Practical2

Hadoop HDFS Commands
1) Hadoop Version->hadoop version
The Hadoop fs shell command version prints the Hadoop version.

```
[cloudera@quickstart ~]$ hadoop version
Hadoop 2.6.0-cdh5.4.2
Subversion http://github.com/cloudera/hadoop -r 15b703c8725733b7b2813d2325659
d57e7a3f
Compiled by jenkins on 2015-05-20T00:03Z
Compiled with protoc 2.5.0
From source with checksum de74f1adb3744f8ee85d9a5b98f90d
This command was run using /usr/jars/hadoop-common-2.6.0-cdh5.4.2.jar
```

2) LS Command

->hdfs dfs -ls /
HDFS Command to display the list of Files and Directories in HDFS. It Lists the contents of the directory specified by path,showing the names,permissions, owner, size and modification datefor each entry.

hdfs dfs is the command that is specific to HDFS.

```
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 5 items
drwxr-xr-x   - hbase supergroup          0 2022-02-10 20:00 /hbase
drwxr-xr-x   - solr  solr                0 2015-06-09 03:38 /solr
drwxrwxrwx   - hdfs  supergroup          0 2022-02-07 21:01 /tmp
drwxr-xr-x   - hdfs  supergroup          0 2015-06-09 03:38 /user
drwxr-xr-x   - hdfs  supergroup          0 2015-06-09 03:36 /var
```

->hadoop fs -ls /
hadoop fs is more "generic" command that allows you to interact with multiple file systems
including Hadoop. we are using the ls command to enlist the files and directories present in
HDFS. The Hadoop fs shell command ls displays a list of the contents of a directory specified in
the path provided by the user. It shows the name, permissions, owner, size, and modification date
for each file or directories in the specified directory.
Using the ls command, we can check for the directories in HDFS.

2) MKDIR Command

HDFS Command to create the directory in HDFS.

Usage: hdfs dfs –mkdir /directory_name

Here I am trying to create a directory named "rjc" in HDFS.

After creating ,Using the ls command, we can check for the directories in HDFS or Using ls command we listed the directory 'rjc" created using mkdir.

```
[cloudera@quickstart ~]$ hdfs dfs -mkdir /rjc
[cloudera@quickstart ~]$ hadoop fs -ls /
Found 6 items
drwxr-xr-x   - hbase    supergroup         0 2022-02-14 18:34 /hbase
drwxr-xr-x   - cloudera supergroup         0 2022-02-14 18:44 /rjc
drwxr-xr-x   - solr     solr               0 2015-06-09 03:38 /solr
drwxrwxrwx   - hdfs     supergroup         0 2022-02-07 21:10 /tmp
drwxr-xr-x   - hdfs     supergroup         0 2015-06-09 03:38 /user
drwxr-xr-x   - hdfs     supergroup         0 2015-06-09 03:36 /var
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 6 items
drwxr-xr-x   - hbase    supergroup         0 2022-02-14 18:34 /hbase
drwxr-xr-x   - cloudera supergroup         0 2022-02-14 18:44 /rjc
drwxr-xr-x   - solr     solr               0 2015-06-09 03:38 /solr
drwxrwxrwx   - hdfs     supergroup         0 2022-02-07 21:10 /tmp
drwxr-xr-x   - hdfs     supergroup         0 2015-06-09 03:38 /user
drwxr-xr-x   - hdfs     supergroup         0 2015-06-09 03:36 /var
[cloudera@quickstart ~]$
```

3) copyFromLocal Command

First we will Create a document.

