```
hduser@student-ThinkCentre-M700:~$ start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [student-ThinkCentre-M700]
2024-03-06 15:10:46,109 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable

hduser@student-ThinkCentre-M700:~$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers

hduser@student-ThinkCentre-M700:~$ jps
5953 Jps
5155 SecondaryNameNode
5349 ResourceManager
5493 NodeManager
4730 NameNode
4895 DataNode

hduser@student-ThinkCentre-M700:~$ cd Documents/
hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -mkdir /my_dir
2024-03-06 15:18:30,871 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -put test.txt /my_dir/
2024-03-06 15:21:00,931 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -cat /my_dir/test.txt
2024-03-06 15:22:53,499 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
This is a hadoop file

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -help
2024-03-06 15:26:26,471 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Usage: hadoop fs [generic options]
[-appendToFile <localsrc> ... <dst>]
[-cat [-ignoreCrc] <src> ...]
[-checksum [-v] <src> ...]
[-chgrp [-R] GROUP PATH...]
[-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
[-chown [-R] [OWNER][:[GROUP]] PATH...]
[-concat <target path> <src path> <src path> ...]
[-copyFromLocal [-f] [-p] [-l] [-d] [-t <thread count>] [-q <thread pool queue size>] <localsrc>
... <dst>]
[-copyToLocal [-f] [-p] [-crc] [-ignoreCrc] [-t <thread count>] [-q <thread pool queue size>]
```

<src> ... <localdst>]
[-count [-q] [-h] [-v] [-t [<storage type>]] [-u] [-x] [-e] [-s] <path> ...]
[-cp [-f] [-p | -p[topax]] [-d] [-t <thread count>] [-q <thread pool queue size>] <src> ... <dst>]
[-createSnapshot <snapshotDir> [<snapshotName>]]
[-deleteSnapshot <snapshotDir> <snapshotName>]
[-df [-h] [<path> ...]]
[-du [-s] [-h] [-v] [-x] <path> ...]
[-expunge [-immediate] [-fs <path>]]
[-find <path> ... <expression> ...]
[-get [-f] [-p] [-crc] [-ignoreCrc] [-t <thread count>] [-q <thread pool queue size>] <src> ...
<localdst>]
[-getfacl [-R] <path>]
[-getfattr [-R] {-n name | -d} [-e en] <path>]
[-getmerge [-nl] [-skip-empty-file] <src> <localdst>]
[-head <file>]
[-help [cmd ...]]
[-ls [-C] [-d] [-h] [-q] [-R] [-t] [-S] [-r] [-u] [-e] [<path> ...]]
[-mkdir [-p] <path> ...]
[-moveFromLocal [-f] [-p] [-l] [-d] <localsrc> ... <dst>]
[-moveToLocal <src> <localdst>]
[-mv <src> ... <dst>]
[-put [-f] [-p] [-l] [-d] [-t <thread count>] [-q <thread pool queue size>] <localsrc> ... <dst>]
[-renameSnapshot <snapshotDir> <oldName> <newName>]
[-rm [-f] [-r|-R] [-skipTrash] [-safely] <src> ...]
[-rmdir [--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] [-s <sleep interval>] <file>]
[-test -[defswrz] <path>]
[-text [-ignoreCrc] <src> ...]
[-touch [-a] [-m] [-t TIMESTAMP (yyyyMMdd:HHmmss)] [-c] <path> ...]
[-touchz <path> ...]
[-truncate [-w] <length> <path> ...]
[-usage [cmd ...]]

-appendToFile <localsrc> ... <dst> :
  Appends the contents of all the given local files to the given dst file. The dst
  file will be created if it does not exist. If <localSrc> is -, then the input is
  read from stdin.

-cat [-ignoreCrc] <src> ... :
  Fetch all files that match the file pattern <src> and display their content on
  stdout.

-checksum [-v] <src> ... :
  Dump checksum information for files that match the file pattern <src> to stdout.
  Note that this requires a round-trip to a datanode storing each block of the
  file, and thus is not efficient to run on a large number of files. The checksum
  of a file depends on its content, block size and the checksum algorithm and

parameters used for creating the file.

-chgrp [-R] GROUP PATH... :
This is equivalent to -chown ... :GROUP ...

-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH... :
Changes permissions of a file. This works similar to the shell's chmod command with a few exceptions.

-R modifies the files recursively. This is the only option currently supported.
<MODE> Mode is the same as mode used for the shell's command. The only letters recognized are 'rwxXt', e.g. +t,a+r,g-w,+rwx,o=r.
<OCTALMODE> Mode specifed in 3 or 4 digits. If 4 digits, the first may be 1 or 0 to turn the sticky bit on or off, respectively. Unlike the shell command, it is not possible to specify only part of the mode, e.g. 754 is same as u=rwx,g=rx,o=r.

