

Assignment 11

Make sure you have set up Hadoop on your Linux machine.

You can check if Hadoop is installed properly by using the following command in the terminal:

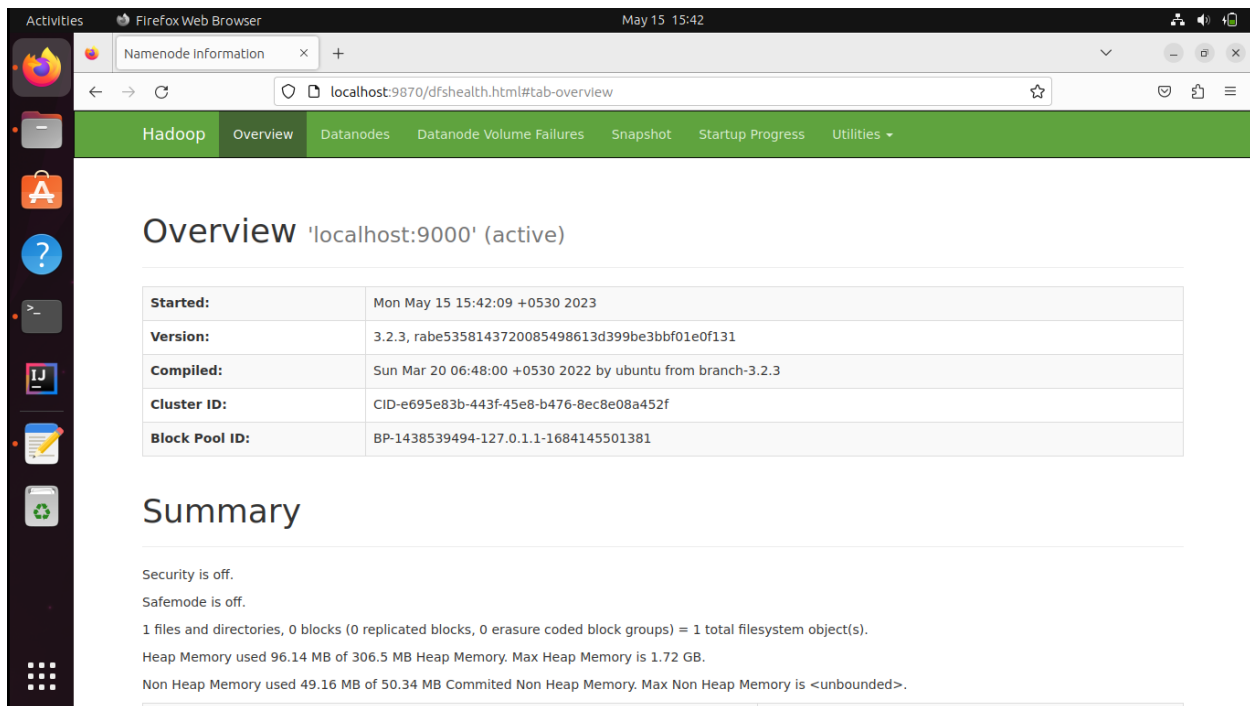
```
hadoop version
```

Start the Hadoop server using the following command:

```
start-all.sh
```

After the server boots up you can open any browser and enter the following URL to view the Hadoop user interface.

URL: localhost:/9870



The screenshot shows a Firefox web browser window displaying the Hadoop Overview page. The browser's address bar shows the URL `localhost:9870/dfshealth.html#tab-overview`. The page has a green header with navigation tabs: Hadoop, Overview, Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, and Utilities. The main content area is titled "Overview 'localhost:9000' (active)". Below the title is a table with the following information:

Started:	Mon May 15 15:42:09 +0530 2023
Version:	3.2.3, rabe5358143720085498613d399be3bbf01e0f131
Compiled:	Sun Mar 20 06:48:00 +0530 2022 by ubuntu from branch-3.2.3
Cluster ID:	CID-e695e83b-443f-45e8-b476-8ec8e08a452f
Block Pool ID:	BP-1438539494-127.0.1.1-1684145501381