Steps: Right click anywhere on desktop->empty file->file_01

Put some information in the file file_03.
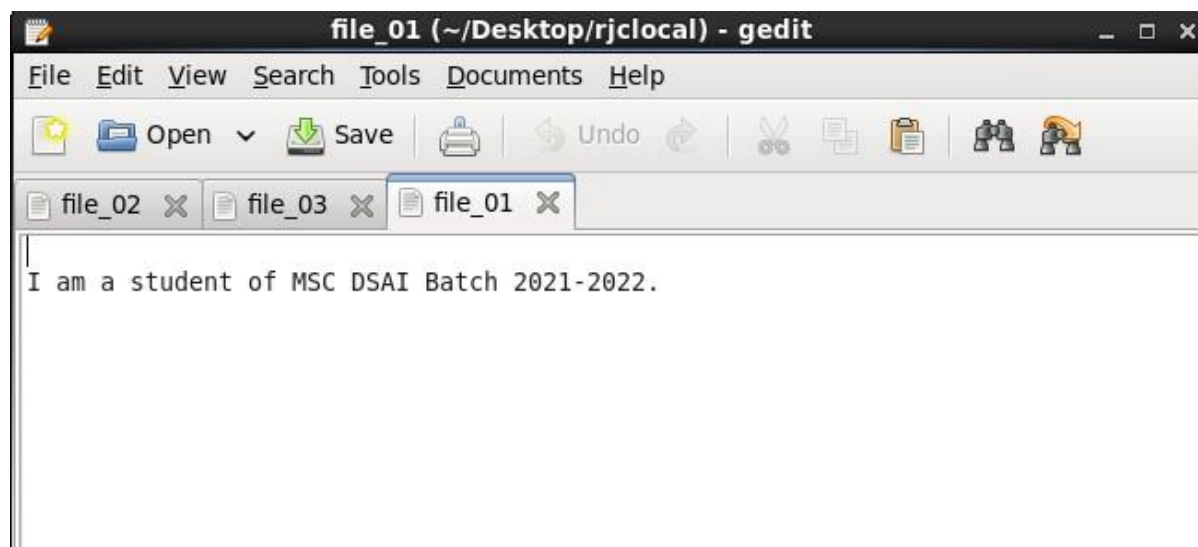
Now we are trying to copy the 'file_01' file present in the local file system to the 'rjc'

directory of Hadoop. Below command copies the file from the local file system

to HDFS.

```
[cloudera@quickstart ~]$ hdfs dfs -copyFromLocal /home/cloudera/Desktop/rjclocal/file_03 /rjc
[cloudera@quickstart ~]$ hdfs dfs -ls /rjc
Found 2 items
-rw-r--r--   1 cloudera supergroup         45 2022-02-14 18:58 /rjc/file_01
-rw-r--r--   1 cloudera supergroup         29 2022-02-14 20:06 /rjc/file_03
```

4) If getting any error due to permissions

Use-> export HADOOP_USER_NAME=hdfs

5) Put Command

-> hdfs dfs -put /home/cloudera/Desktop/ file_02 /rjc

Here in this example, we are trying to copy "file_02" of the local file system to the Hadoop file system.

The Hadoop fs shell command put is similar to the copyFromLocal, which copies files or directory from the local filesystem to the destination in the Hadoop filesystem.

Now using the ls command, we can check for the directories in HDFS.

```
[cloudera@quickstart ~]$ hdfs dfs -put /home/cloudera/Desktop/file_02 /rjc
[cloudera@quickstart ~]$ hdfs dfs -ls /rjc
Found 2 items
-rw-r--r--   1 cloudera supergroup         45 2022-02-14 18:58 /rjc/file_01
-rw-r--r--   1 cloudera supergroup         39 2022-02-14 19:09 /rjc/file_02
```

6) copyToLocal Command

->hdfs dfs -copyToLocal /rjc/Sample /home/cloudera/Desktop

copyToLocal command copies the file from HDFS to the local file system
Here in this example, we are trying to copy the 'file_01 file present in the rjc directory of HDFS
to the local file system.
Deleted Sample file from desktop. If it is already exist. And then again run the command.

```
[cloudera@quickstart ~]$ hdfs dfs -copyToLocal /rjc/file_01 /home/cloudera/De
op/rjclocal/
[cloudera@quickstart ~]$
```

7) CAT Command
->hdfs dfs -cat /rjc/file_01

we are using the cat command to display the content of the 'Sample_01' file present in rjc directory of HDFS

The cat command reads the file in HDFS and displays the content of the file on console or stdout.

```
[cloudera@quickstart ~]$ hdfs dfs -cat /rjc/file_01

I am a student of MSC DSAI Batch 2021-2022.
[cloudera@quickstart ~]$ █
```

8) Cp Command

First we will create 'rjcnew' inside hdfs and then we copy 'file_01 file which is present in 'rjc' folder inside this 'rjcnew directory in hdfs.

->hdfs dfs -cp /rjc/Sample /newdir

We are copying the 'sample' present in rjc directory in HDFS to the rjcnew of HDFS.

