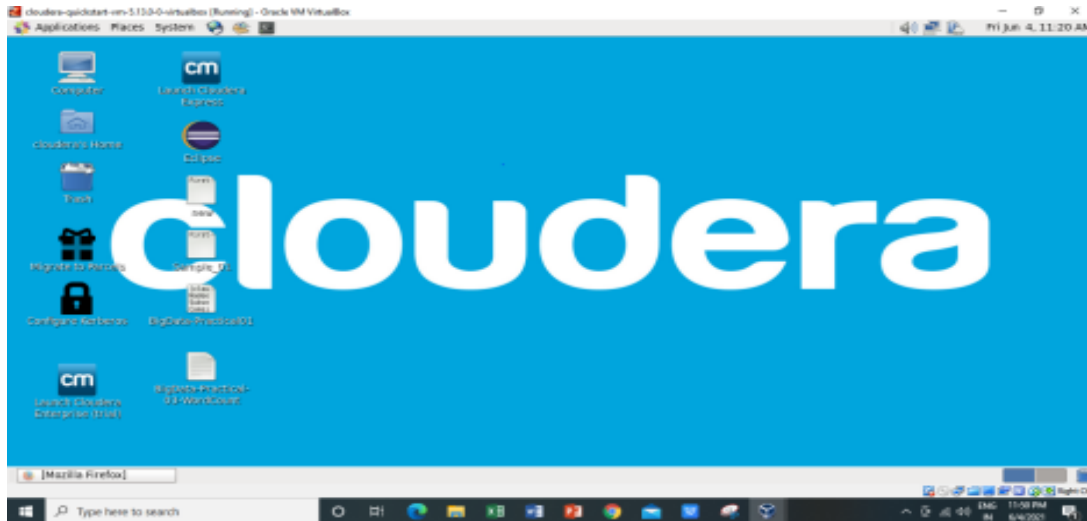


Name:  
Roll No:

## Steps for Word Count in Cloudera: (Without Combiner)

1) Open virtual box and then start cloudera quickstart.

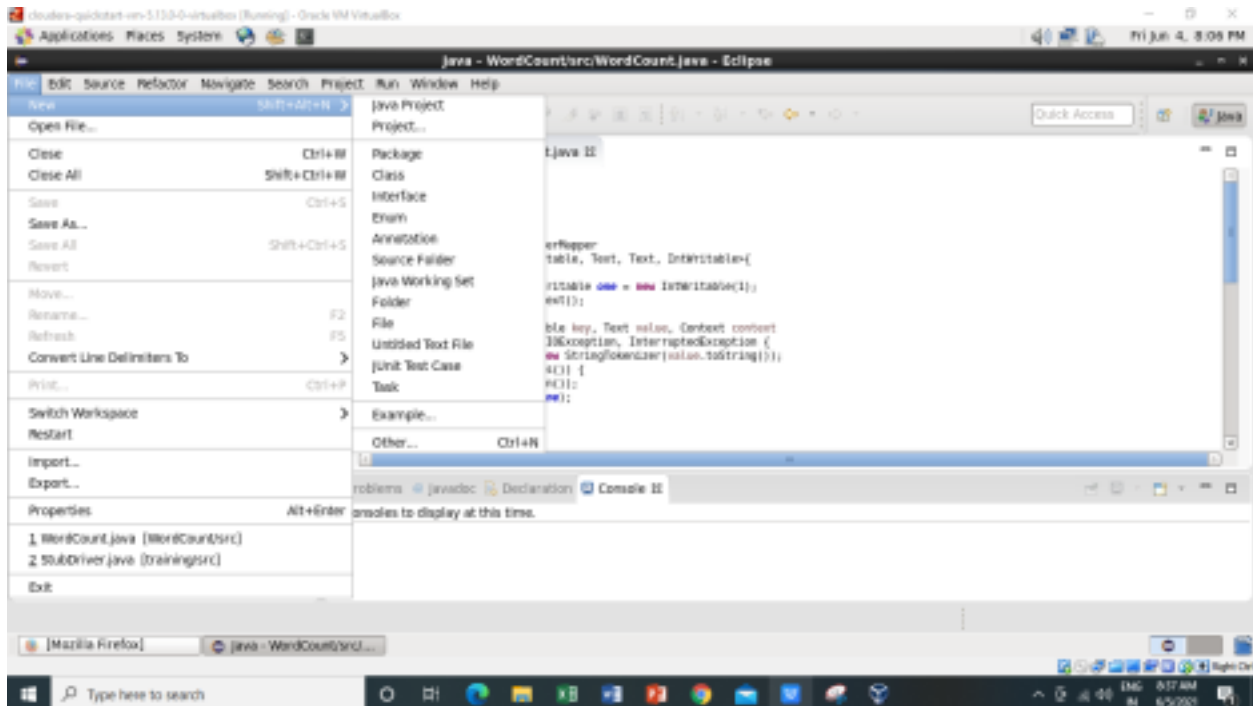


2) Open Eclipse present on the cloudera desktop.

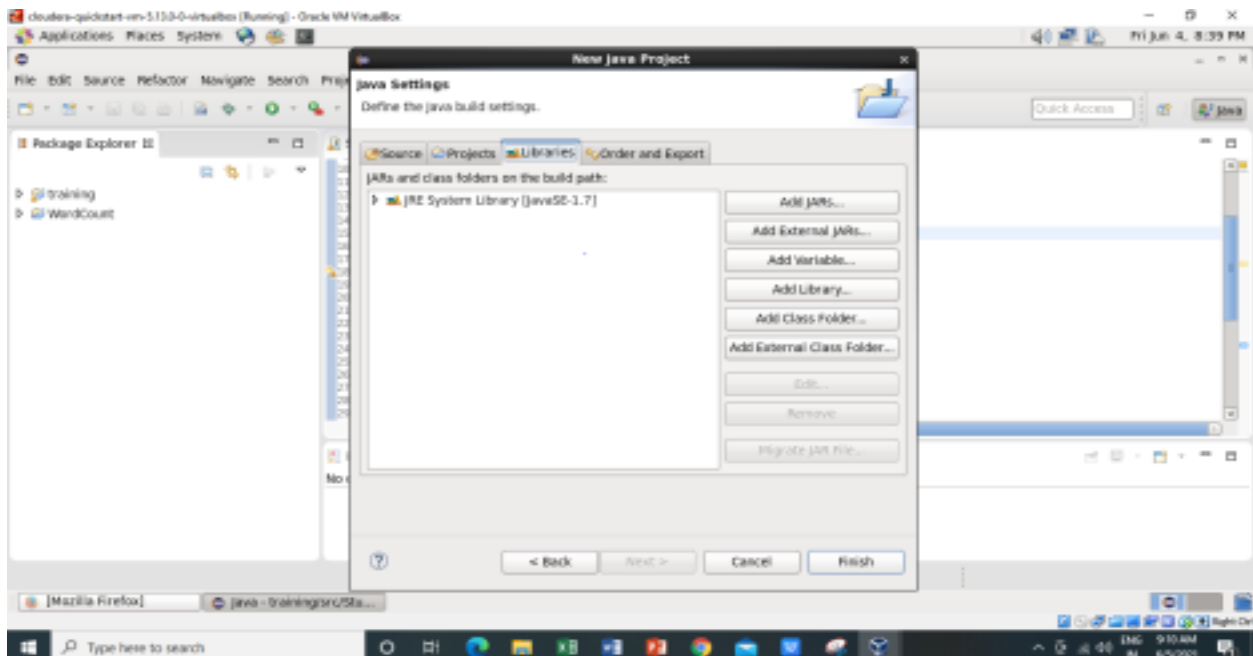


3) Create a new Java project clicking: File -> New -> Project -> Java Project -> Next ("WordCount" is the project name).

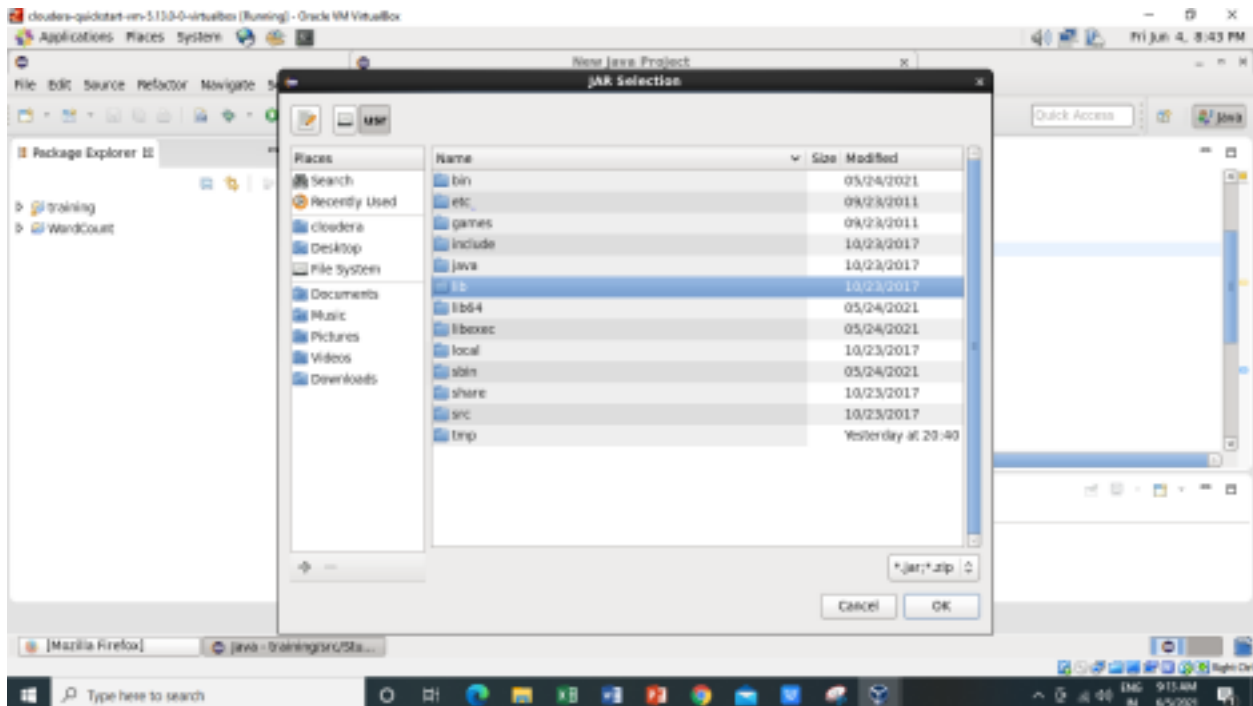
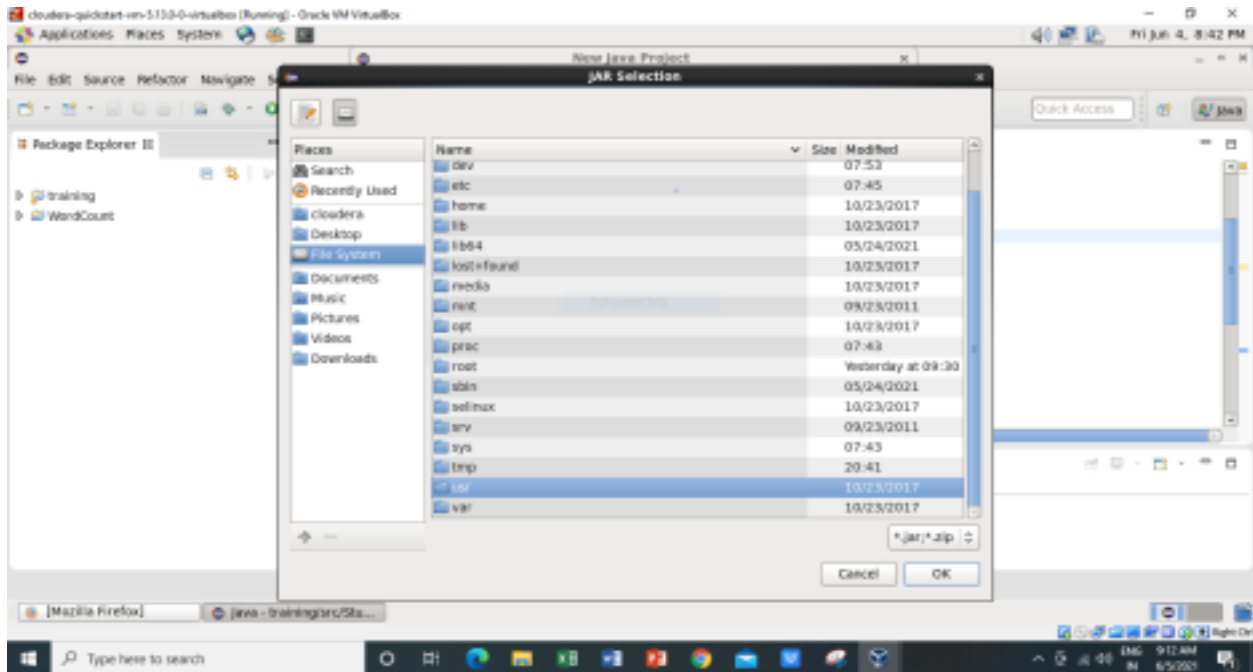
Name:  
Roll No:



4) Adding the Hadoop libraries to the project Click on Libraries -> Add External JARs Click on File System -> usr -> lib -> hadoop Select all the libraries (JAR Files) -> click OK Click on Add External jars, -> client -> select all jar files -> ok -> Finish