Below the table is a "Summary" section. It contains the following text:

Security is off.
Safemode is off.
1 files and directories, 0 blocks (0 replicated blocks, 0 erasure coded block groups) = 1 total filesystem object(s).
Heap Memory used 96.14 MB of 306.5 MB Heap Memory. Max Heap Memory is 1.72 GB.
Non Heap Memory used 49.16 MB of 50.34 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Open Ubuntu Software Store and install IntelliJ Idea Community Edition

Open IntelliJ Idea Community Edition and click on 'Create New Project.'

Enter the name of the project as 'WordCountExample'

Select 'Maven' as the build system while creating new project.

Click on 'Advanced Settings' and keep note of your 'GroupId' which will be required later.

Click on Create then the maven project will be created.

On the left hand side of the file explorer, Delete Main.java which is present inside

src/java/org.example/Main.java

Open pom.xml

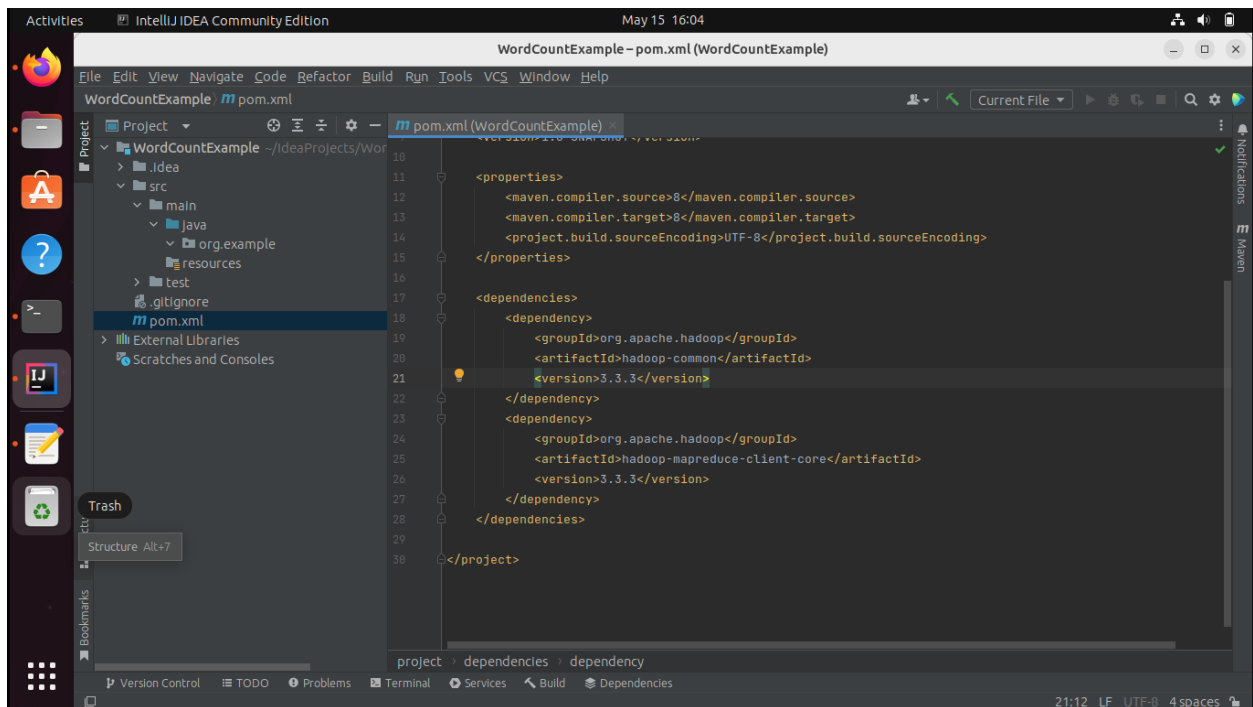
Add the following dependencies:

```
<dependencies>

    <dependency>
        <groupId>org.apache.hadoop</groupId>
        <artifactId>hadoop-common</artifactId>
        <version>3.3.3</version>
    </dependency>

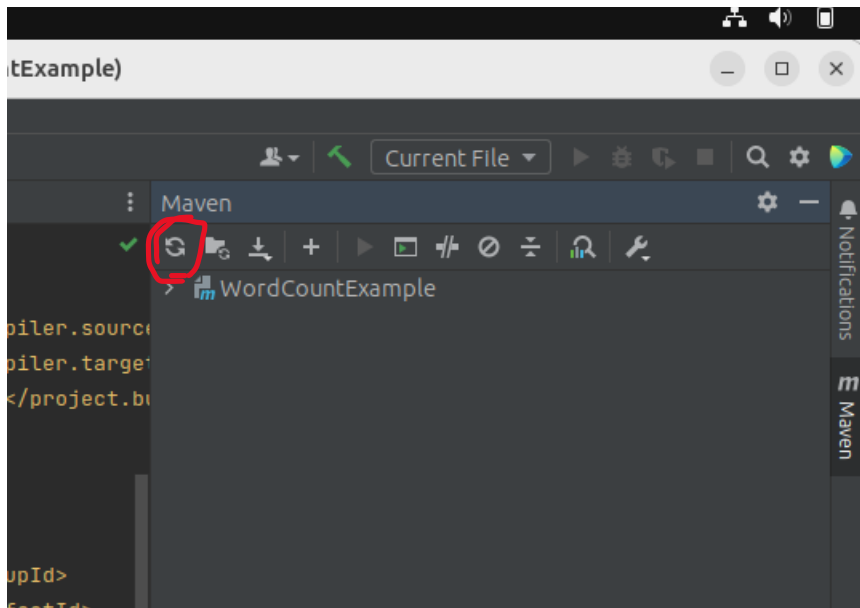
    <dependency>
        <groupId>org.apache.hadoop</groupId>
        <artifactId>hadoop-mapreduce-client-core</artifactId>
        <version>3.3.3</version>
    </dependency>

</dependencies>
```



