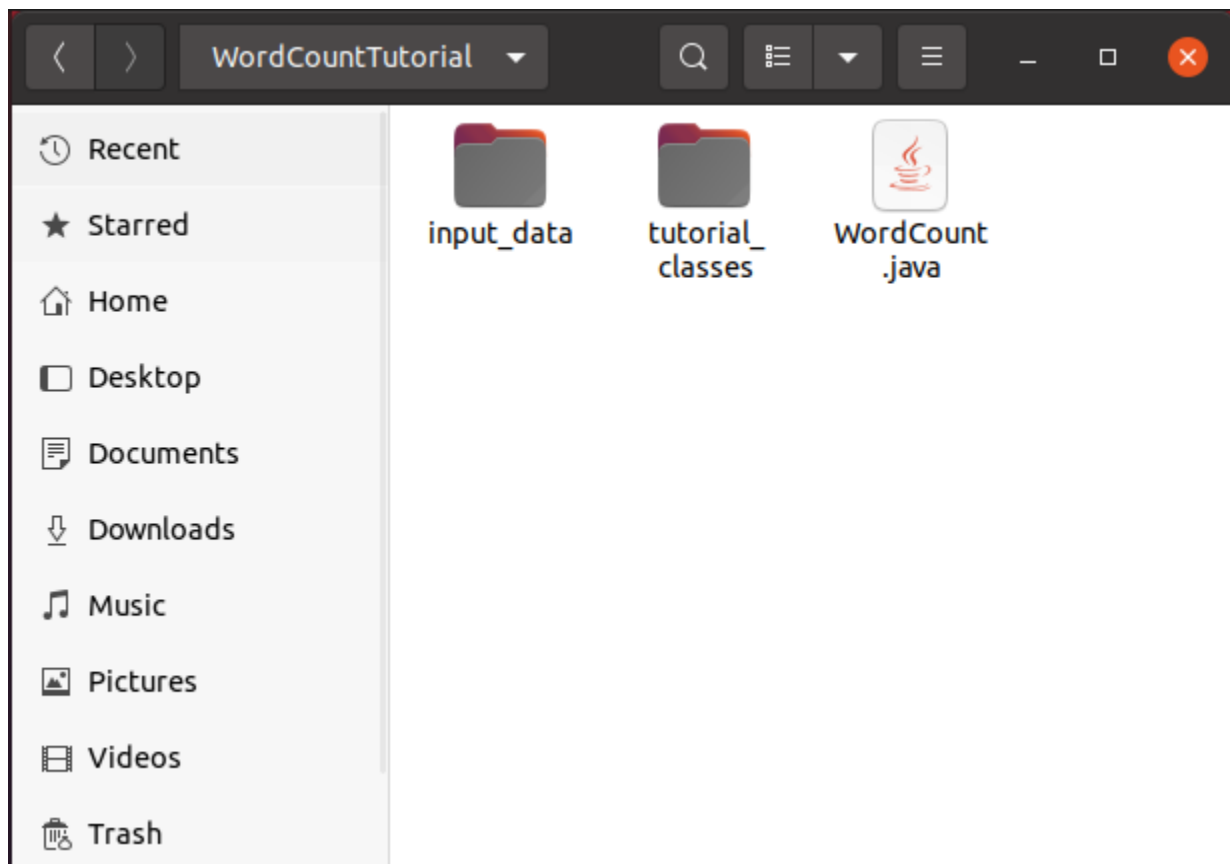


Running WordCount on Hadoop (Linux)

Step 1: Download the [WordCount.java](#) file.

```
java -version
```

Step 2: Create a folder called WordCountTutorial. Create two subfolders within this directory called 'input_data' and 'tutorial_classes'. Place the WordCount.java file from above into the WordCountTutorial Directory.



Step 3: Create a txt file called 'input' and save it in the input_data folder. Add some sample text to the input file.

