1. **mkdir:**

**Description:** mkdir command in Linux allows the user to create directories (also referred to as folders in some operating systems ).

**Command Functionality:** This command can create multiple directories at once as well as set the permissions for the directories.

**Syntax:** $ hadoop fs -mkdir  [-p] <path>

**Example:** $ hadoop fs -mkdir /user/hadoop/

1. **ls:**

**Description:** We use the **ls** command to enlist the files and directories present in HDFS.

**Command Functionality:** 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.

**Syntax:** $ hadoop fs -ls [-d] [-h] [-R] <path>

**Example:** $ hadoop fs -ls /

1. **put:**

**Description:** Copies files from local file system to HDFS.

**Command Functionality:** 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.

**Syntax:** $ hadoop fs -put [-f] [-p] <localsrc> ... <dst>

**Example:** $ hadoop fs -put sample.txt /user/data/

1. **get:**

**Description:** Copies files from HDFS to local file system. This is similar to **-copyToLocal** command.

**Command Functionality:** The Hadoop fs shell command [**get**](https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html#get) copies the file or directory from the Hadoop file system to the local file system.

**Syntax:** $ hadoop fs -get <src> <localdest>

**Example:** $ hadoop fs -get /user/data/sample.txt workspace/

1. **cat:**

**Description:** similar to Unix cat command, it is used for displaying contents of a file.

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

**Syntax:** $ hadoop fs –cat /path\_to\_file\_in\_hdfs

**Example:** $ hadoop fs -cat /user/data/sample.txt

1. **cp:**

**Description:** Similar to Unix cp command, it is used **for copying files** from one directory to another **within HDFS.**

**Command Functionality:** The **cp** command copies a file from one directory to another directory within the HDFS.

**Syntax:** $ hadoop fs -cp <src> <dest>

**Example:** $ hadoop fs -cp /user/data/sample.txt /user/hadoop

1. **mv:**

**Description:** Similar to Unix mv command, it is used **for moving a file** from one directory to another **within HDFS**.

**Command Functionality:** The HDFS mv command moves the files or directories from the source to a destination within [HDFS](http://data-flair.training/blogs/comprehensive-hdfs-guide-introduction-architecture-data-read-write-tutorial/)**.**

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

**Example:** $ hadoop fs -mv /user/hadoop/sample.txt /user/test/

1. **rm:**

**Description:** Remove a file from HDFS, similar to Unix rm command. This command does not delete directories. For recursive delete, use command -rm -r.

**Command Functionality:** Delete files specified as args.

**Syntax:** hadoop fs -rm [-f] [-r |-R] [-skipTrash] [-safely] URI [URI ...]

**Example:** hadoop fs -rm hdfs://nn.example.com/file /user/hadoop/emptydir

1. **getmerge:**

**Description:** used to merge multiple files in an HDFS(Hadoop Distributed File System) and then put it into one single output file in our local file system.

**Command Functionality:** Takes a source directory and a destination file as input and concatenates files in src into the destination local file. Optionally -nl can be set to enable adding a newline character (LF) at the end of each file. -skip-empty-file can be used to avoid unwanted newline characters in case of empty files.

**Syntax:** hadoop fs -getmerge [-nl] <src> <localdst>

**Example:** hadoop fs -getmerge -nl /src /opt/output.txt

1. **setrep:**

**Description:** This command is **used to change the replication factor of a file/directory in HDFS**.

**Command Functionality:** Changes the replication factor of a file. If *path* is a directory then the command recursively changes the replication factor of all files under the directory tree rooted at *path*. The EC files will be ignored when executing this command.

**Syntax:** hadoop fs -setrep [-R] [-w] <numReplicas> <path>

**Example:** hadoop fs -setrep -w 3 /user/hadoop/dir1

1. **touchz:**

**Description:** It creates an empty file.

**Command Functionality:** Create a file of zero length. An error is returned if the file exists with non-zero length.

**Syntax:** hadoop fs -touchz <pathname>

**Example:** Hadoop fs -touchz /student/file1.txt

1. **test:**

**Description:** Test is used as part of the conditional execution of shell commands.

**Command Functionality:** The test command is used to check file types and compare values. Test is used in conditional execution.It is used for:

1. File attributes comparisons.

2. Perform string comparisons.

3. Basic arithmetic comparisons.

**Syntax:** test *EXPRESSION*

**Example:** test 100 -gt 99 && echo "Yes, that's true." || echo "No, that's false."

1. **appendToFile:**

**Description:** The HDFS fs shell command appendToFile appends the content of single or multiple local files specified in the localsrc to the provided destination file on the HDFS.

**Command Functionality:** The destination file gets created if it does not exist earlier. Append single src, or multiple srcs from local file system to the destination file system. Also reads input from stdin and appends to destination file system.

**Syntax:** hadoop fs -appendToFile <localsrc> <dest>

**Example:** hadoop fs -appendToFile - hdfs://nn.example.com/hadoop/hadoopfile Reads the input from stdin.

1. **df:**

**Description:** Class DF. java.lang.Object org.apache.hadoop.util.Shell org.apache.hadoop.fs.DF public class DF extends Shell. Filesystem disk space usage statistics. Uses the unix 'df' program to get mount points, and java.

**Command Functionality:** The Hadoop fs shell command df shows the capacity, size, and free space available on the HDFS file system. The -h option formats the file size in the human-readable format.

**Syntax:** hadoop fs -df [-h] <path>

**Example:** hadoop dfs -df /user/hadoop/dir1

1. **du:**

**Description:** this Hadoop fs shell command du prints a summary of the amount of disk usage of all files/directories in the path.

**Command Functionality:** Displays sizes of files and directories contained in the given directory or the length of a file in case its just a file.

**Syntax:** hadoop fs -du [-s] [-h] [-v] [-x] URI [URI ...]

**Example:** hadoop fs -du /user/hadoop/dir1 /user/hadoop/file1 hdfs://nn.example.com/user/hadoop/dir1

1. **count:**

**Description:** The Hadoop fs shell command count counts the number of files, directories, and bytes under the paths that matches the specified file pattern.

**Options:**  
**-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

**Command Functionality:** counts the number of files, directiories and bytes

**Syntax:** hadoop fs -count [options] <path>

**Example:** hadoop fs -count -v/

1. **chgrp:**

**Description:** The Hadoop fs shell command chgrp changes the group of the file specified in the path. The user must be the owner of the file or superuser.

**Command Functionality:** **chgrp command** in Linux is used to change the group ownership of a file or directory. All files in Linux belong to an owner and a group. You can set the owner by using “[chown](https://www.geeksforgeeks.org/chown-command-in-linux-with-examples/" \t "_blank)” command, and the group by the “chgrp” command.

**Syntax:** hadoop fs -chgrp <group> <path>

**Example:** hadoop fs -chgrp newgroup /sample

1. **chmod:**

**Description:** *-chmod* that stands for change mode command is used for changing the permission for the files in our HDFS.

**Command Functionality:** The Hadoop fs shell command chmod changes the permissions of a file.

The -R option recursively changes files permissions through the directory structure.

The user must be the owner of the file or superuser.

**Syntax:** hadoop fs -chmod [-R] <mode> <path>

**Example:** hadoop fs -chmod -r /testfile

1. **chown:**

**Description:** The Hadoop fs shell command chown changes the owner of the file.

The -R option recursively changes files permissions through the directory structure. The user must be the owner of the file or superuser.

**Command Functionality:** changes the owner of the file

**Syntax:** hadoop fs -chown [-R] [owner] [:[group]] <path>

**Example:** hadoop fs -chown newdataflair /sample