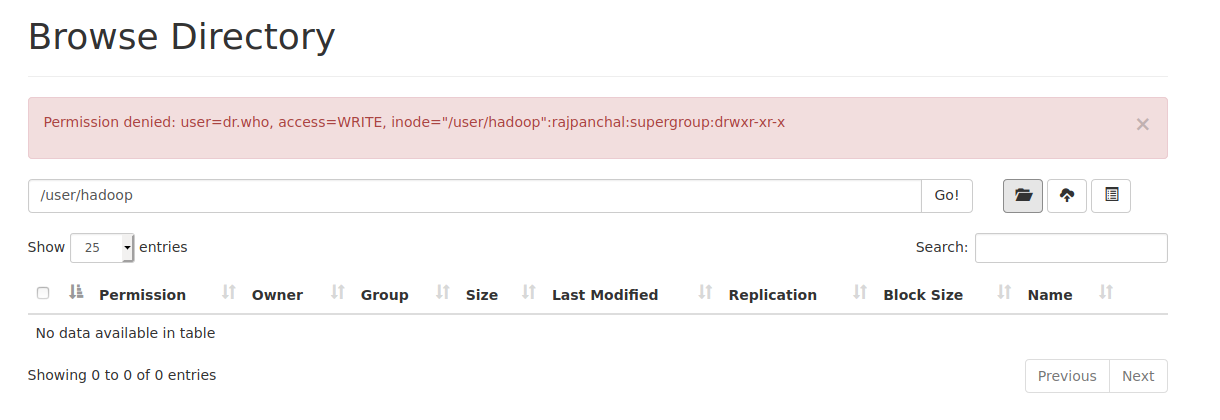
**Aim: Interfacing to Distributed File System – HDFS commands and Web GUI file system browser usage.**

**Create a directory named ‘Test’ from HDFS web UI in ‘/user/hadoop’. It should give an error message. Read this message and solve the error.**

Step 1: When you try to directory from the web UI it gives an error like below.

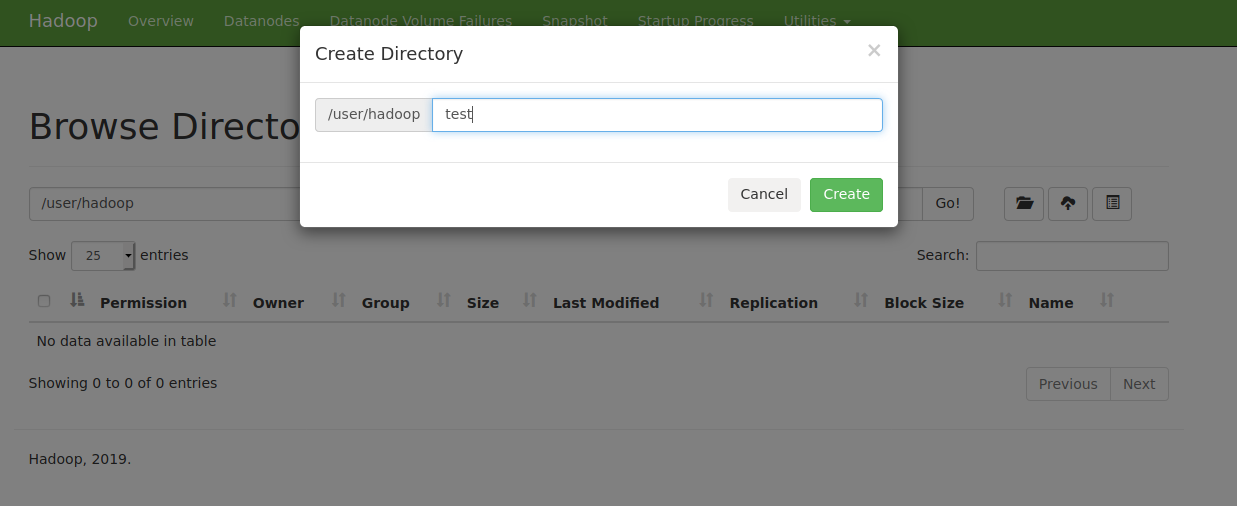


Step 2: To solve this error use the following command.

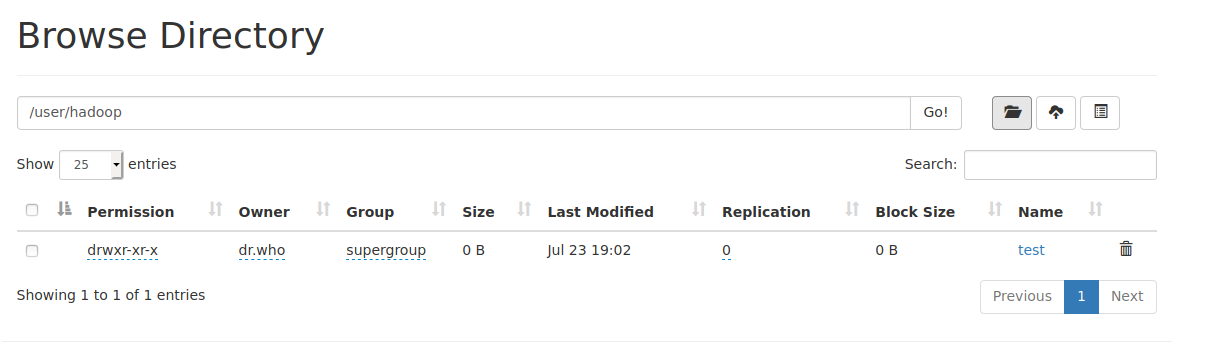
**-hadoop fs -chmod 777 <folder\_name>**