Click on Maven tab on the right hand side of the IDE.

Click on 'reload' icon to download all the dependencies.



Right click on your package name and select New > Java Class

Enter the class name as 'WC_Mapper' and hit enter.

Add the code given below to the WC_Mapper class

```
import java.io.IOException;
import java.util.StringTokenizer;
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;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;

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

    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();
```

```

public void map(LongWritable key, Text value,OutputCollector<Text,IntWritable> output,

        Reporter reporter) throws IOException{

String line = value.toString();

StringTokenizer tokenizer = new StringTokenizer(line);

while (tokenizer.hasMoreTokens()){

    word.set(tokenizer.nextToken());

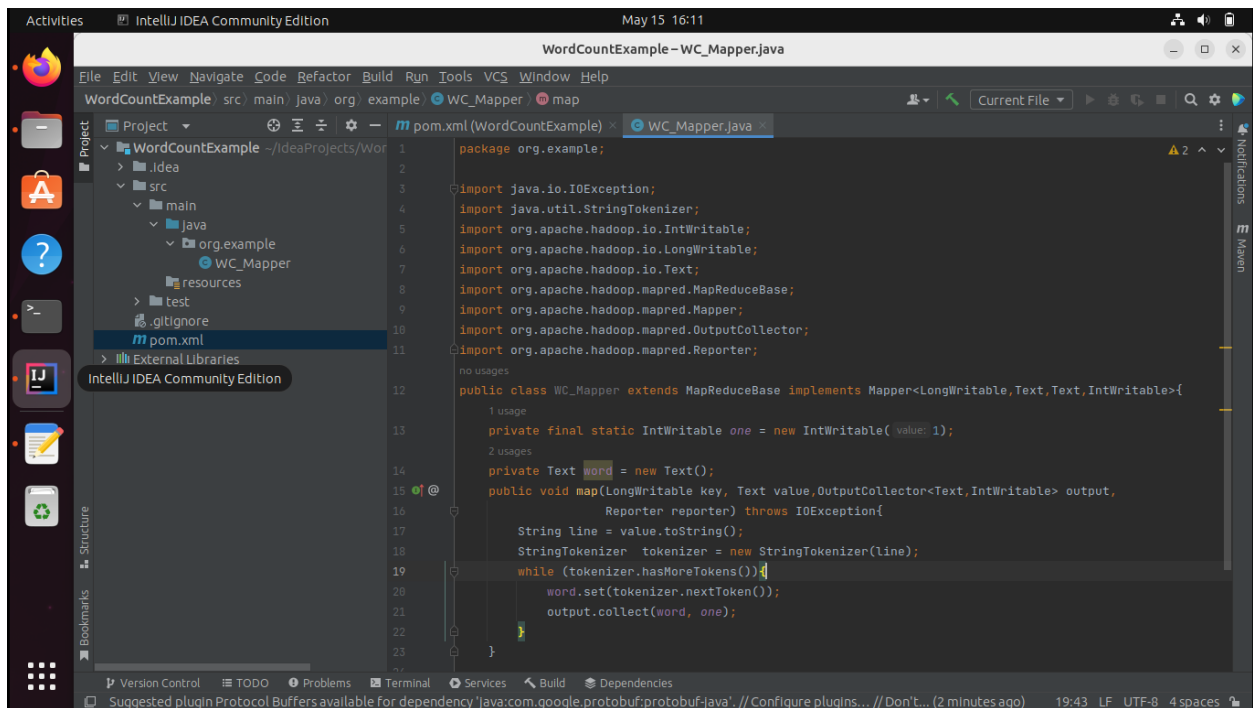
    output.collect(word, one);

}

}

}

```



Create new class WC_Reducer and add the following code inside it:

```

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 WC_Reducer extends MapReduceBase implements
Reducer<Text,IntWritable,Text,IntWritable> {

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

        int sum=0;

        while (values.hasNext()) {

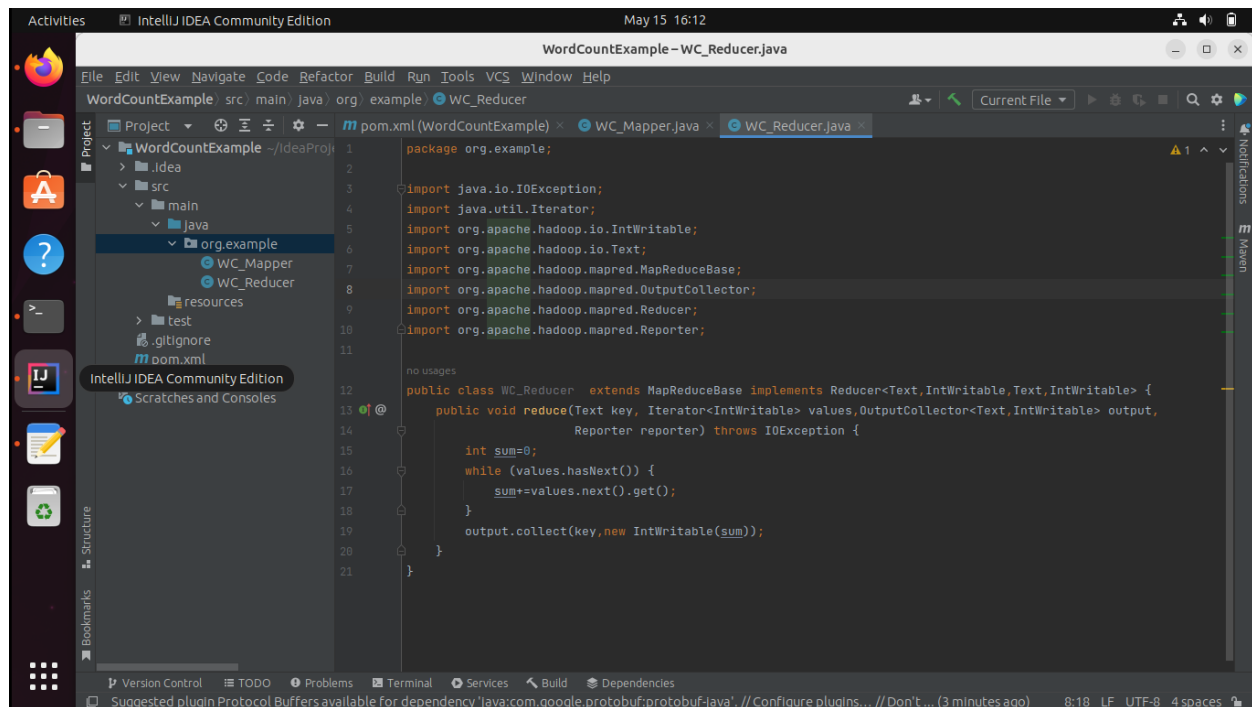
            sum+=values.next().get();

        }

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

    }

}
```



