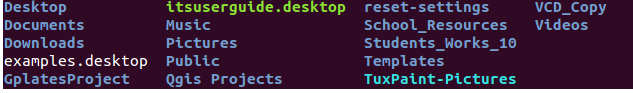
**Submission date :** 04-09-2019

**Linux Commands**

Linux is a Unix-Like operating system. All the Linux/Unix commands are run in the terminal provided by the Linux system. This terminal is just like command prompt of Windows OS. Linux/Unix commands are *case-sensitive.* The terminal can be used to accomplish all Administrative tasks. This includes package installation, file manipulation, and user management. Linux terminal is user-interactive. The terminal outputs the results of commands which are specified by the user itself. Execution of typed command is done only after you press the Enter key.  
**15 essential Linux commands :**

1. pwd
2. ls
3. cd
4. mkdir & rmdir
5. rm
6. touch
7. man & --help
8. cp
9. mv
10. locate
11. Echo
12. Cat
13. nano, vi, jed
14. Sudo
15. du

Description:  
**1. pwd** — When you first open the terminal, you are in the home directory of your user. To know which directory you are in, you can use the **“pwd”** command. It gives us the absolute path, which means the path that starts from the root. The root is the base of the Linux file system. It is denoted by a forward slash( / ). The user directory is usually something like "/home/username".  
   
**2. Is –**Usethe **"ls"** command to know what files are in the directly you are in.You can see all the hidden files by using the command**“ls –a”.**



**3. cd—**Use the**"cd ”** command to go to a directory.For example you are in the home folder,and you want to go to the downloads folder,then you can type in **“cd Downloads”.**

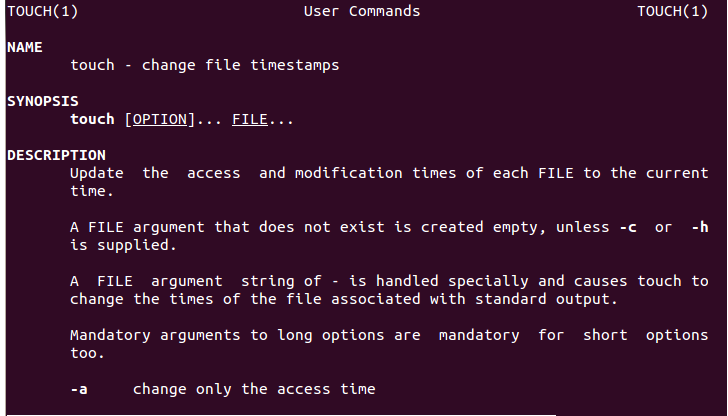


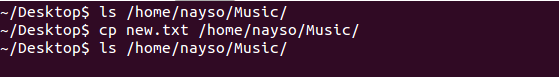
**4. mkdir & rmdir**— Use the **mkdir** command when you need to create a folder or a directory. For example, if you want to make a directory called “DIY”, then you can type **“mkdir DIY**”. Remember, as told before, if you want to create a directory named “DIY Hacking”, then you can type “mkdir **DIY\ Hacking**”. Use **rmdir** to delete a directory. But **rmdir** can only be used to delete an empty directory. To delete a directory containing files, use **rm**.  


**5. rm-**Use the ‘rm’command to delete files and directions.  

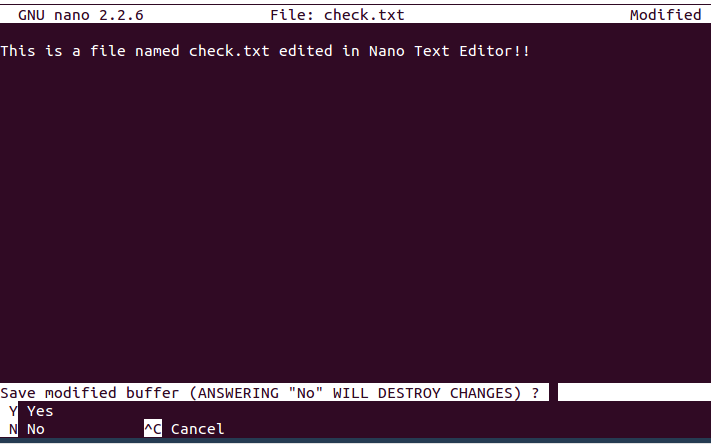

**6.Touch-**the Touch command is used to create a file.  


**7. man & --help** — To know more about a command and how to use it, use the **man** command. It shows the manual pages of the command. For example, “**man cd**” shows the manual pages of the **cd**command. Typing in the command name and the argument helps it show which ways the command can be used (e.g., **cd –help**).



**8. cp** — Use the **cp**command to copy files through the command line. It takes two arguments: The first is the location of the file to be copied, the second is where to copy.  


**9. mv** — Use the **mv** command to move files through the command line. We can also use the **mv** command to rename a file. For example, if we want to rename the file “**text**” to “**new**”, we can use “**mv text new**”. It takes the two arguments, just like the**cp** command.  
**10. locate** — The **locate** command is used to locate a file in a Linux system, just like the search command in Windows. This command is useful when you don't know where a file is saved or the actual name of the file. Using the -i argument with the command helps to ignore the case (it doesn't matter if it is uppercase or lowercase). So, if you want a file that has the word “hello”, it gives the list of all the files in your Linux system containing the word "hello" when you type in “**locate -i hello**”. If you remember two words, you can separate them using an asterisk (\*). For example, to locate a file containing the words "hello" and "this", you can use the command “**locate -i \*hello\*this”.  
11. echo**— The "**echo**" command helps us move some data, usually text into a file. For example, if you want to create a new text file or add to an already made text file, you just need to type in, “**echo hello, my name is mukit>> new.txt**”. You do not need to separate the spaces by using the backward slash here, because we put in two triangular brackets when we finish what we need to write.  
**12. Cat-**Use the Cat command to display the contents of a file.

**13. nano, vi, jed — nano** and **vi** are already installed text editors in the Linux command line. The **nano** command is a good text editor that denotes keywords with color and can recognize most languages. And **vi** is simpler than **nano**. You can create a new file or modify a file using this editor. For example, if you need to make a new file named **"check.txt**", you can create it by using the command “**nano check.txt**”. You can save your files after editing by using the sequence Ctrl+X, then Y (or N for no). In my experience, using **nano**for HTML editing doesn't seem as good, because of its color, so I recommend **jed**text editor. We will come to installing packages soon.  


**14. sudo** — A widely used command in the Linux command line, **sudo** stands for "SuperUser Do". So, if you want any command to be done with administrative or root privileges, you can use the **sudo** command. For example, if you want to edit a file like **viz. alsa-base.conf**, which needs root permissions, you can use the command – **sudo nano alsa-base.conf**. You can enter the root command line using the command “**sudo bash**”, then type in your user password. You can also use the command “**su**” to do this, but you need to set a root password before that. For that, you can use the command “**sudo passwd**”(not misspelled, it is **passwd**). Then type in the new root password.  
**15. du** — Use **du** to know the disk usage of a file in your system. If you want to know the disk usage for a particular folder or file in Linux, you can type in the command **df** and the name of the folder or file. For example, if you want to know the disk space used by the documents folder in Linux, you can use the command “**du Documents**”. You can also use the command “**ls -lah**” to view the file sizes of all the files in a folder.  