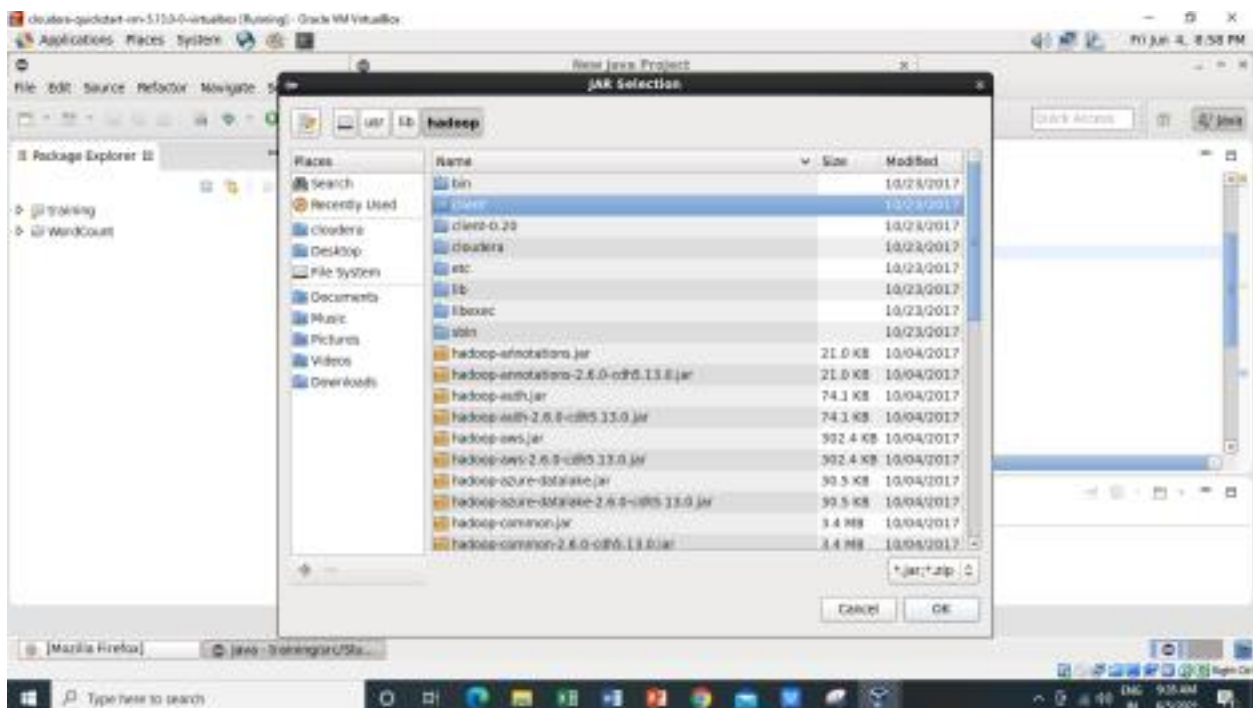
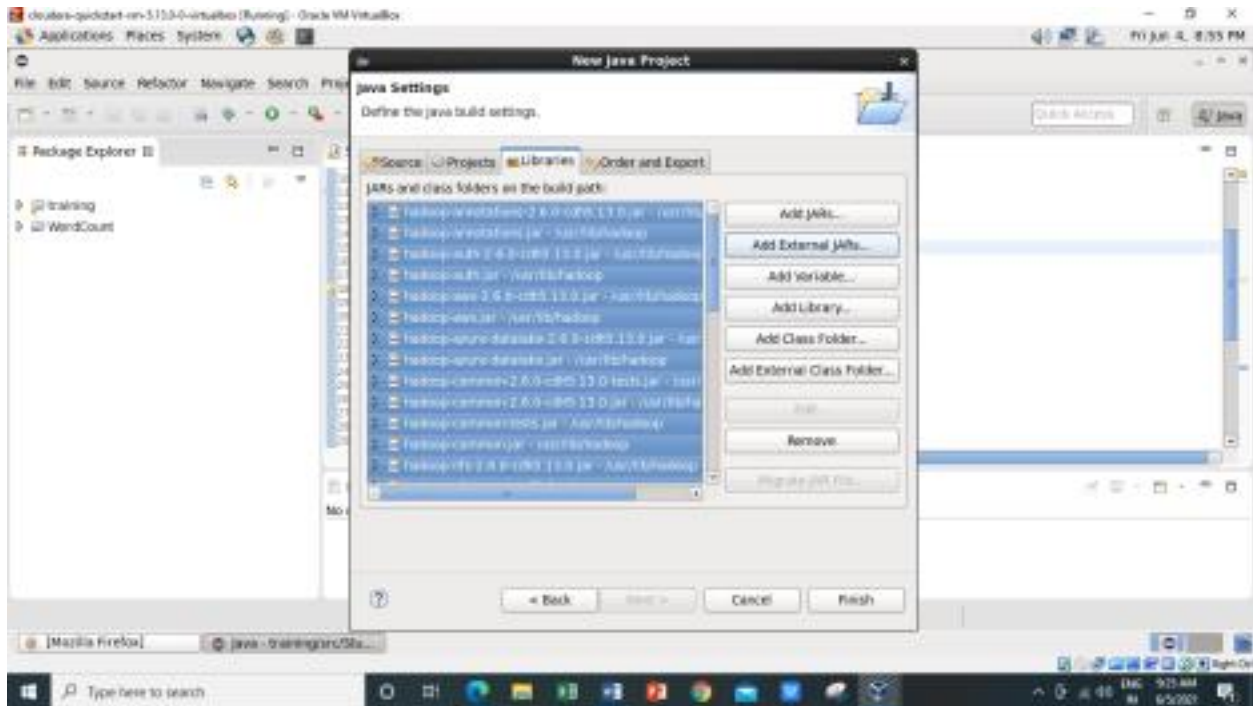


Name:  
Roll No:

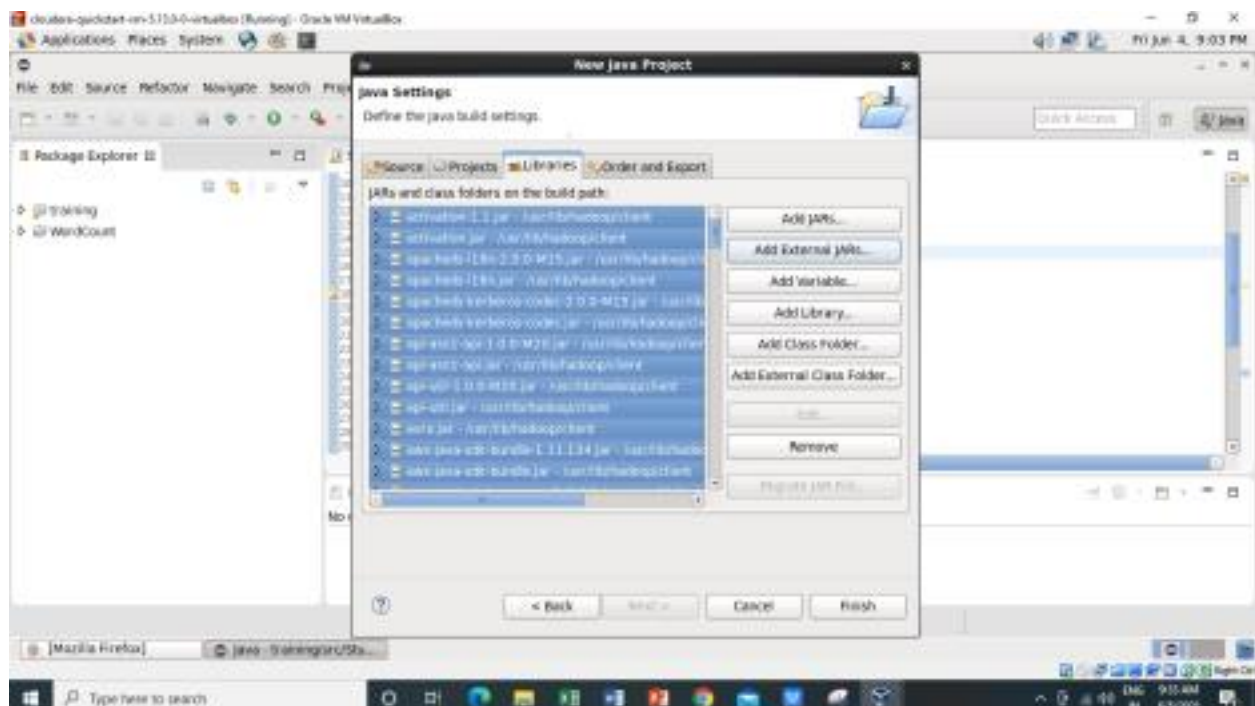
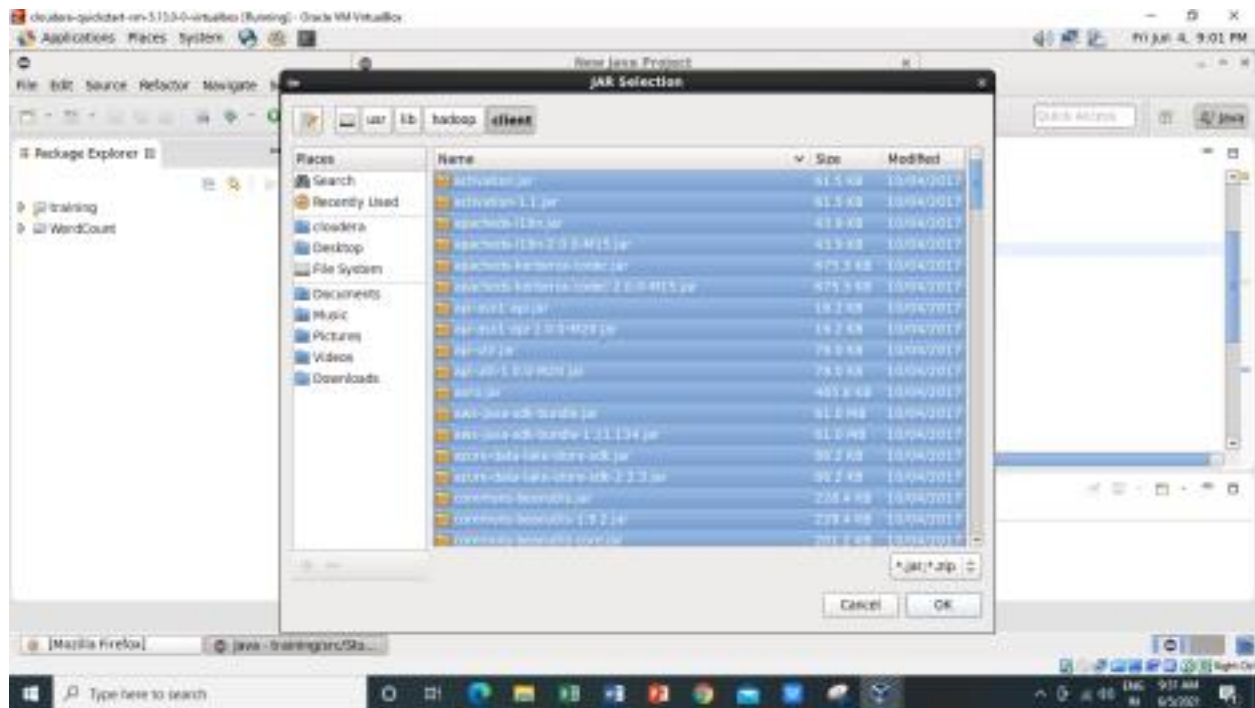




**Name:**  
**Roll No:**



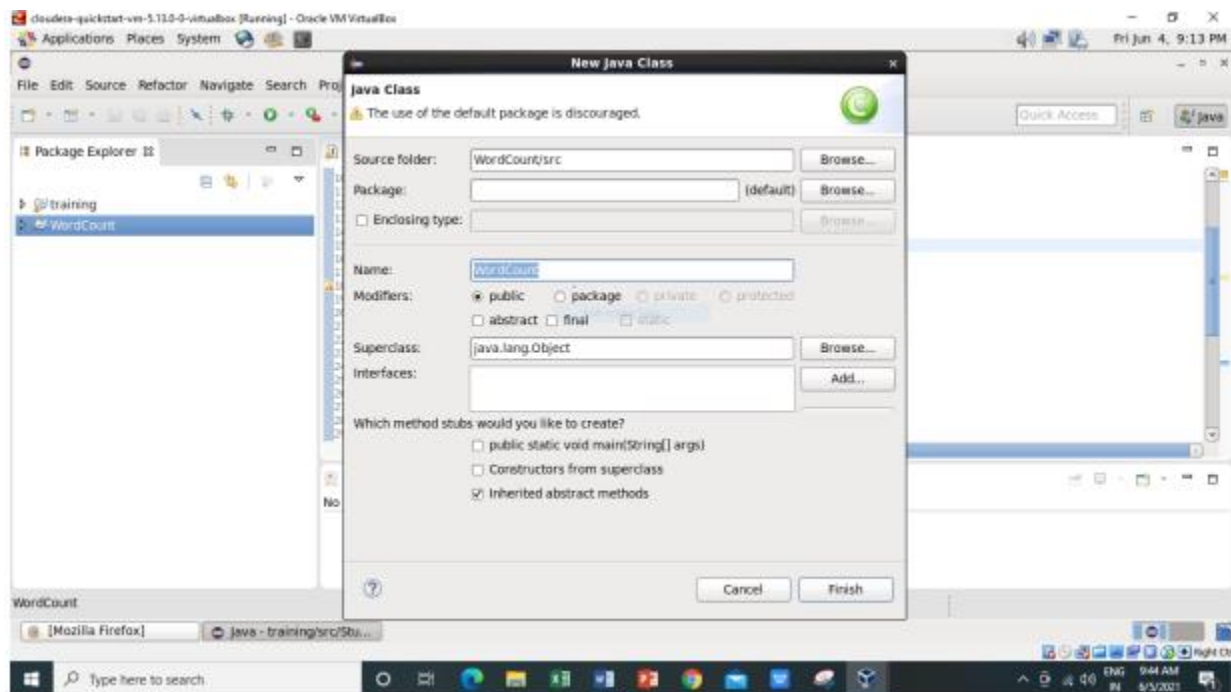
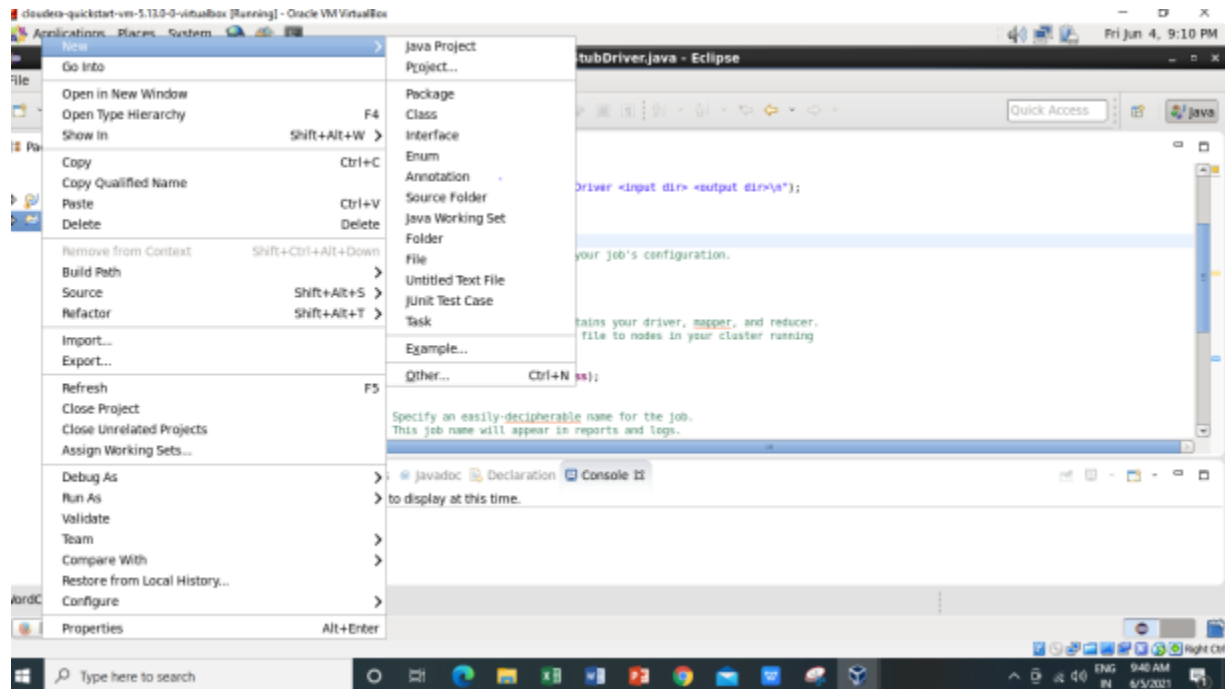
**Name:**  
**Roll No:**



