Linux Commands

— continuation

$ who ami

> display the current logged-in user

File Operations

$ mv (move/rename)

— moves files/directories or rename them

moving files

```
-rw-r--r-- 1 nandyalapsai3443 nandyalapsai3443    0 Jul  1 17:08 test.txt
drwxr-xr-x 2 nandyalapsai3443 nandyalapsai3443 4.0K Jul  1 17:08 files
nandyalapsai3443@cluster-b233-m:~$ mv test.txt files/
```

rename files

```
-rw-r--r-- 1 nandyalapsai3443 nandyalapsai3443 0 Jul  1 17:08 test.txt
nandyalapsai3443@cluster-b233-m:~/files$ mv test.txt renamed_test.txt
nandyalapsai3443@cluster-b233-m:~/files$ ll -lrth
-bash: ll: command not found
nandyalapsai3443@cluster-b233-m:~/files$ ls -lrth
total 0
-rw-r--r-- 1 nandyalapsai3443 nandyalapsai3443 0 Jul  1 17:08 renamed_test.txt
```

move multiple files

```
nandyalapsai3443@cluster-b233-m:~$ ls -lrth
total 4.0K
drwxr-xr-x 2 nandyalapsai3443 nandyalapsai3443 4.0K Jul  1 17:10 files
-rw-r--r-- 1 nandyalapsai3443 nandyalapsai3443    0 Jul  1 17:11 file2.txt
-rw-r--r-- 1 nandyalapsai3443 nandyalapsai3443    0 Jul  1 17:11 file1.txt
nandyalapsai3443@cluster-b233-m:~$ mv file1.txt file2.txt files/
```

$ cp (copy)

— Copy files or directories

Common Options

  -r   — copy directories recursively

  -p   — Preserve metadata (timestamps, permissions)

Examples

```
cp file.txt directory/          # copy file to directory

cp -r directory/ backup/        # copy directory recursively

cp -p file.txt copy.t           # copy with preserved metadata
```

$rm (Remove)

— delete files & directories

Common options:

- r — remove directories recursively

-f — force removal (ignore warnings)

examples

rm file.txt   # remove file

rm -r directory/   # remove directory & contents

rm -rf directory/   # force remove directory

rm -rf *   # removes everything in pwd

Note : use rm -rf with extreme caution as it permanently deletes files

File Creation & Editing

$ touch

— Creates empty files | Update timestamps

touch newfile.txt   # creates empty file

vi (Text Editor) / vim

— edit files using the vi text editor

vi file.txt   # open file for editing

Basic Commands

press ↘ i — enter insert mode (for editing)

ESC — Exit from insert mode

:wq — Save and quit

:q! — Quit without saving

file Content viewing

$ cat

display entire file content

→ cat file.txt   # show all content

$ head

display first/starting few lines of a file [default : 10 lines)

options

—n   Show specific no. of lines

examples

head   file.txt   # shows   first 10 lines

head -n 5   file.txt   # show first 5 lines

$ tail

— display last lines of a file (default : 10 lines)

Options

—n   show specific no. of lines

—f   follow file changes   (useful for logs)

examples

tail file.txt   # show last 10 lines

tail -n 5 file.txt   # show last 5 lines

tail -f application.log   # monitor log file in real-time

Search & Text Processing

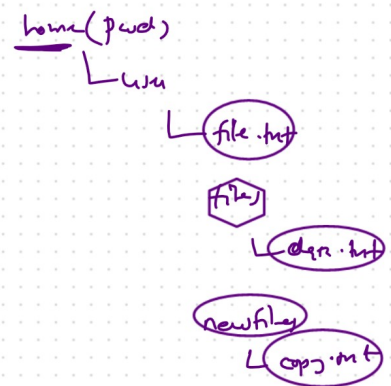## $ grep

— search for text patterns in files

options

—i — ignore case search

—r — search recursively in directories

—c — count occurrences

grep "^error^" file.txt   # search for "error" in file

grep —i "^ErroR" log.txt   # case insensitive search

grep —c "^error" log.txt   # count occurrences

find [path] [options] [expression]

— search for files and directories

Options

— name — search by name

— type ⟨ f ⟹ find files only
        d ⟹ find directories only

examples

find —name "file.txt"   find file.txt

find —name "*.txt"   find all .txt files

find directory —type d —name "log*" # find directories starting with log

Wild cards & pattern matching

* (Asterisk)

matches any no. of characters

home (pwd)
└ user
  └ file.txt
  files
  └ dqn.txt
  newfiles
  └ opg.mt

ls   *.txt   # List out all .txt files

ls log*   # List files starting with "log"

ls *file*   # List files containing "file"

$ wc   (word count)

— count lines, words, & characters

Options
   - l   — count lines only
   - w   — count words only
   - c   — count characters only

```
nandyalapsai3443@cluster-b233-m:~$ wc -l log.txt
219 log.txt
nandyalapsai3443@cluster-b233-m:~$ wc -w log.txt
1095 log.txt
nandyalapsai3443@cluster-b233-m:~$ wc -c log.txt
8871 log.txt
```


$ clear

Clears the terminal screen

Note:

```
nandyalapsai3443@cluster-b233-m:~$ man wc
nandyalapsai3443@cluster-b233-m:~$
nandyalapsai3443@cluster-b233-m:~$ wc --help
```

helps or guides us to see the option for a command

→ "tab" key for auto-completion

→ use up/down arrows to navigate command history

→ ctrl + l   clear screen

→ ctrl + c → cancel running command