**ASSIGNMENT 3**

**PART 2 - MongoDB indexing**

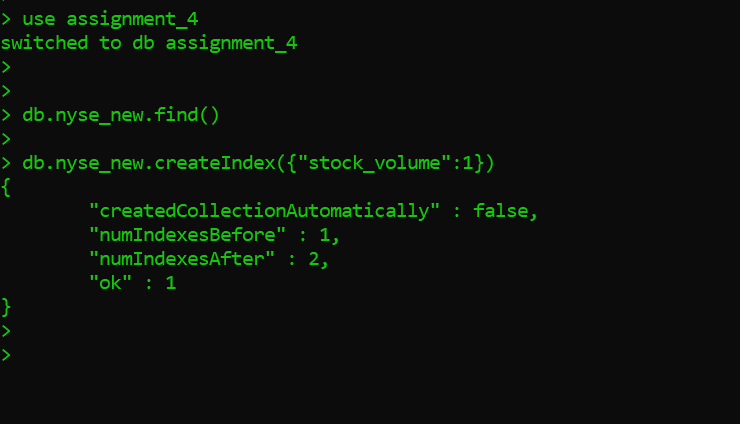
**Most of the time, you’ll want to declare your indexes before putting your application into production. This allows indexes to be built incrementally, as the data is inserted. But there are two cases where you might choose to build an index after the fact. The first case occurs when you need to import a lot of data before switching into production. For instance, you might be migrating an application to MongoDB and need to seed the database with user information from a data warehouse. You could create the indexes on your user data in advance but doing so after you have imported the data will ensure an ideally balanced and compacted index from the start. This will also minimize the net time to build the index. Use the NYSE dataset to declare your indexes before putting your application into production.**

**ANS-**

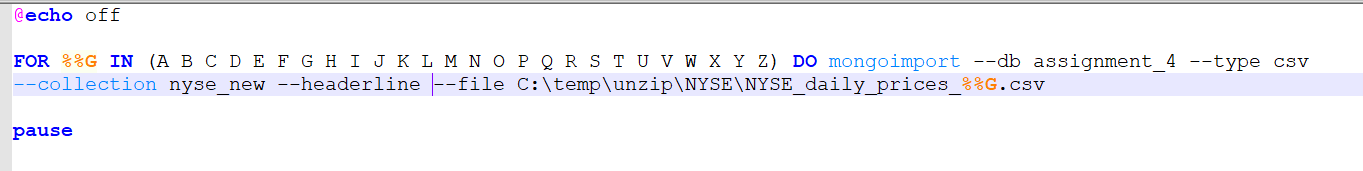
Created new DB- Assignment\_4

Then created a collections nyse\_new

Created index on key: stock\_volume in ascending order



**Then imported the nyse data to this collection using bat file**



**The indexes are correctly created.**



**PART 3 - MongoDB Indexing**

**Insert the NYSE dataset into a new database. You may use the existing NYSE database created before.**

**Now, create indexes on existing data sets.**

**Ans—**

**Imported all the nyse data to new collection stocks**

**Then after that created index** on key: stock\_symbol in ascending order



**PART 4 – Programming Assignment**

**All hadoop commands are invoked by the bin/hadoop script. Running the hadoop script without any arguments prints the description for all commands.**

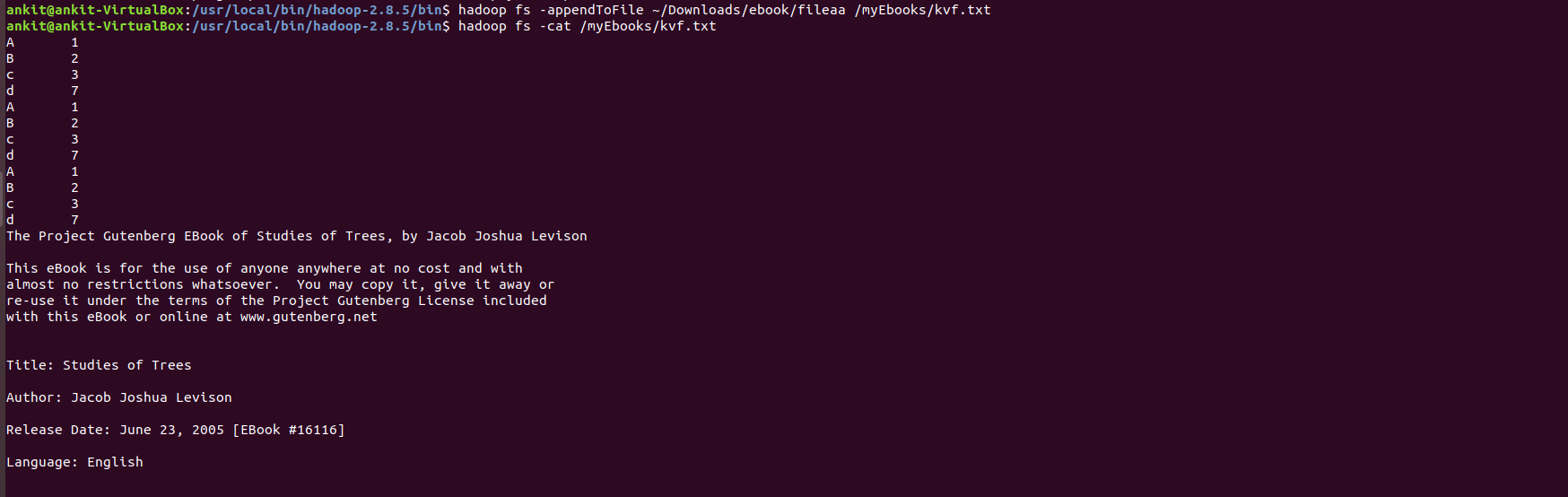
**Usage: hadoop [--config confdir] [--loglevel loglevel] [COMMAND] [GENERIC\_OPTIONS] [COMMAND\_OPTIONS]**

**Execute each hadoop command once and place the screenshots into a word file. If a command cannot be executed for any reason (such as, a distributed environment is needed), you may write the definition of the command, and skip execution.**

[**http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html**](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html)

**Ans:**

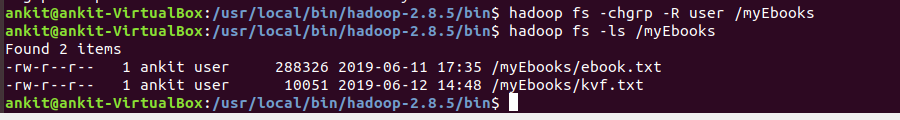
* appendToFile
* cat



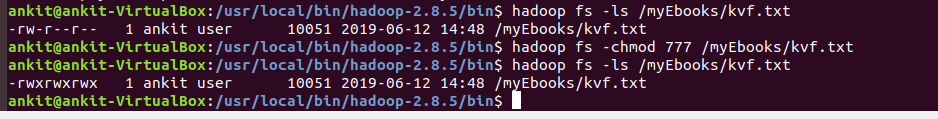
* [checksum](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#checksum)



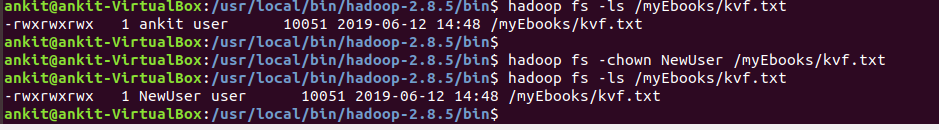
* [chgrp](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#chgrp)



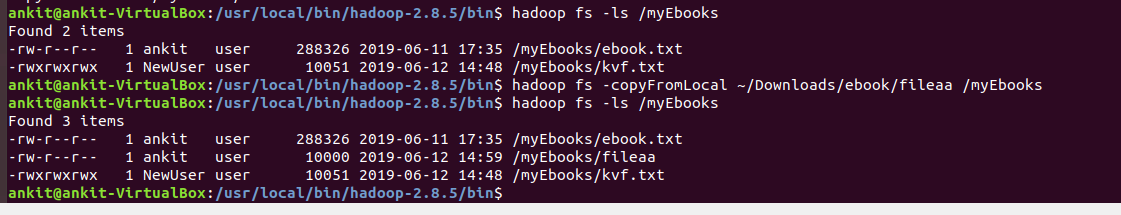
* [chmod](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#chmod)