If none of 'augo' is specified, 'a' is assumed and unlike the shell command, no umask is applied.

-chown [-R] [OWNER][:[GROUP]] PATH... :
Changes owner and group of a file. This is similar to the shell's chown command with a few exceptions.

-R modifies the files recursively. This is the only option currently supported.

If only the owner or group is specified, then only the owner or group is modified. The owner and group names may only consist of digits, alphabet, and any of [-_./@a-zA-Z0-9]. The names are case sensitive.

WARNING: Avoid using '.' to separate user name and group though Linux allows it. If user names have dots in them and you are using local file system, you might see surprising results since the shell command 'chown' is used for local files.

-concat <target path> <src path> <src path> ... :
Concatenate existing source files into the target file. Target file and source files should be in the same directory.

-copyFromLocal [-f] [-p] [-l] [-d] [-t <thread count>] [-q <thread pool queue size>] <localsrc> ... <dst> :
Identical to the -put command.

-copyToLocal [-f] [-p] [-crc] [-ignoreCrc] [-t <thread count>] [-q <thread pool queue size>] <src> ... <localdst> :
Identical to the -get command.

-count [-q] [-h] [-v] [-t [<storage type>]] [-u] [-x] [-e] [-s] <path> ... :
Count the number of directories, files and bytes under the paths that match the specified file pattern. The output columns are:

DIR_COUNT FILE_COUNT CONTENT_SIZE PATHNAME
or, with the -q option:
QUOTA REM_QUOTA SPACE_QUOTA REM_SPACE_QUOTA
DIR_COUNT FILE_COUNT CONTENT_SIZE PATHNAME
The -h option shows file sizes in human readable format.
The -v option displays a header line.
The -x option excludes snapshots from being calculated.
The -t option displays quota by storage types.
It should be used with -q or -u option, otherwise it will be ignored.
If a comma-separated list of storage types is given after the -t option,
it displays the quota and usage for the specified types.
Otherwise, it displays the quota and usage for all the storage
types that support quota. The list of possible storage types(case insensitive):
ram_disk, ssd, disk and archive.
It can also pass the value '', 'all' or 'ALL' to specify all the storage types.
The -u option shows the quota and
the usage against the quota without the detailed content summary.The -e option
shows the erasure coding policy.The -s option shows snapshot counts.

-cp [-f] [-p | -p[topax]] [-d] [-t <thread count>] [-q <thread pool queue size>] <src> ... <dst> :
Copy files that match the file pattern <src> to a destination. When copying
multiple files, the destination must be a directory.
Flags :

-p[topax] Preserve file attributes [topx] (timestamps,
ownership, permission, ACL, XAttr). If -p is
specified with no arg, then preserves timestamps,
ownership, permission. If -pa is specified, then
preserves permission also because ACL is a
super-set of permission. Determination of whether
raw namespace extended attributes are preserved is
independent of the -p flag.
-f Overwrite the destination if it already exists.
-d Skip creation of temporary file(<dst>._COPYING_).
-t <thread count> Number of threads to be used, default is 1.
-q <thread pool queue size> Thread pool queue size to be used, default is
1024.

-createSnapshot <snapshotDir> [<snapshotName>] :
Create a snapshot on a directory

-deleteSnapshot <snapshotDir> <snapshotName> :
Delete a snapshot from a directory

-df [-h] [<path> ...] :
Shows the capacity, free and used space of the filesystem. If the filesystem has
multiple partitions, and no path to a particular partition is specified, then
the status of the root partitions will be shown.

-h Formats the sizes of files in a human-readable fashion rather than a number
of bytes.

-du [-s] [-h] [-v] [-x] <path> ... :
  Show the amount of space, in bytes, used by the files that match the specified
  file pattern. The following flags are optional:

  -s Rather than showing the size of each individual file that matches the
      pattern, shows the total (summary) size.
  -h Formats the sizes of files in a human-readable fashion rather than a number
      of bytes.
  -v option displays a header line.
  -x Excludes snapshots from being counted.

  Note that, even without the -s option, this only shows size summaries one level
  deep into a directory.

  The output is in the form
  size disk space consumed name(full path)

-expunge [-immediate] [-fs <path>] :
  Delete files from the trash that are older than the retention threshold

-find <path> ... <expression> ... :
  Finds all files that match the specified expression and
  applies selected actions to them. If no <path> is specified
  then defaults to the current working directory. If no
  expression is specified then defaults to -print.

