

# List of Experiments

# Practice on basic Linux commands.

# 2. Implement the following file management tasks in Hadoop:

# a. Adding files and Directories

# b. Retrieving files

# c. Deleting files

# d. Copying files from local filesystem to HDFS and vice versa.

# e. Moving files

# 3. Write driver code, mapper code, reducer code to count number of words in a given file(Hint: WordCount Map- Reduce Program)

# 4. Write a MapReduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather a large volume of log data, which is a good candidate for analysis with MapReduce, since it is semi structured and record-oriented.

# 5. Implement Matrix Multiplication with Hadoop Map Reduce

# 6. Install and run Pig then write Pig Latin scripts to Load, Store and Filter data.

# 7. Write Pig Latin scripts to perform data processing operations

# a. Grouping and joining data

# b. Sorting data

# c. Combining and Splitting data

# 8. Implement User Defind Funcitons in Pig.

# 9. Install Hive and and use Hive to create databases and tables.

# a. Create and drop databases

# b. Create, alter, and drop tables

# c. Insert, Update and delete records

# 10. Perform data processing operations using Hive

# a. Sort and Aggregation of data

# b. Joins

# 11. Perform data processing operations using Hive

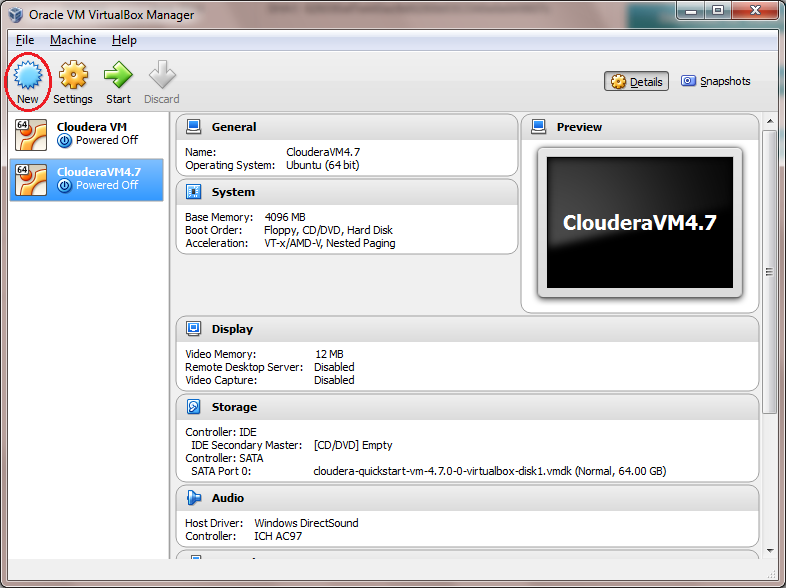
# a. Views

# b. Indexes

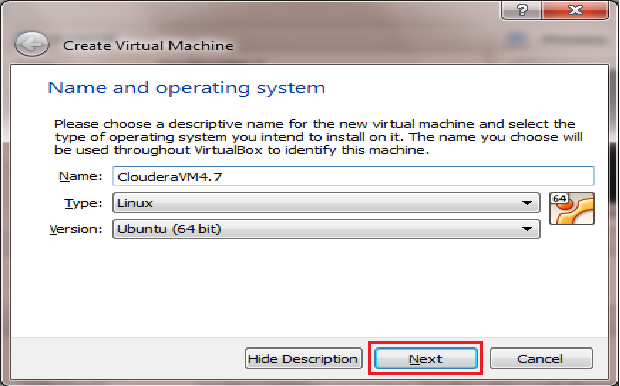
# 12. Implement User Defind Funcitons in Hive

**Cloudera quick start VM 4.7.x Installation procedure:**

1. Download and install Oracle Virtual Box from <http://dlc.sun.com.edgesuite.net/virtualbox/4.3.12/VirtualBox-4.3.12-93733-Win.exe>.
2. Open Oracle Virtual Box and click on **New**.

