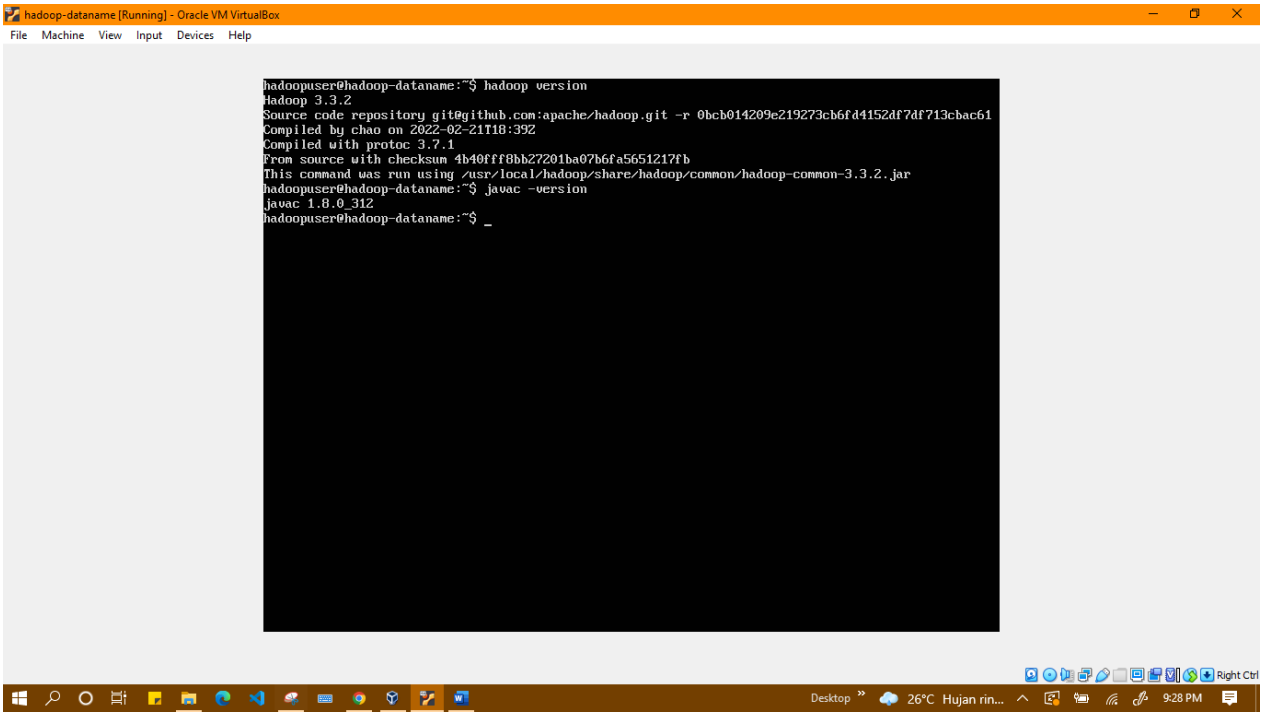
JURUSAN TEKNOLOGI INFORMASI

PROGRAM STUDI TEKNIK INFORMATIKA

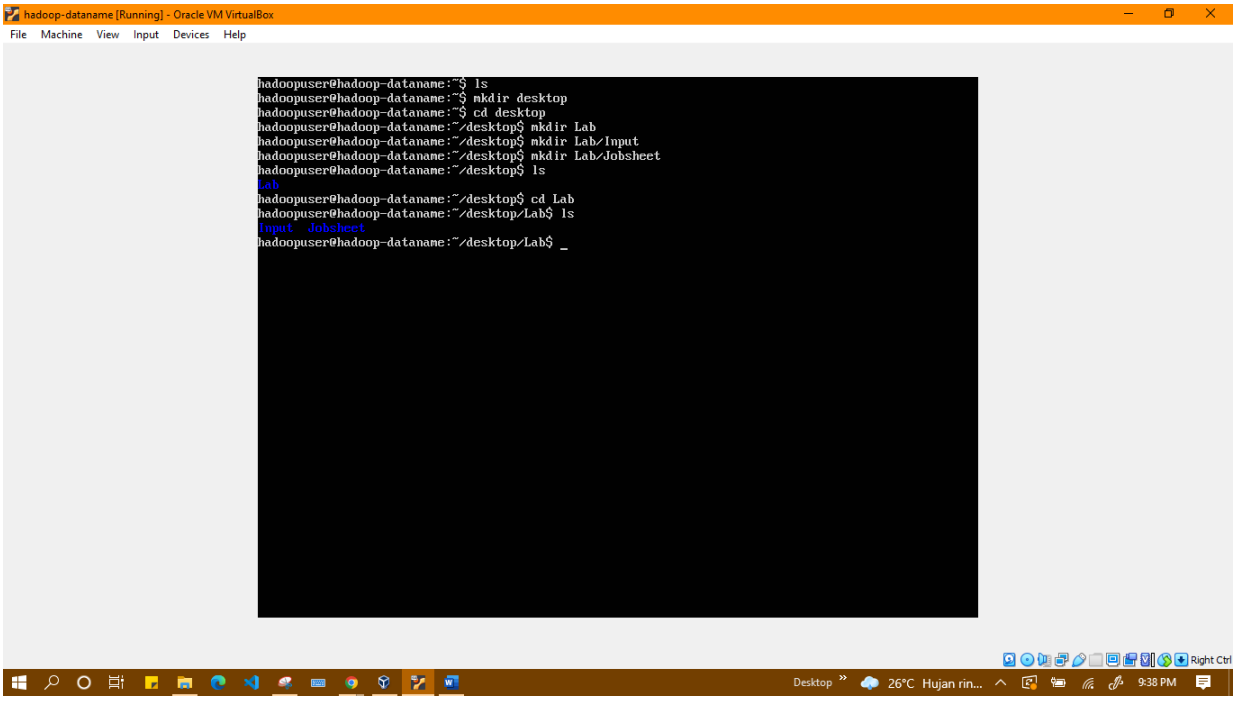