* [chown](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#chown)

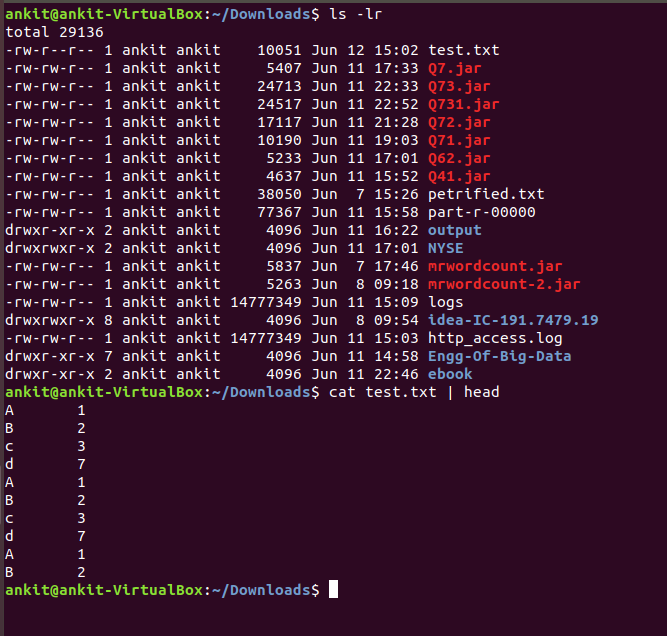


* [copyFromLocal](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#copyFromLocal)

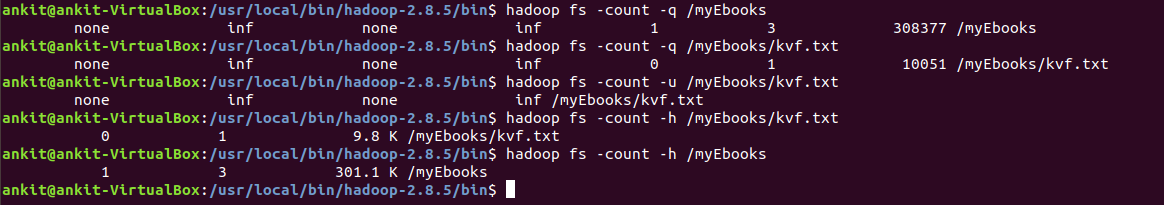


* [copyToLocal](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#copyToLocal)

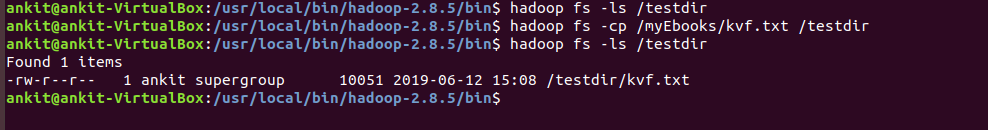




* [count](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#count)



* [cp](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#cp)



* [createSnapshot](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#createSnapshot)

Create a snapshot of a snapshottable directory. This operation requires owner privilege of the snapshottable directory.

**Command:**

hdfs dfs -createSnapshot <path> [<snapshotName>]

* [deleteSnapshot](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#deleteSnapshot)

Delete a snapshot of from a snapshottable directory. This operation requires owner privilege of the snapshottable directory.

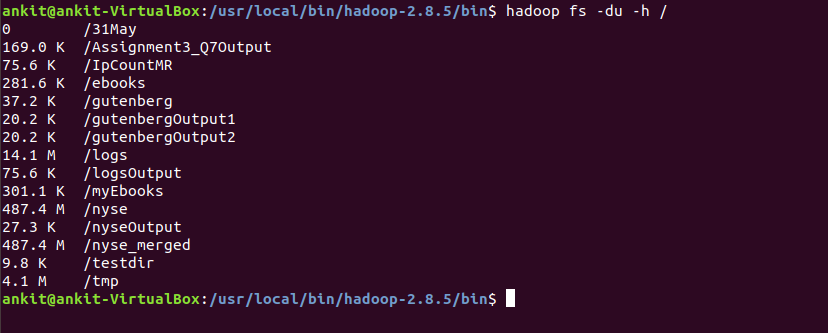
**Command:**

hdfs dfs -deleteSnapshot <path> <snapshotName>

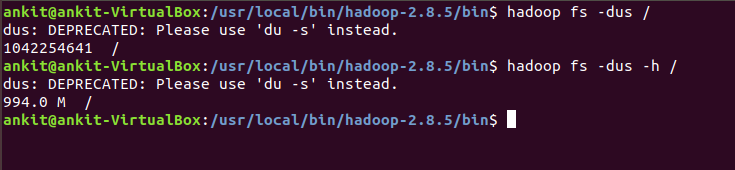
* [df](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#df)



* [du](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#du)



* [dus](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#dus)



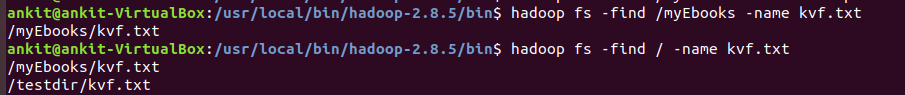
* [expunge](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#expunge)

Usage: hadoop fs -expunge

Permanently delete files in checkpoints older than the retention threshold from trash directory and create new checkpoint.

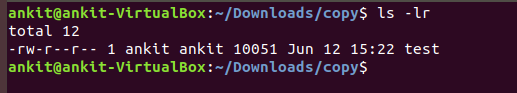
When checkpoint is created, recently deleted files in trash are moved under the checkpoint. Files in checkpoints older than fs.trash.interval will be permanently deleted on the next invocation of -expunge command.

* [find](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#find)

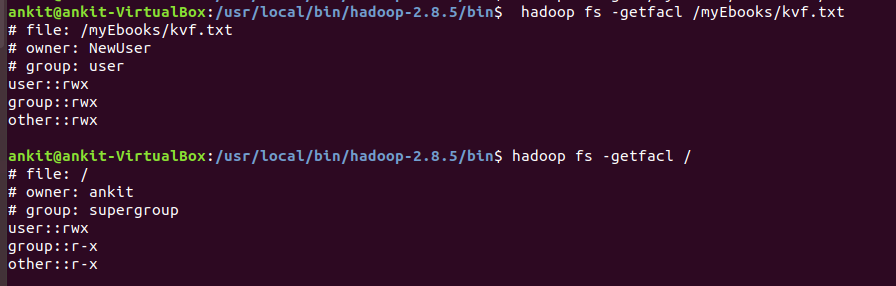


* [get](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#get)

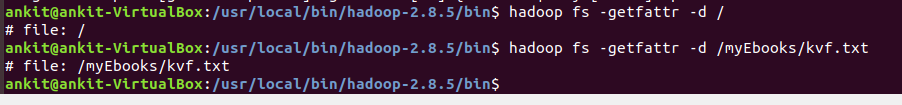




* [getfacl](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#getfacl)

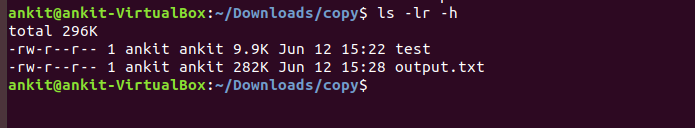


* [getfattr](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#getfattr)

