

- Examples given below are basic and simple. Intension here is to introduce to you the necessary and helpful Linux bash commands. If you intend on becoming a command line Ninja, please use Google/man pages ☺
- Make Directory (**mkdir**):
  - `mkdir test_dir`
  - `mkdir -p test_dir/test_dir2/test_dir3`
- Remove Directory (**rmdir**)
  - `rmdir test_dir/test_dir2/test_dir3` (Removes dir only if it's empty)
  - `rm -rf test_dir` (Removes everything in the dir recursively. Be careful!!!)
- Navigation shortcuts:
  - `ctrl a` (Brings cursor to beginning)
  - `ctrl e` (Brings cursor to end)
  - `ctrl u` (Deletes entire command line)
  - `ctrl w` (Deletes with white space character delimiter)
  - `ctrl c` (Cancels current command/process)
  - `ctrl r` (Start typing to bring up commands from history)
- **pwd** (Print working directory)\
- **history** (Provides a list of previously executed commands)
- Change Directory (**cd**):
  - `cd test_dir`
  - `cd test_dir/test_dir2/test_dir3` (Use tab to auto complete. Tab twice to list)
  - `cd -` (Change to previous dir)
  - `cd ~` (Change to home dir)
  - `cd ..` (Change to dir one level up)
- List contents (**ls**)
  - `ls`
  - `ls -lh` (Detailed list with size in human readable format)
  - `ls -lah` (Detailed list (including hidden files) with size in human readable format)
- Aliases (**alias**)
  - `alias ls="ls -lah"` (ls works as ls -lah henceforth. Put this in .profile file in home directory to have this enabled as soon as you login)
- Manual Pages (**man**)
  - `man ls` (Manual page gives lots of useful information about the command)
- Apropos (**apropos**)
  - `apropos directory` (Search for key words)
- Copy (**cp**)
  - `cp file1 test_dir/file2` (Copy files)
  - `cp -pr dir1 dir2` (Copy directories)
- Rename or move (**mv**)
  - `mv file1 test_dir/file2` (Moves and renames file1 to file2 in test\_dir)
  - `mv dir1 dir2` (Renames dir1 to dir2)
- Print on standard output (**cat**)

- cat file (Print file content)
- cat file1 file2 (Concatenate and print contents of file1 and file2)
- **less and more command**
  - more file1 (Read text one screenful at a time. Only forward)
  - less file1 (Same as above but also allows backward movement)
- **Tail a file (tail)**
  - tail filename (Print last 10 lines of filename)
  - tail -n 20 filename (Print last 20 lines of filename)
  - tail -f filename (Follow filename and print new lines)
- **Print starting lines of a file (head)**
  - head filename (Print 10 starting lines of filename)
  - head -n 20 filename (Print 20 starting lines of filename)
- **Word Count (wc)**
  - wc -l file\_name (Gives line count)
  - cat file\_name |wc -l (Another way to do the same using a pipe)
- **awk**
  - cat file |awk '{print \$2}' (Prints 2<sup>nd</sup> field. White space character delimiter)
  - cat file |awk '{print \$1, \$2}' (Prints 1<sup>st</sup> and 2<sup>nd</sup> field)
  - cat file |awk '{print \$(NF-1), \$NF}' (Prints last but one field and last field)
  - cat file |awk '\$1 == 1 {print \$1, \$2}' (Prints 1<sup>st</sup> and 2<sup>nd</sup> field if 1<sup>st</sup> field equal to 1)
- **sed**
  - cat file |sed 's/1/9/' (Substitute 1<sup>st</sup> occurrence of 1 with 9 in each line of file)
  - cat file |sed 's/1/9/g' (Substitute all occurrences of 1 with 9 in each line of file)
  - sed -i 's/1/9/' (Edit and substitute in place)
- **stdin, stdout, stderr**
  - 0 -> stdin
  - 1 -> stdout
  - 2 -> stderr
  - tail filename > filename2 (Redirect stdout to a file instead of display)
  - tail filename 1>filename2 (Another way to do the above)
  - long\_tail filename 2>filename2 (Redirect stderr to a file)
  - long\_tail filename > filename2 2>&1 (Redirect both stderr and stdout to a file)
  - long\_tail filename >> filename2 2>&1 (Same as above but append to file)
  - tail filename > /dev/null (Redirect output to a black hole!)
- **jobs**
  - ctrl z to suspend current process.
  - End a command/script with & (ampersand) to run it in background.
  - jobs (To list all jobs)
  - bg %<job number> (To run job in background)
  - fg %<job number> (To run job in foreground)
  - kill -19 %<job number> (To suspend a job)
  - kill %<job number> (To kill a job)