POLITEKNIK NEGERI MALANG

2022

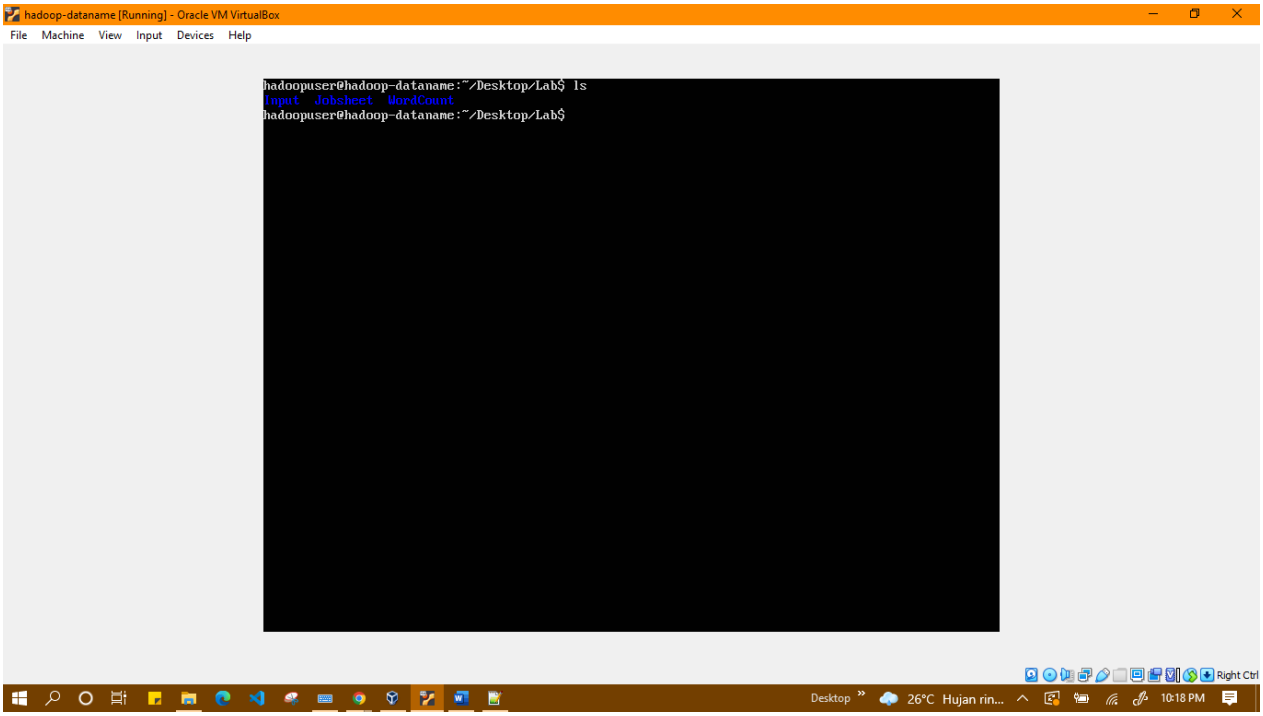
1. Pastikan Hadoop dan Java diinstal dengan benar