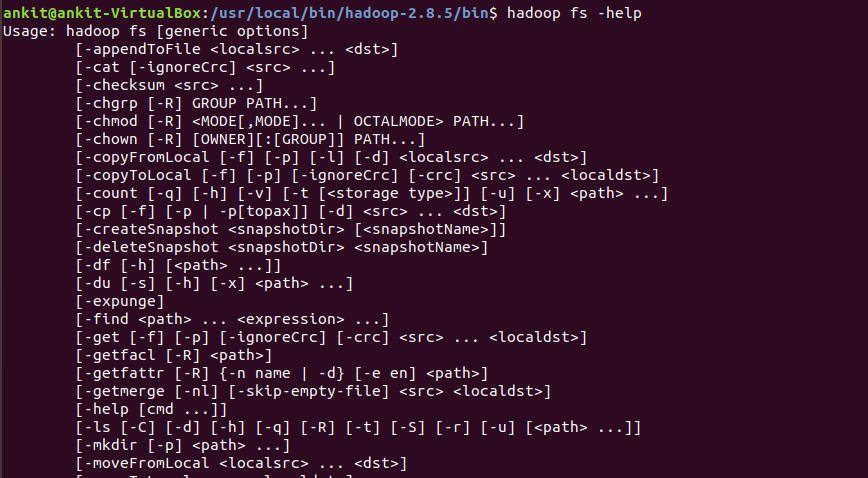


* [getmerge](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#getmerge)

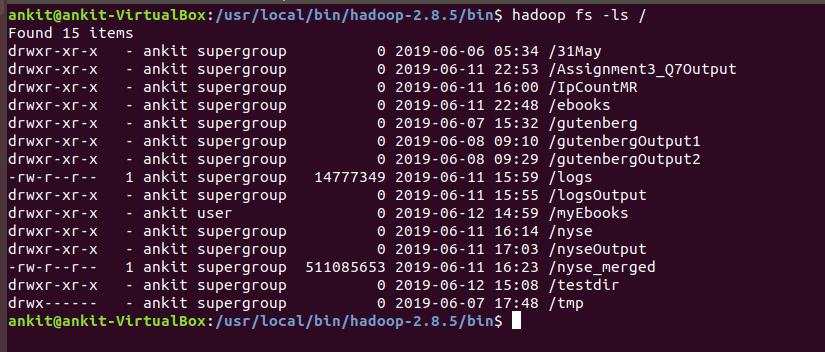




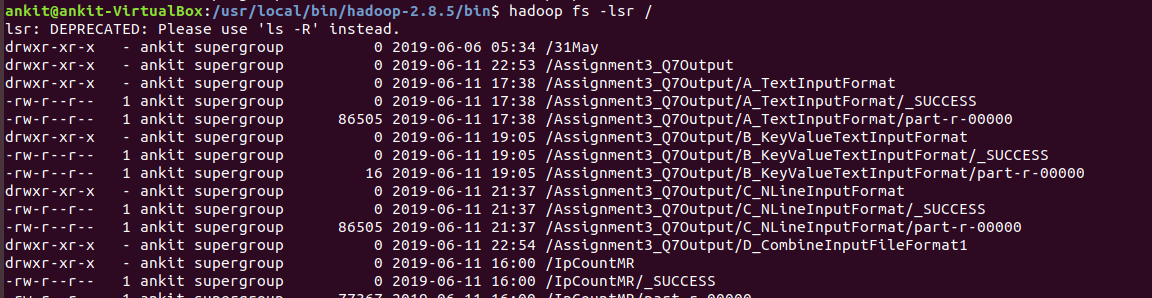
* [help](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#help)



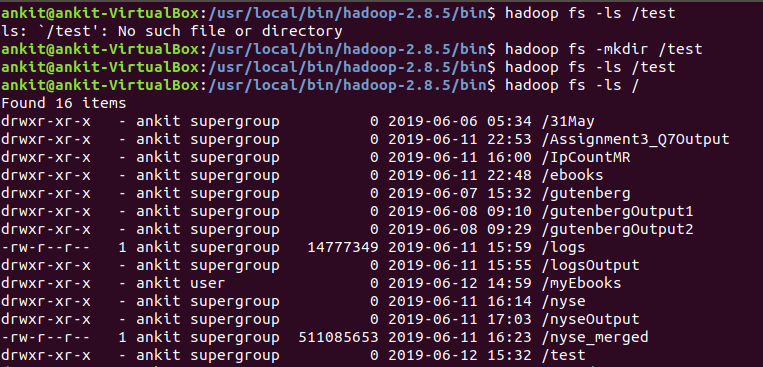
* [ls](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#ls)



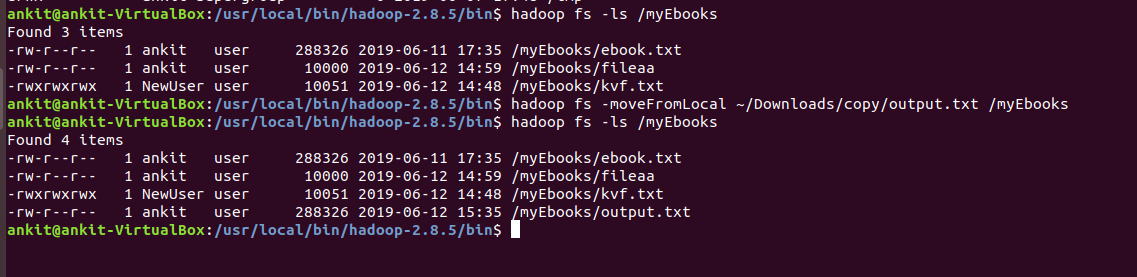
* [lsr](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#lsr)

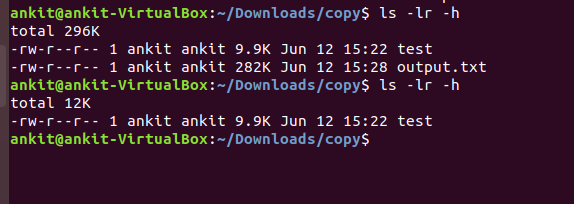


* [mkdir](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#mkdir)

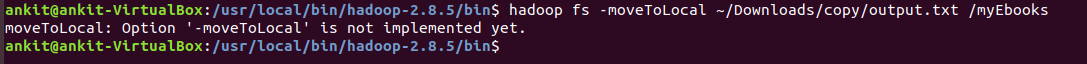


* [moveFromLocal](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#moveFromLocal)

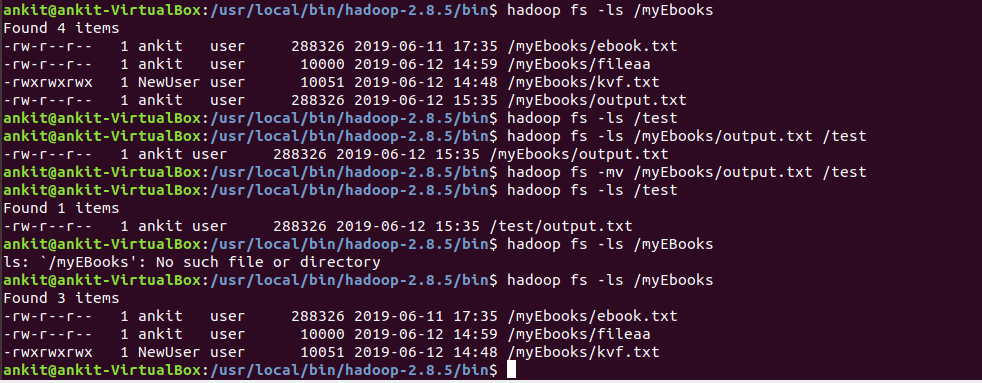




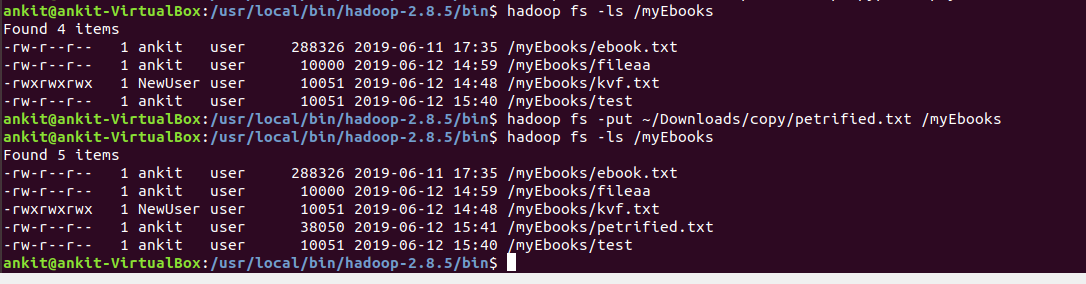
* [moveToLocal](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#moveToLocal)



