

PRACTICAL -03

AIM:- Implement the following file management tasks in Hadoop:-

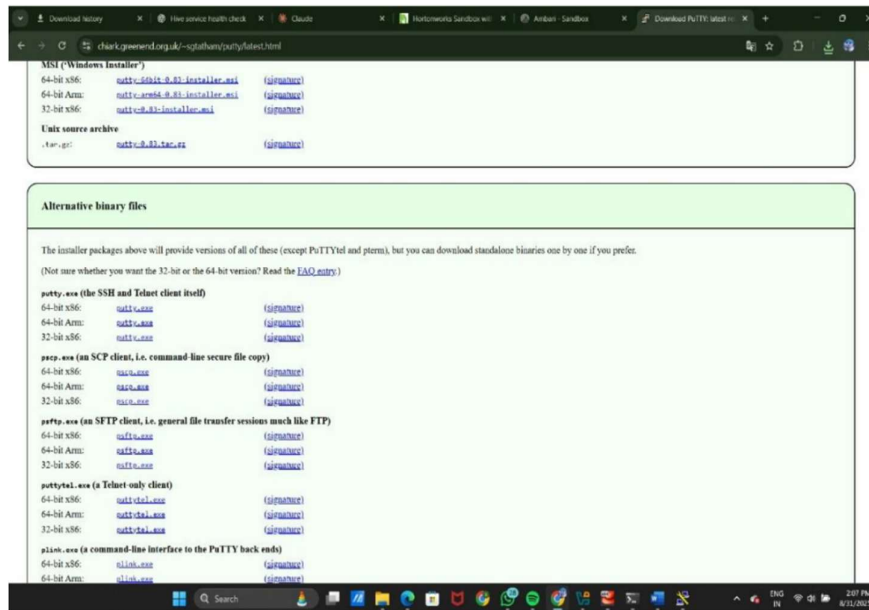
> Adding files and directories

> Retrieving files from HDFS to local file system

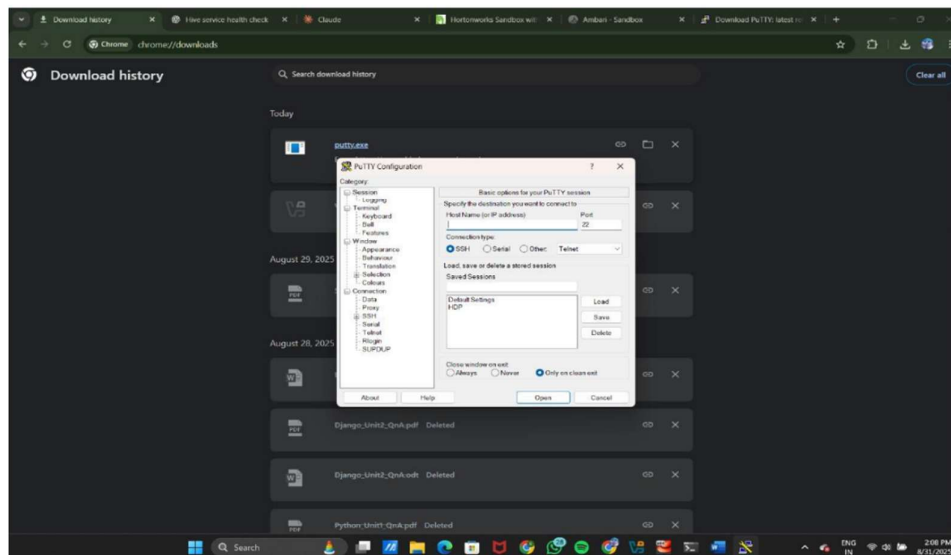
> Deleting files from HDFS

1) To give commands in HDFS download the platform putty it gets directly connected with the HDFS dashboard and from where you can give commands to add & delete the files

Download Links- <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>



After downloading open the file and give following details



Host name- 127.0.0.1

Port- 2222

Connection type- SSH

Load server- HDP & Save

After saving you will get to see the command prompt where you have to enter the password which you have been set for your browser dashboard