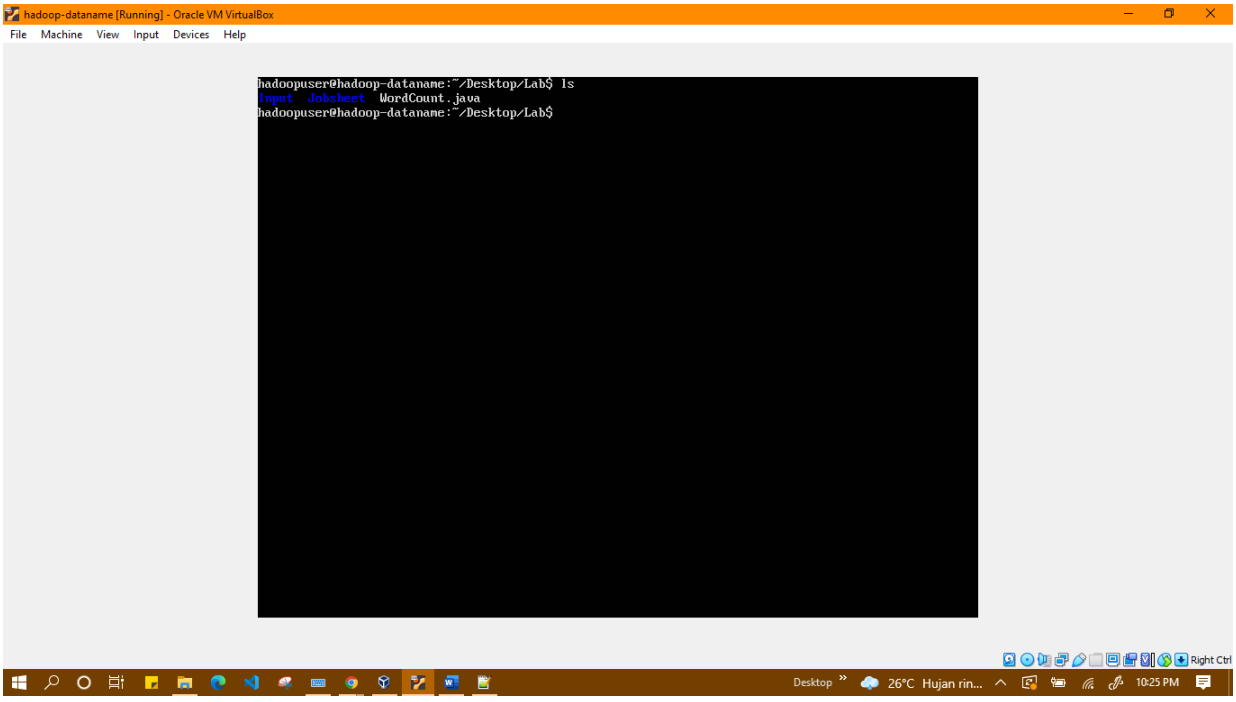


2. Buat direktori di Desktop bernama Lab dan di dalamnya buat dua folder; satu disebut "Input" dan yang lainnya disebut “Jobsheet”.



3. Buat dan Tambahkan file .java "WordCount.java" di lab direktori





hadoop-dataname [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
 GNU nano 2.9.3                          WordCount.java

import java.io.IOException;
import java.util.regex.Pattern;
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;

import org.apache.log4j.Logger;

public class WordCount extends Configured implements Tool {

  private static final Logger LOG = Logger.getLogger(WordCount.class);

  public static void main(String[] args) throws Exception {
    int res = ToolRunner.run(new WordCount(), args);
    System.exit(res);
  }