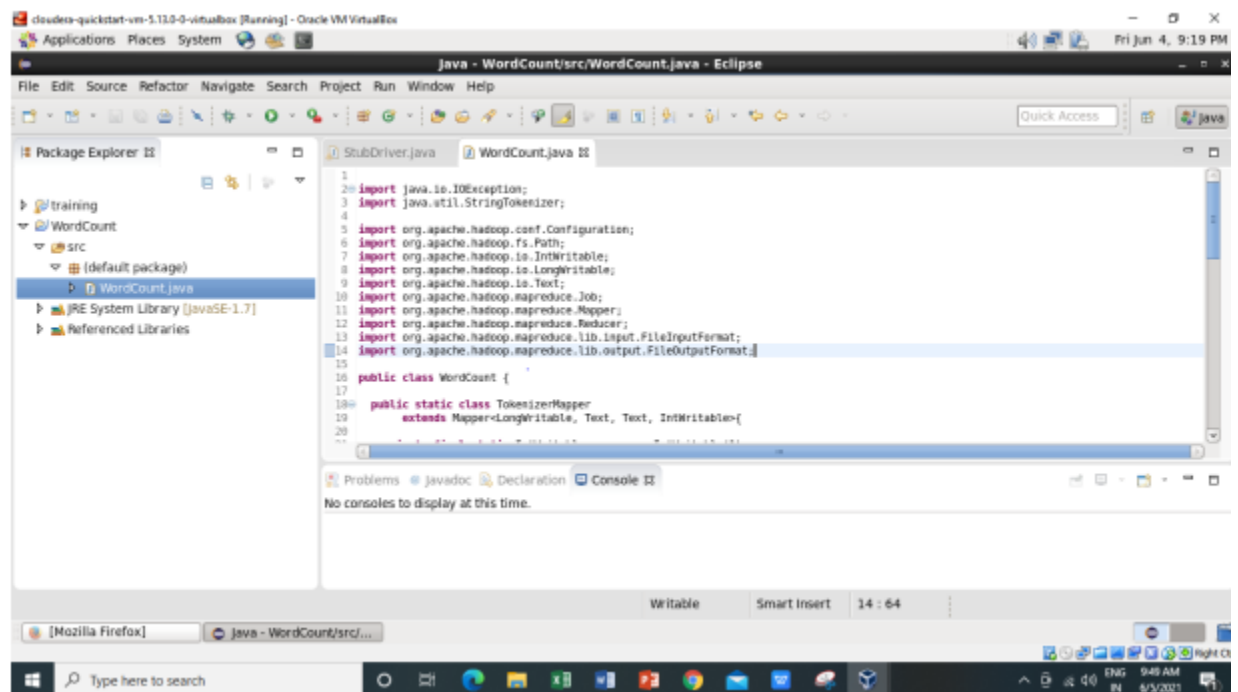
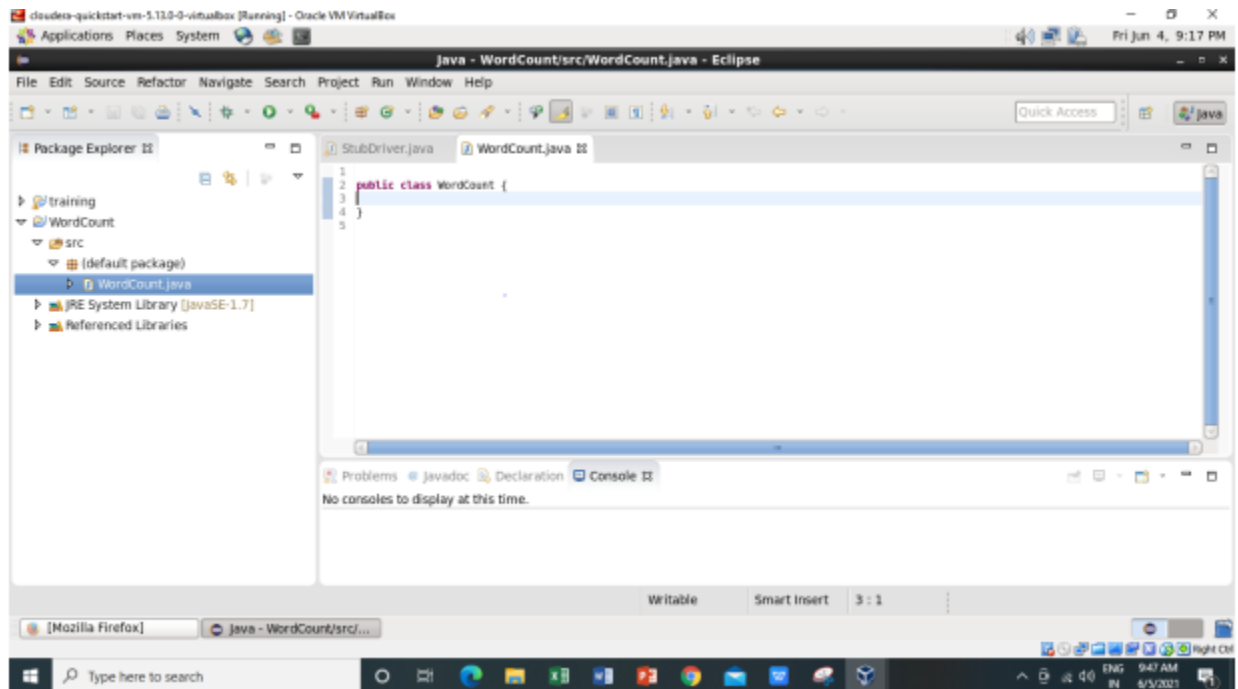
5) Right Click on the name of Project “WordCount” -> New -> class Don’t write anything for package Write Name Textbox write “WordCount” -> Finish Then WordCount.java window will pop up



Name:  
Roll No:



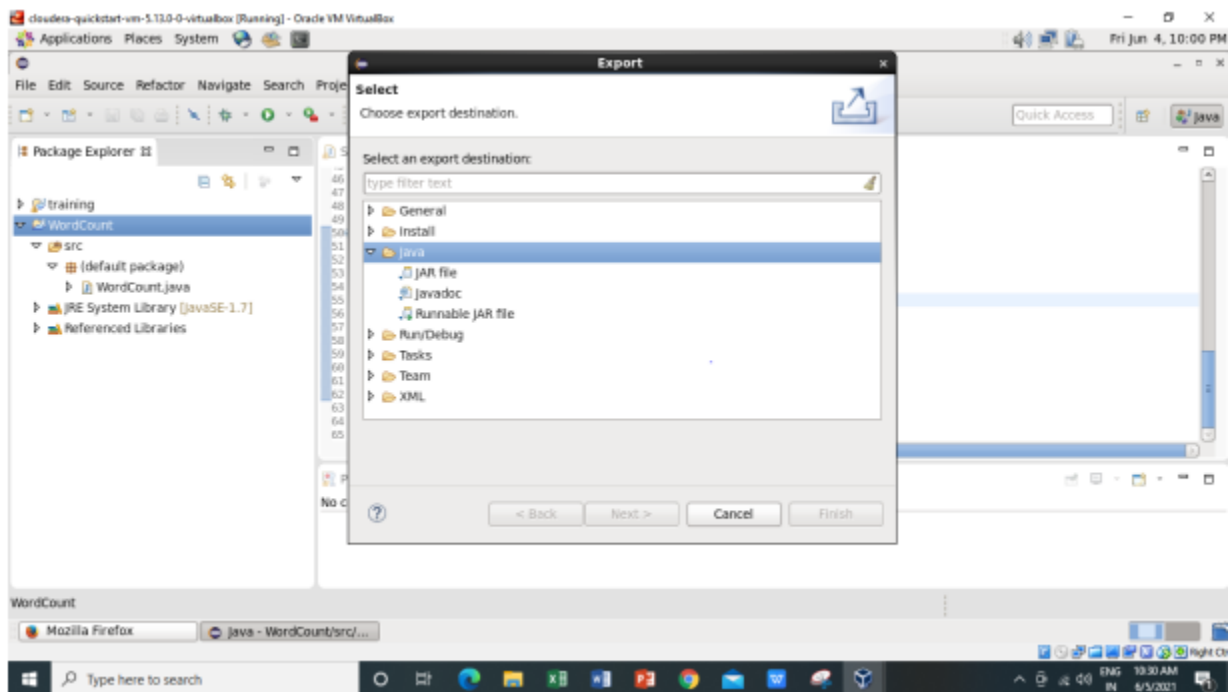
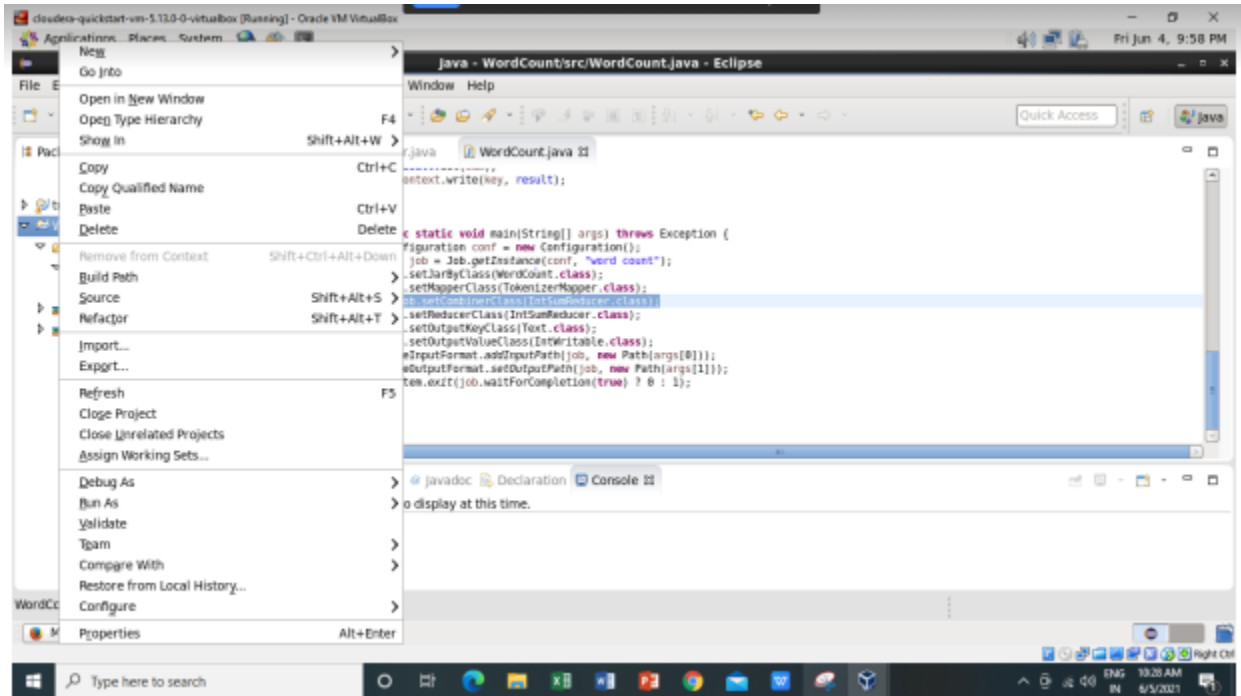
Name:  
Roll No:



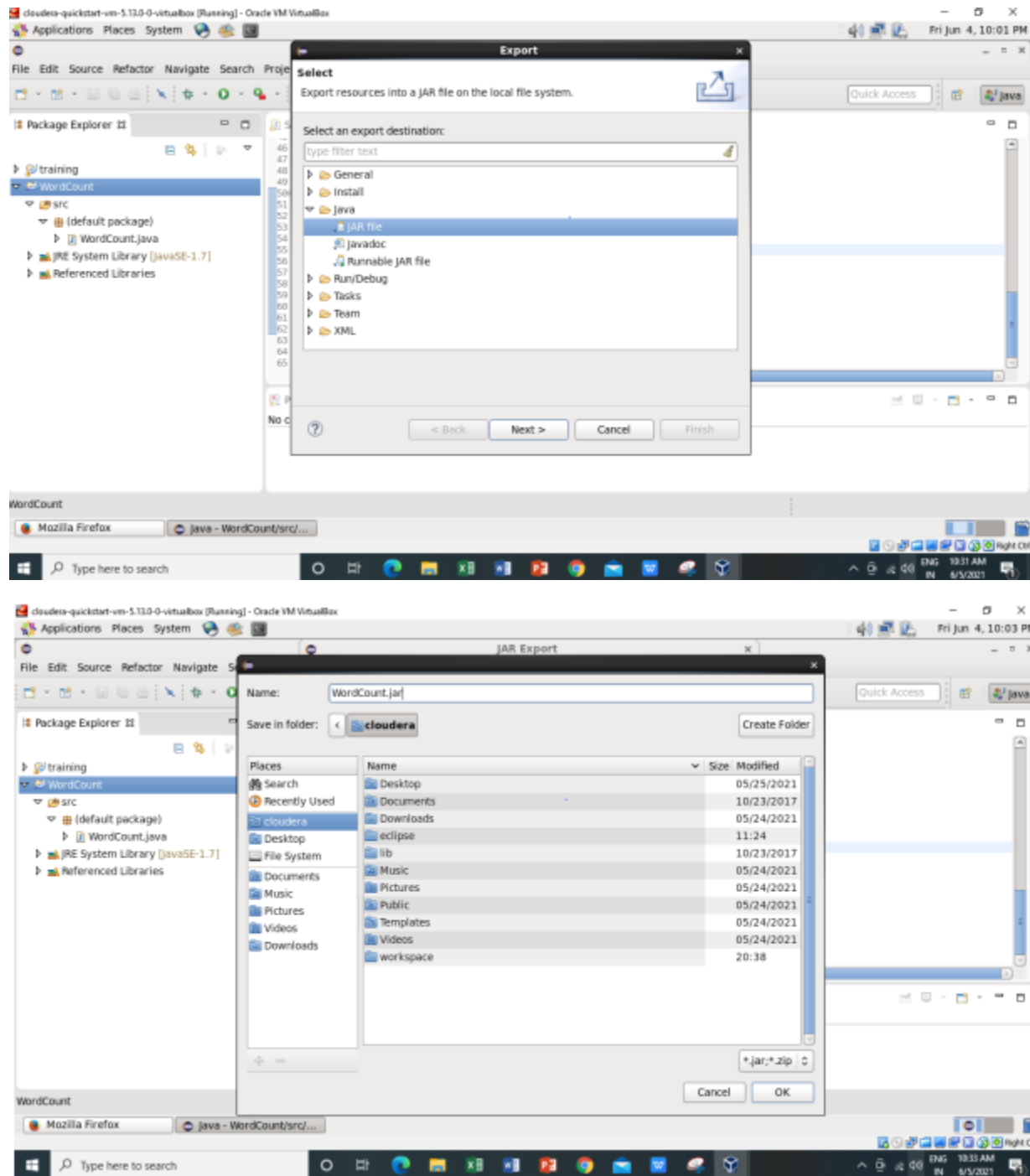


Name:  
Roll No:

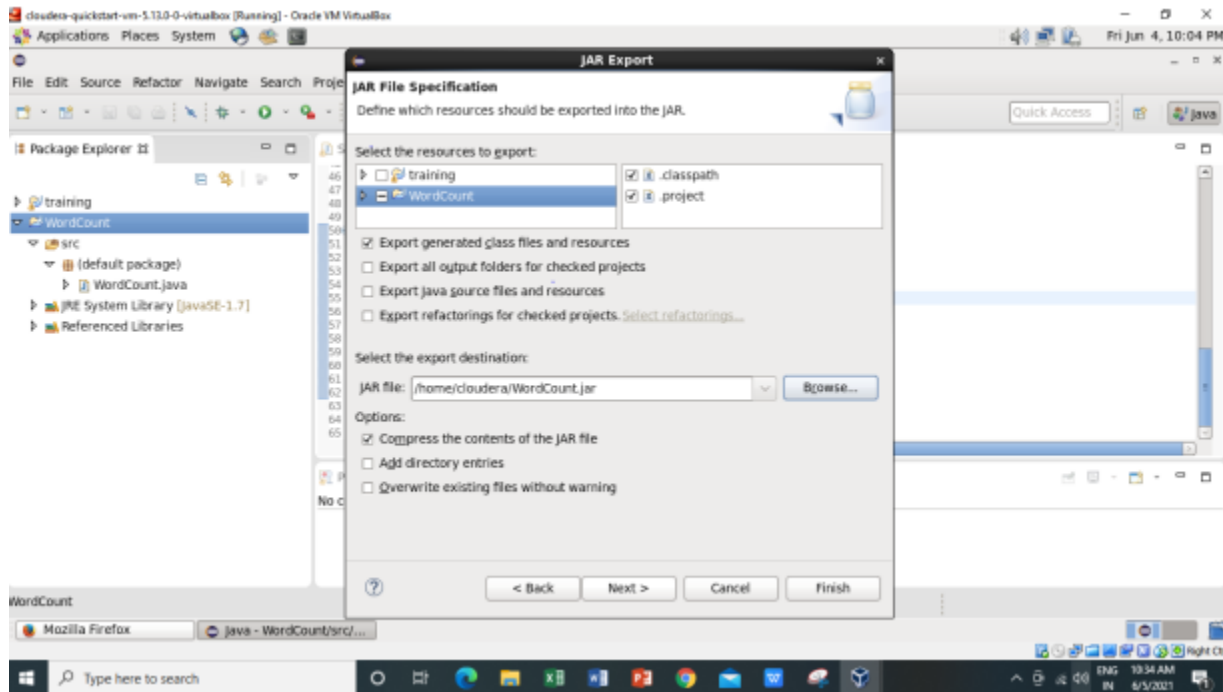
6) Right Click on the project name WordCount -> Export -> Java -> JAR File -> Next -> for select the export destination for JAR file: browse -> Name : WordCount.jar -> save in folder -> cloudera -> Finish -> OK