Step 3: Now try to create a directory from the web UI.



Step 4: Directory has been successfully created.



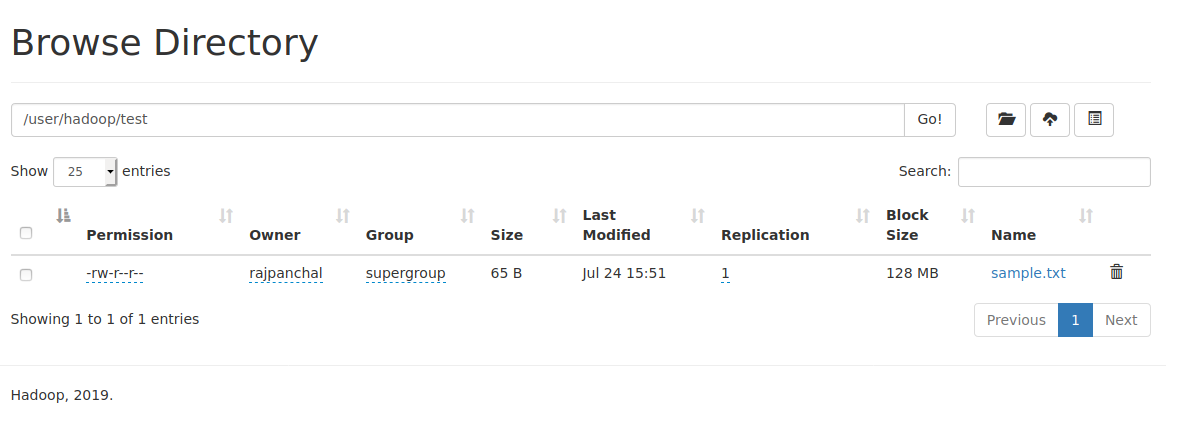
**Create a text file in the local file system and put the same file in HDFS using appropriate command.**

Step 1: Create the sample.txt file on Desktop and add some content into it. Then run the following command to put the same file in HDFS.

**-hadoop fs -put <source> <dest>**



Step 2: Locate the file through hdfs web GUI on localhost:9870.



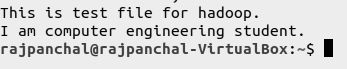
**Display the contents of this file of HDFS on the terminal.**

Step 1: Using the following command you can display the contents file which you have just created.

**-hadoop fs -cat <file path>**



Step 2: See the content of the file.



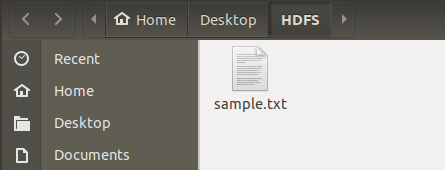
**Download this file of HDFS on a local machine and change its contents. Upload this modified file again on HDFS and verify its contents. Also make sure that this file should be deleted from the local machine automatically.**

Step 1: Download the file of HDFS on a local machine.

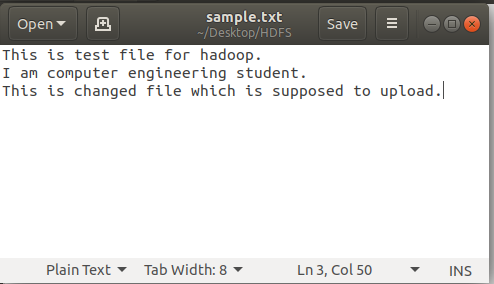
**-hadoop fs -get <source> <dest>**



Step 2: Locate the file in your local system.



Step 3: Change it content by using text editor and rename to **usample.txt**.



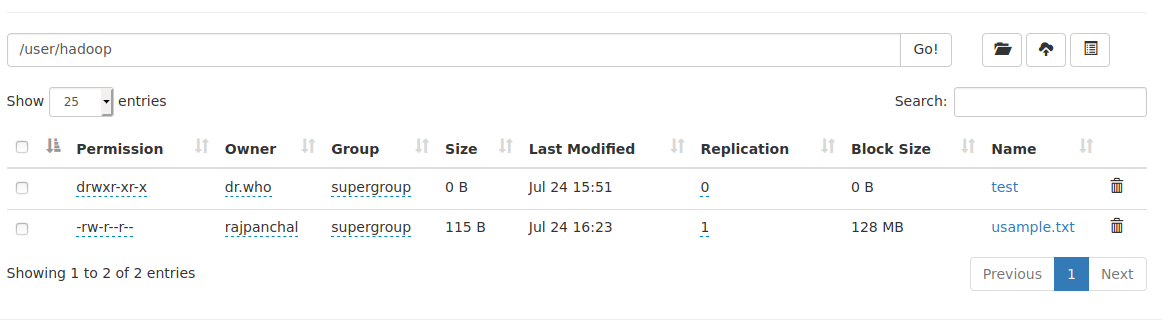
Step 4: Use the following command to move file from the local machine to hdfs .