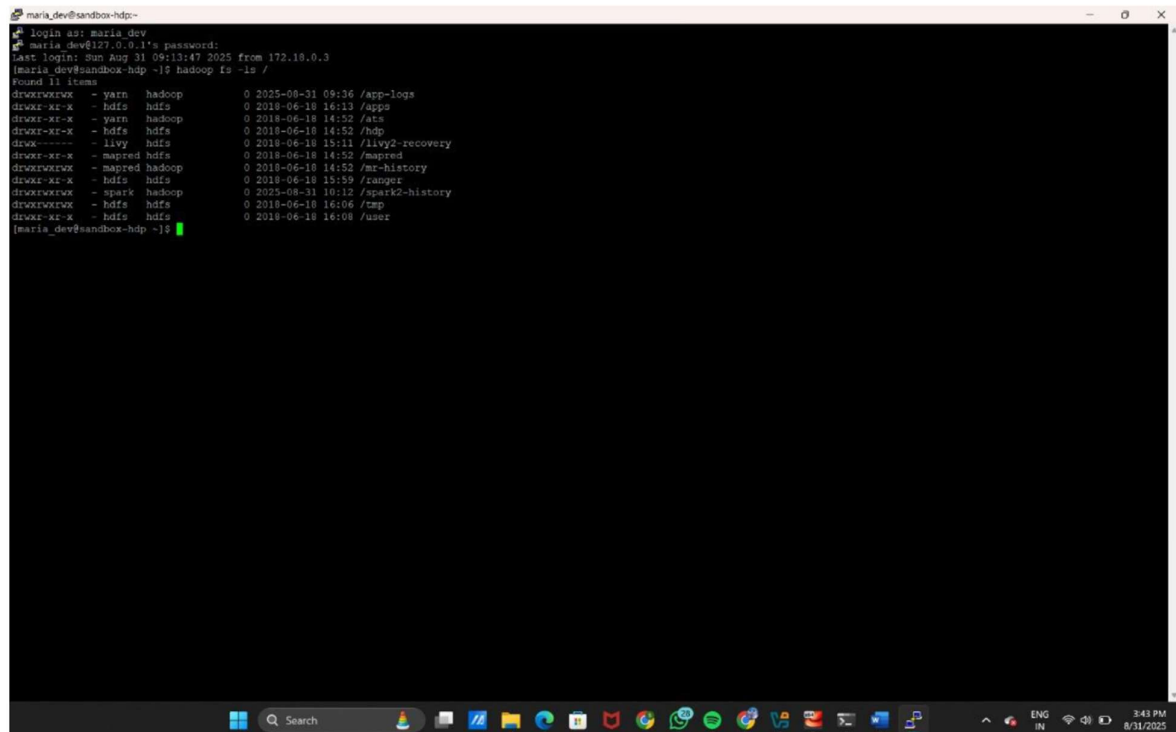
Password- maria_dev

2) To go in the Hadoop system give the command-

*hadoop fs -ls

The command **hadoop fs -ls** is used to **list files and directories stored in Hadoop Distributed File System (HDFS)** or other supported file systems (like local FS, S3, etc., depending on configuration).

Shows the **files and directories** at the given path.



```
maria_dev@sandbox-hdp:~$ login as: maria_dev
maria_dev@127.0.0.1's password:
Last login: Sun Aug 31 09:13:47 2025 from 172.18.0.3
maria_dev@sandbox-hdp:~$ hadoop fs -ls /
Found 11 items
drwxr-xr-x - yarn  hadoop      0 2025-08-31 09:36 /app-logs
drwxr-xr-x - hdfs  hdfs      0 2019-06-18 16:13 /apps
drwxr-xr-x - yarn  hadoop      0 2019-06-18 14:52 /ats
drwxr-xr-x - hdfs  hdfs      0 2019-06-18 14:52 /hdp
drwxr-xr-x - livy   hdfs      0 2019-06-18 15:11 /livy2-recovery
drwxr-xr-x - mapred hdfs      0 2019-06-18 14:52 /mapred
drwxr-xr-x - mapred hadoop      0 2019-06-18 14:52 /mr-history
drwxr-xr-x - hdfs  hdfs      0 2019-06-18 15:59 /ranger
drwxr-xr-x - spark hadoop      0 2025-08-31 10:12 /spark2-history
drwxr-xr-x - hdfs  hdfs      0 2019-06-18 16:06 /tmp
drwxr-xr-x - hdfs  hdfs      0 2019-06-18 16:08 /user
maria_dev@sandbox-hdp:~$
```

Displays **metadata**:

- File permissions
- Replication factor
- Owner & group
- File size (in bytes)
- Last modification date & time
- Path

*Hadoop fs -mkdir

The **hadoop fs -mkdir** command is used to **create new directories in Hadoop Distributed File System (HDFS)** (or any other file system supported by Hadoop, like S3, local FS, etc., depending on your configuration)

Purpose

- To create a **new directory** in HDFS.