Name:  
Roll No:

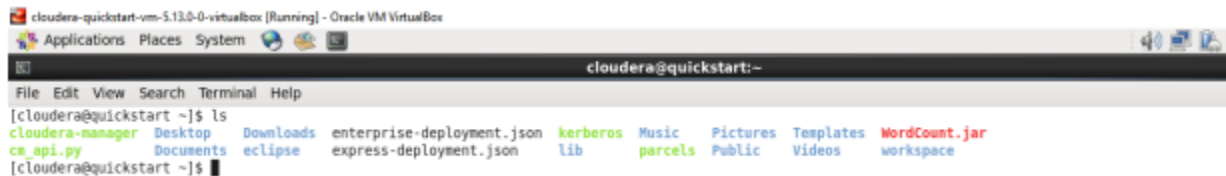


Name:  
Roll No:



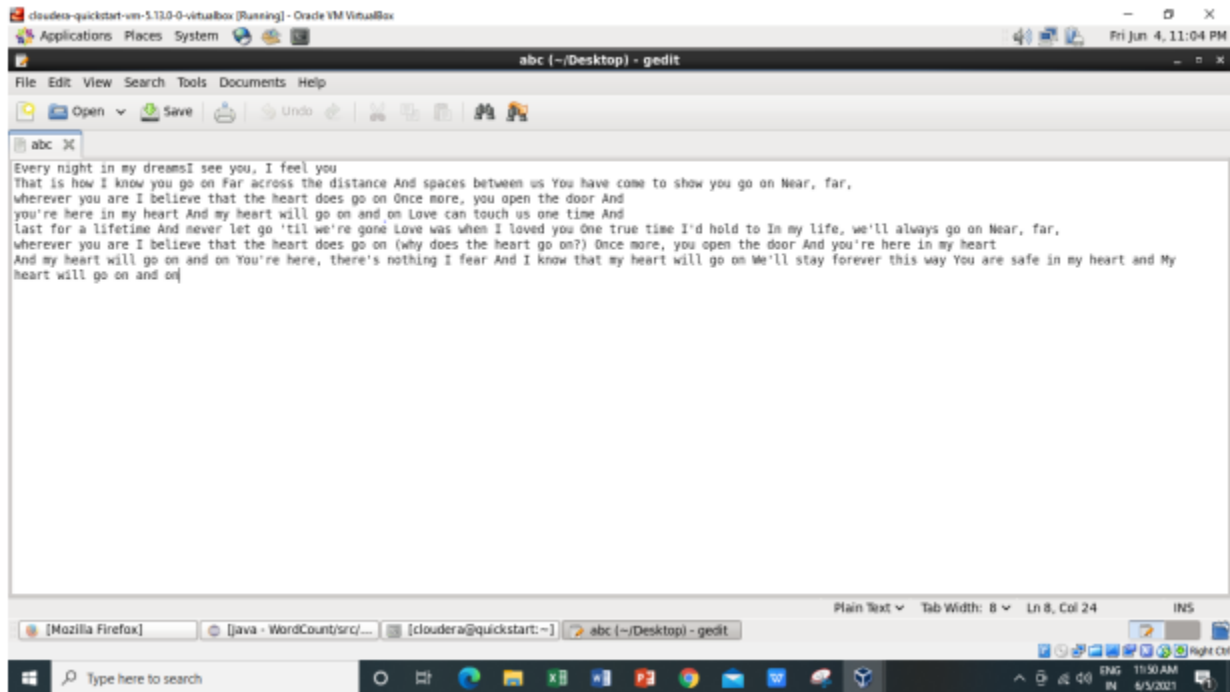
7) Verify jar file from terminal by using Open terminal & type “ls” There it will show WordCount.jar  
Check current working directory ->pwd

```
[cloudera@quickstart ~]$ pwd
/home/cloudera
[cloudera@quickstart ~]$
```



8) We need to create an input file in local file system  
Creating an input file named as “abc”.

Name:  
Roll No:



Here listing all the directory present in hdfs using `hdfs dfs -ls /` command

```
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 10 items
-rw-r--r-- 1 cloudera supergroup          27 2021-05-24 12:04 /Sample 01
drwxrwxrwx - hdfs supergroup              0 2017-10-23 09:15 /benchmarks
drwxr-xr-x - cloudera supergroup           0 2021-05-24 13:58 /forcopy
drwxr-xr-x - hbase supergroup              0 2021-06-04 07:57 /hbase
drwxr-xr-x - cloudera supergroup           0 2021-05-24 13:20 /newdir
drwxr-xr-x - cloudera supergroup           0 2021-05-24 13:36 /rjc
drwxr-xr-x - cloudera supergroup           0 2021-05-24 13:55 /solr
drwxrwxrwt - hdfs supergroup               0 2021-05-24 10:39 /tmp
drwxr-xr-x - hdfs supergroup               0 2017-10-23 09:17 /user
drwxr-xr-x - hdfs supergroup               0 2017-10-23 09:17 /var
[cloudera@quickstart ~]$
```

9) Now we have to move this input file to hdfs. For this we create a directory on hdfs using command `hdfs dfs -mkdir /inputnew`.

```
[cloudera@quickstart ~]$ hdfs dfs -mkdir /inputdir
[cloudera@quickstart ~]$
```

Then we can verify whether this directory is created or not using `ls` command `hdfs dfs -ls /`

**Roll No:**

Move the input file to this directory created in hdfs by using either put command or copyFromLocal command.

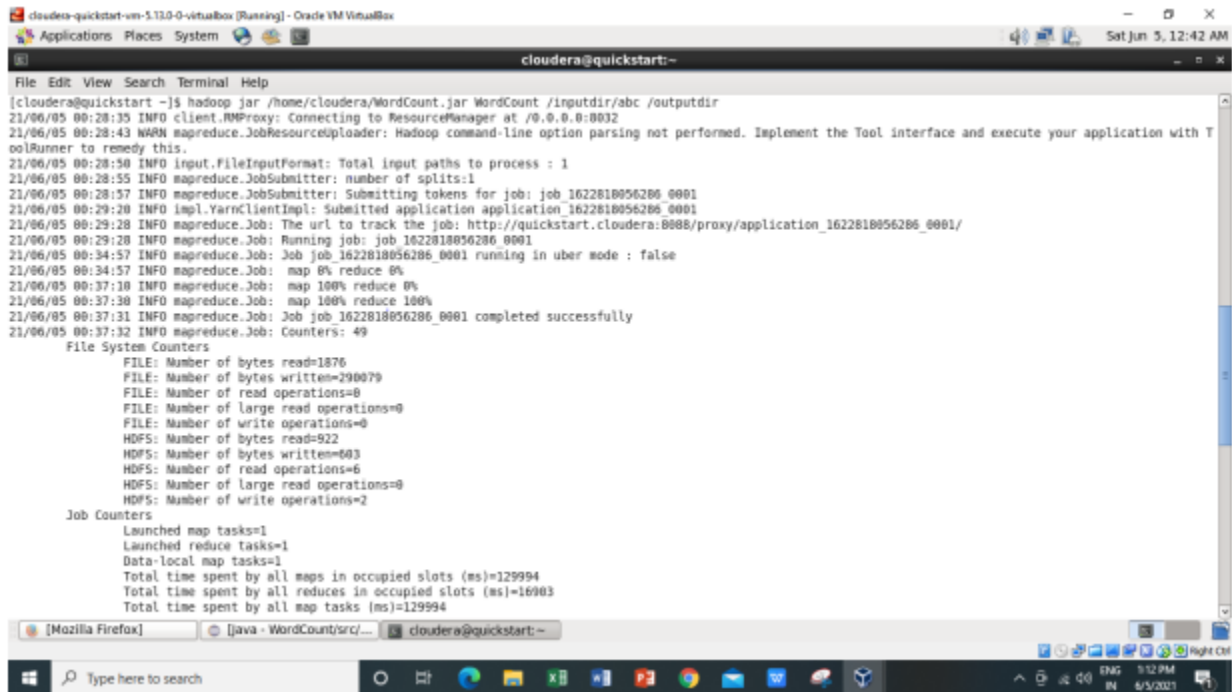
Now checking whether the “abc” present in /inputdir directory of hdfs or not using hdfs dfs -ls /inputdir command

As we can see “abc” file is present in /inputdir directory of hdfs. Now we will see the content of this file using `hdfs dfs -cat /inputdir/abc` command

10) Running Mapreduce Program on Hadoop, syntax is `hadoop jar jarFileName.jar ClassName /InputFileAddress /outputdir`

BDT

Name:  
Roll No:

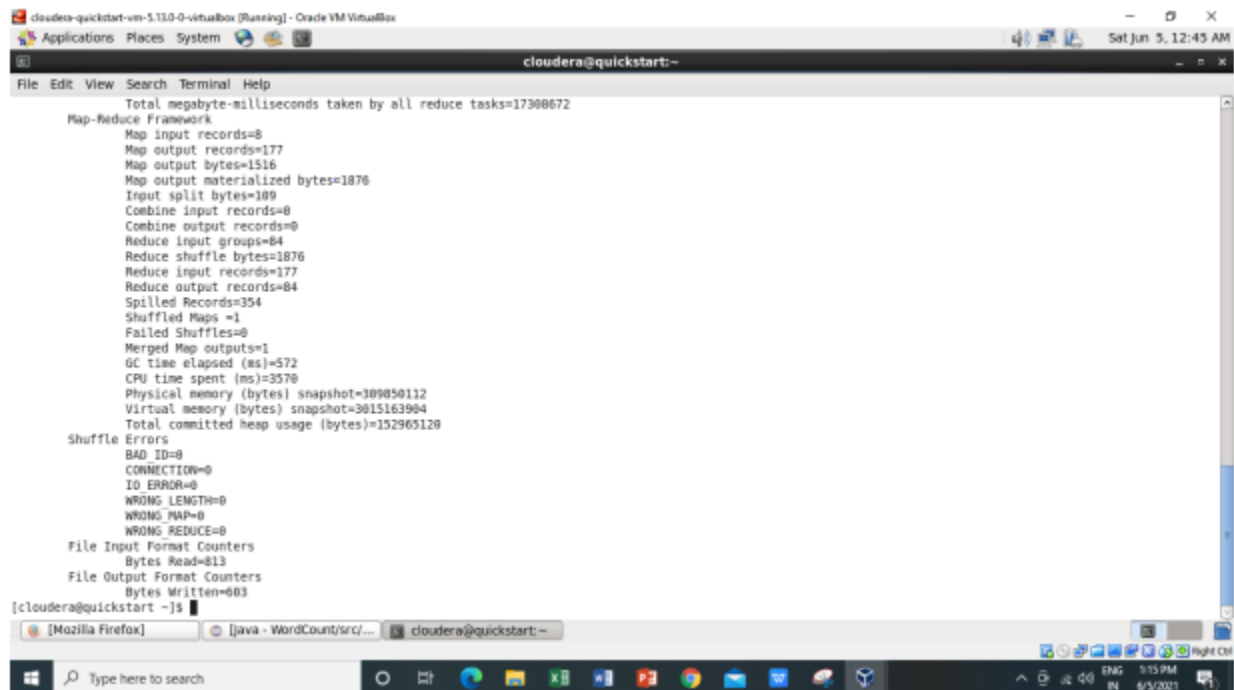


```
cloudera@quickstart:~$ hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/abc /outputdir
21/06/05 00:28:35 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/06/05 00:28:43 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
21/06/05 00:28:50 INFO input.FileInputFormat: Total input paths to process : 1
21/06/05 00:28:55 INFO mapreduce.JobSubmitter: number of splits:1
21/06/05 00:28:57 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1622818056286_0001
21/06/05 00:29:20 INFO impl.YarnClientImpl: Submitted application application_1622818056286_0001
21/06/05 00:29:28 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1622818056286_0001/
21/06/05 00:29:28 INFO mapreduce.Job: Running job: job_1622818056286_0001
21/06/05 00:34:57 INFO mapreduce.Job: Job job_1622818056286_0001 running in uber mode : false
21/06/05 00:34:57 INFO mapreduce.Job: map 0% reduce 0%
21/06/05 00:37:10 INFO mapreduce.Job: map 100% reduce 0%
21/06/05 00:37:10 INFO mapreduce.Job: map 100% reduce 100%
21/06/05 00:37:31 INFO mapreduce.Job: Job job_1622818056286_0001 completed successfully
21/06/05 00:37:32 INFO mapreduce.Job: Counters: 49

File System Counters
  FILE: Number of bytes read=1876
  FILE: Number of bytes written=290079
  FILE: Number of read operations=0
  FILE: Number of large read operations=0
  FILE: Number of write operations=0
  HDFS: Number of bytes read=922
  HDFS: Number of bytes written=603
  HDFS: Number of read operations=6
  HDFS: Number of large read operations=0
  HDFS: Number of write operations=2

Job Counters
  Launched map tasks=1
  Launched reduce tasks=1
  Data-local map tasks=1
  Total time spent by all maps in occupied slots (ms)=129994
  Total time spent by all reduces in occupied slots (ms)=16903
  Total time spent by all map tasks (ms)=129994
```

## Map-Reduce Framework



```
cloudera@quickstart:~$ hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/abc /outputdir
Total megabyte-milliseconds taken by all reduce tasks=17300672

Map-Reduce Framework
  Map input records=8
  Map output records=177
  Map output bytes=1516
  Map output materialized bytes=1876
  Input split bytes=109
  Combine input records=0
  Combine output records=0
  Reduce input groups=84
  Reduce shuffle bytes=1876
  Reduce input records=177
  Reduce output records=84
  Spilled Records=354
  Shuffled Maps =1
  Failed Shuffles=0
  Merged Map outputs=1
  GC time elapsed (ms)=572
  CPU time spent (ms)=3570
  Physical memory (bytes) snapshot=309050112
  Virtual memory (bytes) snapshot=3015163904
  Total committed heap usage (bytes)=152903120

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=813
File Output Format Counters
  Bytes Written=603

[cloudera@quickstart ~]$
```

As we can see in the above output,

Combine input records=0

Combine output records=0



Name:  
Roll No:

We are getting this because we have commented the Combiner line in main function.

And Reduce shuffle bytes coming as,

Reduce shuffle bytes=1876

So when we are not using combiner 1876 bytes acting as an input for the reducer.

11) Then we can verify the content of outputdir directory and in that part-r file has the actual output by using the command `Hdfs dfs -cat /outputdir/part-r-00000` This will give us final output.