* [mv](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#mv)



* [put](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#put)



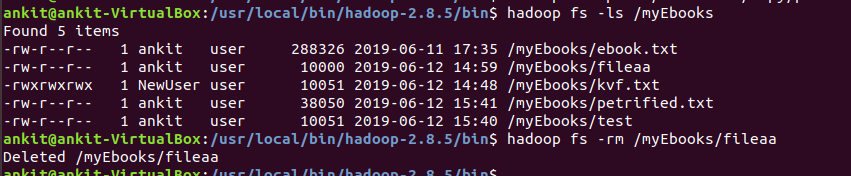
* [renameSnapshot](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#renameSnapshot)

Rename a snapshot. This operation requires owner privilege of the snapshottable directory.

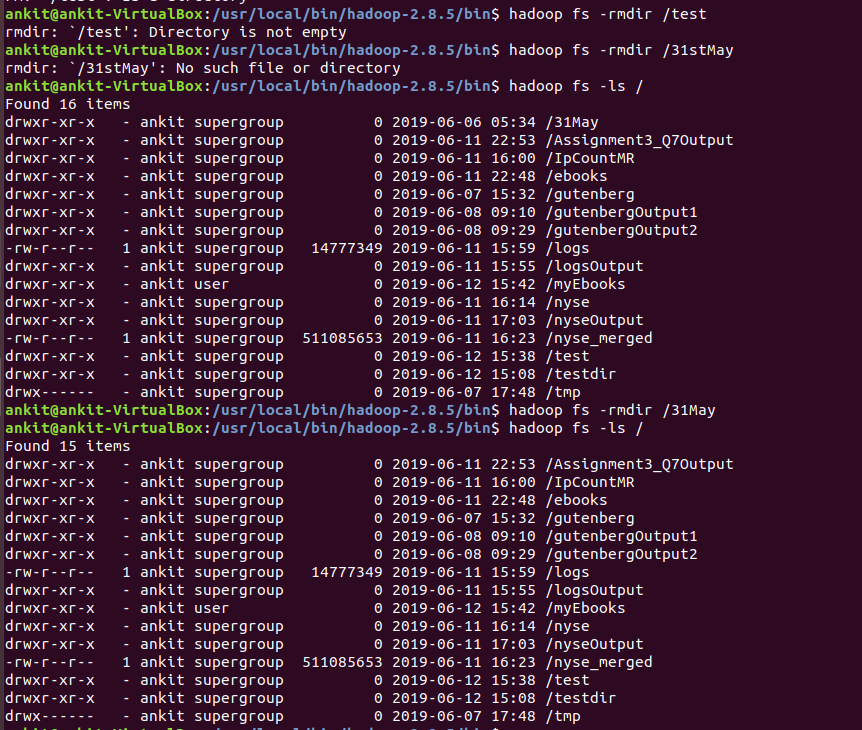
**Command:**

hdfs dfs -renameSnapshot <path> <oldName> <newName>

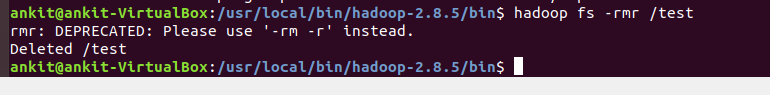
* [rm](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#rm)



* [rmdir](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#rmdir)



* [rmr](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#rmr)

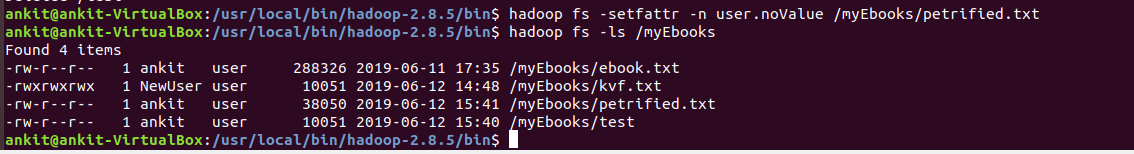


* [setfacl](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#setfacl)

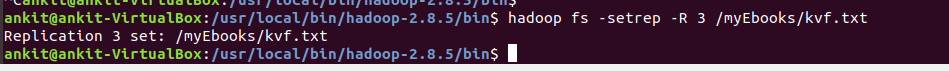
Usage: hadoop fs -setfacl [-R] [-b |-k -m |-x <acl\_spec> <path>] |[--set <acl\_spec> <path>]

Sets Access Control Lists (ACLs) of files and directories.

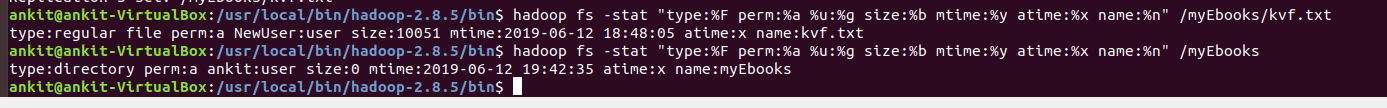
* [setfattr](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#setfattr)



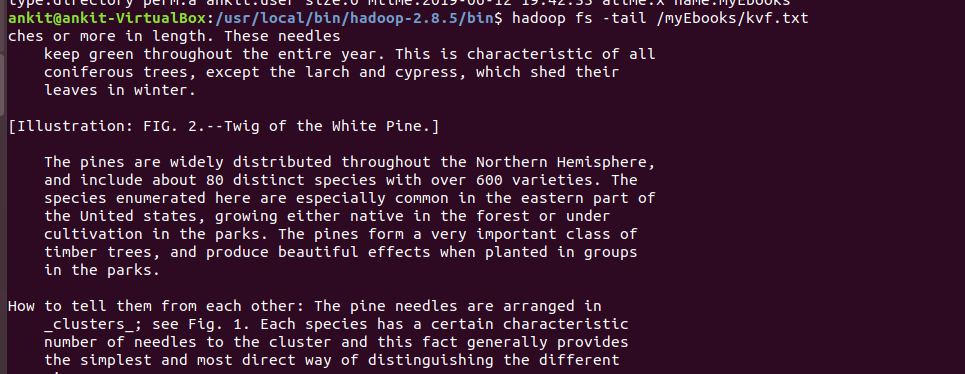
* [setrep](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#setrep)



* [stat](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#stat)



* [tail](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#tail)



* [test](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#test)

Usage: hadoop fs -test -[defsz] URI

Options:

-d: f the path is a directory, return 0.

-e: if the path exists, return 0.

-f: if the path is a file, return 0.

-s: if the path is not empty, return 0.

-r: if the path exists and read permission is granted, return 0.

-w: if the path exists and write permission is granted, return 0.

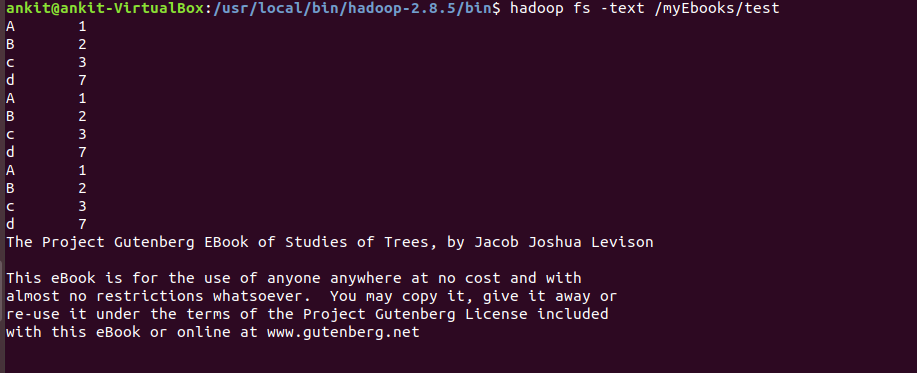
-z: if the file is zero length, return 0.