**-hadoop fs -moveFromLocal <source> <dest>**

This command delete file from the local machine directly.



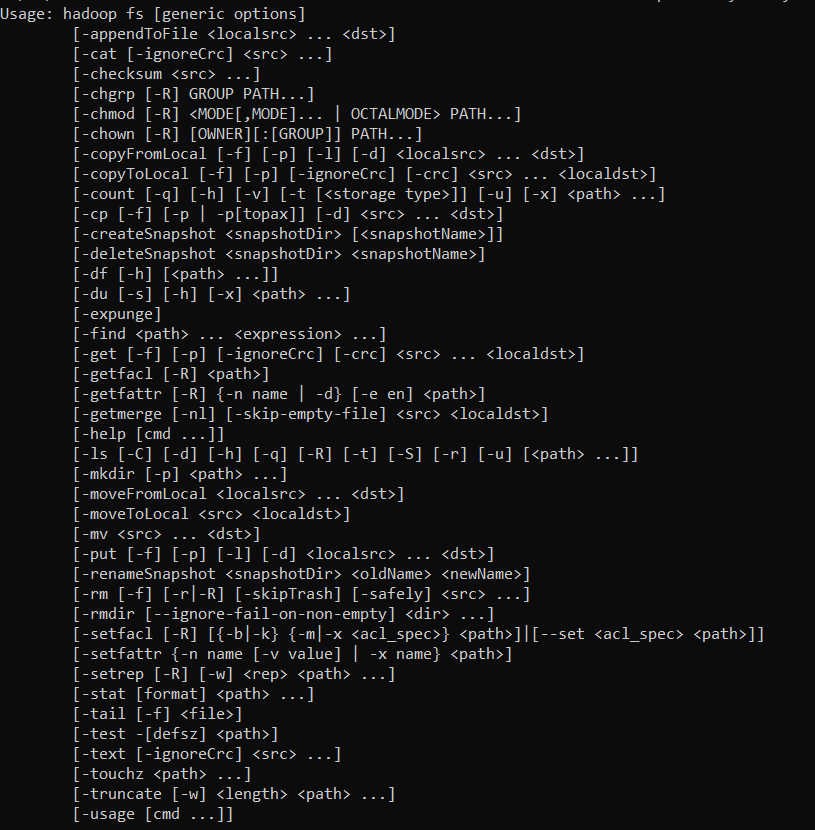
Step 5: Now you are able to see that usample.txt file delete from your local machine and locate the file in web UI.



**Find out a CLI way to display all file system commands supported by hdfs. Record the list.**

Step 1: Use the following command to get a list of all file system commands.

**-hdfs** **dfs** or **hadoop fs**

****

* **mkdir** : similar to Unix mkdir command, it is used for creating directories in HDFS.
* **ls** : similar to Unix ls command, it is used for listing directories in HDFS. The **-lsr** command can be used for **recursive listing**.
* **put**: copies files from local file system to HDFS. This is similar to **-copyFromLocal** command.
* **get**: copies files from HDFS to local file system. This is similar to **-copyToLocal** command.
* **cat**: similar to Unix cat command, it is used for displaying contents of a file.
* **cp**: similar to Unix cp command, it is used **for copying files** from one directory to another **within HDFS.**
* **mv**: similar to Unix mv command, it is used **for moving a file** from one directory to another **within HDFS**.
* **rm**: similar to Unix rm command, it is used **for removing a file** from HDFS. The command **-rmr** can be used for recursive delete.
* **getmerge**: **it is one of the important and useful command** when trying to read the contents of map reduce job or pig job’s output files. It is used for **merging a list of files in one directory on HDFS into a single file on a local file system.**
* **setrep:** this command is used to change the replication factor of a file to a specific instead of the default of replication factor for the remaining in HDFS.
* **touchz**: this command can be used to **create a file of zero length** in HDFS.
* **test**: this command can be used to **test a hdfs file’s existence or zero length or is it a directory**.
* **expunge**: this command is used to empty the trash in hadoop file system.
* **appendToFile**: appends the contents of all the given local files to the given destination file on HDFS. The destination file will be created if it does not exist.
* **tail**: shows the last 1KB of the file.
* **stat**: this option prints statistics about the file/directory at <path> in the specified format.
* **setfattr**: sets an extended attribute name and value for a file or directory.