Suppose we will give the command for creating a directory for a movielens dataset

Command- `hadoop fs -mkdir ml-100k`

```
[maria_dev@sandbox-hdp ~]$ hadoop fs -mkdir /ml-100k
[maria_dev@sandbox-hdp ~]$ hadoop fs -ls /
```

*hadoop fs -ls

The **hadoop fs -ls** command is used to **list files and directories in Hadoop Distributed File System (HDFS)** or in any other file system supported by Hadoop (like local FS, S3, etc., depending on configuration)

Purpose

- To **view the contents** of a directory in HDFS.
- To **see metadata** of files/directories such as:
 - **Permissions** (read, write, execute)
 - **Replication factor** (for files in HDFS)
 - **Owner and Group**
 - **File size** (in bytes)
 - **Modification date & time**
 - **File/Directory name (path)**

```
[maria_dev@sandbox-hdp ~]$ hadoop fs -mkdir /ml-100k
[maria_dev@sandbox-hdp ~]$ hadoop fs -ls /
Found 12 items
drwxrwxrwx - yarn      hadoop      0 2025-08-31 09:36 /app-logs
drwxr-xr-x - hdfs      hdfs       0 2018-06-18 16:13 /apps
drwxr-xr-x - yarn      hadoop      0 2018-06-18 14:52 /ats
drwxr-xr-x - hdfs      hdfs       0 2018-06-18 14:52 /hdp
drwx----- - livy      hdfs       0 2018-06-18 15:11 /livy2-recovery
drwxr-xr-x - mapred    hdfs       0 2018-06-18 14:52 /mapred
drwxr-xr-x - maria_dev hdfs       0 2025-09-01 16:40 /ml-100k
drwxrwxrwx - mapred    hadoop      0 2018-06-18 14:52 /mr-history
drwxr-xr-x - hdfs      hdfs       0 2018-06-18 15:59 /ranger
drwxrwxrwx - spark     hadoop      0 2025-09-01 16:41 /spark2-history
drwxrwxrwx - hdfs      hdfs       0 2018-06-18 16:06 /tmp
drwxr-xr-x - hdfs      hdfs       0 2018-06-18 16:08 /user
```

*ls

In **Hadoop**, the **ls** command is used to **list files and directories** in the Hadoop Distributed File System (**HDFS**)—similar to the **ls** command in Linux, but it operates on HDFS paths instead of local file system paths.

Purpose:

- To display the list of files/directories in a given HDFS directory.
- To view metadata like **permissions, owner, group, file size, replication factor, modification date, and path**.

*pwd

```
[maria_dev@sandbox-hdp ~]$ pwd
/home/maria_dev
[maria_dev@sandbox-hdp ~]$
```

Purpose of pwd in Hadoop

- pwd stands for **Print Working Directory**.
- It shows the **current working directory in HDFS** where you are operating.
- Useful to confirm your present location before running file operations like ls, put, or get.

*ls

Command to display the directory

*wget <http://media.sundog-soft.com/hadoop/ml-100k/u.data>

The above command is used to copy the data from web server to the Hadoop file system

```
[maria_dev@sandbox-hdp ~]$ wget http://media.sundog-soft.com/hadoop/ml-100k/u.data
--2025-09-01 16:42:05-- http://media.sundog-soft.com/hadoop/ml-100k/u.data
Resolving media.sundog-soft.com (media.sundog-soft.com)... 52.217.113.17, 52.217.118.249, 16.
15.176.17, ...
Connecting to media.sundog-soft.com (media.sundog-soft.com)|52.217.113.17|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2079229 (2.0M) [application/octet-stream]
Saving to: 'u.data.2'

100%[=====>] 2,079,229    419KB/s   in 5.2s

2025-09-01 16:42:22 (394 KB/s) - 'u.data.2' saved [2079229/2079229]

[maria_dev@sandbox-hdp ~]$ ls
u.data u.data.1 u.data.2 u.item
```

*ls

Give the command ls to see whether the data is imported in hdfs

Once it is imported you will see the name as u.data

```
[maria_dev@sandbox-hdp ~]$ ls
u.data u.data.1 u.item
```

*ls -la

Purpose of ls -la (Linux vs Hadoop)

- In **Linux**, ls -la lists **all files including hidden ones** (those starting with .), with detailed information (long format).