  public int run(String[] args) throws Exception {
    Job job = Job.getInstance(getConf(), "wordcount");
    job.setJarByClass(this.getClass());
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    job.setMapperClass(Map.class);
                                [ Read 70 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
```

Right Ctrl

Desktop 26°C Hujan rin... 10:28 PM

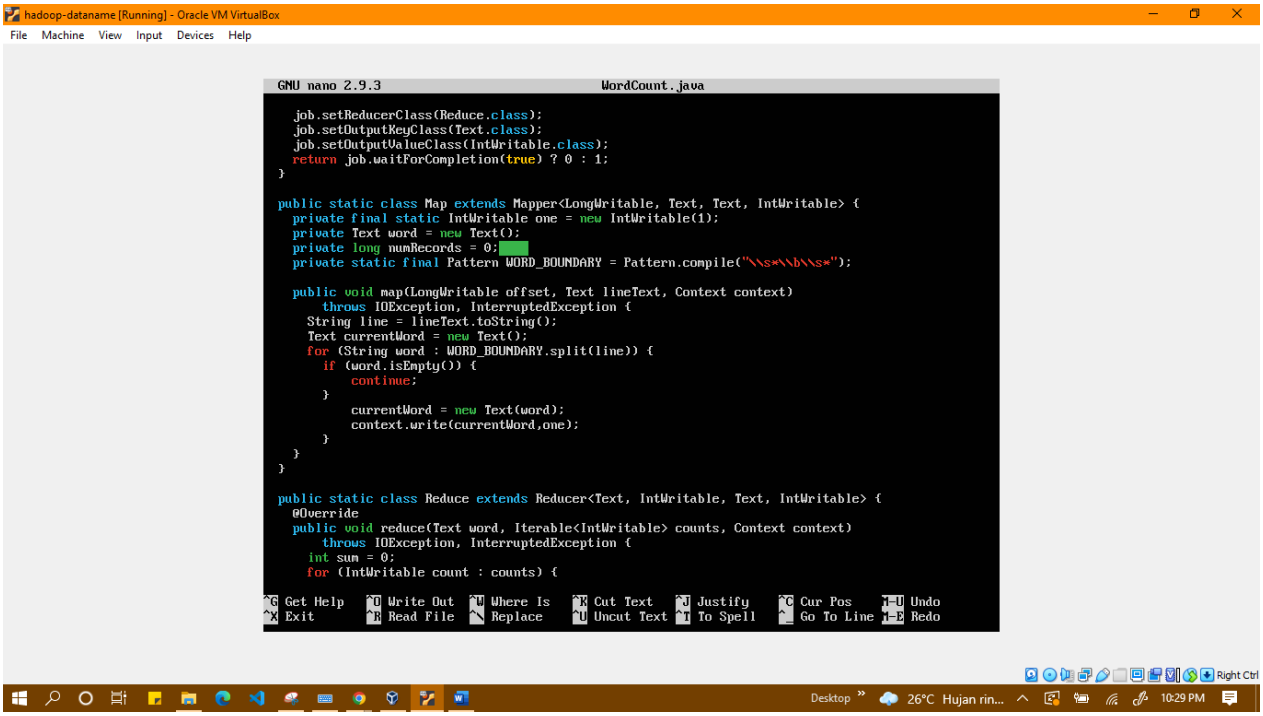
hadoop-dataname [Running] - Oracle VM VirtualBox

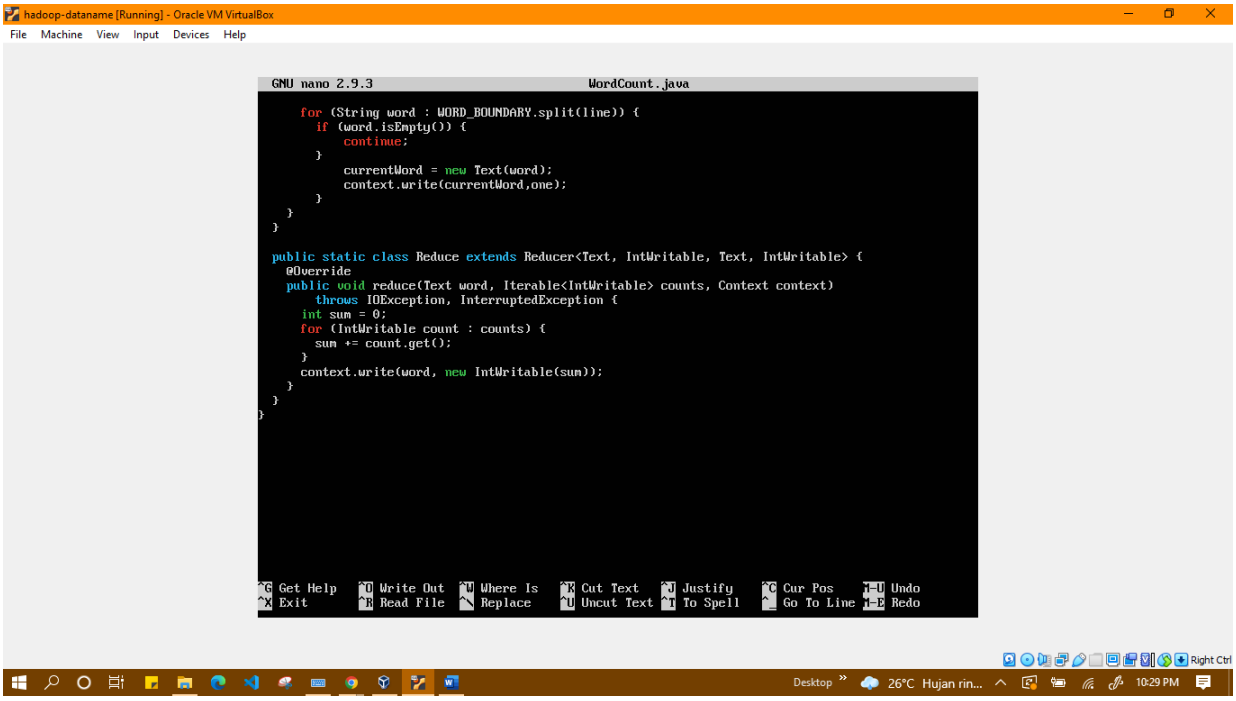
File Machine View Input Devices Help