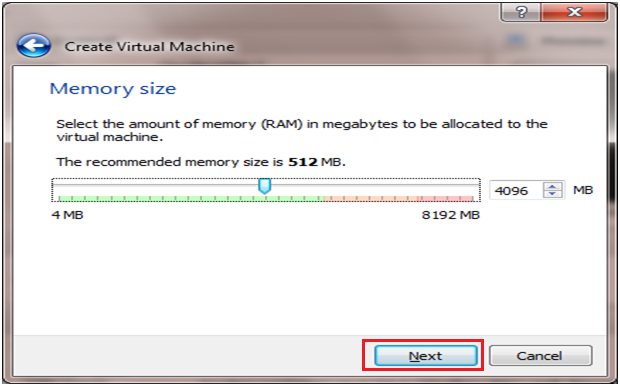


1. Give the details as given below:

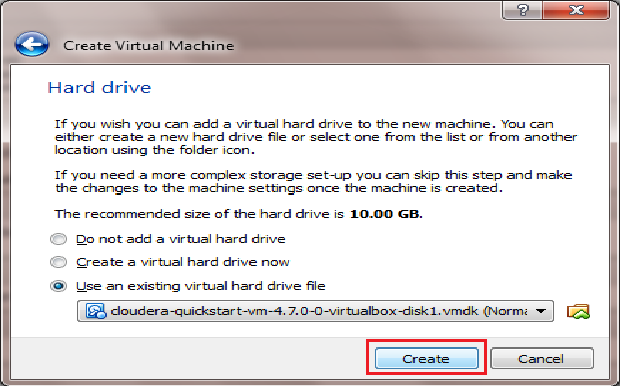


And click on **Next.**

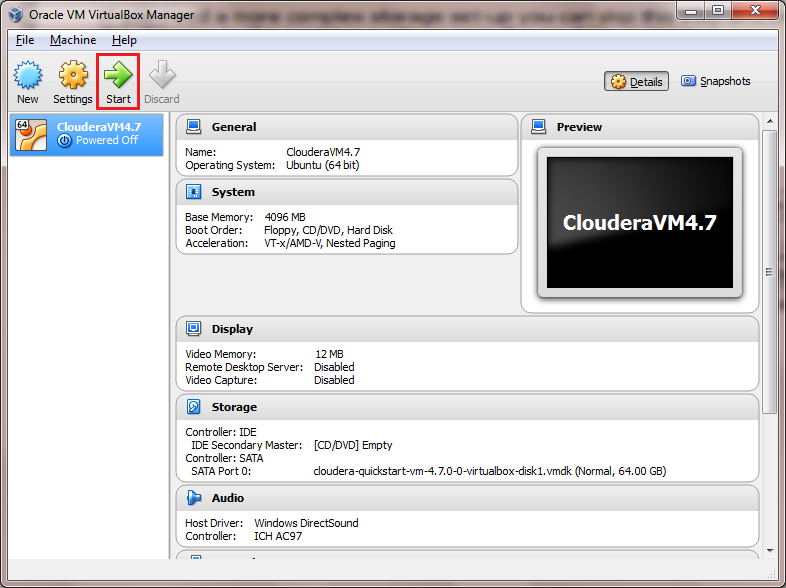
1. Set the RAM memory as given below and click on **Next**. Approximately Half of the RAM need to be allocated to Virtual Box Instance



1. Select Use an existing virtual hard drive file and click on **Create.**



1. Now, select ClouderaVM4.7 and click on **Start** button.



**System requirements:**

This requires a 64-bit host OS and a virtualization product that can support a 64-bit guest OS.