Create new class 'WC_Runner' and add the following code:

```
import java.io.IOException;

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.mapred.TextInputFormat;

import org.apache.hadoop.mapred.TextOutputFormat;

public class WC_Runner {

    public static void main(String[] args) throws IOException{
```

```

JobConf conf = new JobConf(WC_Runner.class);

conf.setJobName("WordCount");

conf.setOutputKeyClass(Text.class);

conf.setOutputValueClass(IntWritable.class);

conf.setMapperClass(WC_Mapper.class);

conf.setCombinerClass(WC_Reducer.class);

conf.setReducerClass(WC_Reducer.class);

conf.setInputFormat(TextInputFormat.class);

conf.setOutputFormat(TextOutputFormat.class);

FileInputFormat.setInputPaths(conf,new Path(args[0]));

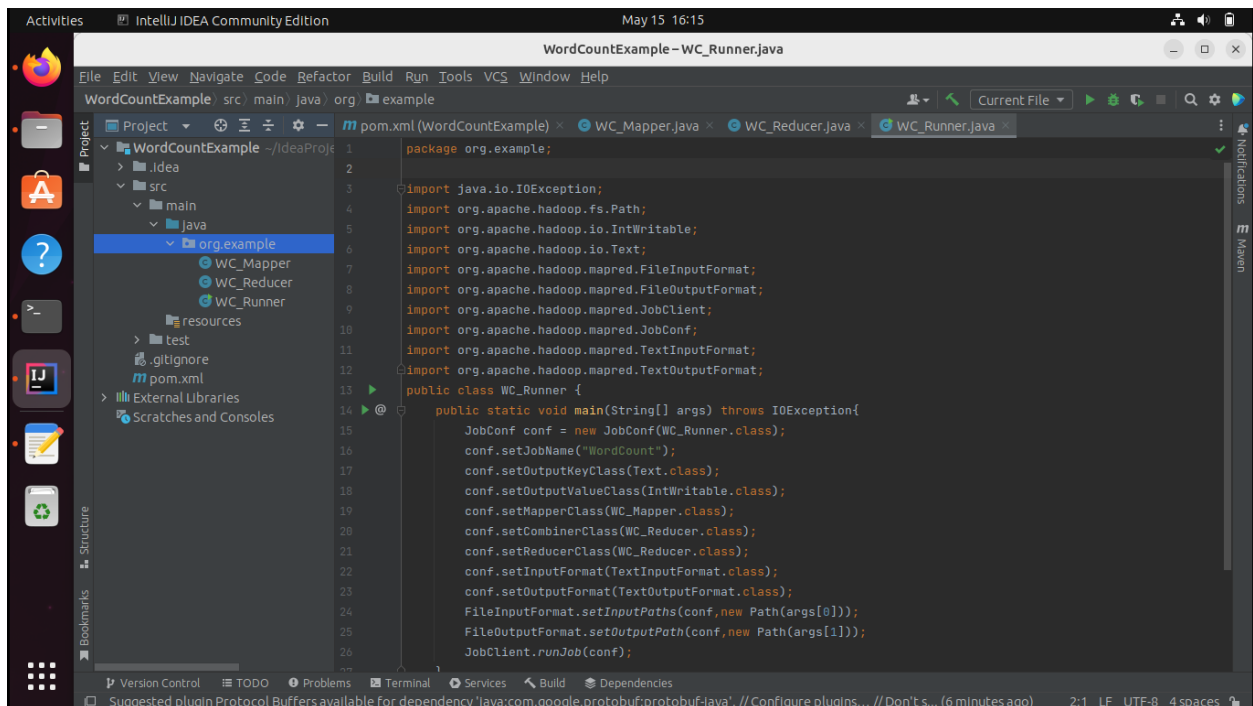
FileOutputFormat.setOutputPath(conf,new Path(args[1]));

JobClient.runJob(conf);

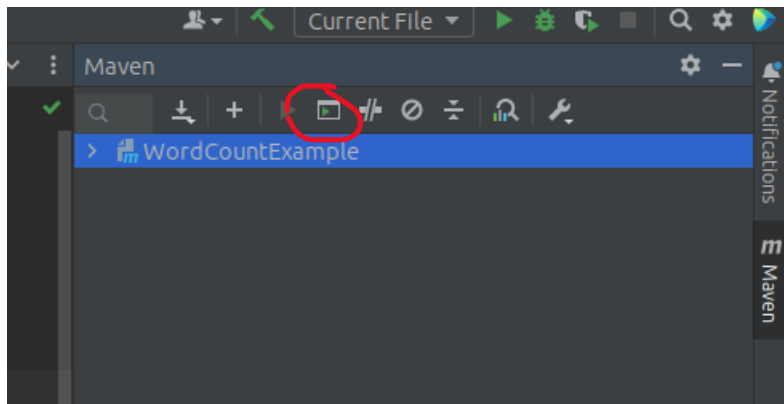
}

}