```
 GNU nano 2.9.3                          WordCount.java


public class WordCount extends Configured implements Tool {

  private static final Logger LOG = Logger.getLogger(WordCount.class);

  public static void main(String[] args) throws Exception {
    int res = ToolRunner.run(new WordCount(), args);
    System.exit(res);
  }

  public int run(String[] args) throws Exception {
    Job job = Job.getInstance(getConf(), "wordcount");
    job.setJarByClass(this.getClass());
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    job.setMapperClass(Map.class);
    job.setReducerClass(Reduce.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    return job.waitForCompletion(true) ? 0 : 1;
  }

  public static class Map extends Mapper<LongWritable, Text, Text, IntWritable> {
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();
    private long numRecords = 0;
    private static final Pattern WORD_BOUNDARY = Pattern.compile("\\s*\\b\\s*");

    public void map(LongWritable offset, Text lineText, Context context)
        throws IOException, InterruptedException {
      String line = lineText.toString();
      Text currentWord = new Text();

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
```

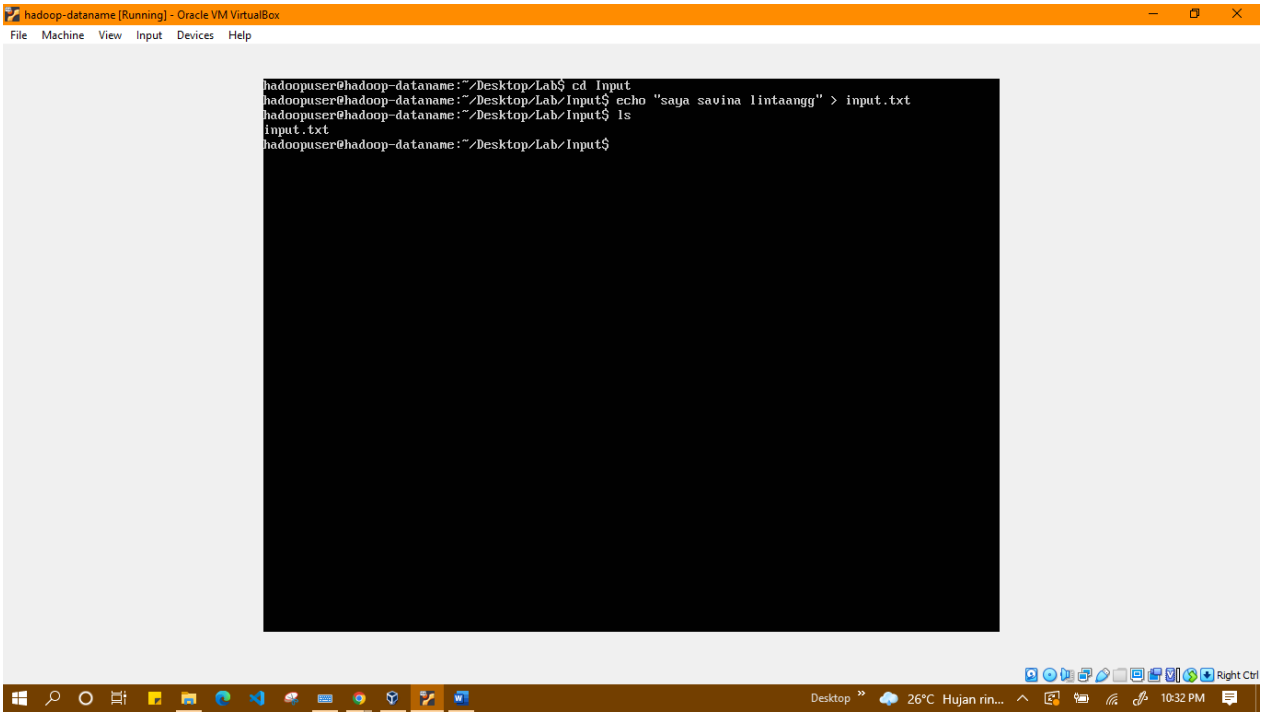
Right Ctrl

Desktop 26°C Hujan rin... 10:28 PM

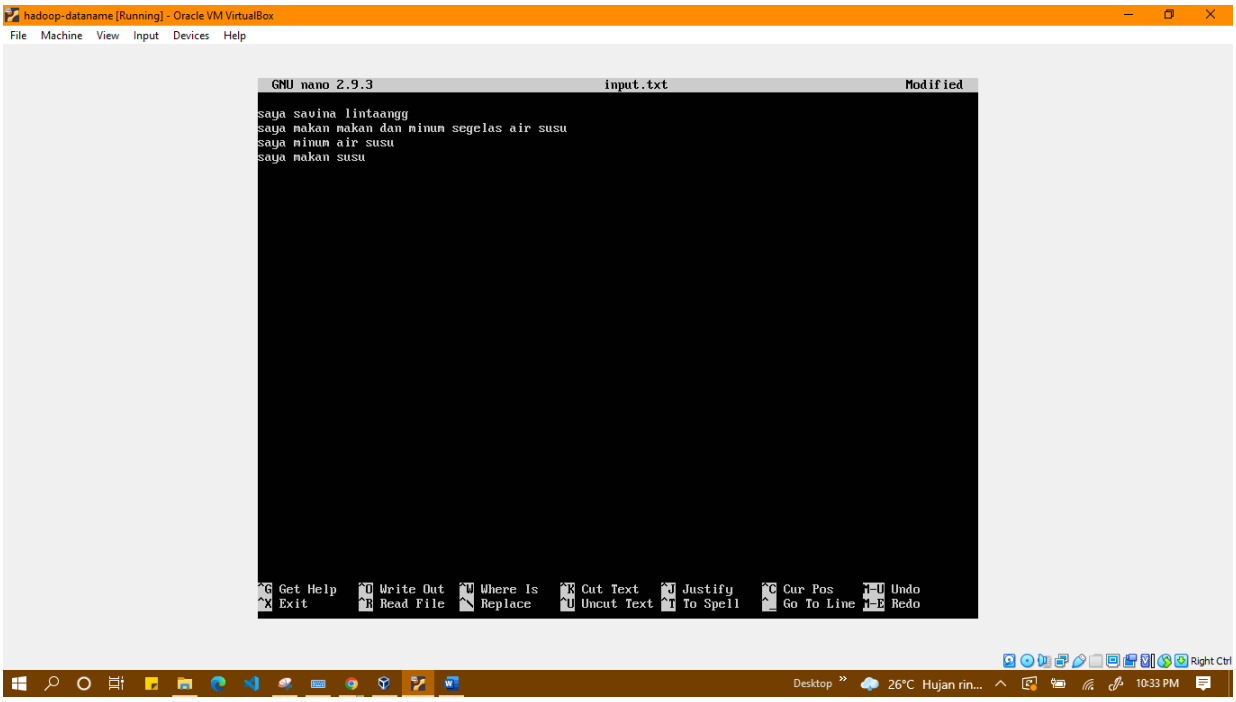

hadoop-dataname [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 WordCount.java
job.setReducerClass(Reduce.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);
return job.waitForCompletion(true) ? 0 : 1;
}
public static class Map extends Mapper<LongWritable, Text, Text, IntWritable> {
private final static IntWritable one = new IntWritable(1);