Better to have 8GM RAM since we are using virtual box but 4GB is also fine for practice

Double Click on the “poweroff” button and you will be accessing Cloudera Manager

Cloudera Manager UserId/Password: cloudera/cloudera

# EXPERIMENT-1

# GetImage (3).pngGetImage (2).pngGetImage (1).pngGetImage.png

# EXPERIMENT-2

## 2)Implement the following file management tasks in Hadoop:

* + **Adding files and directories**
  + **Retrieving files**
  + **Deleting files**

**Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.**

**HDFS basic Command-line file operations**

1. Create a directory in HDFS at given path(s):

**Command:** hadoop fs -mkdir <paths>

1. List the contents of a directory:

**Command:** hadoop fs -ls <args>

1. Upload and download a file in HDFS:

*Upload:*

**Command:** hadoop fs -put <localsrc> <HDFS\_dest\_path>

*Downloa:*

**Command:** hadoop fs -get <HDFS\_src> <localdst>

1. See contents of a file:

**Command:** hadoop fs -cat <path[filename]>

1. Copy a file from source to destination:

**Command:** hadoop fs -cp <source> <dest>

1. Copy a file from/To Local file system to HDFS:

**Command:** hadoop fs -copyFromLocal <localsrc> URI

**Command:** hadoop fs -copyToLocal [-ignorecrc] [-crc] URI <localsrc>

1. Move file from source to destination:

**Command:** hadoop fs -mv <src> <dest>

1. Remove a file or directory in HDFS:

Remove files specified as argument. Delete directory only when it is empty.

**Command:** hadoop fs -rm <arg> Recursive version of delete

**Command:** hadoop fs -rmr <arg>

1. Display last few lines of a file:

**Command:** hadoop fs -tail <path[filename]>

1. Display the aggregate length of a file:

**Command:** hadoop fs -du <path>

1. Getting help:

**Command:** hadoop fs -help

## Adding files and directories:

* Creating a directory

**Command:** hadoop fs -mkdir input/

* Copying the files from localfile system to HDFS

**Command:** hadoop fs -put inp/file01 input/

## Retrieving files:

**Command:** hadoop fs -get input/file01 localfs

## Deleting files and directories:

**Command:** hadoop fs -rmr input/file01

Hadoop provides a set of command line utilities that work similarly to the Linux file commands.

**Default directories**

Local file system : **/home/cloudera**

HDFS : **/user/cloudera**

**Basic file commands:**

Hadoop file command take the form of

**hadoop  fs  -cmd  <args>**

where **cmd** is the specific file command and **<args>** is the variable number of arguments

**Example:**

     Command for listing files is:

**hadoop   fs   –ls**

 Most common file management tasks in hadoop are—

•   Adding files and directories

•    Retrieving files

•    Deleting files

**a)   Adding files and directories**:  Before running hadoop programs need to put the data into HDFS first .

1. **mkdir : Create a directory in HDFS at given path(s**).

        hadoop fs  -mkdir  <paths>

         Example:

                         hadoop  fs  -mkdir  /user/cloudera/myfolder1

                                             (absolute path)

                                                       Or

                         hadoop fs –mkdir myfolder1

                                             (relative path)

**Create a sub directory**

   Example:

         hadoop fs –mkdir  /user/cloudera/myfolder1/subfolder1

1. **ls  : List the contents of a directory.**

           hadoop fs -ls <args>

           Example:

  hadoop  fs  – ls

  hadoop  fs  –ls  /   (list the contents of root directory)

  hadoop  fs  –lsr  /   (recursively displays entries in all subdirectories of                                         path)

  hadoop  fs  –ls  –R

  hadoop fs  –lsr   /user/cloudera/myfolder1

1. **put  or copyFromLocal : Upload a file in HDFS**

  hadoop  fs  -**put**  localsrc  dst

                                    or

  hadoop  fs –**copyFromLocal**  localsrc  dst

Copy single src file, or multiple src files from local file system to the Hadoop distributed file system

Example

create two files in local filesystem using cat or using any editor nano or gedit

                           cat > file1

                                      This is Hadoop Lab

                            Ctrl+Z

            cat > file2

                                     This is Bigdata Lab

                            Ctrl+Z

         hadoop  fs  - put  file1 /user/cloudera/myfolder1

  hadoop  fs  -copyFromLocal file2 /user/cloudera/myfolder1/subfolder1

  hadoop  fs  -put file3  .   (put the file in the default directory

**checking:**

 hadoop  fs – lsr /user/cloudera/myfolder1

  hadoop fs –ls   /

**b)Retriving files**

     copy files from HDFS to local filesystem.