Sample text:

Hello I am Qasim

How can I help you

How can I assist you

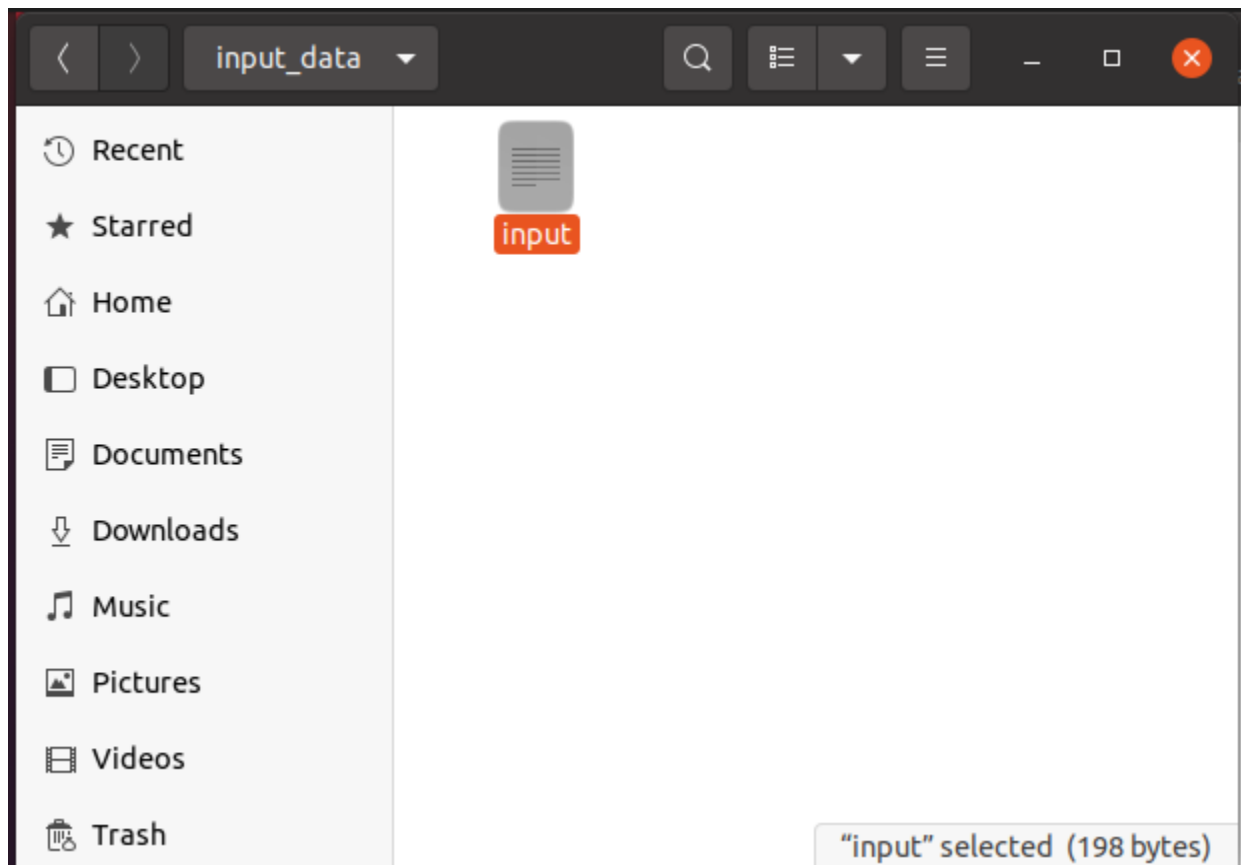
Are you an engineer

Are you looking for coding

Are you looking for interview questions

what are you doing these days

what are your strengths



Step 4: Check if you have javac installed.

```
javac -version
```

```
hadoopusr@ubuntu:~/Desktop$ javac -version
javac 11.0.14
```

Step 5: Make sure hadoop is running.

```
start-dfs.sh
```

```
start-yarn.sh
```

Step 6: Set HADOOP_CLASSPATH environment variable.

```
export HADOOP_CLASSPATH=$(hadoop classpath)
```

