

## **HDFS Exercises**

**Create a hdfs directory called proj1**

```
hdfs dfs -mkdir proj1
```

**Create a file sample1.txt and write some content and save it in the current directory in your local file system. Copy the sample1.txt to the proj1 directory of hdfs**

```
hdfs dfs -put sample1.txt proj1
```

**List the contents of proj1 directory**

```
hdfs dfs -ls proj1
```

**Create another directory in hdfs home directory called proj2**

```
hdfs dfs -mkdir proj2
```

**Create another file sample2.txt and enter some content and save it. Write this file to proj2 directory of hdfs**

```
hdfs dfs -put sample2.txt proj2
```

**Read the file sample2.txt from proj2 directory and print the contents on the screen.**

```
hdfs dfs -cat proj2/sample2.txt
```

**Copy the hdfs file(sample1.txt) from proj1 directory to proj2 directory in hdfs**

```
hdfs dfs -cp proj1/sample1.txt proj2
```

**Returns all the available files and recursively lists all the subdirectories under HDFS home directory**

```
hdfs dfs -ls -R <HDFS Home dir>
```

**Delete the sample1.txt file from proj1 directory of hdfs**

```
hdfs dfs -rm proj1/sample1.txt
```

**Move the sample1.txt file from proj2 directory of hdfs to proj1 of hdfs**

```
hdfs dfs -mv proj2/sample1.txt proj1
```

**Copy a directory in to hdfs. Create a directory in your local file system(test) with a few files. Then copy the whole directory to HDFS. If no target is specified, directory will be copied to HDFS Home directory**

```
hdfs dfs -put test
```

**Change the permissions of the file1.txt in proj1 directory to give complete access**

```
hdfs dfs -chmod 777 proj1/sample1.txt
```

**Make some changes in your local copy of sample1.txt file and write it back to the hdfs proj1 directory**

```
hdfs dfs -put sample1.txt proj1  
hdfs dfs -put -f sample1.txt proj1
```

**Get a copy of the sample1.txt in to your local system with a different name “sample-local.txt” in to a directory called data in your local file system**

```
hdfs dfs -get proj1/sample1.txt data/sample-local.txt
```

**Remove the proj1 directory**

```
hdfs dfs -rm -r proj1
```

**Remove the test2 directory**

```
hdfs dfs -rm -r proj2
```

**Create multiple directories with a single command in hdfs**

```
hdfs dfs -mkdir maindir/subdir  
hdfs dfs -mkdir -p maindir/subdir  
hdfs dfs -mkdir -p proj1/module1
```

**Change the permissions of the subdir directory above to 755**

```
hdfs dfs -chmod 755 proj1/module1
```

**Displays the disk usage for all the files available under a given directory.**

```
hdfs dfs -du /
```

**Displays disk usage of current hadoop distributed file system.**

```
hdfs dfs -df
```

**This hadoop command will show the last kilobyte of the file to stdout.**

```
hdfs dfs -tail <hdfs Path>
```

Note : -f option shows page wise format

**Create an empty file at the specified location.**

```
hdfs dfs -touchz <hdfs Path>/newfile.txt
```

**This hadoop command is used to set the replication for a specific file.**

```
hdfs dfs -setrep -w 1 /user/bigdata/Sample1.txt
```

**This hadoop command is basically used to change the group name**

```
hdfs dfs -chgrp -R hadoop /tmp
```

**This hadoop command is used to change the owner**

```
hdfs dfs -chown -R hadoop /tmp
```

**Returns the checksum information of a file.**

```
hdfs dfs -checksum output/part-r-00000
```

**Count the number of directories, files, and bytes under the paths that match the specified file pattern. The output columns with -count are: DIR\_COUNT, FILE\_COUNT, CONTENT\_SIZE, PATHNAME**

```
hdfs dfs -count /user/ubuntu
```

**Finds all files that match the specified expression with -name.  
If no path is specified then defaults to the current working  
directory.**

```
hdfs dfs -find [path to search]-name <file to search>
```