Example:

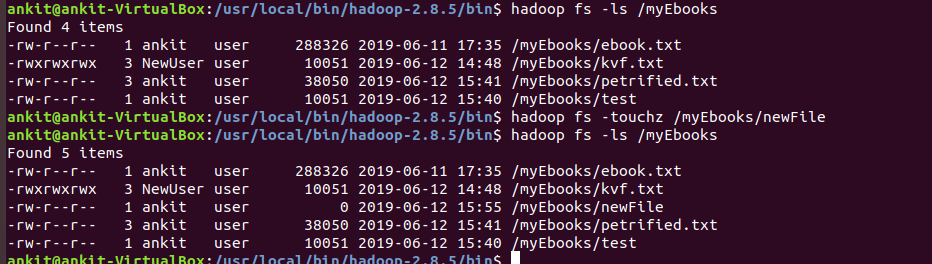
hadoop fs -test -e filename



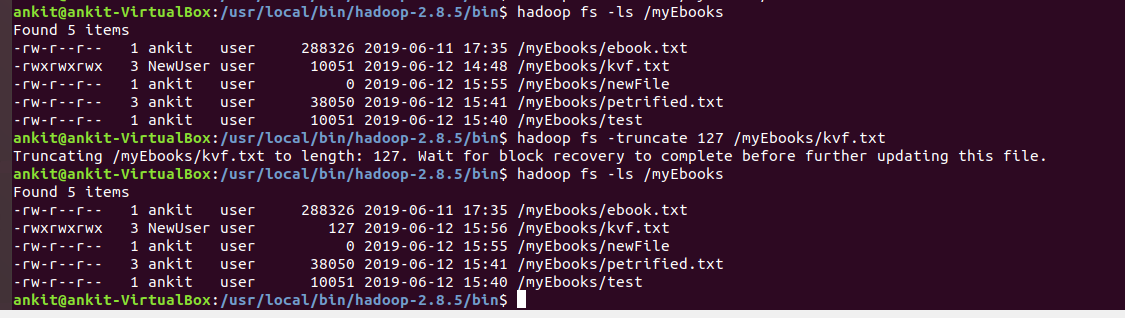
* [text](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#text)



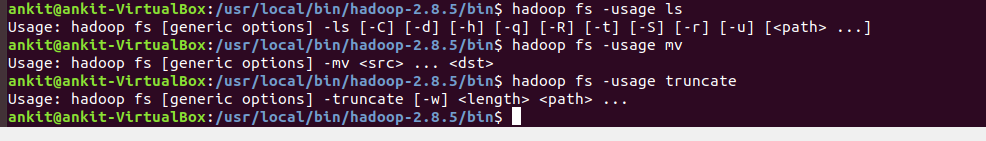
* [touchz](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#touchz)



* [truncate](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#truncate)



* [usage](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#usage)



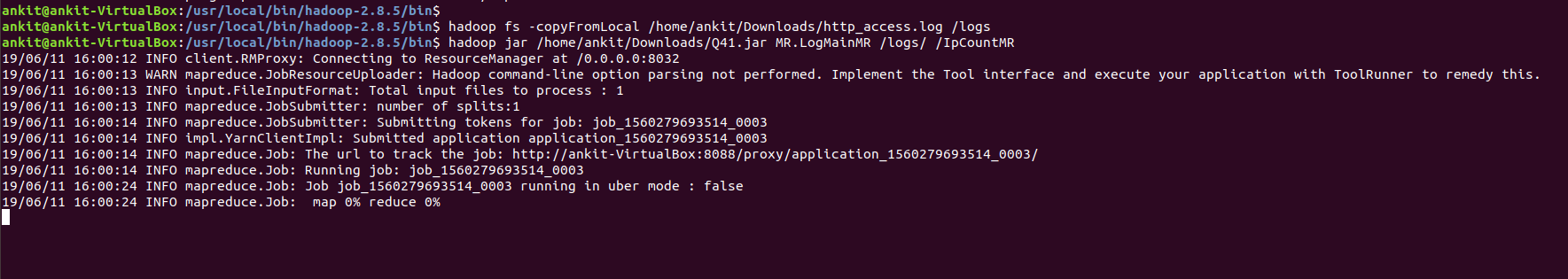
**PART 5 – Programming Assignment**

**Copy the attached ‘http\_access.log’ file into HDFS under /logs directory.**

**Using the access.log file stored in HDFS, implement MapReduce in Hadoop to find the number of times each IP accessed the website.**