The same file can also be accessed using a browser. For every execution of this program we need to delete the output directory or give a new name to the output directory every time.

1st we are checking whether the outputdir directory is created in hdfs or not using command

**hdfs dfs -ls /**

```
bytes written=000
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 12 items
-rw-r--r-- 1 cloudera supergroup 27 2021-05-24 12:04 /Sample_01
drwxrwxrwx - hdfs supergroup 0 2017-10-23 09:15 /benchmarks
drwxr-xr-x - cloudera supergroup 0 2021-05-24 13:58 /forcopy
drwxr-xr-x - hbase supergroup 0 2021-06-04 07:57 /hbase
drwxr-xr-x - cloudera supergroup 0 2021-06-05 00:06 /inputdir
drwxr-xr-x - cloudera supergroup 0 2021-05-24 13:20 /newdir
drwxr-xr-x - cloudera supergroup 0 2021-06-05 00:37 /outputdir
drwxr-xr-x - cloudera supergroup 0 2021-05-24 13:36 /rjc
drwxr-xr-x - cloudera supergroup 0 2021-05-24 13:55 /solr
drwxrwxrwt - hdfs supergroup 0 2021-05-24 10:39 /tmp
drwxr-xr-x - hdfs supergroup 0 2017-10-23 09:17 /user
drwxr-xr-x - hdfs supergroup 0 2017-10-23 09:17 /var
[cloudera@quickstart ~]$
```

Now let's check what we have inside this **outputdir** directory using command as **hdfs dfs -ls**

**/outputdir**

```
[cloudera@quickstart ~]$ hdfs dfs -ls /outputdir
Found 2 items
-rw-r--r-- 1 cloudera supergroup 0 2021-06-05 00:37 /outputdir/_SUCCESS
-rw-r--r-- 1 cloudera supergroup 603 2021-06-05 00:37 /outputdir/part-r-00000
[cloudera@quickstart ~]$
```

Now we want to read the content of the **part-r-00000** file which present inside the **outputdir** using command **hdfs dfs -cat /outputdir/part-r-00000**

**Name:**  
**Roll No:**

```
[cloudera@quickstart ~]$ hdfs dfs -cat /outputdir/part-r-00000
'til 1
(why 1
And 8
Every 1
Far 1
I 7
I'd 1
In 1
Love 2
My 1
Near, 2
Once 2
One 1
That 1
We'll 1
You 2
You're 1
a 1
across 1
always 1
and 4
are 3
believe 2
between 1
can 1
come 1
distance 1
does 3
door 2
dreamsI 1
```

**Name:**  
**Roll No:**

```
more, 2
my 8
never 1
night 1
nothing 1
on 12
on?) 1
one 1
open 2
safe 1
see 1
show 1
spaces 1
stay 1
that 3
the 6
there's 1
this 1
time 2
to 2
touch 1
true 1
us 2
was 1
way 1
we'll 1
we're 1
when 1
wherever 2
will 4
you 8
you're 2
you, 1
[cloudera@quickstart ~]$
```

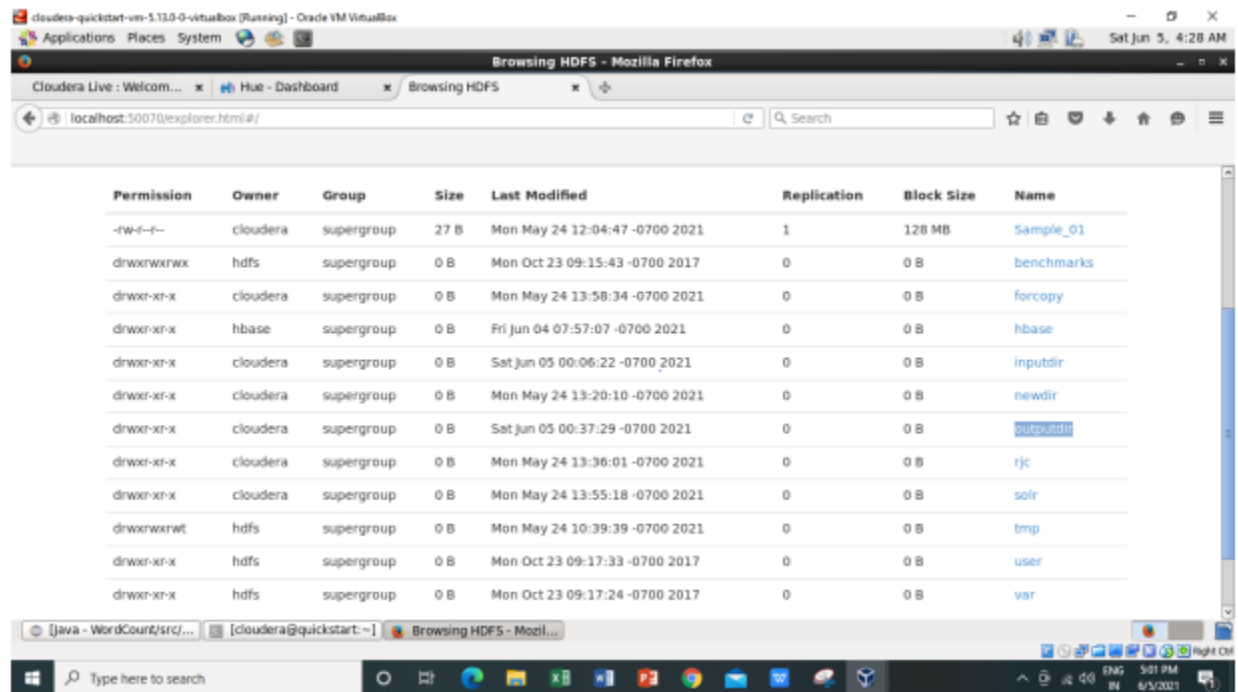
It will give the count of number of times each word has occurred as output.

**12) The same file can also be accessed using a browser.**

Browse the Directory by

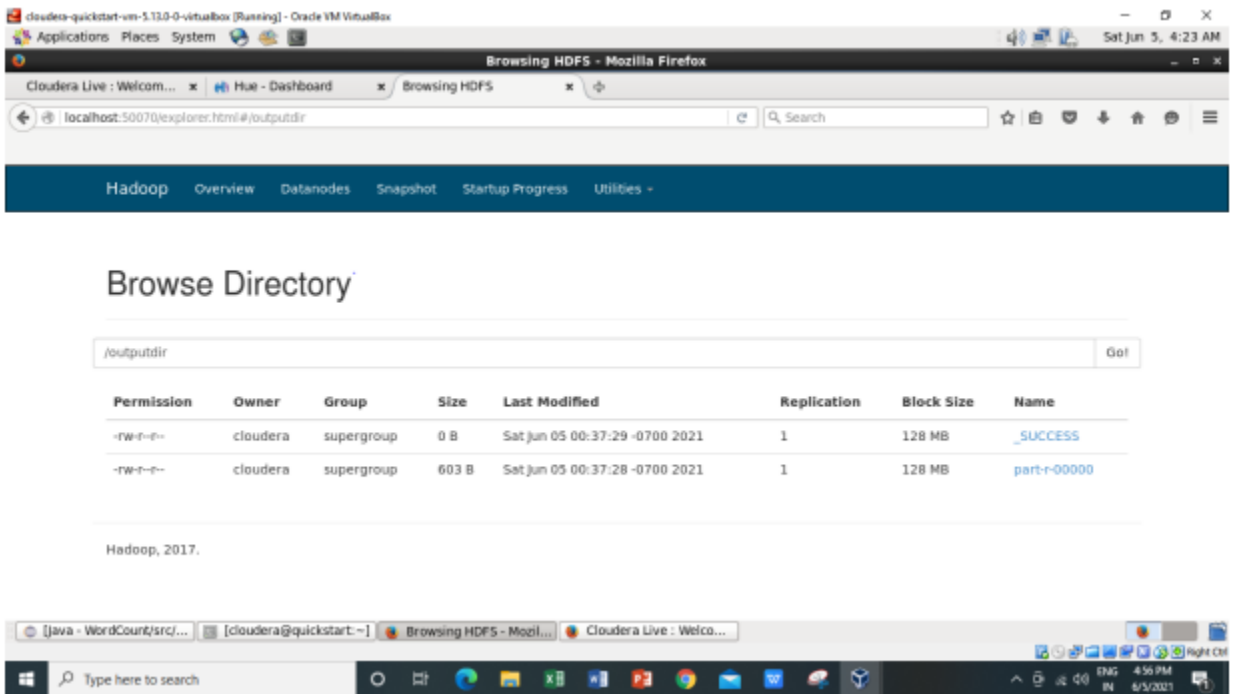
**Hadoop->HDFS Namenode->Utilities ->Browse the file system**

Name:  
Roll No:



The screenshot shows the Hue web interface for browsing HDFS. The browser address bar displays 'localhost:50070/explorer.html#/'. The main content area shows a table of files and directories in the HDFS filesystem.

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	cloudera	supergroup	27 B	Mon May 24 12:04:47 -0700 2021	1	128 MB	Sample_01
drwxrwxrwx	hdfs	supergroup	0 B	Mon Oct 23 09:15:43 -0700 2017	0	0 B	benchmarks
drwxr-xr-x	cloudera	supergroup	0 B	Mon May 24 13:58:34 -0700 2021	0	0 B	forcopy
drwxr-xr-x	hbase	supergroup	0 B	Fri Jun 04 07:57:07 -0700 2021	0	0 B	hbase
drwxr-xr-x	cloudera	supergroup	0 B	Sat Jun 05 00:06:22 -0700 2021	0	0 B	inputdir
drwxr-xr-x	cloudera	supergroup	0 B	Mon May 24 13:20:10 -0700 2021	0	0 B	newdir
drwxr-xr-x	cloudera	supergroup	0 B	Sat Jun 05 00:37:29 -0700 2021	0	0 B	outputdir
drwxr-xr-x	cloudera	supergroup	0 B	Mon May 24 13:36:01 -0700 2021	0	0 B	rjc
drwxr-xr-x	cloudera	supergroup	0 B	Mon May 24 13:55:18 -0700 2021	0	0 B	solr
drwxrwxrwt	hdfs	supergroup	0 B	Mon May 24 10:39:39 -0700 2021	0	0 B	tmp
drwxr-xr-x	hdfs	supergroup	0 B	Mon Oct 23 09:17:33 -0700 2017	0	0 B	user
drwxr-xr-x	hdfs	supergroup	0 B	Mon Oct 23 09:17:24 -0700 2017	0	0 B	var



The screenshot shows the Hue web interface for browsing HDFS, specifically the 'outputdir' directory. The browser address bar displays 'localhost:50070/explorer.html#/outputdir'. The main content area shows a table of files and directories in the HDFS filesystem.

### Browse Directory

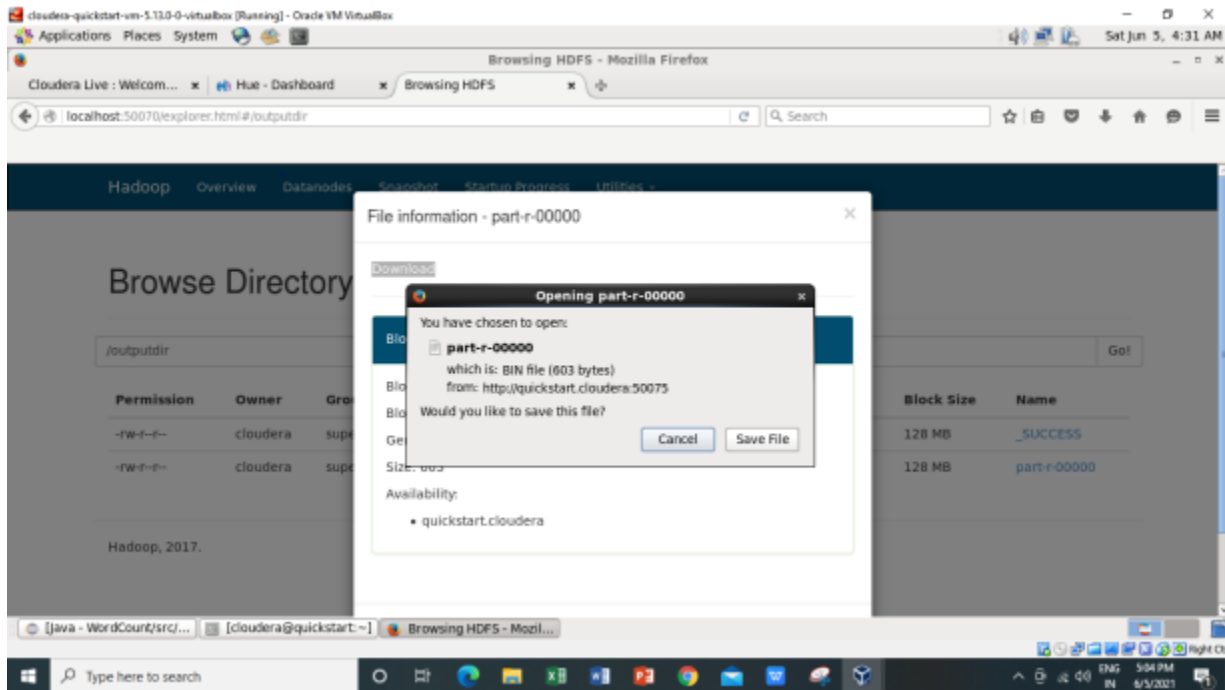
/outputdir

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	cloudera	supergroup	0 B	Sat Jun 05 00:37:29 -0700 2021	1	128 MB	_SUCCESS
-rw-r--r--	cloudera	supergroup	603 B	Sat Jun 05 00:37:28 -0700 2021	1	128 MB	part-r-00000

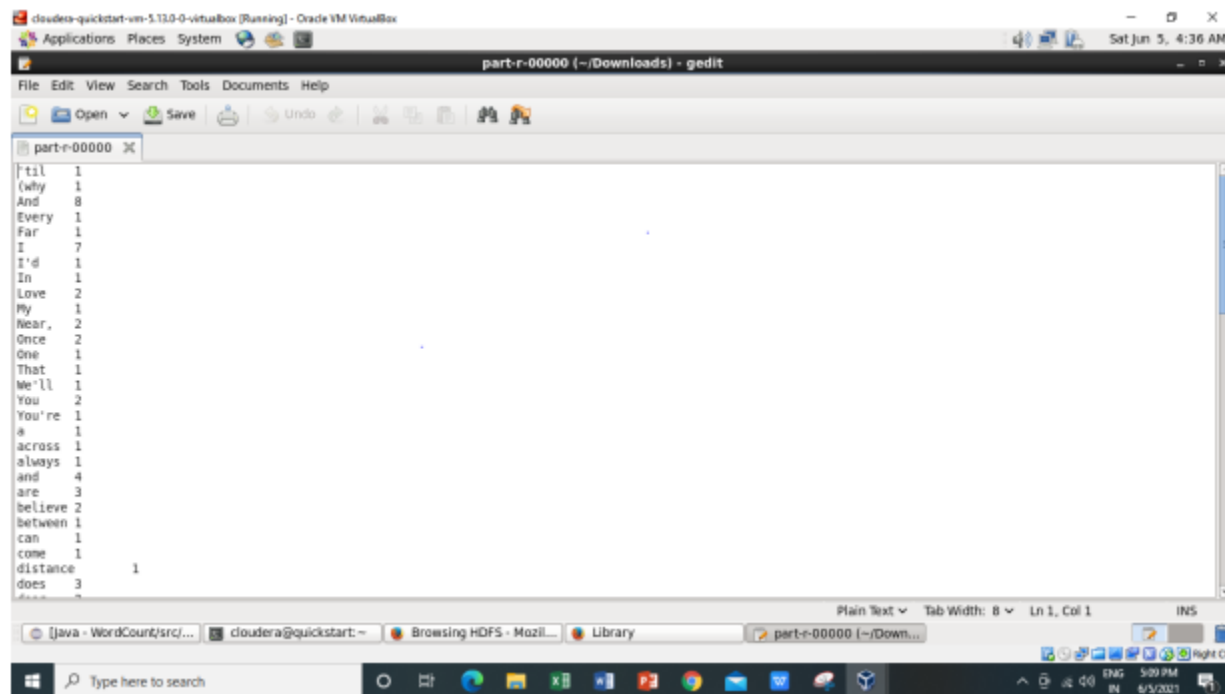
Hadoop, 2017.

