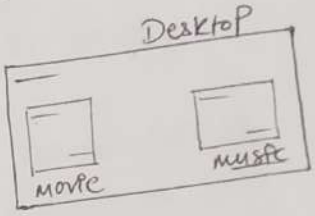
Linux Commands - Intermediate.

terminal emulator
- a program that will let us use the terminal in graphical way.

shell
- command line interface that will take all our commands (executable file) as input and interprete it to tell operation system what to do.

command prompt
- where we write commands.

(A) change directory - use to change the directories
- [cd] - change working directory to Home.
- [cd foldername] - enter inside the directory/sub-directory.
- [cd ..] - go to the previous directory.
- [cd ../subdirectory] - first come out of working directory and then enter another directory.



Example ~ cd.
~ cd Desktop - (Pwd - desktop)
~ cd music - (Pwd - music)
~ cd ../movie - 1st it will come out of music and enter another directory movie.

(B) Print working directory - shows the current working directory.
- [Pwd].

Example ~ cd Desktop.
~ Pwd
- Home/Desktop

(C) Make directory - use to create directories/folders.
- [mkdir foldername]
- [mkdir f1/f3] - this will make f3 folder inside f1 but f1 must exist already.
- [mkdir -p f1/f2/f3] - folder f2 will be created as - if create the directory and if required, all parent directories.
  (-p: Parent.)

(D) touch command - use to create the files.
- [touch filename.extention] -
- [touch f1/file1.txt] - Make the file inside already existing subfolder f1.

(E) list command - Shows all the unhidden (hidden) items in pwd.

- [ls] - shows the unhidden items in pwd.
- [ls -a] - shows all the hidden items in pwd.
- [ls -l] - shows more info about unhidden items in pwd.
- [ls -al] - shows more info about hidden items in pwd.
- [ls -R] - shows all the files and sub-directories in all the subdirectories.
- [ls subdirectory] - shows all the files/folders in mentioned directories.

(f) clear command [clear] - clean the terminal prompt / command prompt.

(g) [where executablefile] - gives the location of mentioned file.

(h) [man command] (or) [command --help] - gives info about command.

(i) echo command - use to display line of text/string passed as an argument and to override the text inside the file.

- [echo "set of strings"] - set of strings will display on command prompt
- [echo filename > "string"] - string will override whatever written in file.
- [echo $PATH] - displays all paths environment variables.

Example. ~ echo "Sukant Tekade"
Sukant Tekade
~ echo file.txt > "Hey"

(j) concatenate - displays the contents of one or more files without having to open the file for editing.

- [cat filename] - display all the content inside file.
- [cat > filename] - it will create a new file in pwd.
  - write content of the file → content inside of file.
  - [ctrl + c] → disable the controls (or) come out of the command.
- [cat file1.txt > file.txt] - copy content of f1 to f.

Example. ~ cat > file1.txt. ~ echo file1.txt > "Hey"

~ cat file.txt file1.txt. - display content inside f1 & f.
Hello Hey
~ cat file.txt > file1.txt.
- Hello Hello.

(k) translate

- [cat filename | tr a-z A-Z] → convert lowercase to uppercase.

Example
~ cat file.txt | tr a-z A-Z.
HELLO

(c) copy command - copy content of one file to another

- [cp file1.txt file2.txt] - f1 ⟶ f2

(m) move command - move file/folder to another folder.

- [mv file1.txt f1] file1.txt ⟶ f1

* mv file.txt [filechange.txt] rename the file/folder.

remove command

* rm file.txt ← remove files permenently.
* rm -R folder ← remove folder permenently.
* rm -f file.txt ← remove file forcefully
* rm -Rf folder ← remove folder forcefully
* rm -d folder ← remove empty folder

cp -R folder1 folder2 ← copy entire directory including its sub directory & files to another directory.

df command - use to check system disk space usage.

-[df] - shows space in kbps
-[df -m] - shows space in mbps
-[df -h] - shows space in human readable form in gigabyte

du command - display disk usage statistics

- [du] - shows in kbps
- [du -h] - shows in human readable format

head command - use to veiw first few lines of the files (by default 10 lines)

-[head <filename.extention>] - syntax.
-[head file.txt] - shows first 10 lines of file.txt)
-[head -n 4 file.txt] - shows first 4 lines of file.txt.

tail command - use to view last few lines of the files (by default 10 lines)

-[tail file.txt] - shows last 10 line of file.txt
-[tail -n 4 file.txt] - shows last 4 lines of file.txt.

#diff command - compares the content of the file line by line and output the line which not matter the same/matches.

 # locate command — It gives location of the files

- [locate "*.txt"] - It will gives all the files of txt extention.

 # find command - use to find stuff (like files, directories - - -)

* find <foldername> - listing folders & files inside the mentioned folder.
* find . -type f(<foldername>) - It will list only files inside the mentioned folder.
* find . -type d(<foldername>) - It will list only folders of current directory or of mention directory.
  (sub directory)
* find . -type f -name "file.txt" -
  ↖ file name is case sensetive.

To make it non sensetive. we can write. find . -type -iname "file.txt".

* find . -type f -mmin 20 - displays all the files modified or created 20 mins ago.
* find . -type f -mtime 10 - displays all the files modified or created 10 days ago.
* find -size +1k - display files greater than 1kb -

 # file permissions

- Their are three types of people who are uring the computer.
  ① user ② people ③ other.

For this people their are three types of permissions - read, write & execute.

- drwx rwx rwx
  - d → Indicates d for only directory.
  - rwx → permission given to usees. r-read w-write x-eexecutes
  - rwx → permission given to people.
  - rwx → permission given to other

* read (r) - 4
* write (w) - 2
* execute (x) - 1
* read (r) + write (w) - 6
* read (r) + execute (x) - 5 - - -

 # changing the permission

- chmod 777 <file name> ← gives all types of permissions to all people.
- chmod 500 <file name> ← gives write & execute permission to user only.