The cp command copies a file from one directory to another directory within the HDFS.

```
[cloudera@quickstart ~]$ hdfs dfs -mkdir /rjcnew
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 7 items
drwxr-xr-x   - hbase    supergroup          0 2022-02-10 20:00 /hbase
drwxr-xr-x   - cloudera supergroup          0 2022-02-14 19:09 /rjc
drwxr-xr-x   - cloudera supergroup          0 2022-02-14 19:36 /rjcnew
drwxr-xr-x   - solr     solr                0 2015-06-09 03:38 /solr
drwxrwxrwx   - hdfs     supergroup          0 2022-02-07 21:01 /tmp
drwxr-xr-x   - hdfs     supergroup          0 2015-06-09 03:38 /user
drwxr-xr-x   - hdfs     supergroup          0 2015-06-09 03:36 /var


[cloudera@quickstart ~]$ hdfs dfs -cp /rjc/file_01 /rjcnew
[cloudera@quickstart ~]$ hdfs dfs -ls /rjcnew
Found 1 items
-rw-r--r--   1 cloudera supergroup         45 2022-02-14 19:40 /rjcnew/file_0
```

9) MV Command

->hdfs dfs -mv /rjc/file_02 /rjcnew

we have a directory 'rjc' in HDFS. We are using mv command to move the rjc directory to the rjcnew directory in HDFS.

The HDFS mv command moves the files or directories from the source to a destination within HDFS.

```
[cloudera@quickstart ~]$
[cloudera@quickstart ~]$ hdfs dfs -mv /rjc/file_02 /rjcnew
[cloudera@quickstart ~]$ hdfs dfs -ls /rjcnew
Found 2 items
-rw-r--r--   1 cloudera supergroup         45 2022-02-14 19:40 /rjcnew/file_0
-rw-r--r--   1 cloudera supergroup         39 2022-02-14 19:09 /rjcnew/file_0
```

10) RM Command

->hdfs dfs –rm /rjcnew /file name

The hadoop dfs -rm command deletes objects and directories full of objects.

```
[cloudera@quickstart ~]$ hdfs dfs -rm /rjcnew/file_02
22/02/14 19:59:14 INFO fs.TrashPolicyDefault: Namenode trash configuration: D
Deleted /rjcnew/file_02
[cloudera@quickstart ~]$ hdfs dfs -ls /rjcnew
Found 1 items
-rw-r--r--   1 cloudera supergroup         45 2022-02-14 19:40 /rjcnew/file_0
[cloudera@quickstart ~]$ █
```

11)->hdfs dfs -rm -r /filename or hdfs dfs –rmdir /filename

In case, we want to delete a directory which contains files, -rm will not be able to delete the directory. In that case we can use recursive option for removing all the files from the directory following by removing the directory when it is empty.

```
[cloudera@quickstart ~]$ hdfs dfs -rm /rjcnew/file_01
22/02/14 19:59:56 INFO fs.TrashPolicyDefault: Namenode trash configuration: D
eletion interval = 0 minutes, Emptier interval = 0 minutes.
Deleted /rjcnew/file_01
[cloudera@quickstart ~]$ hdfs dfs -rmdir /rjcnew
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 6 items
drwxr-xr-x   - hbase    supergroup          0 2022-02-10 20:00 /hbase
drwxr-xr-x   - cloudera supergroup          0 2022-02-14 19:55 /rjc
drwxr-xr-x   - solr     solr                0 2015-06-09 03:38 /solr
drwxrwxrwx   - hdfs     supergroup          0 2022-02-07 21:01 /tmp
drwxr-xr-x   - hdfs     supergroup          0 2015-06-09 03:38 /user
drwxr-xr-x   - hdfs     supergroup          0 2015-06-09 03:36 /var
[cloudera@quickstart ~]$ █
```

12) MoveFromLocal Command
->hdfs dfs -moveFromLocal /home/cloudera/Desktop/new /outputdir
The Hadoop fs shell command moveFromLocal moves the file or directory from the local filesystem to the destination in Hadoop HDFS.

```
[cloudera@quickstart ~]$ hdfs dfs -moveFromLocal /home/cloudera/Desktop/rjclocal/file_03 /rjc
[cloudera@quickstart ~]$ ls Desktop/rjclocal
file_01  file_03~
[cloudera@quickstart ~]$ hdfs dfs -ls /rjc
Found 2 items
-rw-r--r--   1 cloudera supergroup         45 2022-02-14 18:58 /rjc/file_01
-rw-r--r--   1 cloudera supergroup         29 2022-02-14 20:09 /rjc/file_03
```