To check if it has been set properly use:

```
echo $HADOOP_CLASSPATH
```

```
hadoopusr@ubuntu:~/Desktop$ export HADOOP_CLASSPATH=$(hadoop classpath)
hadoopusr@ubuntu:~/Desktop$ echo $HADOOP_CLASSPATH
/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/*:/usr/local/hadoop/share/hadoop/common/*:/usr/local/hadoop/share/hadoop/hdfs:/usr/local/hadoop/share/hadoop/hdfs/lib/*:/usr/local/hadoop/share/hadoop/hdfs/*:/usr/local/hadoop/share/hadoop/yarn:/usr/local/hadoop/share/hadoop/yarn/lib/*:/usr/local/hadoop/share/hadoop/yarn/*:/usr/local/hadoop/share/hadoop/mapreduce/lib/*:/usr/local/hadoop/share/hadoop/mapreduce/*:/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/common/lib/*:/usr/local/hadoop/share/hadoop/common/*:/usr/local/hadoop/share/hadoop/hdfs:/usr/local/hadoop/share/hadoop/hdfs/lib/*:/usr/local/hadoop/share/hadoop/hdfs/*:/usr/local/hadoop/share/hadoop/yarn:/usr/local/hadoop/share/hadoop/yarn/lib/*:/usr/local/hadoop/share/hadoop/yarn/*:/usr/local/hadoop/share/hadoop/mapreduce/lib/*:/usr/local/hadoop/share/hadoop/mapreduce/*:/usr/local/hadoop/contrib/capacity-scheduler/*.jar:/usr/local/hadoop/contrib/capacity-scheduler/*.jar
```

Step 7: Create Directory for the Program with a subdirectory for the input data:

Format:

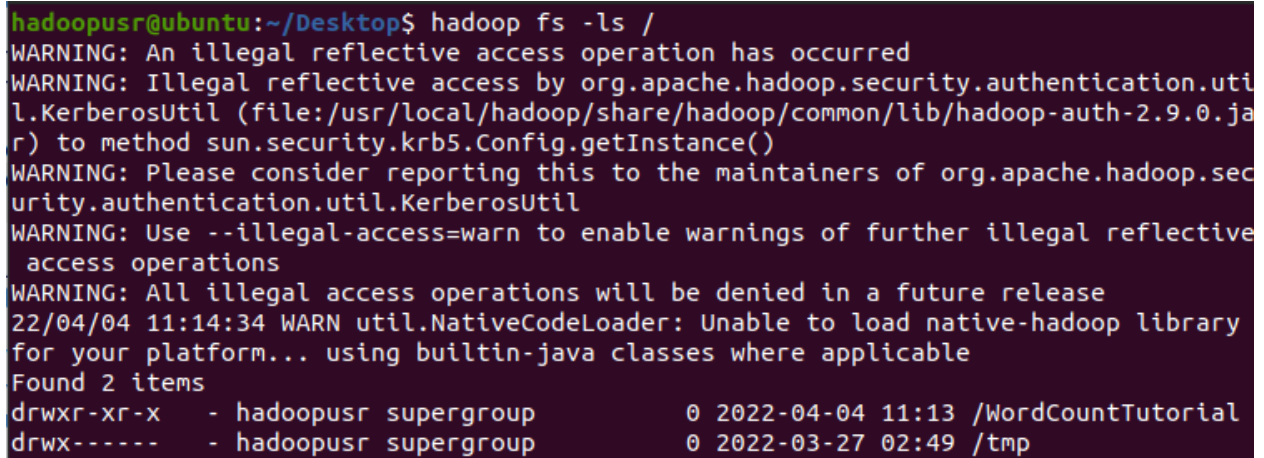
```
hadoop fs -mkdir <DIRECTORY_NAME>
```

```
hadoop fs -mkdir <HDFS_INPUT_DIRECTORY>
```

```
hadoop fs -mkdir /WordCountTutorial
hadoop fs -mkdir /WordCountTutorial/Input
```