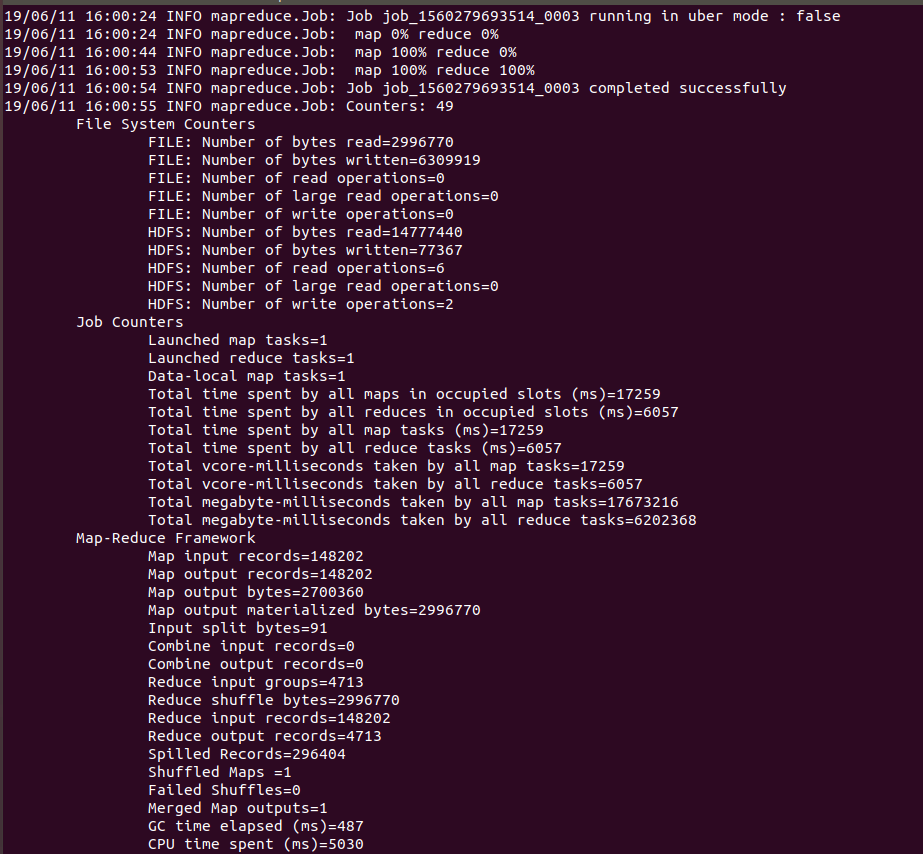
**Ans--**

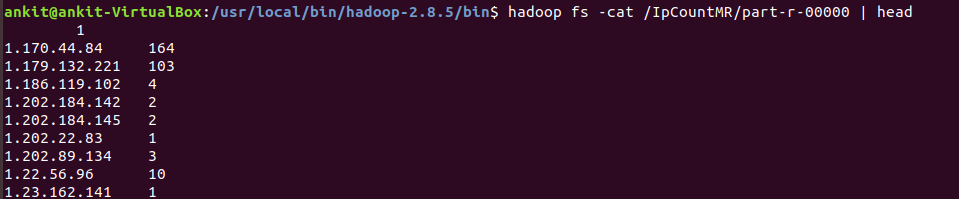
**copy http\_access.log to /logs**

****

Also ran MAP reduce on the logs to count the number of times ip is used

The program folder is inside the Assignment folder

Output head is as follows



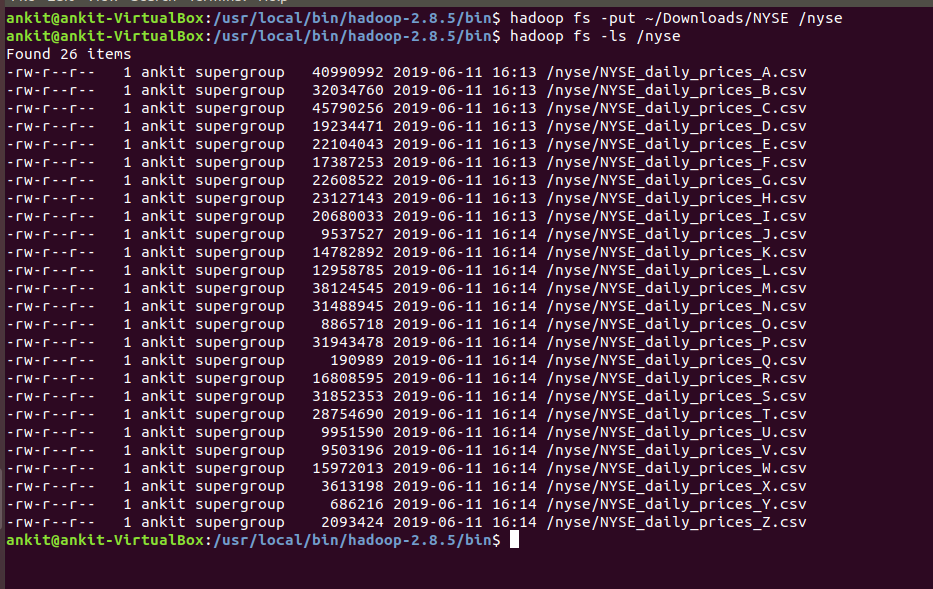
**PART 6 – Programming Assignment**

**Download and Copy all the files (http://msis.neu.edu/nyse/) (DailyPrices\_A to DailyPrices\_Z) to a folder in HDFS.**

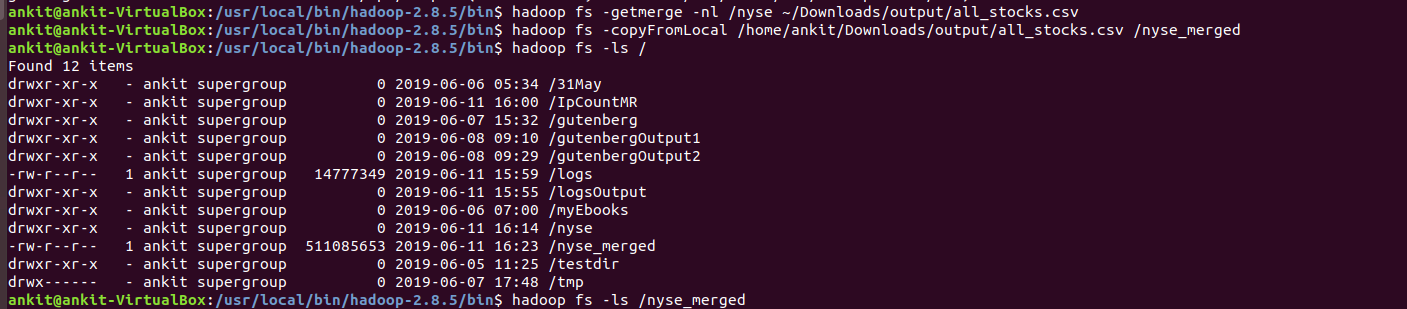
**Write a MapReduce to find the Max price of stock\_price\_high for each stock. Capture the running time programmatically (or manually using a wristwatch or smartphone).**

**Ans--**

Download and Copy all the files (http://msis.neu.edu/nyse/) (DailyPrices\_A to DailyPrices\_Z) to a folder in HDFS.



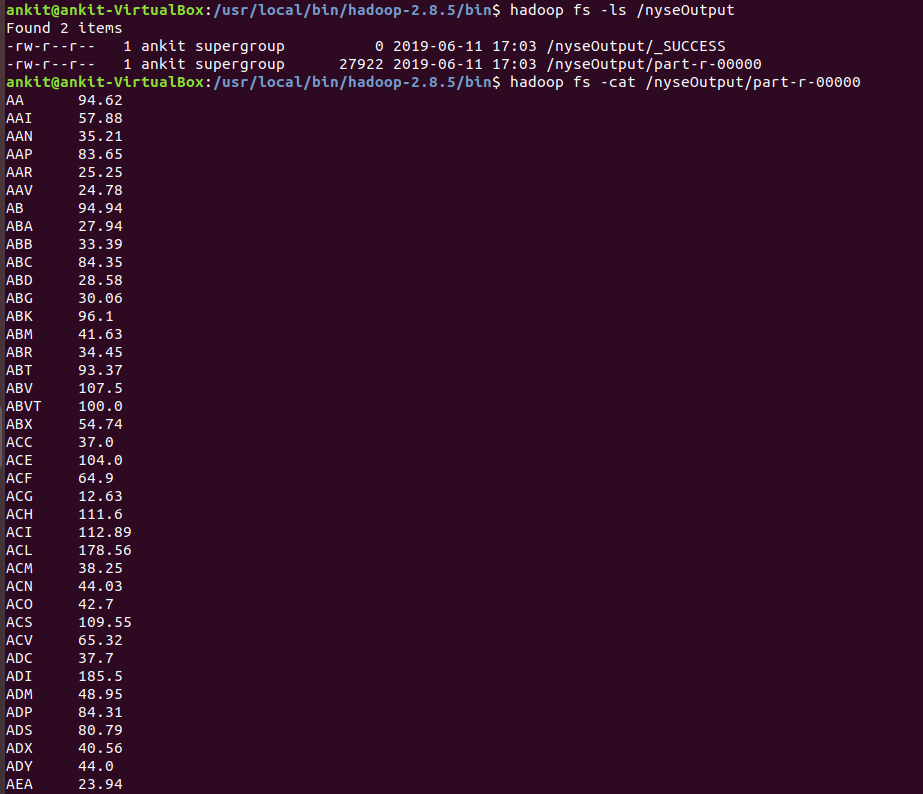
Merging all records in single file and saving in HDFS



The java program is in the assignment folder

Output of MapReduce for maximum stock\_price\_high for each stock is as follows:





**PART 7 – Programming Assignment**

**Write one MapReduce program using each of the classes that extend FileInputFormat<k,v>**

**(CombineFileInputFormat, FixedLengthInputFormat, KeyValueTextInputFormat, NLineInputFormat, SequenceFileInputFormat, TextInputFormat)**

[**http://hadoop.apache.org/docs/current/api/org/apache/hadoop/mapreduce/lib/input/FileInputFormat.html**](http://hadoop.apache.org/docs/current/api/org/apache/hadoop/mapreduce/lib/input/FileInputFormat.html)