private Text word = new Text();
private long numRecords = 0;
private static final Pattern WORD_BOUNDARY = Pattern.compile("\\s*\\b\\s*");
public void map(LongWritable offset, Text lineText, Context context)
throws IOException, InterruptedException {
String line = lineText.toString();
Text currentWord = new Text();
for (String word : WORD_BOUNDARY.split(line)) {
if (word.isEmpty()) {
continue;
}
currentWord = new Text(word);
context.write(currentWord,one);
}
}
}
public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {
@Override
public void reduce(Text word, Iterable<IntWritable> counts, Context context)
throws IOException, InterruptedException {
int sum = 0;
for (IntWritable count : counts) {
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
Right Ctrl
Desktop 26°C Hujan rin... 10:29 PM


hadoop-dataname [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 WordCount.java
for (String word : WORD_BOUNDARY.split(line)) {
if (word.isEmpty()) {
continue;
}
currentWord = new Text(word);
context.write(currentWord,one);
}
}
}
public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {
@Override
public void reduce(Text word, Iterable<IntWritable> counts, Context context)
throws IOException, InterruptedException {
int sum = 0;
for (IntWritable count : counts) {
sum += count.get();
}
context.write(word, new IntWritable(sum));
}
}
}
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
Right Ctrl
Desktop 26°C Hujan rin... 10:29 PM

4. Buat & Tambahkan file .txt dengan penamaan "input.txt" di direktori Lab/Input.



Tambahkan kata-kata lain pada file input



5. Ketik command berikut untuk mengekspor classpath hadoop ke bash.

Pastikan telah berhasil diekspor.