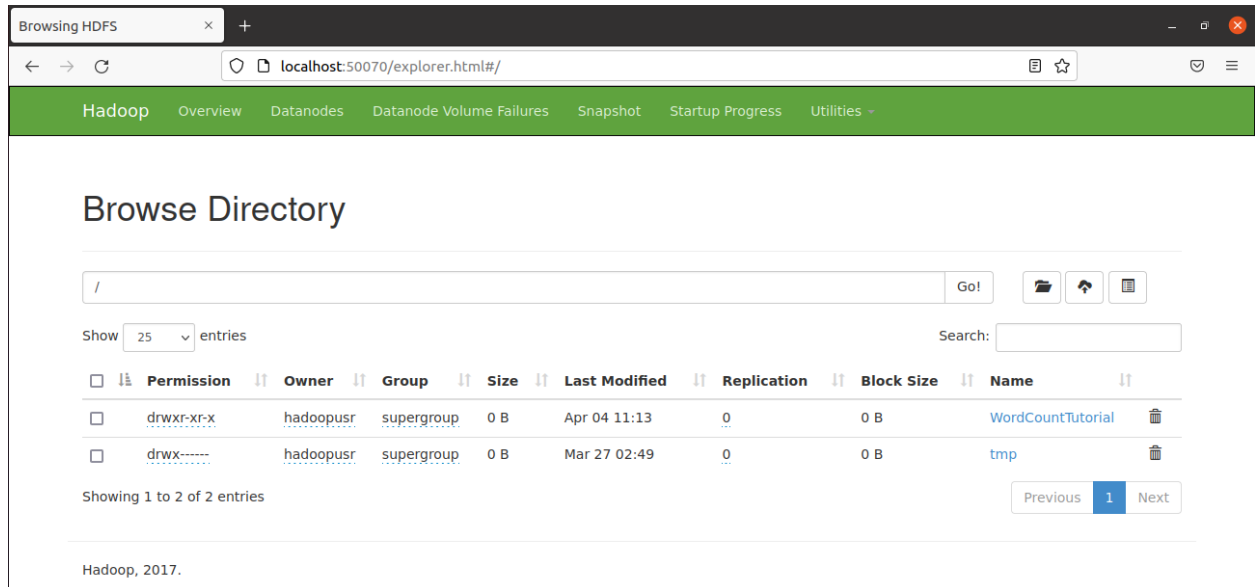
You can check if the directories have been created using:

```
hadoop fs -ls /
hadoop fs -ls /WordCountTutorial
```

A terminal window with a dark background and light-colored text. The prompt is 'hadoopusr@ubuntu:~/Desktop\$'. The command 'hadoop fs -ls /' is entered. The output shows several warning messages from Hadoop regarding reflective access and native code loading. After the warnings, it says 'Found 2 items' and lists two directories: '/WordCountTutorial' with permissions 'drwxr-xr-x' and '/tmp' with permissions 'drwx-----'.

```
hadoopusr@ubuntu:~/Desktop$ hadoop fs -ls /
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.9.0.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
22/04/04 11:14:34 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
drwxr-xr-x   - hadoopusr supergroup          0 2022-04-04 11:13 /WordCountTutorial
drwx----- - hadoopusr supergroup          0 2022-03-27 02:49 /tmp
```

You can also go to localhost:50070 > Utilities > Browse the file System.



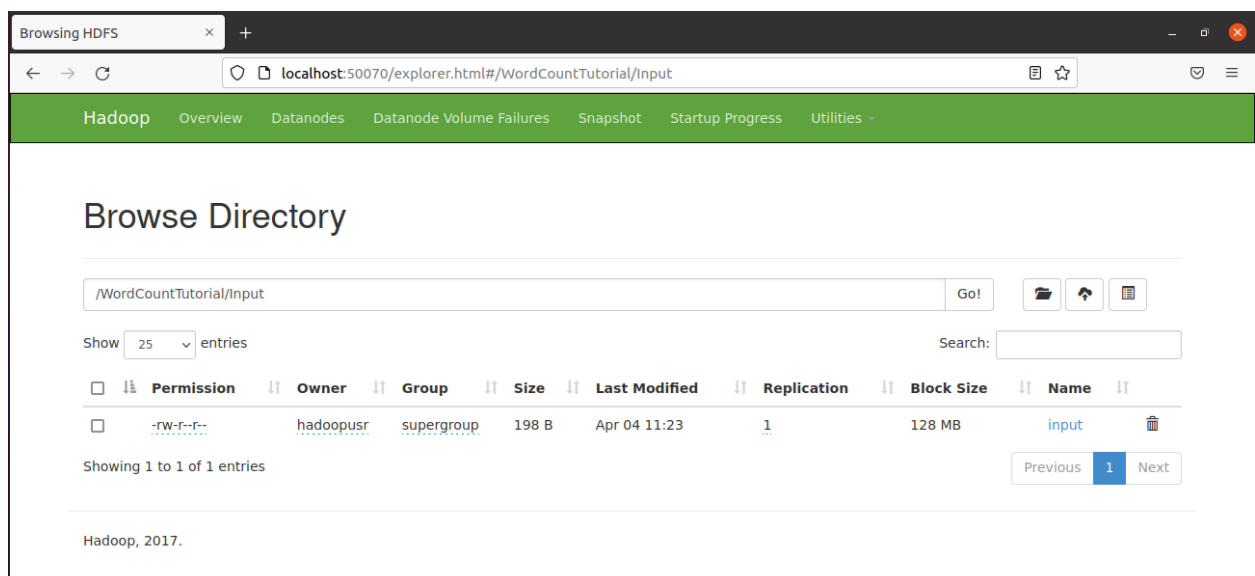
Step 8: Upload the input file to the filesystem:

