

Unix & shell part 2

/ => root

... / grandparent

=> file is any kind of data

./ => (current, ... / parent)

=> files are arranged in directory

=> files are hierarchically arranged (upside-down tree)

Ex: /usr/ghansham) Ap/gis.txt

Commands: ls, ls -l => to differentiate betw files & folders

ls -C => alphabetical order ls -l => more detail

- => file d => directory

=> pwd => cd

only cd => home directory

Filename & File type

a-z, A-Z, 0-9, ., @, -, +, =, :

Avoid some character = space, ", ', !, #, \$, %, ^, &, *

=> you never use () in file name

=> a filename / folder name having "/" with it

=> Unix don't have file type

(file *) => give type of each file in folder

(only name not used)
use full
name + extension

wildcard

* => all

? => for only one character

using wildcard we can perform action on a group

ls -d ls -d => don't list directory just show it

ls hell.*ng

? => only one char

=> hell.??? => find hell containing extension with 3 char

you want find, bt.*xy

ls bt.??? will search it

=> you can also find ls using regular expression

ls [!s]* => not containing s at all

\$(!s).xy

↑ second character to be s

Form (x) Displaying file content

→ cat ⇒ more (large fast can seen with it)
(space new page) (Enter one line)

→ head top 10 line

→ tail last 10 line

we can use more to display multiple files

(cp)

↓ copy

(mv)

↑ move

with ~~cp~~ mv we can also do rename

⇒ you can copy multiple files in directory

⇒ copy & store filename using different name

(x) rm used to delete

rmdir ⇒ delete empty folder

rm -r folder-y

⇒ delete folder containing files

rm -f ⇒ force

rm -i ⇒ interactive ask before delete

-r ⇒ recursive (to delete folder & files)

hidden file ⇒ (.) ⇒ start name with dot

⇒ It not display on normal ls

⇒ ls used to display hidden files

or ls -a or ls -la

-a ⇒ all file
normal + hidden

⇒ you can rename a file start with . & that will be hidden

⇒ Hidden files are some configuration files & not should be for normal user

• ⇒ current

• ⇒ parent

• ⇒ grandparent

• / .. / ... / ⇒ Relative path

/usr/ghansham/document/hell.txt ⇒ Absolute position

- ⇒ absolute path for a file never changes
- ⇒ Relative path of file change with location

⊛ working with directory

→ mkdir → cd → rmdir → rm -r ^{-r ⇒ recursive}

⊛ finding files

→ find command

⊛ find folder/ -name hell.txt -print ^{unlike name of screen on terminal}

↳ find will search in all entire file system if you not provide folder

⇒ In find if you want to use wildcard in close the in " " ⇒ single quotes

find . -name !*.txt (enclose wildcard in single quotes)
else you get wrong (clp)

⇒ you can also find using type

⇒ Archiving file (zipping file)

tar, cpio, gzip, compress used to provide archive & compression