  The following primary expressions are recognised:
    -name pattern
    -iname pattern
      Evaluates as true if the basename of the file matches the
      pattern using standard file system globbing.
      If -iname is used then the match is case insensitive.

    -print
    -print0
      Always evaluates to true. Causes the current pathname to be
      written to standard output followed by a newline. If the -print0
      expression is used then an ASCII NULL character is appended rather
      than a newline.

  The following operators are recognised:
    expression -a expression
    expression -and expression
    expression expression
      Logical AND operator for joining two expressions. Returns
      true if both child expressions return true. Implied by the
      juxtaposition of two expressions and so does not need to be
      explicitly specified. The second expression will not be
      applied if the first fails.

-get [-f] [-p] [-crc] [-ignoreCrc] [-t <thread count>] [-q <thread pool queue size>] <src> ... <localdst> :
  Copy files that match the file pattern <src> to the local name. <src> is kept.
  When copying multiple files, the destination must be a directory.
  Flags:

  -p   Preserves timestamps, ownership and the mode.
  -f   Overwrites the destination if it already exists.
  -crc   write CRC checksums for the files downloaded.
  -ignoreCrc   Skip CRC checks on the file(s) downloaded.
  -t <thread count>   Number of threads to be used, default is 1.
  -q <thread pool queue size> Thread pool queue size to be used, default is
                        1024.

-getfacl [-R] <path> :
  Displays the Access Control Lists (ACLs) of files and directories. If a
  directory has a default ACL, then getfacl also displays the default ACL.

  -R   List the ACLs of all files and directories recursively.
  <path> File or directory to list.

-getfattr [-R] {-n name | -d} [-e en] <path> :
  Displays the extended attribute names and values (if any) for a file or
  directory.

  -R   Recursively list the attributes for all files and directories.
  -n name   Dump the named extended attribute value.
  -d   Dump all extended attribute values associated with pathname.
  -e <encoding> Encode values after retrieving them.Valid encodings are "text",
                "hex", and "base64". Values encoded as text strings are enclosed
                in double quotes ("), and values encoded as hexadecimal and
                base64 are prefixed with 0x and 0s, respectively.
  <path>   The file or directory.

-getmerge [-nl] [-skip-empty-file] <src> <localdst> :
  Get all the files in the directories that match the source file pattern and
  merge and sort them to only one file on local fs. <src> is kept.

  -nl   Add a newline character at the end of each file.
  -skip-empty-file Do not add new line character for empty file.

-head <file> :
  Show the first 1KB of the file.

-help [cmd ...] :
  Displays help for given command or all commands if none is specified.