```
hadoopuser@hadoop-dataname:~/Desktop/Lab/Input$ export HADOOP_CLASSPATH=$(hadoop classpath)
hadoopuser@hadoop-dataname:~/Desktop/Lab/Input$ echo $HADDOOP_CLASSPATH

hadoopuser@hadoop-dataname:~/Desktop/Lab/Input$ echo $HADOOP_CLASSPATH
/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/*:/usr/local/hadoop/share/had
oop/common/*:/usr/local/hadoop/share/hadoop/hdfs:/usr/local/hadoop/share/hadoop/hdfs/lib/*:/usr/loca
l/hadoop/share/hadoop/hdfs/*:/usr/local/hadoop/share/hadoop/mapreduce/*:/usr/local/hadoop/share/hado
op/yarn:/usr/local/hadoop/share/hadoop/yarn/lib/*:/usr/local/hadoop/share/hadoop/yarn/*
```

6. Selanjutnya adalah mebuat direktori baru /WordCount/Input di HDFS dan memindahkan input.txt ke HDFS dengan mengetikan perintah berikut.

```
hadoopuser@hadoop-dataname:~/Desktop/Lab/Input$ hadoop fs -mkdir /WordCount
hadoopuser@hadoop-dataname:~/Desktop/Lab/Input$ hadoop fs -mkdir /WordCount/Input
hadoopuser@hadoop-dataname:~/Desktop/Lab/Input$
```

```
hadoopuser@hadoop-dataname:~/Desktop$ hadoop fs -put Lab/Input?input.txt /WordCount/Input
hadoopuser@hadoop-dataname:~/Desktop$ hadoop fs -ls /WordCount/Input
Found 1 items
drwxr-xr-x   - hadoopuser supergroup          0 2022-04-20 00:29 /WordCount/Input/Input
hadoopuser@hadoop-dataname:~/Desktop$ _
```

7. Periksa dan lihat bahwa direktori dan file telah ditempatkan di File System yang tepat.