13) Tail Command
->hdfs dfs -tail /rjc/file_01
The Hadoop fs shell tail command shows the last 1KB of a file on console or stdout.

```
tail:  /rjc : is a directory
[cloudera@quickstart ~]$ hdfs dfs -tail /rjc/file_01

I am a student of MSC DSAI Batch 2021-2022.
```

14) Expunge Command
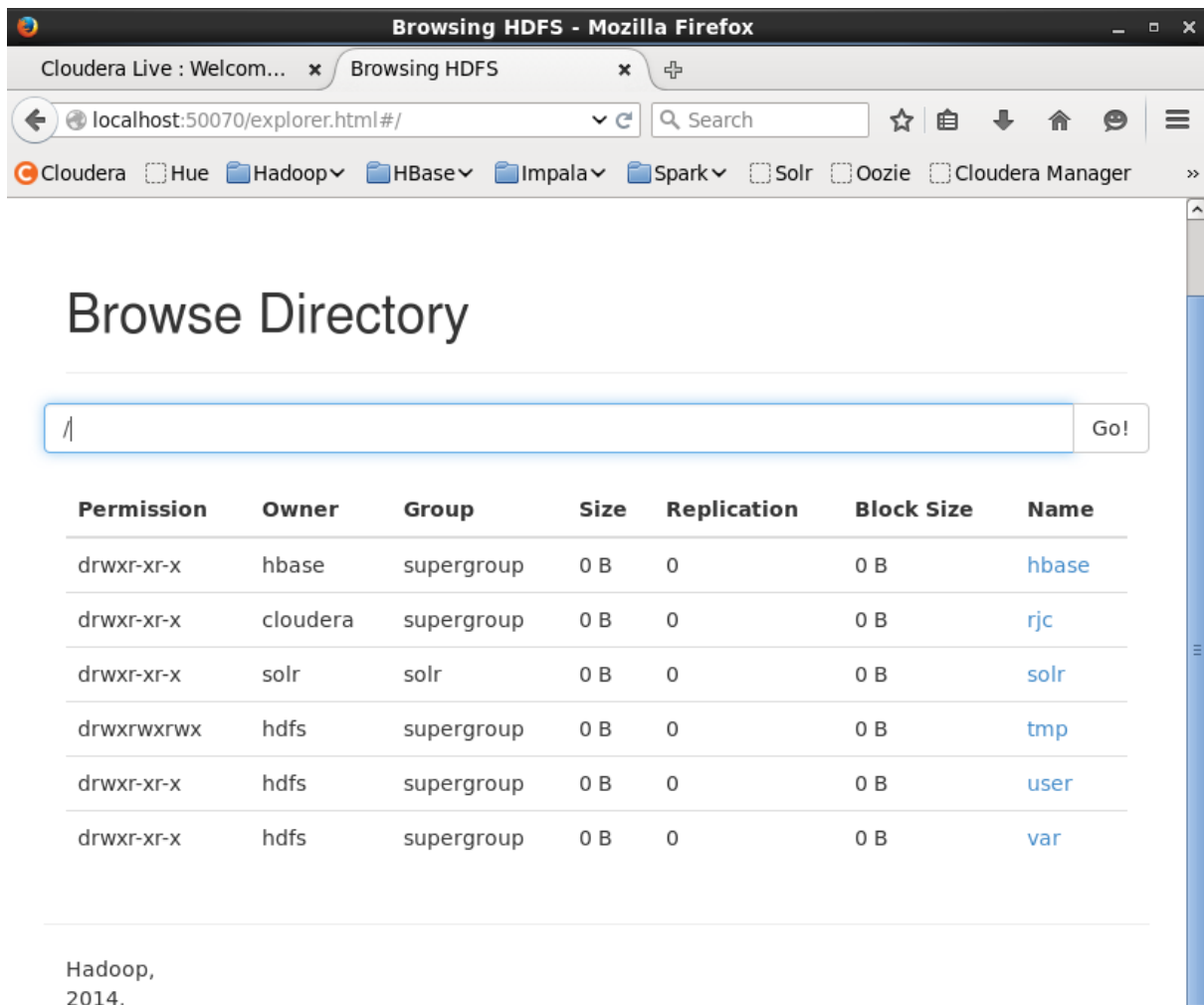**->hdfs dfs -expunge**
This command is used to empty the trash available in an HDFS system.

```
[cloudera@quickstart ~]$ hdfs dfs -expunge
22/02/14 20:24:34 INFO fs.TrashPolicyDefault: Namenode trash configuration: Deletion interval = 0 m
[cloudera@quickstart ~]$
```

15) Replication Command
Earlier Replication Number of file_01 is 1.