Now downloading the **part-r-00000** file.

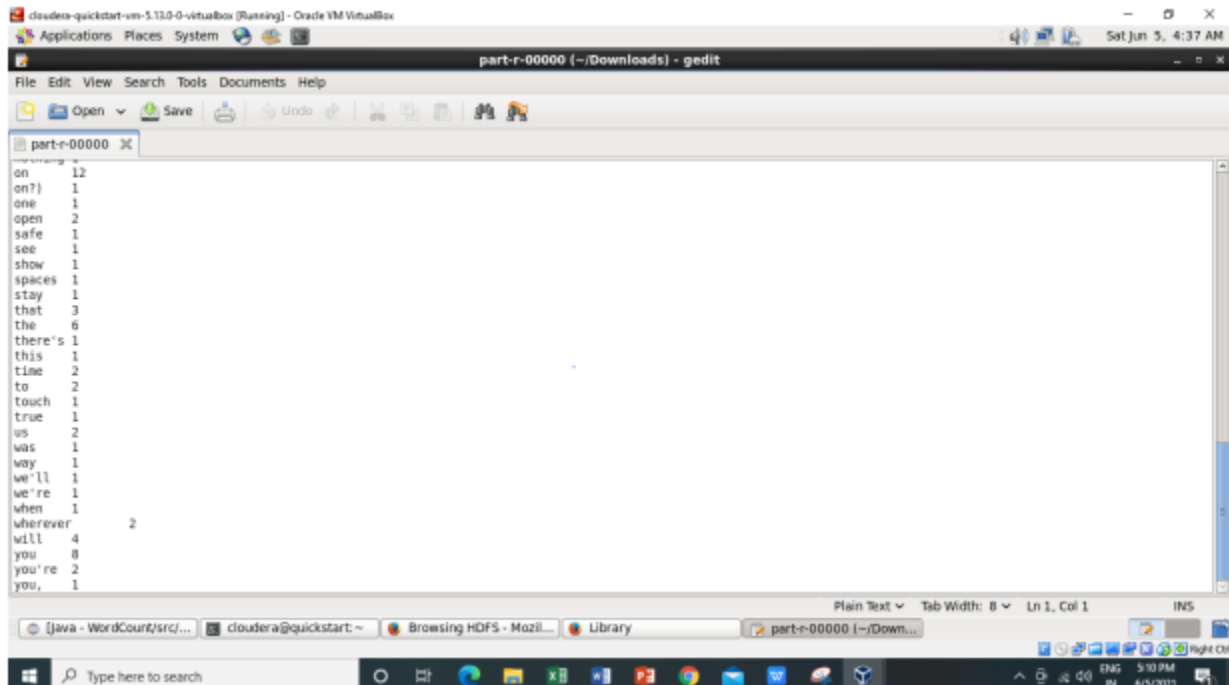
Name:  
Roll No:



Inside the **part-r-00000** file it will have the same output as we are getting after executing using command **hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/abc /op1**



Name:  
Roll No:



The screenshot shows a virtual machine window titled "cloudera-quickstart-vm-5.12.0-0-virtualbox [Running] - Oracle VM VirtualBox". Inside the VM, a gedit editor window titled "part-r-00000 (~/.Downloads) - gedit" is open. The editor contains a list of words and their corresponding counts, such as "on 12", "on7) 1", "one 1", "open 2", "safe 1", "see 1", "show 1", "spaces 1", "stay 1", "that 3", "the 6", "there's 1", "this 1", "time 2", "to 2", "touch 1", "true 1", "us 2", "was 1", "way 1", "we'll 1", "we're 1", "when 1", "wherever 2", "will 4", "you 8", "you're 2", and "you, 1". The bottom status bar of the gedit window indicates "Plain Text", "Tab Width: 8", "Ln 1, Col 1", and "INS". The taskbar at the bottom of the VM shows various application icons and the system clock.

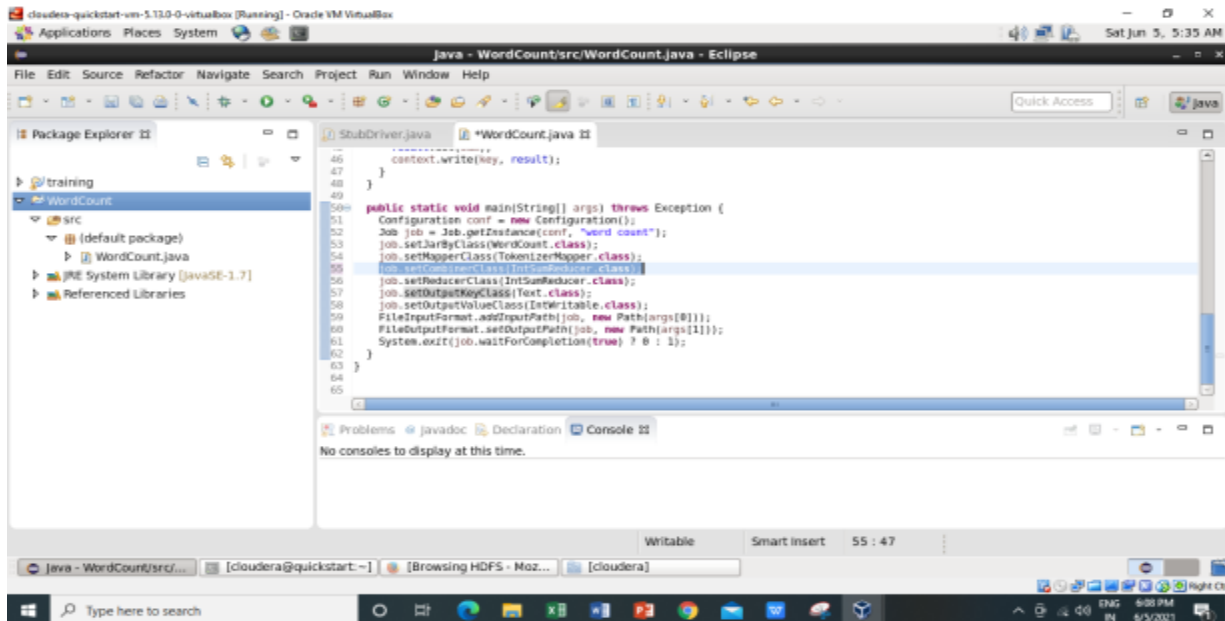
For every execution of this program we need to delete the output directory or give a new name to the output directory every time.

### Implementation of WordCount problem using Hadoop MapReduce (With Combiner) in Eclipse:

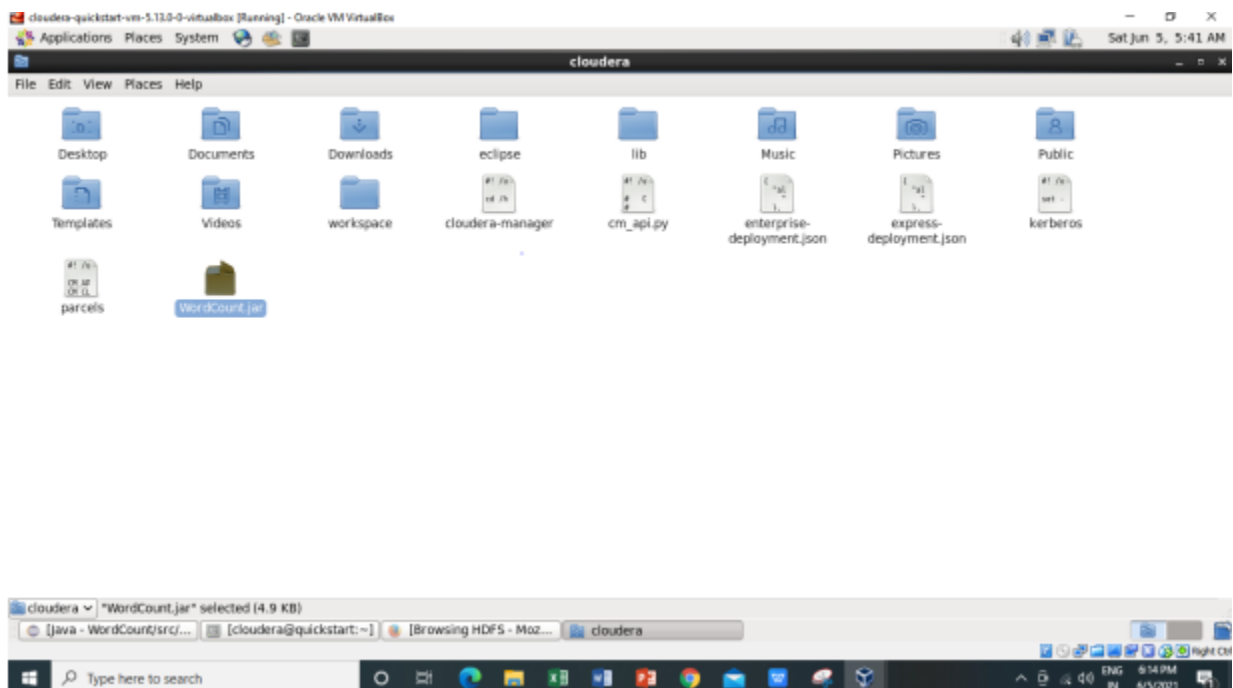
1) We will perform the same steps as we have done above for WordCount (without using combiner) in that we just uncommenting the combiner line in main function.



Name:  
Roll No:

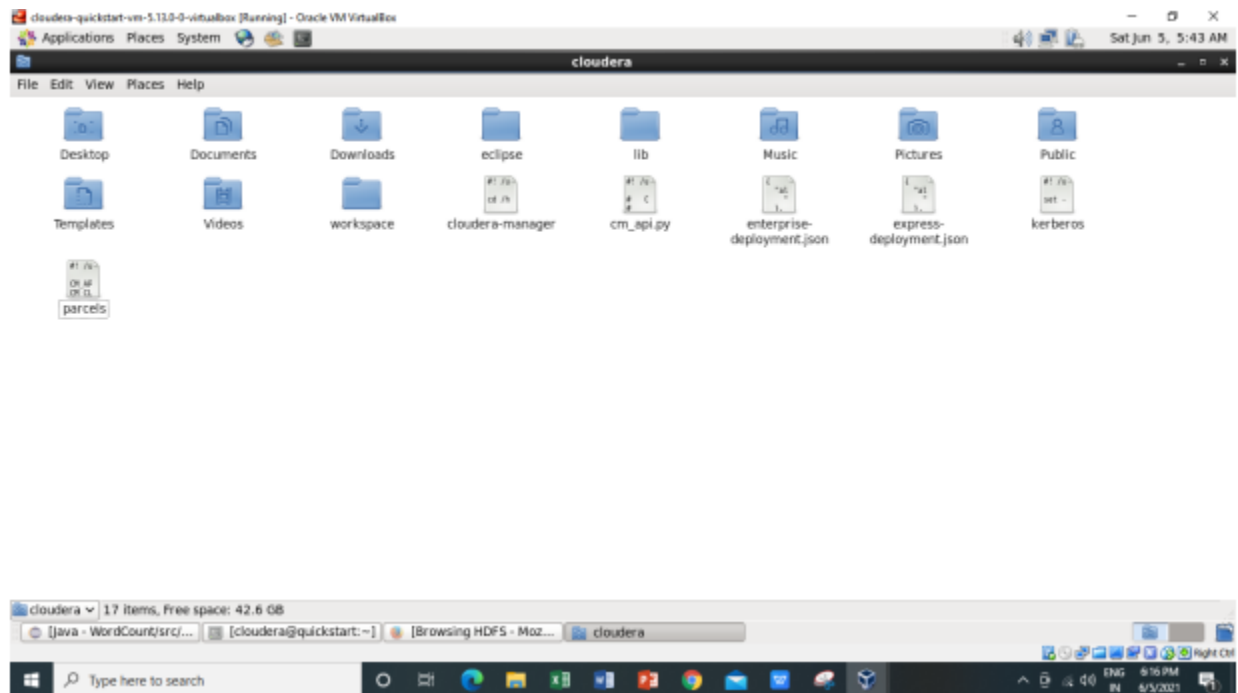


2) And will delete the WordCount.jar file in which all jar files are present from /home/cloudera.

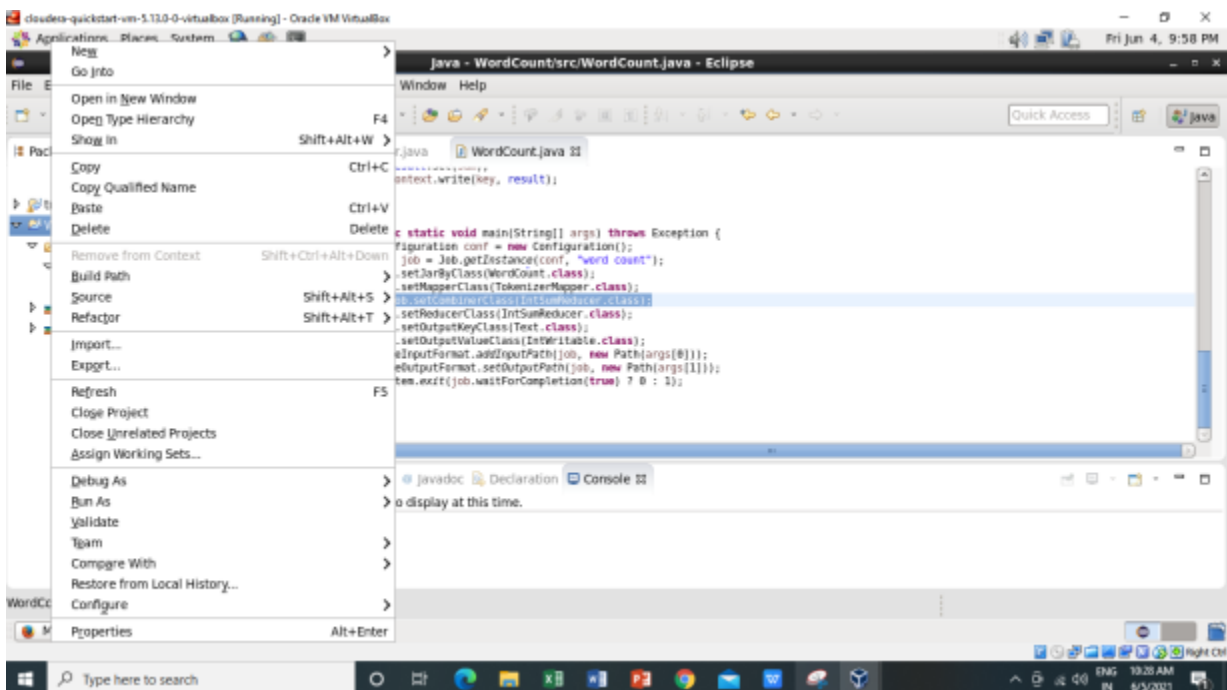


We have successfully deleted the WordCount.jar file.