Format:

`hadoop fs -put <INPUT_FILE> <HDFS_INPUT_DIRECTORY>`

```
hadoop fs -put
' /home/hadoopusr/Desktop/WordCountTutorial/input_data/input
' /WordCountTutorial/Input
```

You can check if the file has been uploaded on the localhost:



Step 9: Change the current directory to the WordCountTutorial directory using:

Format:

```
cd <DIRECTORY_PATH>
```

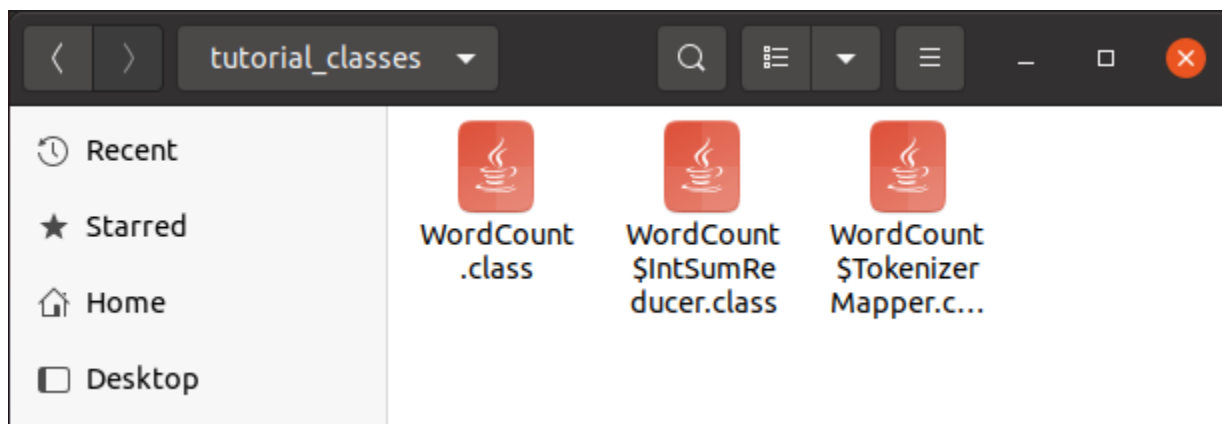
```
cd WordCountTutorial
```

Step 10: Compile the java code using:

Format: `javac -classpath ${HADOOP_CLASSPATH} -d <CLASSES_FOLDER> <JAVA_FILE>`

```
javac -classpath ${HADOOP_CLASSPATH} -d  
'/home/hadoopusr/Desktop/WordCountTutorial/tutorial_classes'  
'/home/hadoopusr/Desktop/WordCountTutorial/WordCount.java'
```

This will produce 3 .class files in the tutorial_classes folder.