```



Now, Click on maven tab and select 'Execute Maven Goal'



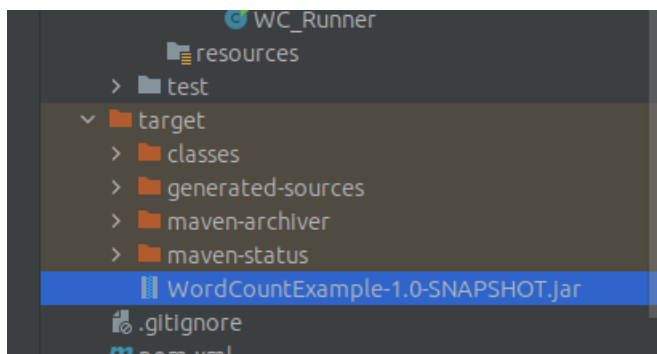
Run

```
mvn clean
```

Again click on 'Execute Maven Goal'

```
mvn install
```

This will create a .jar file which we will use later.

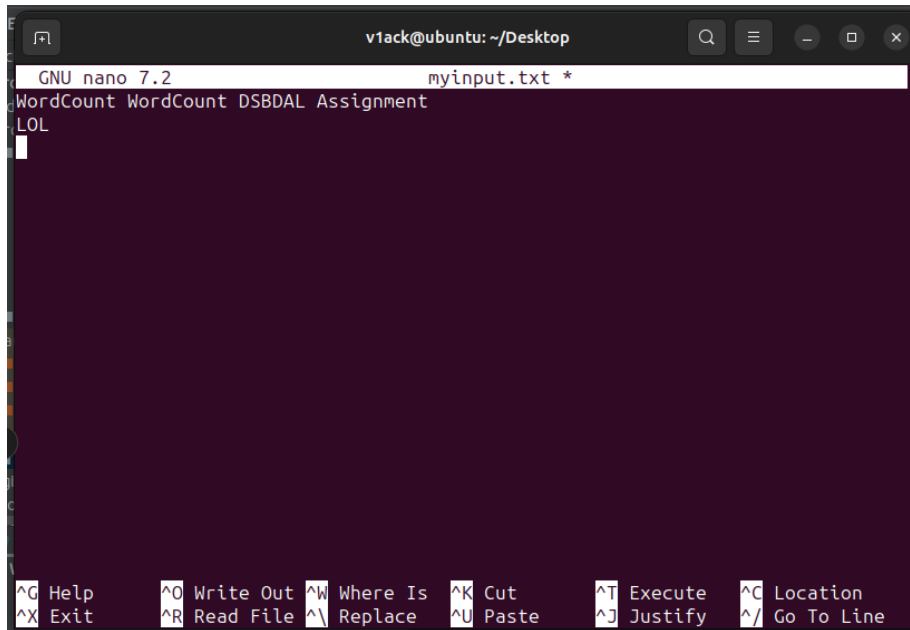


Open your terminal, move to your Desktop.

```
cd Desktop
```

```
nano myinput.txt
```

Add anything that you want.



```
GNU nano 7.2 myinput.txt *
WordCount WordCount DSBDA Assignment
LOL
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

After adding the words or sentences, type

Ctrl + O

Enter

Ctrl + X

Enter

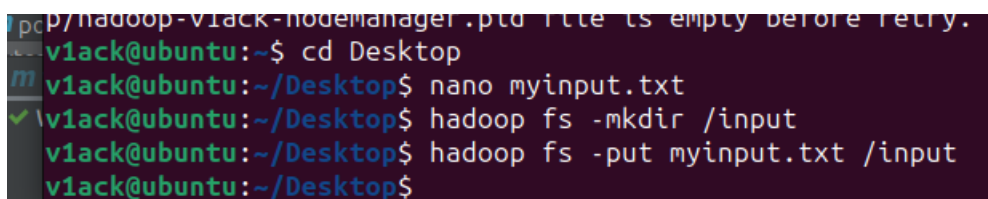
Above commands will then save the text that you wrote into myinput.txt file.

Run the below command to create a new directory inside Hadoop file system.

```
hadoop fs -mkdir /input
```

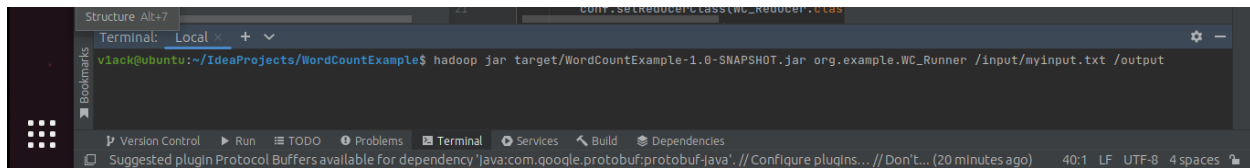
Run the below command to move the myinput.txt to your newly crated input folder inside hadoop file system.

```
hadoop fs -put myinput.txt /input
```



```
pcp/hadoop-v1ack-nodemanager.ptd file is empty before retry.
v1ack@ubuntu:~$ cd Desktop
v1ack@ubuntu:~/Desktop$ nano myinput.txt
v1ack@ubuntu:~/Desktop$ hadoop fs -mkdir /input
v1ack@ubuntu:~/Desktop$ hadoop fs -put myinput.txt /input
v1ack@ubuntu:~/Desktop$
```

Now, open IntelliJ IDE and select terminal from the bottom bar and run the following command.