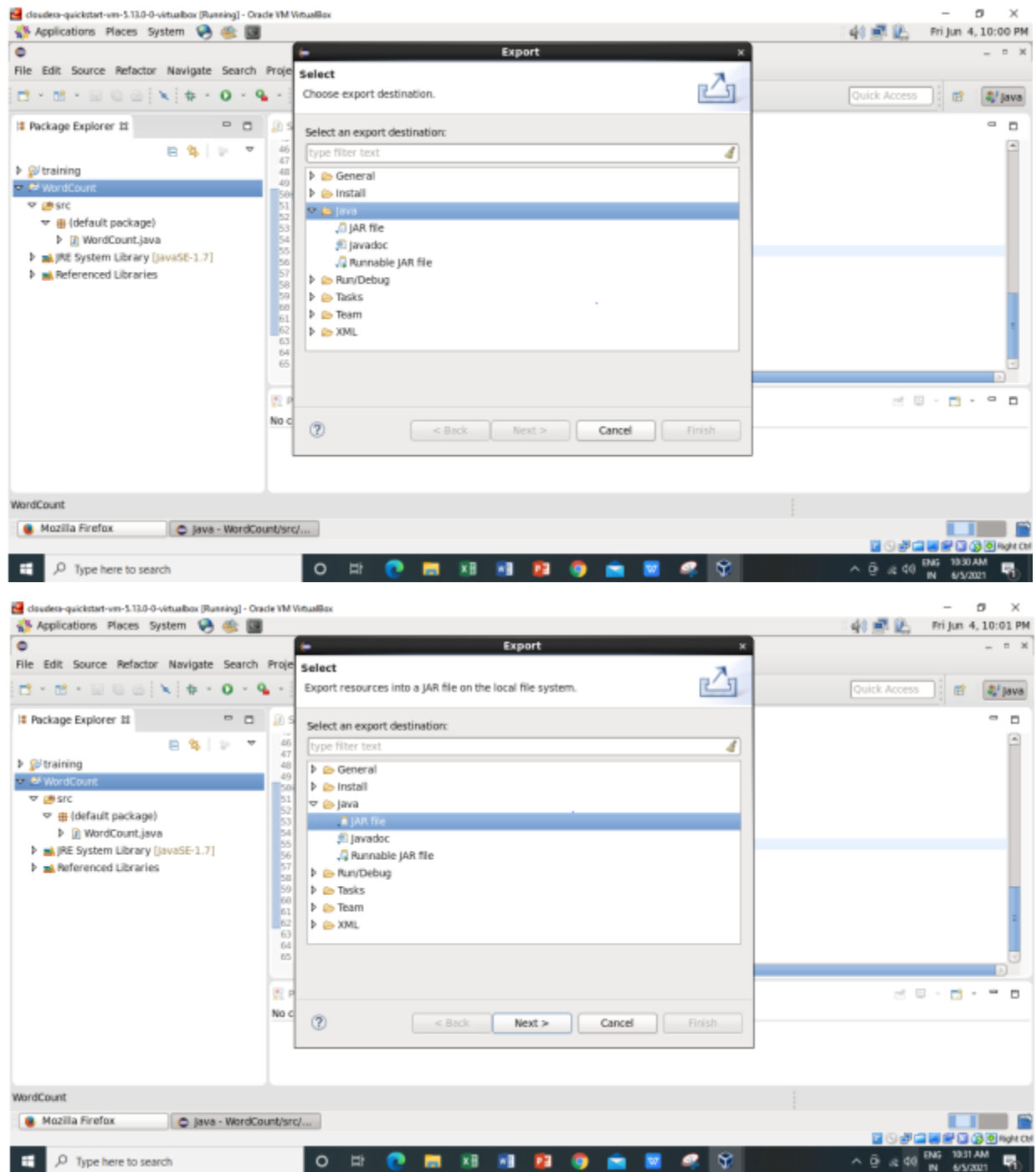
Name:  
Roll No:



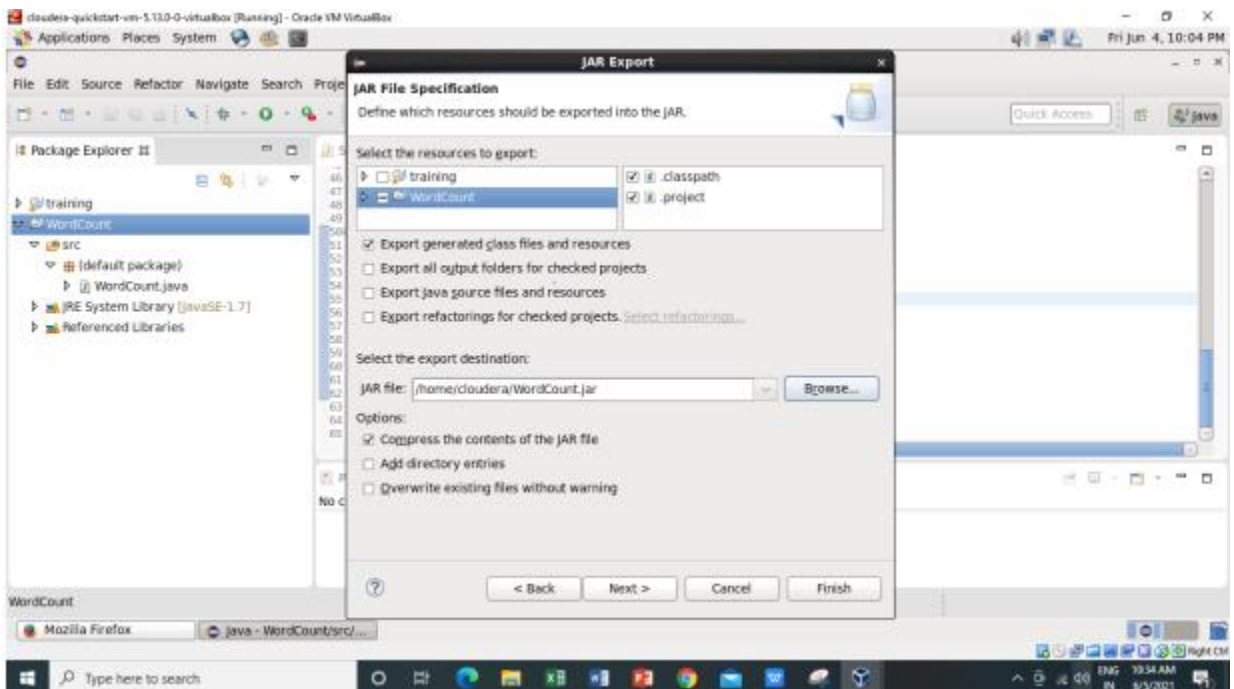
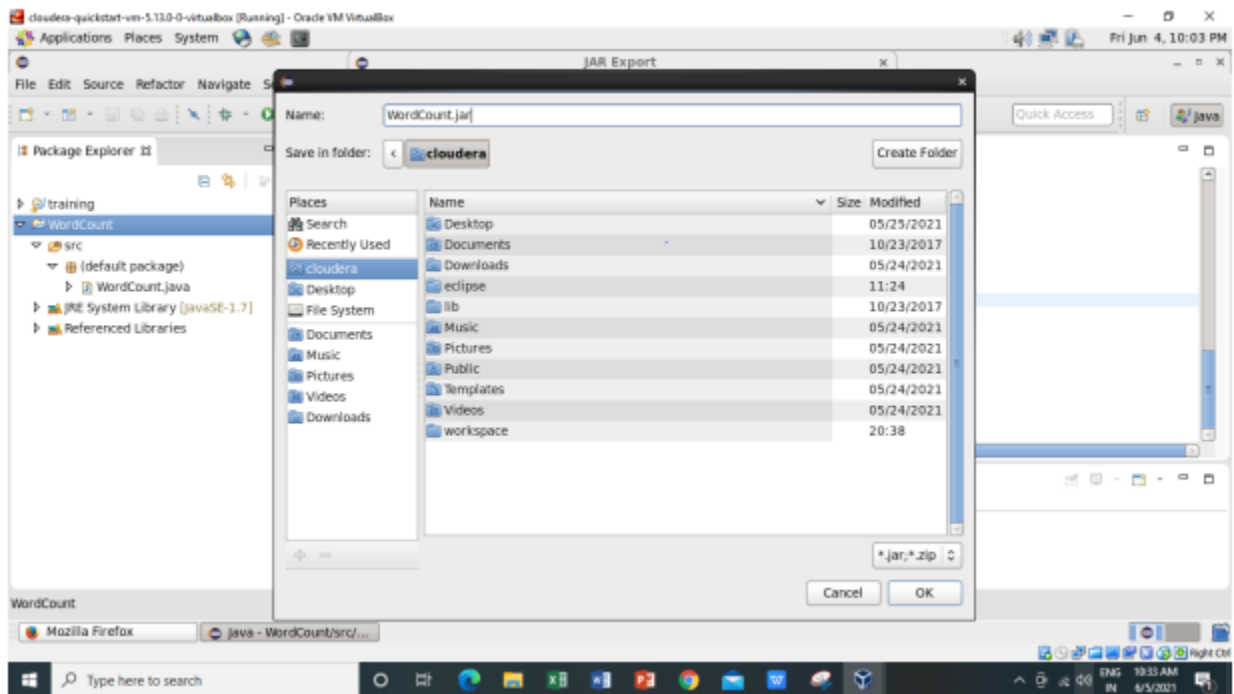
- 3) Now exporting the jar files Right Click on the project name WordCount -> Export -> Java -> JAR File -> Next -> for select the export destination for JAR file: browse -> Name : WordCount.jar -> save in folder -> cloudera -> Finish -> OK



Name:  
Roll No:



Name:  
Roll No:



4) Now checking the WordCount.jar file is created or not using -ls command

```
[cloudera@quickstart ~]$ ls
cloudera-manager Desktop Downloads enterprise-deployment.json kerberos Music Pictures Templates WordCount.jar
cm_api.py Documents eclipse express-deployment.json lib parcels Public Videos workspace
[cloudera@quickstart ~]$
```

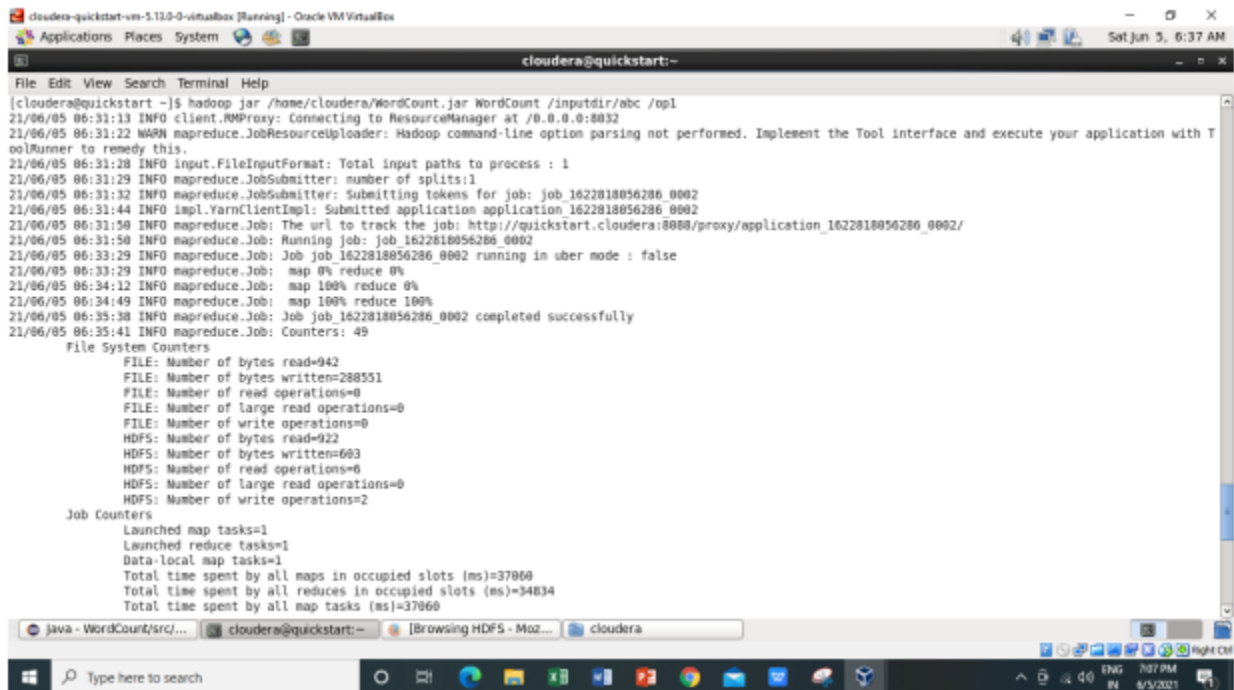
**Name:**  
**Roll No:**

5) Running Mapreduce Program on Hadoop, syntax is `hadoop jar jarFileName.jar ClassName /InputFileAddress /outputdir`

**i.e. `hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/abc /op1`**

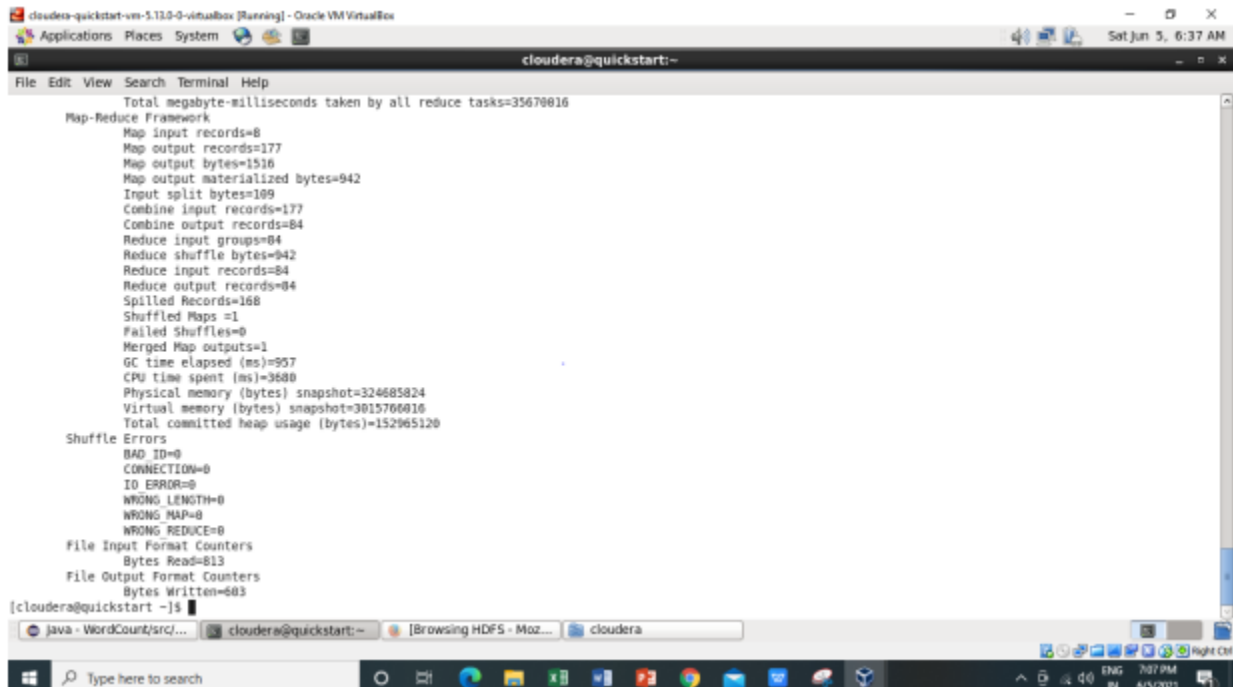
here I am using the same input file 'abc' which I have created earlier for WordCount

example (Without Combiner). **For every execution of this program we need to delete the output directory or give a new name to the output directory every time.** So here I am giving the new name to the output directory as 'op1'.



```
cloudera@quickstart:~$ hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/abc /op1
21/06/05 06:31:13 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/06/05 06:31:22 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
21/06/05 06:31:28 INFO Input.FileInputFormat: Total input paths to process : 1
21/06/05 06:31:29 INFO mapreduce.JobSubmitter: number of splits:1
21/06/05 06:31:32 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1622818056286_0002
21/06/05 06:31:44 INFO impl.YarnClientImpl: Submitted application application_1622818056286_0002
21/06/05 06:31:50 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1622818056286_0002/
21/06/05 06:31:50 INFO mapreduce.Job: Running job: job_1622818056286_0002
21/06/05 06:33:29 INFO mapreduce.Job: Job job_1622818056286_0002 running in uber mode : false
21/06/05 06:33:29 INFO mapreduce.Job:  map 0% reduce 0%
21/06/05 06:34:12 INFO mapreduce.Job:  map 100% reduce 0%
21/06/05 06:34:49 INFO mapreduce.Job:  map 100% reduce 100%
21/06/05 06:35:38 INFO mapreduce.Job: Job job_1622818056286_0002 completed successfully
21/06/05 06:35:41 INFO mapreduce.Job: Counters: 49
  File System Counters
    FILE: Number of bytes read=942
    FILE: Number of bytes written=288551
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
    HDFS: Number of bytes read=922
    HDFS: Number of bytes written=693
    HDFS: Number of read operations=6
    HDFS: Number of large read operations=0
    HDFS: Number of write operations=2
  Job Counters
    Launched map tasks=1
    Launched reduce tasks=1
    Data-local map tasks=1
    Total time spent by all maps in occupied slots (ms)=37868
    Total time spent by all reduces in occupied slots (ms)=34834
    Total time spent by all map tasks (ms)=37868
```