**1.Download: get or copyToLocal :Copies/Downloads files to the local file system**

        hadoop  fs  –**get**    hdfs\_src     localdst

                                or

        hadoop  fs  - **copyToLocal**  hdfs\_src  localdst

Example:

        hadoop   fs  -get  /user/clooudera/myfolder1/file1  .

        hadoop  fs  -copyToLocal  /user/cloudera/myfolder1/file2  .

Another way to access the data is to display it. We can use the Hadoop filecommand with unix pipes to send its output for further processing.

  hadoop   fs –cat file1

  hadoop   fs –cat file1 | head

  hadoop  fs –tail   file1   (display the last 1 kb of file1)

**c)  Deleting files**

Hadoop command for removing files is **rm**

Example :

  hadoop   fs  -**rm**  file1

  hadoop   fs  -**rmr** myfolder1 (remove directory recursively)

**Looking Up Help**

 A list of hadoop file commands together with the usage and description of each command can see by using **help**command.

  hadoop  fs  -**help**  cmd

Example :

  hadoop fs –help ls

**1. cp : Copy a file from source to destination**

  hadoop  fs  -**cp**  <source>  <dest>

Example:

  hadoop   fs  -cp  /user/cloudera/file1   /user/cloudera/myfolder1

**2.mv : Move file from source to destination.**

Note:- Moving files across filesystem is not permitted.

   hadoop  fs  **–mv** <src>  <dest>

Example:

  hadoop  fs  -mv  /user/cloudera/file1  user/cloudera/myfolder1

**3. du : Shows disk usage, in bytes, for all the files which match path; filenames are reported with the full HDFS protocol prefix.**

  hadoop fs  -**du** <path>

Example:

  hadoop  fs  -du  /user/cloudera

**4. dus : Like -du, but prints a summary of disk usage of all files/directories in the path.**

  hadoop fs  - **dus** <path>

Example:

  hadoop  fs  -dus  /user/cloudera

**Moving files across filesystem**

**5. moveFromLocal : files from local file system to the Hadoop distributed file system**

  hadoop  fs –**moveopyFromLocal**  localsrc  dst

Move single src file, or multiple src files from local file system to the Hadoop distributed  file system

Example

create afile in local filesystem using cat or using any editor nano or gedit

                           cat > file4

                                      This is Hadoop and BigdataLab

                            Ctrl+Z

 hadoop  fs  -moveFromLocal  file4 /user/cloudera/myfolder1/subfolder1

 checking:

                 hadoop  fs – lsr /user/cloudera/myfolder1

  hadoop fs –ls  .

**6. moveToLocal: copy files from HDFS to local filesystem.**

  hadoop  fs  - **moveToLocal**  hdfs\_src  localdst

Example:

  hadoop  fs  -moveToLocal  /user/cloudera/myfolder1/file4  .

**7. Chmod : To change permissions  of files/directories**

  hadoop fs  -**chmod**  777 filename/directory name

Example:

  hadoop fs –chmod  666  /user/cloudera/file2

**8.getmerge: concatenates the files in the source directory into the destination file.**

  hadoop fs -**getmerge**  <src>   <localdst>  [addnl]

The addnl option is for adding new line character at the end of each file.

Example :

  hadoop  fs –getmerge file1 file2 mergfile

**9. chown : used to change the ownership of files. The -R option can be used to recursively change the owner of a directory structure.**

hadoop fs -**chown** [-R] <NewOwnerName>[:NewGroupName] <file or dir name>

**10. Expunge : Used to empty the trash**.

   hadoop  fs  -**expunge**

**11.setrep:  used to change the replication factor of a file.**

  hadoop fs -**setrep** -w 4  /user/cloudera /file1

**12. touchz: creates a zero byte file. This is similar to the touch command in unix.**

  hadoop fs -**touchz**  /user/cloudera/filename

Example :

  hadoop fs –touchz  /user/cloudera/file0