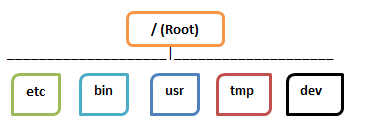
In Windows, files are stored in folders on different data drives like C: D: E:

But in**Linux, files are ordered in a tree structure starting with the root directory**.

**BIN :** Bin contains the executables that is ready to run programs for the purpose of booting.

**SBIN** : Same as that of bin. This is mostly for administrative tools, that should be made available only to the root user.

**ETC :** etc contains configuration files, that is a local file used to control the operation of a program.

**DEV :** Dev is the location of device files.

**VAR :** Var contains variable data files. This includes administrative and logging data.

**TMP :** Tmp is a temporary landing for files. Users also have write access to this directory.

**HOME:** The home directory is for a particular user of the system and consist of individual files. It is also known as the logging directory. It is automatically created as /home for each user in the directory.

**OPT:** Opt is for the installation of add on application software that are not part of the system. For example any external or third party software.

**LIB :** Lib is a library files directory which contains all helpful library files used by the system.

**MNT :** The mnt command instructs the OS that a file system is ready to use.

**MEDIA** : Where removable media devices inserted are mounted. For example when you insert a CD into your system, a directory will automatically be created inside the /media directory. You can access the contents of this CD inside this directory.

**D ->** Directory files solo job is to store file names and the related information. A directory file contains no data but some details of the subdirectories and files that it contains.

**"-" ->** Normal file is a most common file type. It governs all diff. files such as text files, images, binary files etc.

**l ->** A link is a symbolic connection or pointer to a single file that allows you to access it from more than one directory. A symbolic link is created when you link files between directories, allowing access to the file without providing access to the directory.

**C ->** A character special file is a file that provides access to an input/output device.

**B ->** Block special files provide buffered access to hardware devices, and provide some abstraction from their specifics. Unlike character devices, block devices will always allow the programmer to read or write a block of any size including single characters/bytes.

**S ->** A socket is a special file used for inter-process communication, which enables communication between two processes. In addition to sending data, processes can send file descriptors across a Unix domain socket connection using the sendmsg() and recvmsg() system calls.

**Cd: cd** command in linux known as change directory command. It is used to change current working directory.

**cd. :** single dot represents the current working directory. So it will change into the current directory.

**Cd.. :** two dots represents the parent directory. So it will move a level up or to the parent directory.

**Cd - :** To change back to the previous working directory, pass the dash character as an argument to the cd command**.**

**Cd ~ :** To navigate to your home directory, use cd with a tilde character.

**Ls :** used to list computer files.

**Ls -r:** list all the files and directories in reverse order.

**Ls -more:** list the contents of your current working directory.

**Ls -l:** to show long listing information about the file/directory.

**Ls -a:** to show all the hidden files in the directory.

**Ls -A:** to show the hidden file but not the current and parent directory.

**$cal:** to show the date in a calendar format.

**$date:** to show the date in different format.

**Mkdir –** creates a new directory.

**mkdir abc xyz** – creates two directories named abc and xyz.

**mkdir dir-{a..z}**- creates directories naming dir-a upto dir-z

**mkdir -p a/b/c/d-** creates a path. Like d inside c, c inside b, and b inside a.

**rmdir-** remove the directory.

**rmdir dir-{a..z}-** removes the directories from dir-a upto dir-z

**rm -r dir1-** removes the single directory dr1

**Touch:** touch creates an empty file.

**Cat:** cat creates a file with content it or to view the content in a fille.

to create a file – $ cat > a.txt

to show the content of file - $ cat a.txt

to append the content with the already exsisting content of exsisting file –

$ cat>> a.txt

To show the line no. in the file - $ cat -n a.txt

To skip the blank lines - $ cat -b a.txt

To copy content in a new file - $ cat a.txt > c.txt

To reverse content of file - $ tac a.txt

To reverse each line – $ rev a.txt

To print first 10 lines of the file - $ head a.txt

To print first 4 lines of the file - $ head -n 4 a.txt

To print last 10 lines of the file - $ tail a.txt

To print last 4 lines of the file - $ tail -n 4 a.txt