Name:  
Roll No:



```
cloudera-quickstart-vm-5.13.0-0-virtualbox [Running] - Oracle VM VirtualBox
Applications Places System
cloudera@quickstart:~$
File Edit View Search Terminal Help
Total megabyte-milliseconds taken by all reduce tasks=35670016
Map-Reduce Framework
  Map input records=8
  Map output records=177
  Map output bytes=1516
  Map output materialized bytes=942
  Input split bytes=109
  Combine input records=177
  Combine output records=84
  Reduce input groups=84
  Reduce shuffle bytes=942
  Reduce input records=84
  Reduce output records=84
  Spilled Records=168
  Shuffled Maps =1
  Failed Shuffles=0
  Merged Map outputs=1
  GC time elapsed (ms)=957
  CPU time spent (ms)=3680
  Physical memory (bytes) snapshot=324685824
  Virtual memory (bytes) snapshot=3015766016
  Total committed heap usage (bytes)=152965120
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=813
File Output Format Counters
  Bytes Written=603
[cloudera@quickstart ~]$
```

- As we can see from above image the the combiner input and output records coming out as,

**Combine input records=177**

**Combine output records=84**

- Earlier it was coming out as “zero” while executing WordCount (without combiner).

**Combine input records=0**

**Combine output records=0**

- And also here we are getting the Reduce Shuffle bytes as,

**Reduce shuffle bytes=942**

Earlier while executing WordCount (without combiner) it is coming out as,

**Reduce shuffle bytes=1876**

- So Combiner is used to save the Network Bandwidth. So for saving the Network bandwidth we make use of combiner. So instead of sending every word over the network what we do is we incorporate the logic of the reducer at the combiner side so that the less amount of information can be transmitted over the network.
- So when we are not using combiner 1876 bytes acting as an input for the reducer. And when we are making use of the combiner so 942 bytes acting as input for the reducer.

6) The same file can also be accessed using a browser.



Name:  
Roll No:

Browse the Directory by

Hadoop->HDFS Namenode->Utilities ->Browse the file system

The screenshot shows the Hadoop HDFS Namenode web interface in a Mozilla Firefox browser. The address bar shows the URL `localhost:50070/explorer.html#`. The page title is "Browse Directory". Below the title, there is a search bar and a "Go!" button. The main content area displays a table of files and directories in the HDFS file system.

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	cloudera	supergroup	27 B	Mon May 24 12:04:47 -0700 2021	1	128 MB	Sample_01
drwxrwxrwx	hdfs	supergroup	0 B	Mon Oct 23 09:15:43 -0700 2017	0	0 B	benchmarks
drwxr-xr-x	cloudera	supergroup	0 B	Mon May 24 13:58:34 -0700 2021	0	0 B	forcopy
drwxr-xr-x	hbase	supergroup	0 B	Fri Jun 04 07:57:07 -0700 2021	0	0 B	hbase
drwxr-xr-x	cloudera	supergroup	0 B	Sat Jun 05 00:06:22 -0700 2021	0	0 B	inputdir
drwxr-xr-x	cloudera	supergroup	0 B	Mon May 24 13:20:10 -0700 2021	0	0 B	newdir
drwxr-xr-x	cloudera	supergroup	0 B	Sat Jun 05 06:34:49 -0700 2021	1	128 MB	op1
drwxr-xr-x	cloudera	supergroup	0 B	Sat Jun 05 00:37:29 -0700 2021	0	0 B	outputdir

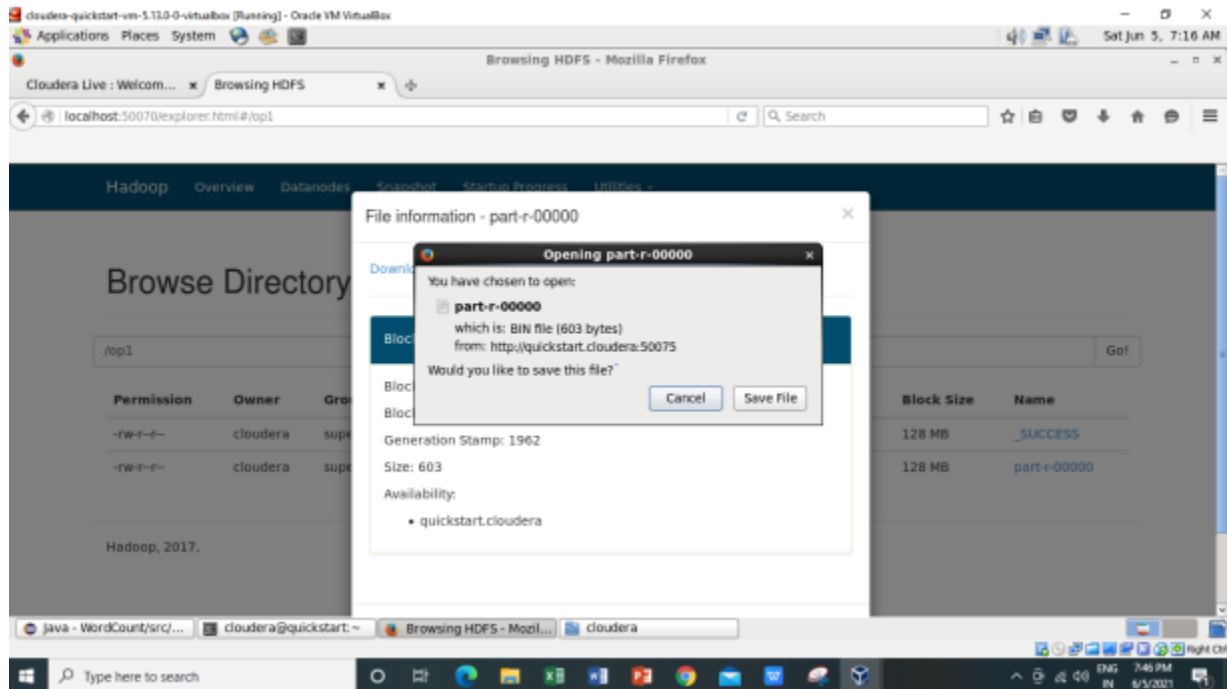
The screenshot shows the Hadoop HDFS Namenode web interface in a Mozilla Firefox browser. The address bar shows the URL `localhost:50070/explorer.html#/op1`. The page title is "Browse Directory". Below the title, there is a search bar and a "Go!" button. The main content area displays a table of files and directories in the HDFS file system.

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	cloudera	supergroup	0 B	Sat Jun 05 06:34:49 -0700 2021	1	128 MB	_SUCCESS
-rw-r--r--	cloudera	supergroup	603 B	Sat Jun 05 06:34:47 -0700 2021	1	128 MB	part-r-00000

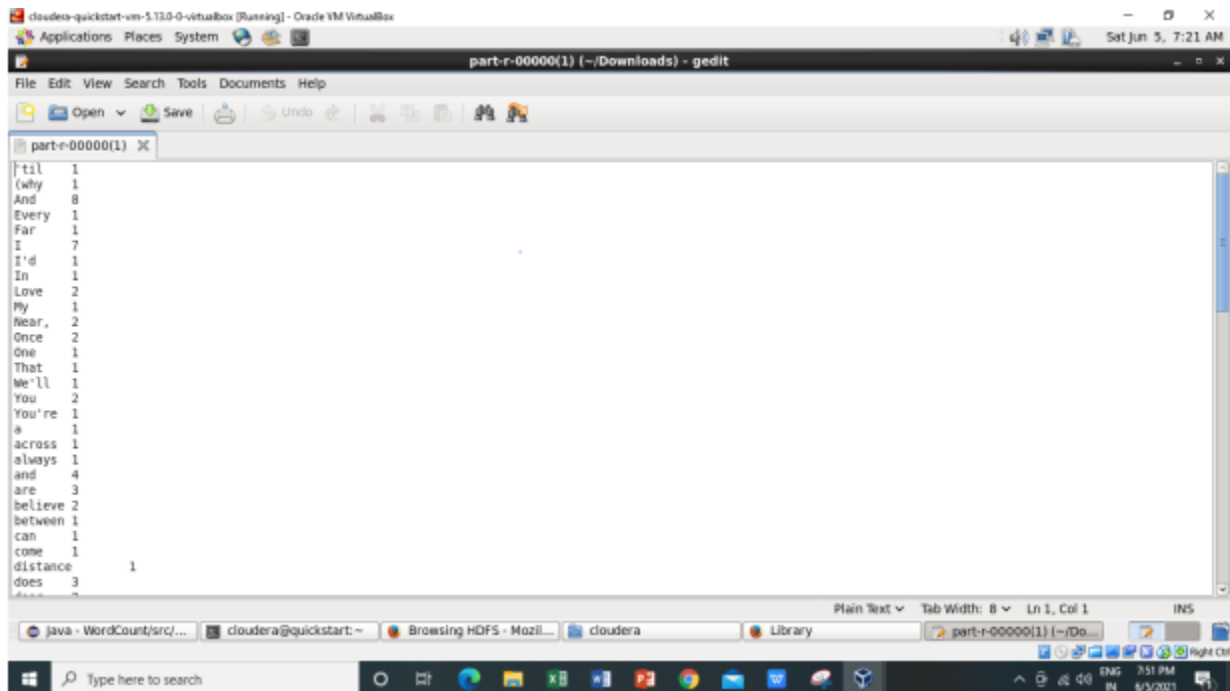
Hadoop, 2017.

Now downloading the **part-r-00000** file.

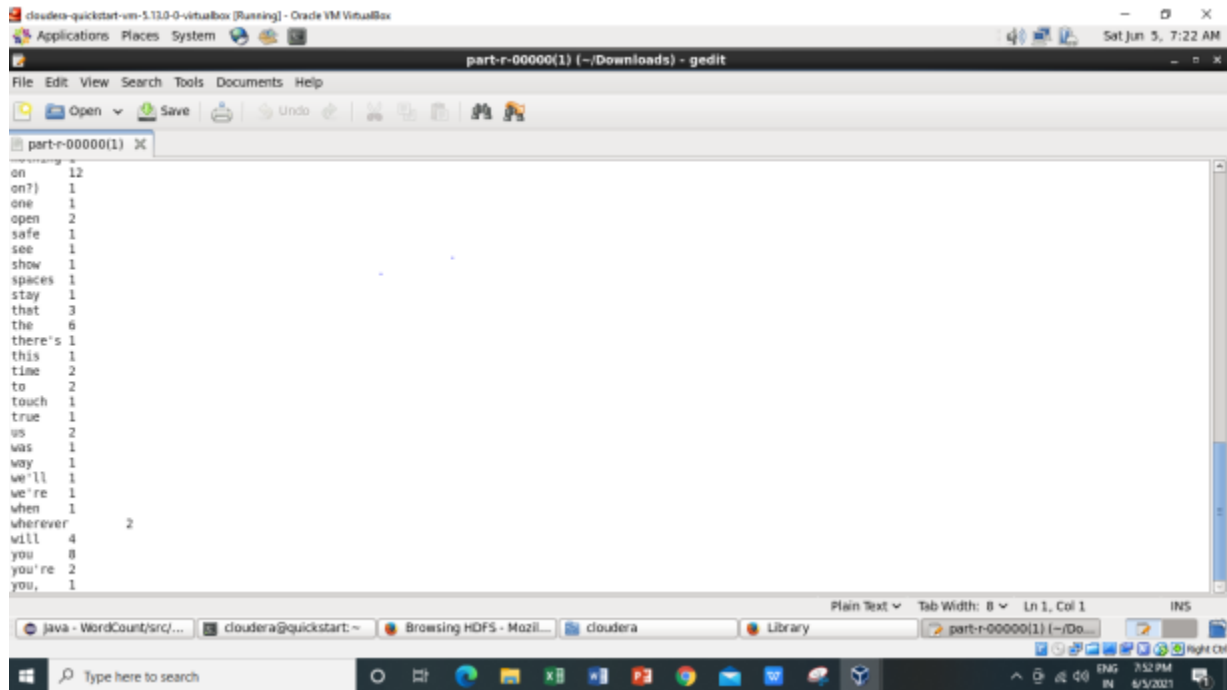
Name:  
Roll No:



Inside the **part-r-00000** file it will have the same output as we are getting after executing using command **hadoop jar /home/cloudera/WordCount.jar WordCount /inputdir/abc /op1**



Name:  
Roll No:



The screenshot shows a virtual machine window titled "cloudes-quickstart-vm-5.13.0-0-virtualbox (Running) - Oracle VM VirtualBox". Inside the VM, a terminal window titled "part-r-00000(1) [~/Downloads] - gedit" is open. The terminal displays the output of a word count program, listing words and their frequencies. The words and their counts are: on (12), on? (1), one (1), open (2), safe (1), see (1), show (1), spaces (1), stay (1), that (3), the (6), there's (1), this (1), time (2), to (2), touch (1), true (1), us (2), was (1), way (1), we'll (1), we're (1), when (1), wherever (2), will (4), you (8), you're (2), and you, (1). The terminal window has a menu bar with File, Edit, View, Search, Tools, Documents, and Help. Below the menu bar is a toolbar with icons for Open, Save, Print, Undo, and other standard text editor functions. The terminal window is running on a Linux system, as indicated by the prompt "cloudera@quickstart: ~". The taskbar at the bottom of the VM shows various application icons, including a web browser, a file manager, and a terminal. The system clock in the bottom right corner shows the date and time as "Sat Jun 5, 7:22 AM".

```
on 12
on? 1
one 1
open 2
safe 1
see 1
show 1
spaces 1
stay 1
that 3
the 6
there's 1
this 1
time 2
to 2
touch 1
true 1
us 2
was 1
way 1
we'll 1
we're 1
when 1
wherever 2
will 4
you 8
you're 2
you, 1
```