*hadoop fs -copyFromLocal u.data ml-100k/u.data

The file will get copied from local file system to the Hadoop named as u.data

```

[maria_dev@sandbox-hdp ~]$ hadoop fs -copyFromLocal u.data /ml-100k/u.data
-copyFromLocal: Unknown command
Usage: hadoop fs [generic options]
    [-appendToFile <localsrc> ... <dst>]
    [-cat [-ignoreCrc] <src> ...]
    [-checksum <src> ...]
    [-chgrp [-R] GROUP PATH...]
    [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
    [-chown [-R] [OWNER][:[GROUP]] PATH...]
    [-copyFromLocal [-f] [-p] [-l] <localsrc> ... <dst>]
    [-copyToLocal [-p] [-ignoreCrc] [-crc] <src> ... <localdst>]
    [-count [-q] [-h] [-v] [-t <storage type>]] [-u] <path> ...]
    [-cp [-f] [-p] [-l] [-p] [-t] <src> ... <dst>]
    [-createSnapshot <snapshotDir> [<snapshotName>]]
    [-deleteSnapshot <snapshotDir> [<snapshotName>]]
    [-df [-h] [<path> ...]]
    [-du [-s] [-h] <path> ...]
    [-expunge]
    [-find <path> ... <expression> ...]
    [-get [-p] [-ignoreCrc] [-crc] <src> ... <localdst>]
    [-getfacl [-R] <path>]
    [-getfattr [-R] [-n name | -d] [-e en] <path>]
    [-getmerge [-nl] <src> <localdst>]
    [-help [cmd ...]]
    [-ls [-C] [-d] [-h] [-q] [-R] [-t] [-S] [-r] [-u] [<path> ...]]
    [-mkdir [-p] <path> ...]
    [-moveFromLocal <localsrc> ... <dst>]
    [-moveToLocal <src> <localdst>]
    [-mv <src> ... <dst>]
    [-put [-f] [-p] [-l] <localsrc> ... <dst>]
    [-renameSnapshot <snapshotDir> <oldName> <newName>]
    [-rm [-f] [-r] [-R] [-skipTrash] [-safely] <src> ...]
    [-rmrdir [-ignore-fail-on-non-empty] <dir> ...]
    [-setfacl [-R] [[-b|-k] [-m|-x <acl_spec>] <path>]] [--set <acl_spec> <path>]]
    [-setfattr [-n name [-v value] | -x name] <path>]
    [-setrep [-R] [-w] <rep> <path> ...]
    [-stat [format] <path> ...]
    [-tail [-f] <file>]
    [-test [-dfs] <path>]
    [-text [-ignoreCrc] <src> ...]
    [-touchz <path> ...]
    [-truncate [-w] <length> <path> ...]
    [-usage [cmd ...]]

Generic options supported are
-conf <configuration file>      specify an application configuration file
-D <property=value>             use value for given property
-fs <local|namenode:port>       specify a namenode
-jt <local|resourcemanager:port> specify a ResourceManager
-files <comma separated list of files> specify comma separated files to be copied to the m
ap reduce cluster
-libjars <comma separated list of jars> specify comma separated jar files to include in th
e classpath.
-archives <comma separated list of archives> specify comma separated archives to be unarch

```

*hadoop fs -ls

The **hadoop fs -ls** command is used to **list files and directories in Hadoop Distributed File System (HDFS)** or in any other file system supported by Hadoop (like local FS, S3, etc., depending on configuration)

*hadoop fs -rm ml-100k/u.data

🔗 Purpose

- To **remove (delete) files** from HDFS.
- Works similar to Linux rm, but operates on HDFS.

*hadoop fs -rmdir ml-100k

The **hadoop fs -rmdir** command is used to **remove (delete) empty directories** from HDFS.