```
hadoopuser@hadoop-dataname:~/Desktop$ hadoop fs -ls /
Found 1 items
drwxr-xr-x   - hadoopuser supergroup          0 2022-04-19 22:52 /WordCount
hadoopuser@hadoop-dataname:~/Desktop$ hadoop fs -ls /WordCount/Input
Found 1 items
drwxr-xr-x   - hadoopuser supergroup          0 2022-04-20 00:29 /WordCount/Input/Input
hadoopuser@hadoop-dataname:~/Desktop$ _
```

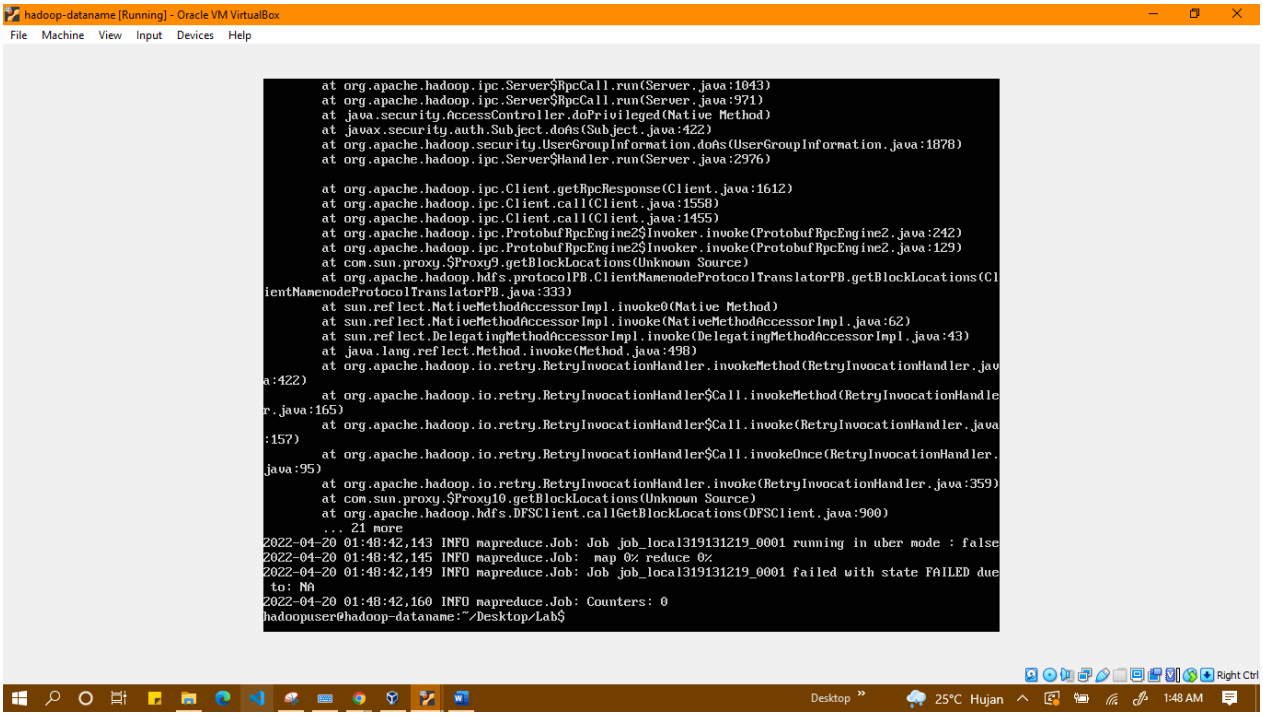
8. Kemudian, kembali ke mesin lokal di mana kita akan mengkompilasi file WordCount.java. Dengan asumsi saat ini berada di direktori Desktop.

Masukkan file output dalam satu file jar

```
hadoopuser@hadoop-dataname:~/Desktop/Lab$ javac -classpath $(hadoop classpath) -d Jobsheet WordCount
.java
hadoopuser@hadoop-dataname:~/Desktop/Lab$ jar -cvf WordCount.jar -C Jobsheet/ .
added manifest
adding: WordCount$Reduce.class(in = 1627) (out= 686)(deflated 57%)
adding: WordCount.class(in = 1946) (out= 981)(deflated 49%)
adding: WordCount$Map.class(in = 2189) (out= 981)(deflated 55%)
hadoopuser@hadoop-dataname:~/Desktop/Lab$ _
```

```
hadoopuser@hadoop-dataname:~/Desktop/Lab$ jar -cvf WordCount.jar -C Jobsheet/ .
added manifest
hadoopuser@hadoop-dataname:~/Desktop/Lab$ ls
Input  Jobsheet  WordCount.jar  WordCount.java
```

9. Sekarang, jalankan file jar di Hadoop



10. Jika Anda ingin menjalankan sampel lagi, Anda harus terlebih dahulu menghapus direktori output. Gunakan perintah berikut.