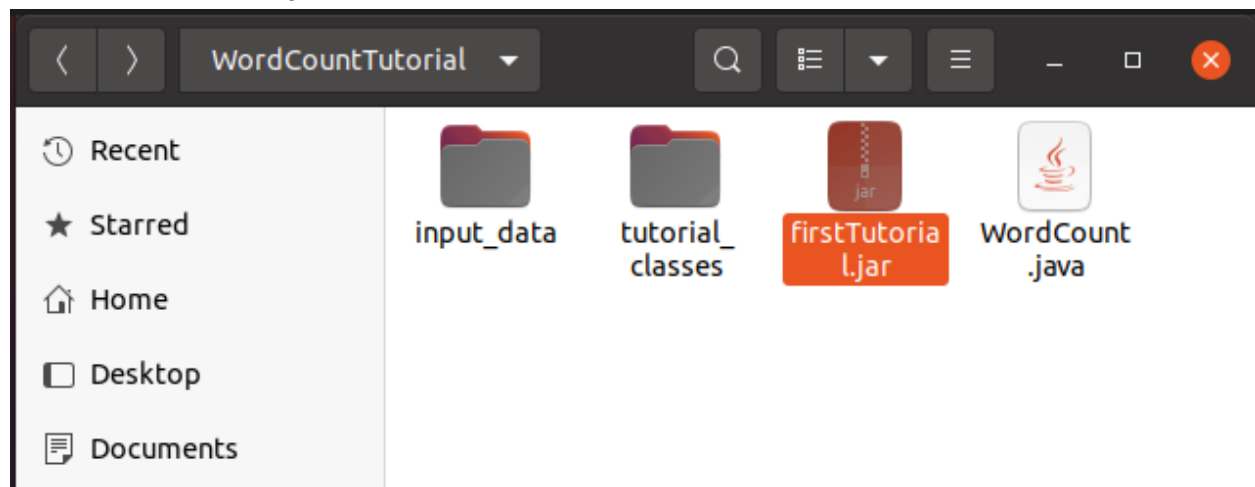
Step 11: Put the output files (.class) into one jar file.

Format:

```
jar -cvf <JAR_FILE_NAME> <CLASSES_FOLDER>
```

```
jar -cvf firstTutorial.jar -C tutorial_classes/ .
```

Now we have the .jar file



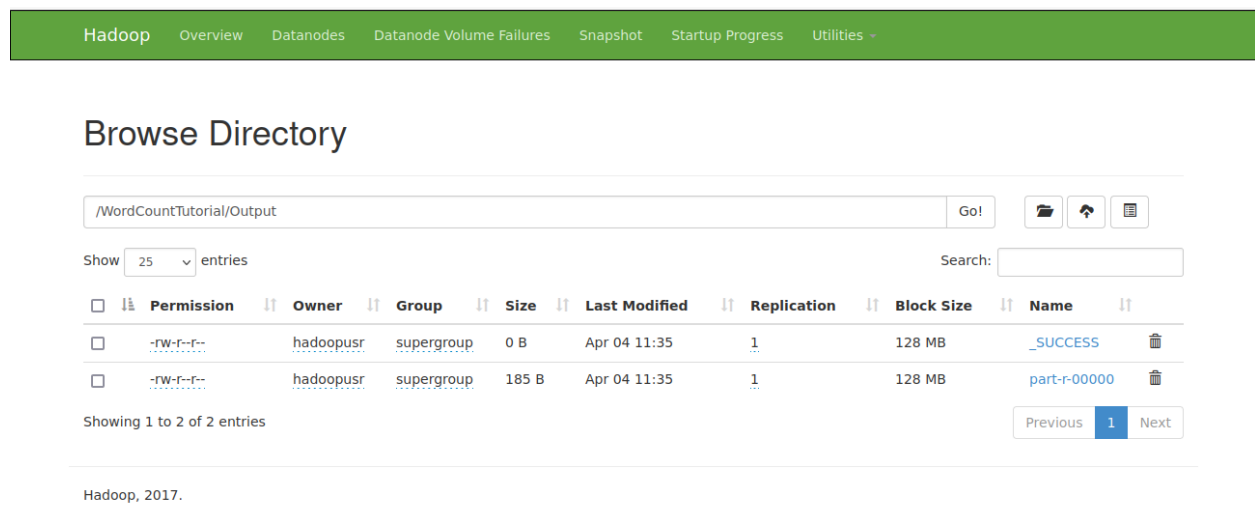
Step 12: Now, to run the file on hadoop:

Format:

```
hadoop jar <JAR_FILE> <CLASS_NAME> <HDFS_INPUT_DIRECTORY>  
<HDFS_OUTPUT_DIRECTORY>
```

```
hadoop jar  
'/home/hadoopusr/Desktop/WordCountTutorial/firstTutorial.jar'  
WordCount /WordCountTutorial/Input  
/WordCountTutorial/Output
```

Output file has been created on the filesystem and can be viewed in the localhost.