🔗 Purpose

- To delete **empty directories** in Hadoop Distributed File System (HDFS).
- It is similar to the Linux rmdir command.
- ⚠️ Unlike -rm -r, it **cannot delete directories that contain files or subdirectories**.

```

rm: /ml-100k/u.data:1: NO such file or directory
[maria_dev@sandbox-hdp ~]$ hadoop fs -rmdir /ml-100k
[maria_dev@sandbox-hdp ~]$

```

*hadoop fs -ls

```
[maria_dev@sandbox-hdp ~]$ hadoop fs -ls /
Found 11 items
drwxrwxrwx - yarn    hadoop    0 2025-08-31 09:36 /app-logs
drwxr-xr-x - hdfs    hdfs    0 2018-06-18 16:13 /apps
drwxr-xr-x - yarn    hadoop    0 2018-06-18 14:52 /ats
drwxr-xr-x - hdfs    hdfs    0 2018-06-18 14:52 /hdp
drwx----- - livy    hdfs    0 2018-06-18 15:11 /livy2-recovery
drwxr-xr-x - mapred  hdfs    0 2018-06-18 14:52 /mapred
drwxrwxrwx - mapred  hadoop    0 2018-06-18 14:52 /mr-history
drwxr-xr-x - hdfs    hdfs    0 2018-06-18 15:59 /ranger
drwxrwxrwx - spark   hadoop    0 2025-09-01 16:53 /spark2-history
drwxrwxrwx - hdfs    hdfs    0 2018-06-18 16:06 /tmp
drwxr-xr-x - hdfs    hdfs    0 2018-06-18 16:08 /user
[maria_dev@sandbox-hdp ~]$
```

The commands checks where the directory is removed from the hadoop

*Hadoop fs

By using this command we may see the activities that we have performed in our Hadoop file system

```
Usage: hadoop fs [generic options]
[-appendToFile <localsrc> ... <dst>]
[-cat [-ignorecrc] <src> ...]
[-checksum <src> ...]
[-chgrp [-R] GROUP PATH...]
[-chmod [-R] <MODE[,MODE]... [OCTALMODE] PATH...]
[-chown [-R] [OWNER][:(GROUP)] PATH...]
[-copyFromLocal [-f] [-p] [-l] <localsrc> ... <dst>]
[-copyToLocal [-p] [-ignorecrc] [-crc] <src> ... <localdst>]
[-count [-q] [-h] [-v] [-t [<storage type>]] [-u] <path> ...]
[-cp [-f] [-p] [-ptopcp] <src> ... <dst>]
[-createSnapshot <snapshotDir> <snapshotName>]
[-deleteSnapshot <snapshotDir> <snapshotName>]
[-df [-h] <path> ...]
[-du [-s] [-h] <path> ...]
[-expunge]
[-find <path> ... <expression> ...]
[-get [-p] [-ignorecrc] [-crc] <src> ... <localdst>]
[-getfacl [-R] <path>]
[-getfattr [-R] [-n name [-d]] [-e en] <path>]
[-getmerge [-n] <src> <localdst>]
[-help [cmd ...]]
[-ls [-c] [-d] [-h] [-q] [-R] [-t] [-S] [-z] [-u] <path> ...]
[-mkdir [-p] <path> ...]
[-moveFromLocal <localsrc> ... <dst>]
[-moveToLocal <src> <localdst>]
[-mv <src> ... <dst>]
[-put [-f] [-p] [-l] <localsrc> ... <dst>]
[-renameSnapshot <snapshotDir> <oldName> <newName>]
[-rm [-f] [-r] [-R] [-skiptrash] [-safely] <src> ...]
[-rmr [-ignore-fail-on-non-empty] <dir> ...]
[-setfacl [-R] [(b|k) (-m|-x <acl_spec>) <path>][--set <acl_spec> <path>]]
[-setfattr [-n name [-v value] | -x name] <path>]
[-setrep [-R] [-w] <rep> <path> ...]
[-stat [format] <path> ...]
[-tail [-f] <file>]
[-test [-d] <path>]
[-test [-ignorecrc] <src> ...]
[-touchr <path> ...]
[-truncate [-w] <length> <path> ...]
[-usage [cmd ...]]

Generic options supported are
-conf <configuration file>      specify an application configuration file
-D <property=value>             use value for given property
-fs <localname>:port            specify a namenode
-it <localresource>:port        specify a ResourceManager
-files <comma separated list of files> specify comma separated files to be copied to the map reduce cluster
-libjars <comma separated list of jars> specify comma separated jar files to include in the classpath.
-archives <comma separated list of archives> specify comma separated archives to be unarchived on the compute machines.

The general command line syntax is
bin/hadoop command [genericOptions] [commandOptions]

[maria_dev@sandbox-hdp ~]$
```