# Linux Commands

**whoami** : gives the user name

**clear** : clears the screen

**cd ~** : takes to the home directory from anywhere

**cd** - : takes to the last visited directory

**ls** : it lists files and directories

**ls -l** : gives long listing of files based on ASCII values of file names (gives more details like permission, number of links, owner, group, access, size, last date modified, file name)

**ls -lt** : gives a long listing of files based on time. Newest comes first.

**ls -ltr** : gives a long listing of files based on time. Oldest comes first.

**ls -lr** : gives a long listing of files based on descending ASCII values of file names.

**ls -lR**: Recursive long listing. Go inside each folder, and gives the files and directories names

**ls -R** : recursive without long listing

**ls -a** : list all the files including hidden files.

**ls -R -a** : recursively lists all the folders and files including the hidden ones

The above command can be written in several ways:
i) ls -R -a     ii) ls -Ra     iii)ls -aR    iv)ls -a -R

All the 4 commands will give the same result and ordering does not matter.
touch filename : it creates an empty file with name filename

There are 3 types of permissions:
**r** - read(it has a weightage of 4)
**w** - write(it has a weightage of 2)
**x** - execute(it has a weightage of 1)

There are three types of users whom permission is given:
i) **owner**
ii) **group**
iii) **other**

**chmod 777 file1** : gives all permission to everyone for file1. chmod is used to change permission.

**chmod 764 file1**: has the following meaning -
i) read, write, execute permission to owner(since 4 + 2 + 1 = 7)
ii) read write permission to group(4 + 2 = 6)
iii) read permission to other(4)

Similarly, **chmod 742 file1**: has the following meaning -
i) read, write, execute permission to owner(since 4 + 2 + 1 = 7)
ii) read permission to group(4)
iii) execute permission to other(2)

**cat file1**: it shows the content of the file1

**mkdir diectory_name** : it creates directory with name diectory_name

**rmdir diectory_name** : It removes/deletes the directory with name directory_name. It has one limitation, it can only delete an empty directory.

**rm file1** : removes/delete file1

If we want to delete a directory which is not empty, we can use the following command:
**rm -R directory_name**: this will delete directory directory_name which is not empty.

**cp location1 location2**: it copies files and directory from location1 to location2

**cp -R directory1 directory2** : recursively copy directory1 to directory2

**mv location1 location2**: it moves files and directory from location1 to location2. It uses cut paste procedure unlike cp command which uses copy paste procedure.

If we want to create a file with name samplefile.txt, write some content inside it and then save it, we can follow the following approach:
i) **vi samplefile.txt** : this will open a file with name samplefile.txt. If not present, it will create it
ii) **i** : this will allow us insert texts inside samplefile.txt
iii) **esc** : escape button helps to escape out of insert mode
iv) **:wq** : to save and exit the file

**head file1** : it gives first 10 lines of file1

**tail file1** : it gives last 10 lines of file1

**cat file1**: gives entire content of the file

**history** : lists all the commands that has been previously executed

**wc file1** : gives word count of file1

**cat > samplefile.txt** : helps in creating or overwriting existing file samplefile.txt

**ctrl + D** : to save the file

**cat >> samplefile.txt** : opens the samplefile.txt in append mode

**Cat file1 file2 >> file3** : takes the content from file1 and file2, and append it to file3

**grep Abhishek *** : search for word Abhishek in all the files present in the current directory