```
Terminal: Local x + v
vjack@ubuntu:~/IdeaProjects/WordCountExample$ hadoop jar target/WordCountExample-1.0-SNAPSHOT.jar org.example.WC_Runner /input/myinput.txt /output
```

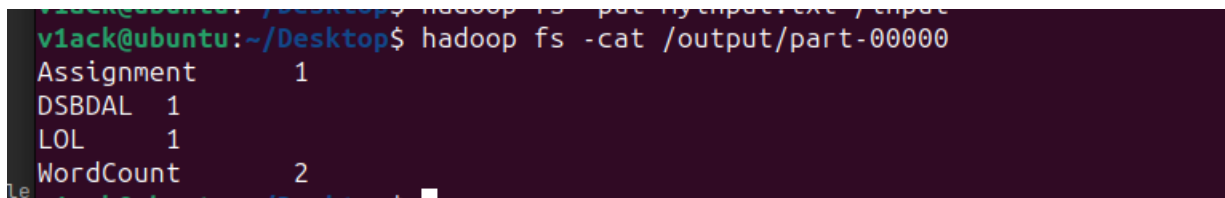
```
hadoop jar target/WordCountExample-1.0-SNAPSHOT.jar org.example.WC_Runner /input/myinput.txt /output
```

This will take 2-3 minutes depending on your processing capabilities.

After successfully executing your output will be ready.

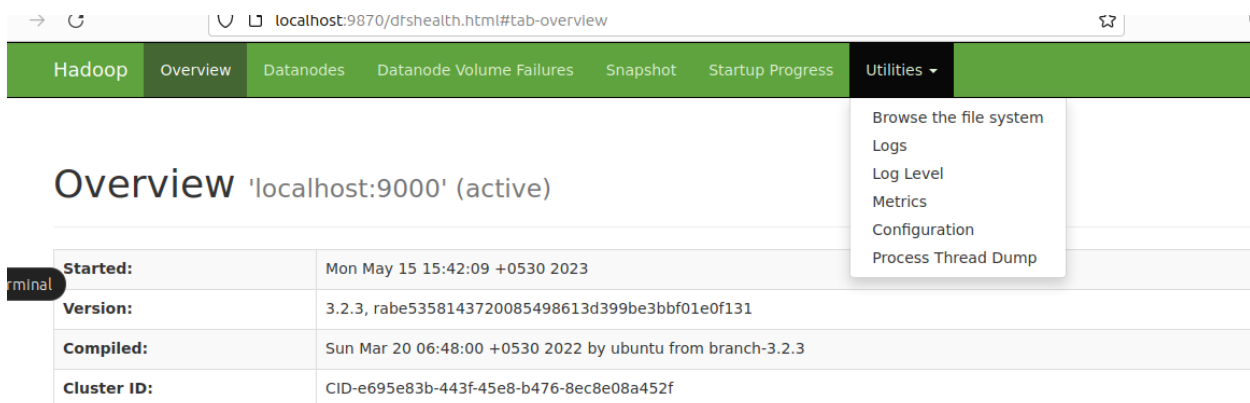
To check the output, run the following command in your terminal:

```
hadoop fs -cat /output/part-00000
```



```
vjack@ubuntu:~/Desktop$ hadoop fs -cat /output/part-00000
Assignment      1
DSBDAL 1
LOL 1
WordCount      2
```

You can also check the output from the Hadoop user interface by going into the Utilities/Browse the file system



The screenshot shows the Hadoop UI with the 'Utilities' dropdown menu open. The 'Browse the file system' option is highlighted. Below the menu, the 'Overview' tab is selected, showing details for 'localhost:9000' (active).

Overview 'localhost:9000' (active)	
Started:	Mon May 15 15:42:09 +0530 2023
Version:	3.2.3, rabe5358143720085498613d399be3bbf01e0f131
Compiled:	Sun Mar 20 06:48:00 +0530 2022 by ubuntu from branch-3.2.3
Cluster ID:	CID-e695e83b-443f-45e8-b476-8ec8e08a452f

Type '/' in the Browse Directory

Select output

Click on part-00000

Click on 'Head the file (first 32K)

File information - part-00000

Download

Head the file (first 32K)

Tail the file (last 32K)

Block information -- Block 0

Block ID: 1073741832

Block Pool ID: BP-1438539494-127.0.1.1-1684145501381

Generation Stamp: 1008

Size: 40

Availability:

- ubuntu

File contents

Assignment 1

DSBDAL 1

LOL 1

WordCount 2