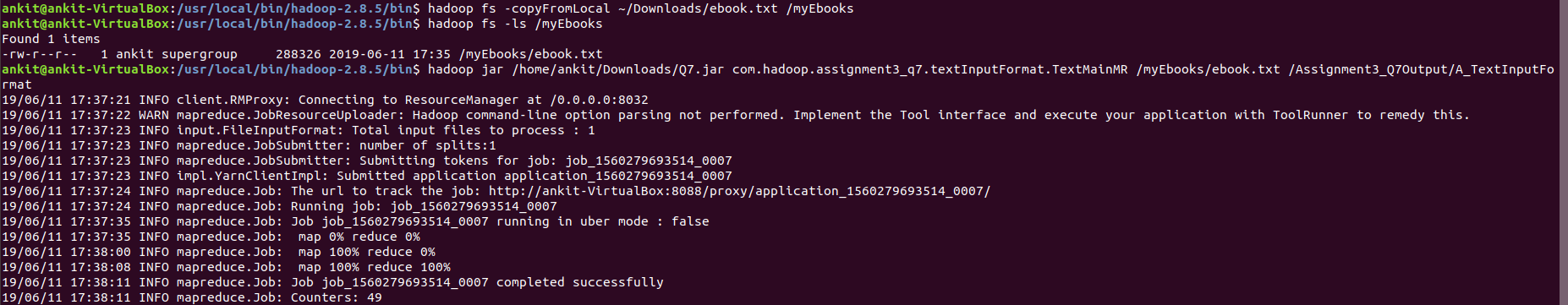
**You could use any input file of your choice. The size of the input files is not important. The MR programs could simply do counting, or any other analysis you choose.**

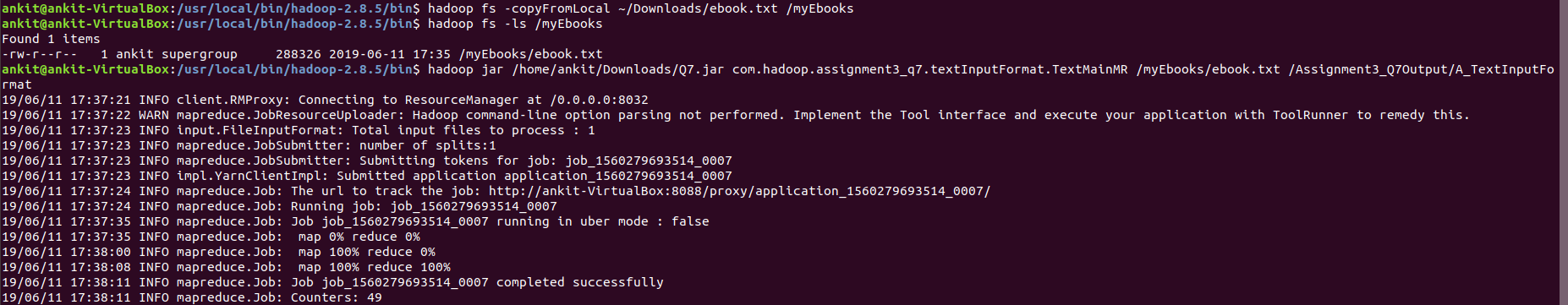
**Ans--**

**1> TextInputFormat**

Code in package textInputFormat

Output

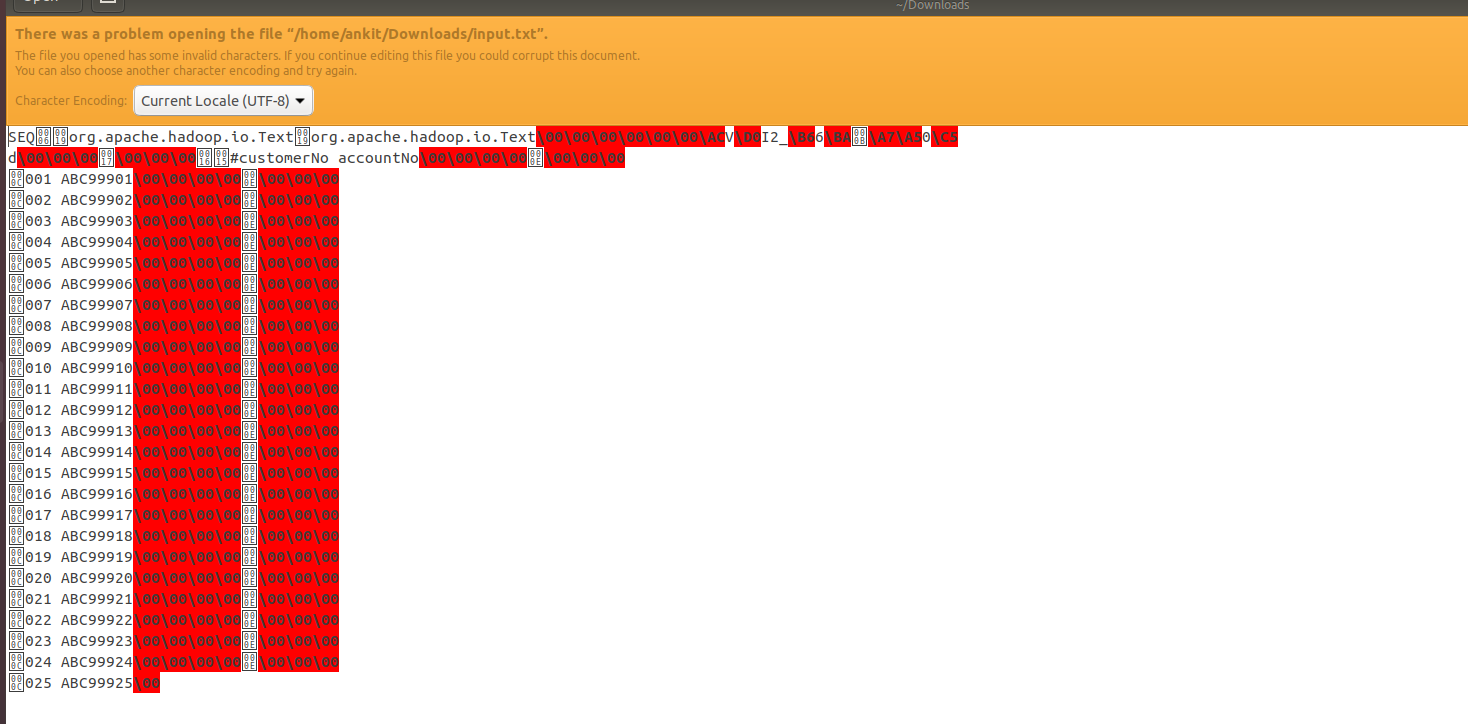




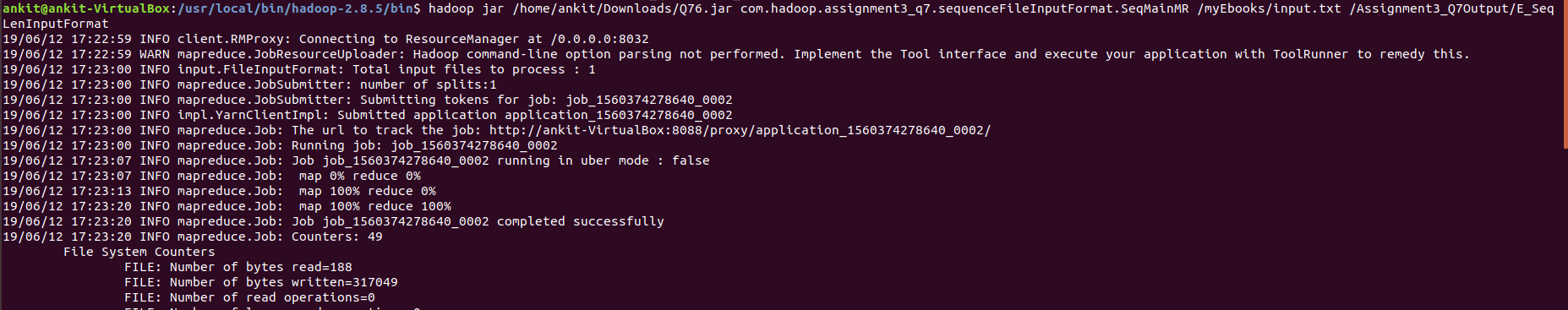
**2> SequenceFileInputFormat**

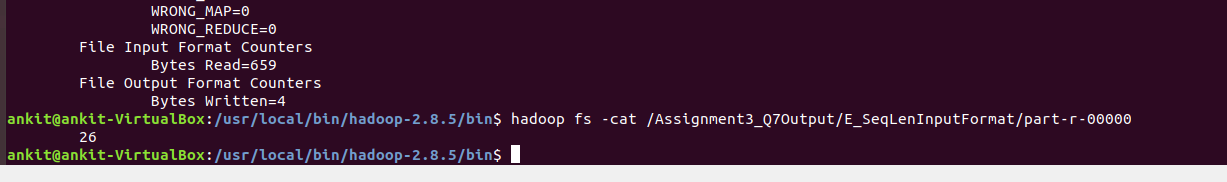
**The code is in package seqFileInputFormat**