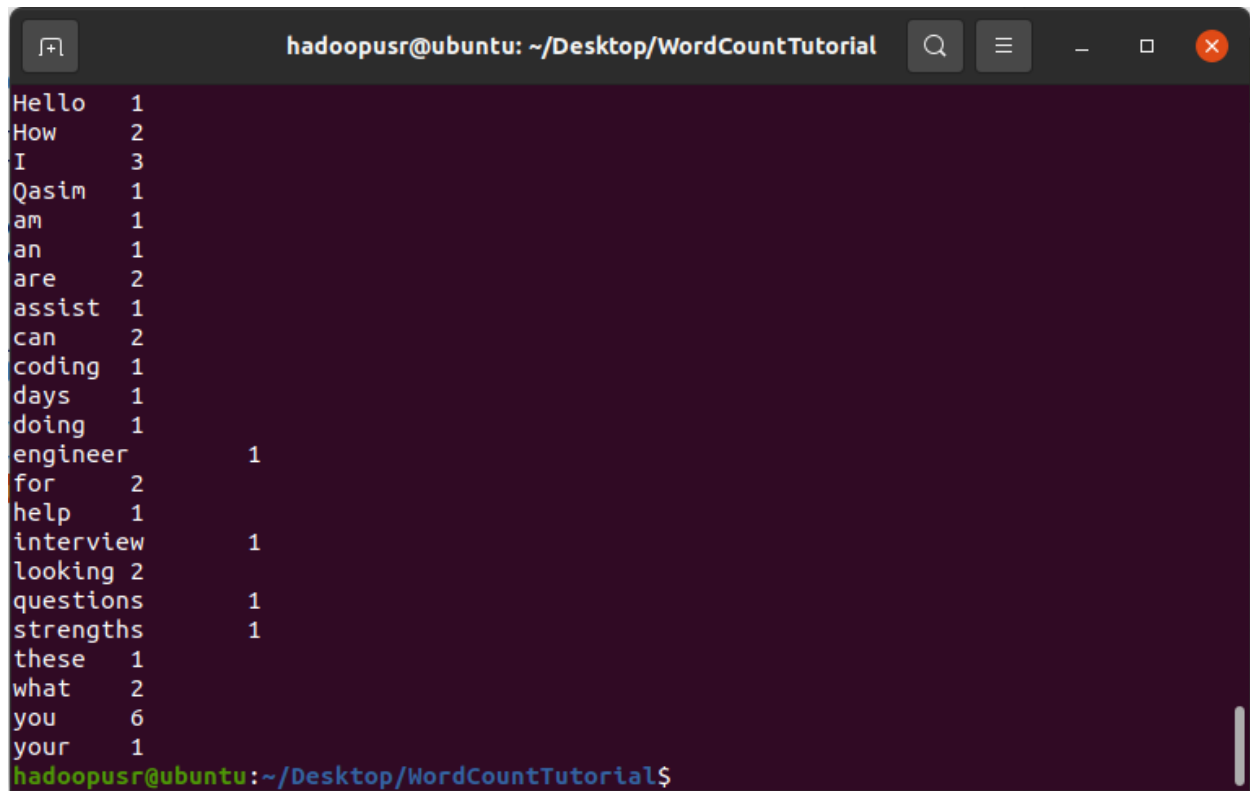


Step 13: To view the output on the terminal:

Format:

```
hadoop dfs -cat <HDFS_OUTPUT_DIRECTORY>*
```

```
hadoop dfs -cat /WordCountTutorial/Output/*
```

A terminal window titled 'hadoopusr@ubuntu: ~/Desktop/WordCountTutorial' with search, menu, and window control icons. It displays the output of the 'hadoop dfs -cat' command, showing a list of words and their counts. The output is as follows:

Hello	1
How	2
I	3
Qasim	1
am	1
an	1
are	2
assist	1
can	2
coding	1
days	1
doing	1
engineer	1
for	2
help	1
interview	1
looking	2
questions	1
strengths	1
these	1
what	2
you	6
your	1

The prompt 'hadoopusr@ubuntu:~/Desktop/WordCountTutorial\$' is visible at the bottom.

Dire