-ls [-C] [-d] [-h] [-q] [-R] [-t] [-S] [-r] [-u] [-e] [<path> ...] :
  List the contents that match the specified file pattern. If path is not
  specified, the contents of /user/<currentUser> will be listed. For a directory a
  list of its direct children is returned (unless -d option is specified).

```
  Directory entries are of the form:
  permissions - userId groupId sizeOfDirectory(in bytes)
  modificationDate(yyyy-MM-dd HH:mm) directoryName

  and file entries are of the form:
  permissions numberOfReplicas userId groupId sizeOfFile(in bytes)
  modificationDate(yyyy-MM-dd HH:mm) fileName

    -C  Display the paths of files and directories only.
    -d  Directories are listed as plain files.
    -h  Formats the sizes of files in a human-readable fashion
        rather than a number of bytes.
    -q  Print ? instead of non-printable characters.
    -R  Recursively list the contents of directories.
    -t  Sort files by modification time (most recent first).
    -S  Sort files by size.
    -r  Reverse the order of the sort.
    -u  Use time of last access instead of modification for
        display and sorting.
    -e  Display the erasure coding policy of files and directories.

-mkdir [-p] <path> ... :
  Create a directory in specified location.

  -p  Do not fail if the directory already exists

-moveFromLocal [-f] [-p] [-l] [-d] <localsrc> ... <dst> :
  Same as -put, except that the source is deleted after it's copied
  and -t option has not yet implemented.

-moveToLocal <src> <localdst> :
  Not implemented yet

-mv <src> ... <dst> :
  Move files that match the specified file pattern <src> to a destination <dst>.
  When moving multiple files, the destination must be a directory.

-put [-f] [-p] [-l] [-d] [-t <thread count>] [-q <thread pool queue size>] <localsrc> ... <dst> :
  Copy files from the local file system into fs. Copying fails if the file already
  exists, unless the -f flag is given.
  Flags:

  -p                     Preserves timestamps, ownership and the mode.
  -f                     Overwrites the destination if it already exists.
  -t <thread count>          Number of threads to be used, default is 1.
  -q <thread pool queue size>  Thread pool queue size to be used, default is
                         1024.
  -l                     Allow DataNode to lazily persist the file to disk.
                         Forces replication factor of 1. This flag will
                         result in reduced durability. Use with care.
```

```
  -d                    Skip creation of temporary file(<dst>._COPYING_).

-renameSnapshot <snapshotDir> <oldName> <newName> :
  Rename a snapshot from oldName to newName

-rm [-f] [-r|-R] [-skipTrash] [-safely] <src> ... :
  Delete all files that match the specified file pattern. Equivalent to the Unix
  command "rm <src>"

  -f          If the file does not exist, do not display a diagnostic message or
              modify the exit status to reflect an error.
  -[rR]       Recursively deletes directories.
  -skipTrash  option bypasses trash, if enabled, and immediately deletes <src>.
  -safely     option requires safety confirmation, if enabled, requires
              confirmation before deleting large directory with more than
              <hadoop.shell.delete.limit.num.files> files. Delay is expected when
              walking over large directory recursively to count the number of
              files to be deleted before the confirmation.

-rmdir [--ignore-fail-on-non-empty] <dir> ... :
  Removes the directory entry specified by each directory argument, provided it is
  empty.

-setfacl [-R] [{-b|-k} {-m|-x <acl_spec>} <path>]|[--set <acl_spec> <path>] :
  Sets Access Control Lists (ACLs) of files and directories.
  Options:

  -b          Remove all but the base ACL entries. The entries for user, group
              and others are retained for compatibility with permission bits.
  -k          Remove the default ACL.
  -R          Apply operations to all files and directories recursively.
  -m          Modify ACL. New entries are added to the ACL, and existing entries
              are retained.
  -x          Remove specified ACL entries. Other ACL entries are retained.
  --set       Fully replace the ACL, discarding all existing entries. The
              <acl_spec> must include entries for user, group, and others for
              compatibility with permission bits. If the ACL spec contains only
              access entries, then the existing default entries are retained. If
              the ACL spec contains only default entries, then the existing
              access entries are retained. If the ACL spec contains both access
              and default entries, then both are replaced.
  <acl_spec>  Comma separated list of ACL entries.
  <path>      File or directory to modify.

-setfattr {-n name [-v value] | -x name} <path> :
  Sets an extended attribute name and value for a file or directory.

  -n name   The extended attribute name.
  -v value  The extended attribute value. There are three different encoding
            methods for the value. If the argument is enclosed in double quotes,
            then the value is the string inside the quotes. If the argument is
```

prefixed with 0x or 0X, then it is taken as a hexadecimal number. If the argument begins with 0s or 0S, then it is taken as a base64 encoding.
-x name Remove the extended attribute.
<path> The file or directory.

-setrep [-R] [-w] <rep> <path> ... :
Set the replication level 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 here.

-w It requests that the command waits for the replication to complete. This can potentially take a very long time.
-R It is accepted for backwards compatibility. It has no effect.

-stat [format] <path> ... :
Print statistics about the file/directory at <path>
in the specified format. Format accepts permissions in
octal (%a) and symbolic (%A), filesize in
bytes (%b), type (%F), group name of owner (%g),
name (%n), block size (%o), replication (%r), user name
of owner (%u), access date (%x, %X).
modification date (%y, %Y).
%x and %y show UTC date as "yyyy-MM-dd HH:mm:ss" and
%X and %Y show milliseconds since January 1, 1970 UTC.
If the format is not specified, %y is used by default.

-tail [-f] [-s <sleep interval>] <file> :
Show the last 1KB of the file.

-f Shows appended data as the file grows.
-s With -f , defines the sleep interval between iterations in milliseconds.

-test -[defswrz] <path> :
Answer various questions about <path>, with result via exit status.
-d return 0 if <path> is a directory.
-e return 0 if <path> exists.
-f return 0 if <path> is a file.
-s return 0 if file <path> is greater than zero bytes in size.
-w return 0 if file <path> exists and write permission is granted.
-r return 0 if file <path> exists and read permission is granted.
-z return 0 if file <path> is zero bytes in size, else return 1.

-text [-ignoreCrc] <src> ... :
Takes a source file and outputs the file in text format.
The allowed formats are zip and TextRecordInputStream and Avro.