->hdfs dfs -setrep 4 /output_abc
This command is used to change the replication factor of a file to a specific count
instead of the default replication factor for the remaining in the HDFS file system.

```
[cloudera@quickstart ~]$
[cloudera@quickstart ~]$ hdfs dfs -setrep 3 /rjc
Replication 3 set: /rjc/file_01
Replication 3 set: /rjc/file_03
```

Now Replication factor will become 3.
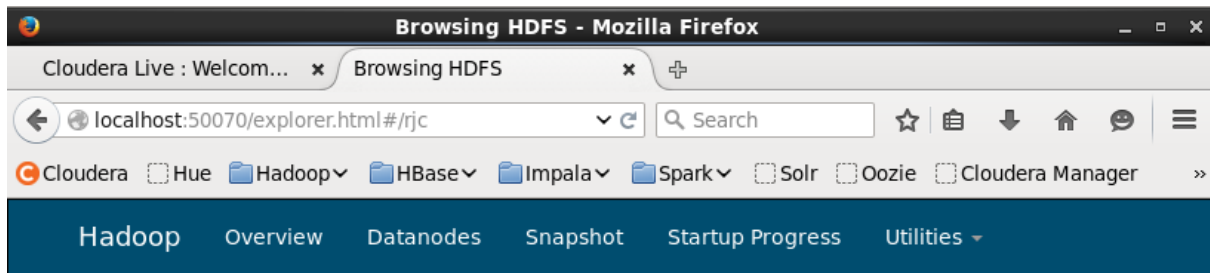
16) DU Command
->hdfs dfs -du /rjc

Use the hdfs du command to get the size of a directory in HDFS du stands for disk usage.

Since the replication factor of file_01 was set as 3 earlier we could see that 45*3=135 and for file_02 will become 29*3 =87.

```
[cloudera@quickstart ~]$ hdfs dfs -du /rjc
45  135  /rjc/file_01
29  87   /rjc/file_03
[cloudera@quickstart ~]$
```

17) appendToFile Command

This command is used to append the text in the file on the Hadoop. We can write the text with the help of echo command.

```
[cloudera@quickstart ~]$ echo "Appendind the tetx in hadoop"|hdfs dfs -appendToFile - /rjc/file1
[cloudera@quickstart ~]$ hdfs dfs -du /rjc/file1
29  29  /rjc/file1
[cloudera@quickstart ~]$
```

18) Df command
->hdfs dfs -df

To get all the space related details of the Hadoop File System we can use df command. It provides the information regarding the amount of space used and amount of space available on the currently mounted filesystem.

```
[cloudera@quickstart ~]$ hdfs dfs -df
Filesystem                          Size        Used   Available  Use%
hdfs://quickstart.cloudera:8020  58665738240  394735616  48010096640   1%
```

->hdfs dfs -df -h

With h parameter the information is human readable.

```
[cloudera@quickstart ~]$ hdfs dfs -df -h
Filesystem                       Size     Used  Available  Use%
hdfs://quickstart.cloudera:8020  54.6 G  376.4 M    44.7 G    1%
[cloudera@quickstart ~]$
```