**Input Sequence File-**



**Output**

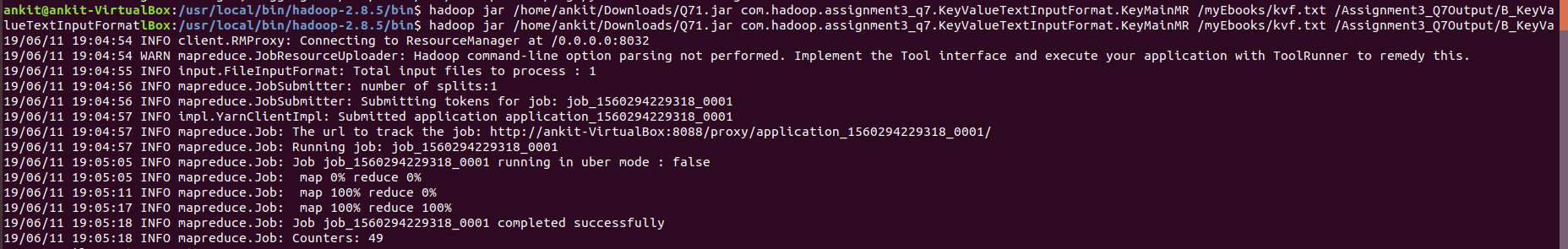


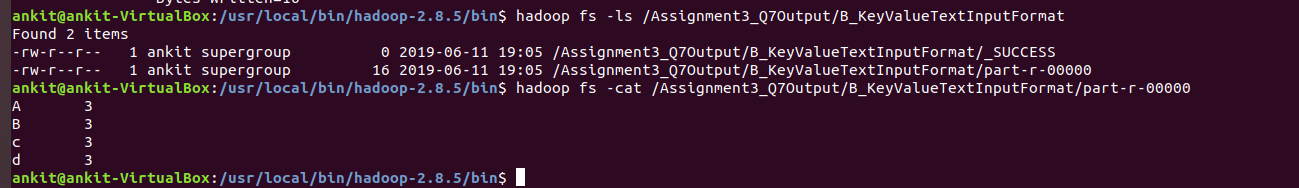


**3>KeyValueTextInputFormat**

Operation- Counting the number of keys

code in package KeyValueInputFormat

**Output**

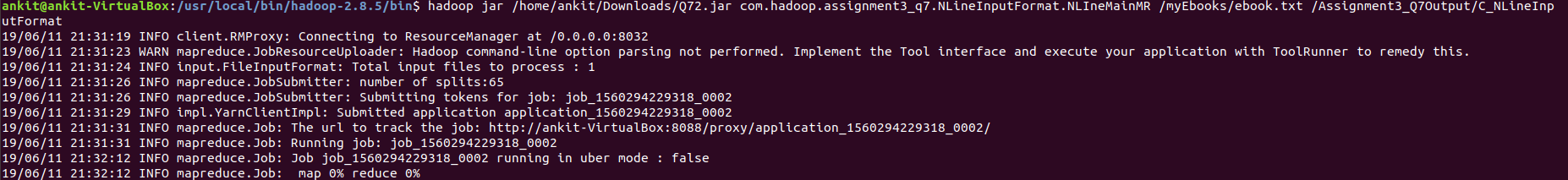
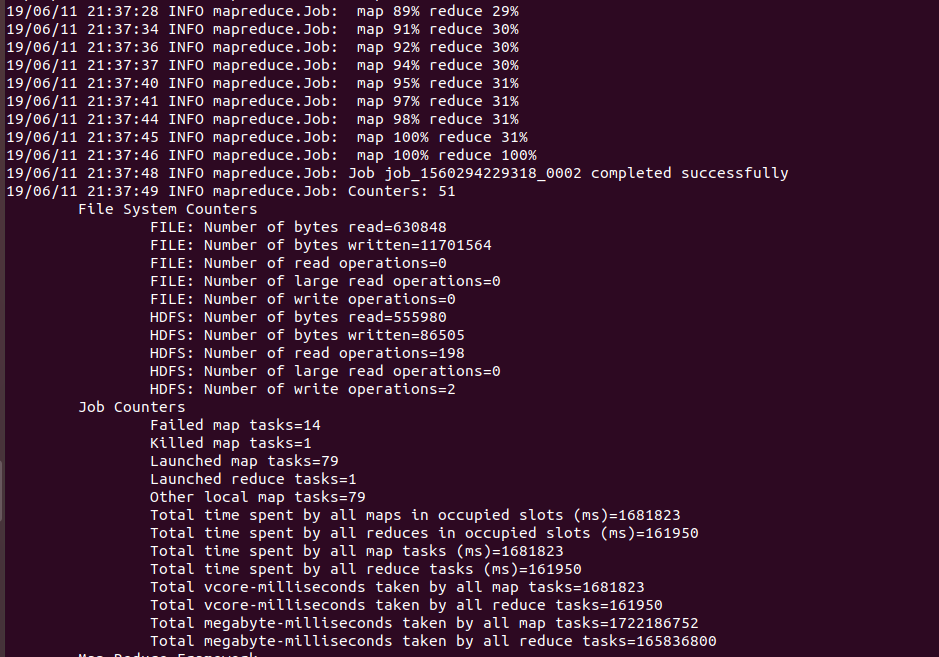
****

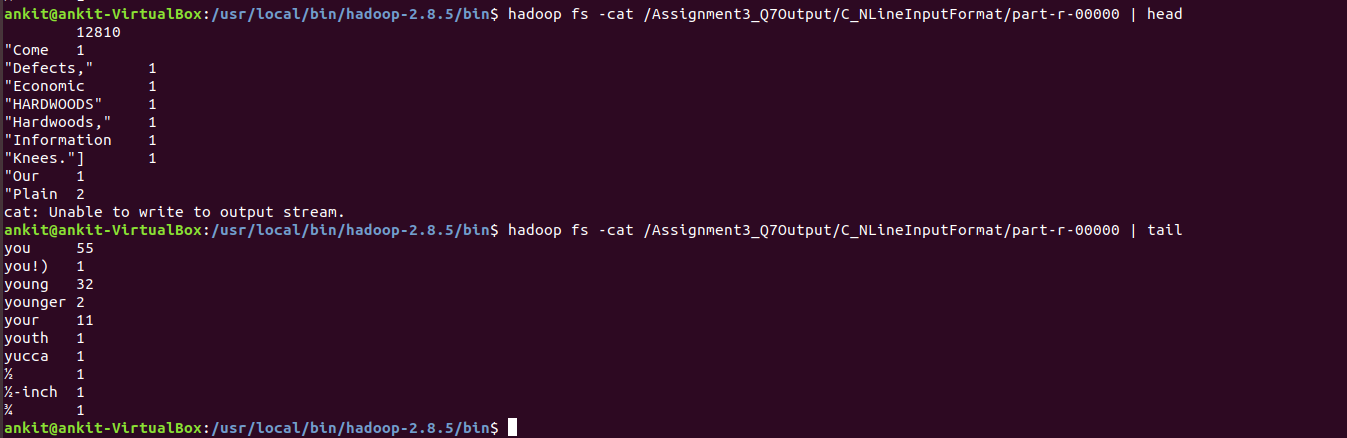
**4> FixedLengthInputFormat**

**5> NLineInputFormat**

**The code is in package NLineInputFormat**

**Output**

****

****

**6> CombineFileInputFormat**