HDFS FILE SYSTEM COMMANDS

hadoop fs –help

- mkdir

Syntax

```
$ hadoop fs -mkdir  [-p] <path>
$ hadoop fs -mkdir test
$ hdfs dfs -mkdir -p /user/hduser
```

- ls

```
$ hdfs dfs -ls /
$ hadoop fs -ls /user/hduser
$ hadoop fs -ls -R /user/hduser
```

- put

Copies files from local path to HDFS path.

```
$ hadoop fs -put test.txt /user/hduser/test/test.txt
```

- get

To get Copies the file from **HDFS** to local file system

```
$ hadoop fs -get /hadoop/hduser/test/new.txt /home/hduser
```

- cat

Displays the content of file from HDFS

```
$ hadoop fs -cat /user/hduser/test/test.txt
```

- touchz

Creates a file at path containing the current time as a timestamp.

Fails if a file already exists at path, unless the file is already size 0.

```
$ hadoop fs -touchz /user/hduser/test/1.txt
```

-cp

Copies file within Hadoop HDFS

```
$ hadoop fs -cp /hadoop/user/test/test.txt
/hadoop/hduser/test/test_new.txt
```

- copyFromLocal

Copies file from Local to HDFS

```
$ hadoop fs -copyFromLocal /home/hduser/filename /hadoop
/hduser/test
```

- copyToLocal

Copy File from HDFS to Local DIR just like 'get' command

```
$ hadoop fs –copyToLocal /hadoop/user/filename.txt /home/hduser/
```

- mv

Move file with in Hadoop HDFS

```
$ hadoop fs –mv /hduser/user/test /hduser/hadoop/test
```

- moveToLocal - moveFromLocal

Move File from HDFS to Local Directory

```
$ hadoop fs –moveToLocal /hadoop/hduser/filename /home/hduser
```

Move File from Local Directory to HDFS

```
$ hadoop fs –moveFromLocal /home/hduser /hadoop/hduser/filename
```

- rm - rmr

Removes file from the destination

```
$ hadoop fs -rm /hadoop/user/filename
```

Removes empty directory from the destination

```
$ hadoop fs -rm /hadoop/user/empty_dir
```

Removes the directory Recursively

```
$ hadoop fs -rmr /hadoop/user/directory
```

- expunge

This Hadoop shell command is used to empty the trash.

hdfs dfs -expunge

- appendToFile

Append to File

```
$ hadoop fs -appendToFile <local files separated by space> <hdfs destination file>
```

```
$ hadoop fs -appendToFile  [localsrc] [destination]
```

- chmod

Changes permissions of a file. This works similar to the Linux shell's chmod command with a few exceptions.

```
$ hadoop -chmod mode /path/filename
```

```
$ hadoop -chmod -R mode /path/filename
```

Mode = +r, +w, +x, +o (octal mode)

- chown

Changes ownership and group of a file

```
$ hadoop fs -chown [-R] [OWNER][:[GROUP]] PATH
```

```
$ hadoop -chown hadoop:hadoop PATH
```

Change ownership recursively

```
$ hadoop -chown -R hadoop:hadoop PATH
```

- chgrp

Changes group of a file or path.

Syntax:

```
$ hadoop fs -chgrp [-R] groupname <path>
```

- setrep

This HDFS command is used ***to change the replication factor of a file***. Enter path is a directory, then this command changes the replication factor of all the files present in the directory tree rooted at path provided by user recursively.

Syntax

```
$ hdfs dfs -setrep 2 /hdfs/dir/
```

```
$ hdfs dfs -setrep 2 /hdfs/dir/filename
```

Syntax

```
$ hdfs dfs -setrep -w 2 /hdfs/dir/filename
```

The -w flag requests that the command waits for the replication process to get completed.

- df:

Shows the capacity, free and used space of the filesystem. If the filesystem has multiple partitions, and no path to a particular partition is specified, then the status of the root partitions will be shown.

Syntax:

```
$ hadoop fs -df [-h] <path>
```

-h Formats the sizes of files in a human-readable fashion rather than a number of bytes.

- du:

Show the amount of space, in bytes, used by the files that match the specified file pattern.

Syntax:

```
$ hadoop fs -du [-s] [-h] <path>
```

-s Rather than showing the size of each individual file that matches the pattern, shows the total (summary) size.

-h Formats the sizes of files in a human-readable fashion rather than a number of bytes.

- stat

HDFS file system command prints information about the path.

```
$ hdfs dfs -stat /hdfs/dir/
```

- count

Hdfs - dfs command counts the number of directories, number of files present and bytes under the paths that match the specified file pattern.

```
$ hdfs dfs -count /user/hdfs/
```

- tail

```
$ hadoop fs -tail <path>
```

- count

```
$ hadoop fs -count [options] <path>
```

```
-q  -  shows quotas(quota is the hard limit on the number of
names and amount of space used for individual directories)

-u  -  it limits output to show quotas and usage only

-h  -  shows sizes in a human-readable format

-v  -  shows header line
```

- expunge

HDFS expunge command makes the trash empty.

```
$ hadoop fs -expunge
```

hadoop fs -getmerge <src> <localdest>

HDFS COMMANDS USING CLI

- find

Finds all files that match the specified expression and performs all the actions to them which are selected. If no path is specified then defaults to the present working directory.

```
hadoop fs -find /user/data/dir/ -name sample -print
```

Find the file name “sample”

- help

```
hadoop fs -help
```

Displays usage information for the commands entered by the user.

- truncate

It truncates (shorts) all the files to a specified length that match the specified file pattern

```
hadoop fs -truncate [-w] <length> <paths>
```

```
hadoop fs -truncate 55 /user/hdfs/dir2/purchase.csv
/user/hdfs/dir1/purchase.csv
```

```
hadoop fs -truncate -w 127 /user/hdfs/dir2/purchase.csv
```

```
hadoop fs -usage mkdir
```