19) Fsck command

The fsck Hadoop command is used to check the health of the HDFS. It moves a corrupted file to the lost+found directory. It deletes the corrupted files present in HDFS. It prints the files being checked**.**

```
[cloudera@quickstart ~]$ hdfs fsck /rjc
Connecting to namenode via http://quickstart.cloudera:50070
FSCK started by cloudera (auth:SIMPLE) from /127.0.0.1 for path /rjc at Mon Feb 14 20:51:48 PST 202
2
.
/rjc/file_01:  Under replicated BP-989008105-127.0.0.1-1433846136903:blk_1073742230_1412. Target Re
plicas is 3 but found 1 replica(s).
.
/rjc/file_03:  Under replicated BP-989008105-127.0.0.1-1433846136903:blk_1073742234_1416. Target Re
plicas is 3 but found 1 replica(s).
Status: HEALTHY
 Total size:    74 B
 Total dirs:    1
 Total files:   2
 Total symlinks:             0
 Total blocks (validated):    2 (avg. block size 37 B)
 Minimally replicated blocks:  2 (100.0 %)
 Over-replicated blocks:      0 (0.0 %)
 Under-replicated blocks:     2 (100.0 %)
 Mis-replicated blocks:       0 (0.0 %)
 Default replication factor:  1
 Average block replication:   1.0
 Corrupt blocks:              0
 Missing replicas:            4 (66.666664 %)
 Number of data-nodes:        1
 Number of racks:             1
FSCK ended at Mon Feb 14 20:51:48 PST 2022 in 3 milliseconds


The filesystem under path '/rjc' is HEALTHY
[cloudera@quickstart ~]$
```

->hdfs fsck /rjc -files

```
[cloudera@quickstart ~]$ hdfs fsck /rjc -files
Connecting to namenode via http://quickstart.cloudera:50070
FSCK started by cloudera (auth:SIMPLE) from /127.0.0.1 for path /rjc at Mon Feb 14 20:53:50 PST 2022
/rjc <dir>
/rjc/file_01 45 bytes, 1 block(s):  Under replicated BP-989008105-127.0.0.1-1433846136903:blk_1073742230_1412. Targ
et Replicas is 3 but found 1 replica(s).
/rjc/file_03 29 bytes, 1 block(s):  Under replicated BP-989008105-127.0.0.1-1433846136903:blk_1073742234_1416. Targ
et Replicas is 3 but found 1 replica(s).
Status: HEALTHY
 Total size:    74 B
 Total dirs:    1
 Total files:   2
 Total symlinks:              0
 Total blocks (validated):    2 (avg. block size 37 B)
 Minimally replicated blocks: 2 (100.0 %)
 Over-replicated blocks:      0 (0.0 %)
 Under-replicated blocks:     2 (100.0 %)
 Mis-replicated blocks:       0 (0.0 %)
 Default replication factor:  1
 Average block replication:   1.0
 Corrupt blocks:              0
 Missing replicas:            4 (66.666664 %)
 Number of data-nodes:        1
 Number of racks:             1
FSCK ended at Mon Feb 14 20:53:50 PST 2022 in 0 milliseconds


The filesystem under path '/rjc' is HEALTHY
```

20) Touchz Command
It creates an empty file.

```
[cloudera@quickstart ~]$ hdfs dfs -touchz /rjc/file1
[cloudera@quickstart ~]$ hdfs dfs -ls /rjc
Found 3 items
-rw-r--r--   1 cloudera supergroup          0 2022-02-14 20:55 /rjc/file1
-rw-r--r--   3 cloudera supergroup         45 2022-02-14 18:58 /rjc/file_01
-rw-r--r--   3 cloudera supergroup         29 2022-02-14 20:09 /rjc/file_03
```

21) Stat Command
The Hadoop fs shell command stat prints the statistics about the file or directory in the specified Format.
It shows the recent date of modification.
->hdfs dfs -stat /rjc

22) ->hdfs dfs -stat %b /rjc/file_01
%b shows byte size of file

```
[cloudera@quickstart ~]$ hdfs dfs -stat %b /rjc/file1
0
[cloudera@quickstart ~]$ hdfs dfs -stat %b /rjc/file_01
45
```

23) Checksum Command

Checksum property, which defaults to 512 bytes. The chunk size is stored as metadata in the crc file, so the file can be read back correctly even if the setting for the chunk size has change.

```
[cloudera@quickstart ~]$ hdfs dfs -checksum /rjc/file1
/rjc/file1      MD5-of-0MD5-of-512CRC32C          000002000000000000000000000498fceae9f36ce61fbec5e48fee2b07d
[cloudera@quickstart ~]$ █
```

24) Help command

->hdfs dfs -help mkdir

Shows the syntax of whereas commands

```
[cloudera@quickstart ~]$ hdfs dfs -help mkdir
-mkdir [-p] <path> ... :
  Create a directory in specified location.

  -p  Do not fail if the directory already exists
```