-touch [-a] [-m] [-t TIMESTAMP (yyyyMMdd:HHmmss)] [-c] <path> ... :
Updates the access and modification times of the file specified by the <path> to the current time. If the file does not exist, then a zero length file is created at <path> with current time as the timestamp of that <path>.

```
  -a Change only the access time
  -m Change only the modification time
  -t TIMESTAMP Use specified timestamp instead of current time
   TIMESTAMP format yyyyMMdd:HHmmss
  -c Do not create any files

-touchz <path> ... :
  Creates a file of zero length at <path> with current time as the timestamp of
  that <path>. An error is returned if the file exists with non-zero length

-truncate [-w] <length> <path> ... :
  Truncate all files that match the specified file pattern to the specified
  length.

  -w  Requests that the command wait for block recovery to complete, if
      necessary.

-usage [cmd ...] :
  Displays the usage for given command or all commands if none is specified.

Generic options supported are:
-conf <configuration file>        specify an application configuration file
-D <property=value>               define a value for a given property
-fs <file:///|hdfs://namenode:port> specify default filesystem URL to use, overrides
'fs.defaultFS' property from configurations.
-jt <local|resourcemanager:port>  specify a ResourceManager
-files <file1,...>                specify a comma-separated list of files to be copied to the map
reduce cluster
-libjars <jar1,...>               specify a comma-separated list of jar files to be included in the
classpath
-archives <archive1,...>          specify a comma-separated list of archives to be unarchived
on the compute machines

The general command line syntax is:
command [genericOptions] [commandOptions]

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -ls /
2024-03-06 15:28:04,978 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Found 4 items
drwxr-xr-x   - hduser supergroup          0 2024-03-05 14:05 /my_data
drwxr-xr-x   - hduser supergroup          0 2024-03-06 15:21 /my_dir
drwxrwxrwx   - hduser supergroup          0 2023-04-24 13:45 /tmp
drwxr-xr-x   - hduser supergroup          0 2023-04-24 13:43 /user


hduser@student-ThinkCentre-M700:~$ cd Downloads/
hduser@student-ThinkCentre-M700:~/Downloads$ hdfs dfs -appendToFile file.txt
/my_dir/test.txt
2024-03-06 15:34:52,224 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
```

```
hduser@student-ThinkCentre-M700:~/Downloads$ hdfs dfs -cat /my_dir/test.txt
2024-03-06 15:35:30,718 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
This is a hadoop file
this is a second file


hduser@student-ThinkCentre-M700:~/Downloads$ cd ..
hduser@student-ThinkCentre-M700:~$ cd Documents/
hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -count /my_dir/test.txt
2024-03-06 15:39:37,406 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
           0           1                 44 /my_dir/test.txt


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -copyFromLocal test.txt
/my_dir/file1.txt
2024-03-06 15:50:54,555 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -touchz /my_dir/file2.txt
2024-03-06 15:56:27,407 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -copyToLocal /my_dir/test.txt
test2.txt
2024-03-06 16:04:28,078 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -get /my_dir/test.txt gettest.txt
2024-03-06 16:07:44,434 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -moveFromLocal move.txt
/my_dir
2024-03-06 16:12:02,387 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -mkdir /project
2024-03-06 16:14:10,172 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable


hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -put code.txt /project/
2024-03-06 16:15:59,172 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
```

```
hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -rm /project/code.txt
2024-03-06 16:17:23,161 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Deleted /project/code.txt

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -rmdir /project
2024-03-06 16:18:21,159 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -stat /my_dir/test.txt
2024-03-06 16:20:30,109 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
2024-03-06 10:04:53

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -stat %n /my_dir/test.txt
2024-03-06 16:22:10,379 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
test.txt

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -stat %F /my_dir/test.txt
2024-03-06 16:22:45,134 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
regular file

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -find /my_dir/test.txt
2024-03-06 16:25:09,123 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
/my_dir/test.txt

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -tail /my_dir/test.txt
2024-03-06 16:27:15,928 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
This is a hadoop file
this is a second file

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -cp /my_dir/gettest.txt
/my_dir/test3.txt
2024-03-06 16:31:10,594 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable

hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -stat %g /my_dir/test.txt
2024-03-06 16:33:53,007 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
supergroup
```

```
hduser@student-ThinkCentre-M700:~/Documents$ hdfs dfs -stat %u /my_dir/test.txt
2024-03-06 16:35:41,466 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
hduser


hduser@student-ThinkCentre-M700:~/Documents$ cd ..
hduser@student-ThinkCentre-M700:~$ stop-df.sh
stop-df.sh: command not found
hduser@student-ThinkCentre-M700:~$ stop-dfs.sh
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [student-ThinkCentre-M700]
2024-03-06 16:39:17,329 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
hduser@student-ThinkCentre-M700:~$ stop-yarn.sh
Stopping nodemanagers
Stopping resourcemanager
hduser@student-ThinkCentre-M700:~$ jps
10791 Jps
hduser@student-ThinkCentre-M700:~$
```