1. Using the `hdfs stat` Command to Get Details about a File

Although the `hdfs dfs –ls` command lets you get the file information you need, there are times when you need specific bits of information from HDFS. When you run the `hdfs dfs –ls` command, it returns the complete path of the file. When you want to see only the base name, you can use the `hdfs –stat` command to view only specific details of a file.

You can format the `hdfs –stat` command with the following options:

```
%b Size of file in bytes
%F Will return "file", "directory", or "symlink" depending on the type of inode
%g Group name
%n Filename
%o HDFS Block size in bytes ( 128MB by default )
%r Replication factor
%u Username of owner
%y Formatted mtime of inode
%Y UNIX Epoch mtime of inode
```

2. Listing HDFS Files and Directories

As with regular Linux file systems, use the `ls` command to list HDFS files. You can specify various options with the `ls` command, as shown here:

```
$ hdfs dfs -usage ls

Usage: hadoop fs [generic options] -ls [-d] [-h] [-R] [<path> ...]
bash-4.2$
Here's what the options stand for:
-d: Directories are listed as plain files.
-h: Format file sizes in a human-readable fashion (eg 64.0m instead of
67108864).
-R: Recursively list subdirectories encountered.
-t: Sort output by modification time (most recent first).
-S: Sort output by file size.
-r: Reverse the sort order.
-u: Use access time rather than modification time for display and sorting.
```

3.Removing HDFS Files and Directories

HDFS file and directory removal commands work similar to the analogous commands in the Linux file system. The `rm` command with the `–R` option removes a directory and everything under that directory in a recursive fashion. Here’s an example.

```
$ hdfs dfs -rm -R /user/alapati
15/05/05 12:59:54 INFO fs.TrashPolicyDefault: Namenode trash configuration:
Deletion interval = 1440 minutes, Emptier interval = 0 minutes.
Moved: 'hdfs://hadoop01-ns/user/alapati' to trash at: hdfs://hadoop01-ns/user/
hdfs/.Trash/Current
$
```

4.Changing HDFS File Permissions

You can use the `chmod` command to change the permissions of a file or directory. You can use standard Linux file permissions. Here's the general syntax for using the `chmod` command:

```
hdfs dfs –chmod [-R] <mode> <file/dir>
```

You must be a super user or the owner of a file or directory to change its permissions.

With the `chgrp`, `chmod` and `chown` commands you can specify the `–R` option to make recursive changes through the directory structure you specify.

In this section, I'm using HDFS commands from the command line to view and manipulate HDFS files and directories. However, there's an even easier way to access HDFS, and that's through Hue, the web-based interface, which is extremely easy to use and which lets you perform HDFS operations through a GUI. Hue comes with a File Browser application that lets you list and create files and directories, download and upload files from HDFS and copy/move files. You can also use Hue's File Browser to view the output of your MapReduce jobs, Hive queries and Pig scripts.

While the `hdfs dfs` utility lets you manage the HDFS files and directories, the `hdfs dfsadmin` utility lets you perform key HDFS administrative tasks. In the next section, you'll learn how to work with the `dfsadmin` utility to manage your cluster.

5.Using the dfsadmin Utility to Perform HDFS Operations

The `hdfs dfsadmin` command lets you administer HDFS from the command line. While the `hdfs dfs` commands you learned about in the previous section help you manage HDFS files and directories, the `dfsadmin` command is useful for performing general HDFS-specific administrative tasks. It's a good idea to become familiar with all the options that are available for the `dfsadmin` utility by issuing the following command:

```
$ hdfs dfsadmin -help
hdfs dfsadmin performs DFS administrative commands.
Note: Administrative commands can only be run with superuser permission.
The full syntax is:
hdfs dfsadmin
        [-report [-live] [-dead] [-decommissioning]]
        [-safemode <enter | leave | get | wait>]
        [-saveNamespace]
...
$
```