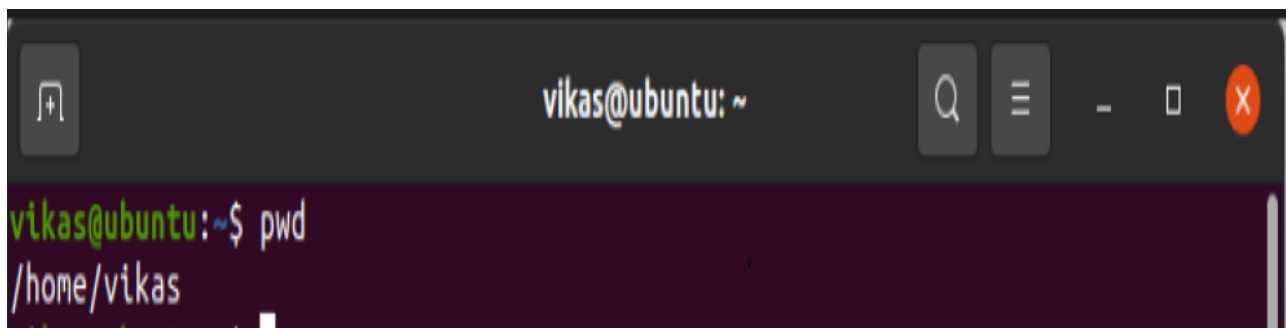


## Linux Commands :-

### 1) pwd :-

When we first open the terminal, we are in the home directory of our system. To know which directory we are in then we can use the “**pwd**” command. It gives us the absolute path, which means the path that starts from the root.

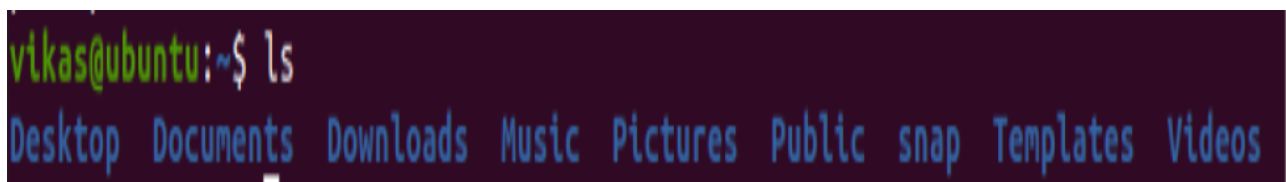
A screenshot of a Linux terminal window. The title bar shows 'vikas@ubuntu: ~'. The terminal prompt is 'vikas@ubuntu:~\$'. The user has entered the command 'pwd'. The output of the command is '/home/vikas', which is displayed on the line immediately following the command.

```
vikas@ubuntu:~$ pwd
/home/vikas
```

### 2) ls :-

We Use the "ls" command to know what files are in the directory we are in.

we can see all the hidden files by using the command “ls -a”.

A screenshot of a Linux terminal window. The title bar shows 'vikas@ubuntu: ~'. The terminal prompt is 'vikas@ubuntu:~\$'. The user has entered the command 'ls'. The output of the command is a list of directories: 'Desktop Documents Downloads Music Pictures Public snap Templates Videos', which is displayed on the line immediately following the command.

```
vikas@ubuntu:~$ ls
Desktop Documents Downloads Music Pictures Public snap Templates Videos
```

```
vikas@ubuntu:~$ ls -a
.          .cache      .gnupg      .profile    Templates
..         .config     .local      Public      Videos
.bash_history Desktop    .mozilla    snap
.bash_logout Documents Music       .ssh
.bashrc     Downloads Pictures    .sudo_as_admin_successful
```

### 3) cd :-

We Use the "**cd**" command to go to a directory. For example, if we are in the home folder and we want to go to the another folder then we can type "**cd filename**".

```
vikas@ubuntu:~$ cd Desktop
vikas@ubuntu:~/Desktop$
```

To go back from a folder to the folder before that then we can type "**cd ..**". The two dots represent back.

### 4) mkdir :-

We Use the **mkdir** command when we need to create a folder or a directory.

```
vikas@ubuntu:~$ mkdir OS
vikas@ubuntu:~$ ls
Desktop  Downloads  OS      Public  Templates
Documents Music      Pictures snap    Videos
```

If we want to create a directory named "DIY Hacking", then we can type "**mkdir DIY\ Hacking**".

```

vikas@ubuntu:~$ mkdir OS\Programming
vikas@ubuntu:~$ ls
Desktop  Downloads  OS          Pictures  snap      Videos
Documents Music      OSProgramming Public    Templates
vikas@ubuntu:~$ mkdir OS\ programming
vikas@ubuntu:~$ ls
Desktop  Downloads  OS          OSProgramming  Public  Templates
Documents Music      'OS programming' Pictures      snap    Videos

```

### 5) rmdir :-

We use **rmdir** to delete a directory. But **rmdir** can only be used to delete an empty directory.

```

vikas@ubuntu:~$ rmdir OS\Programming
vikas@ubuntu:~$ ls
Desktop  Downloads  OS          Pictures  snap      Videos
Documents Music      'OS programming' Public    Templates
vikas@ubuntu:~$ rmdir OS\ programming
vikas@ubuntu:~$ ls
Desktop  Downloads  OS          Public  Templates
Documents Music      Pictures  snap    Videos

```

### 6) rm :-

We use the **rm** command to delete files and directories. We use "**rm -r**" to delete just the directory. It deletes both the folder and the files if we using using only the **rm** command.

```
vikas@ubuntu: ~/Desktop/OS
vikas@ubuntu:~$ cd Desktop
vikas@ubuntu:~/Desktop$ cd OS
vikas@ubuntu:~/Desktop/OS$ ls
vikas@ubuntu:~/Desktop/OS$ mkdir movies
vikas@ubuntu:~/Desktop/OS$ ls
movies
vikas@ubuntu:~/Desktop/OS$ mkdir Folder
vikas@ubuntu:~/Desktop/OS$ ls
Folder  movies
vikas@ubuntu:~/Desktop/OS$ rm movies
rm: cannot remove 'movies': Is a directory
vikas@ubuntu:~/Desktop/OS$ rm -r movies
vikas@ubuntu:~/Desktop/OS$ ls
Folder
vikas@ubuntu:~/Desktop/OS$ rm
rm: missing operand
Try 'rm --help' for more information.
vikas@ubuntu:~/Desktop/OS$ rm -r Folder
vikas@ubuntu:~/Desktop/OS$ ls
vikas@ubuntu:~/Desktop/OS$
```

## 7) Touch :-

The **touch** command is used to create a file. It can be anything, from an empty txt file to an empty zip file.

```
vikas@ubuntu:~/Desktop$ ls
Projects
vikas@ubuntu:~/Desktop$ touch new.txt
vikas@ubuntu:~/Desktop$ ls
new.txt  Projects
vikas@ubuntu:~/Desktop$
```

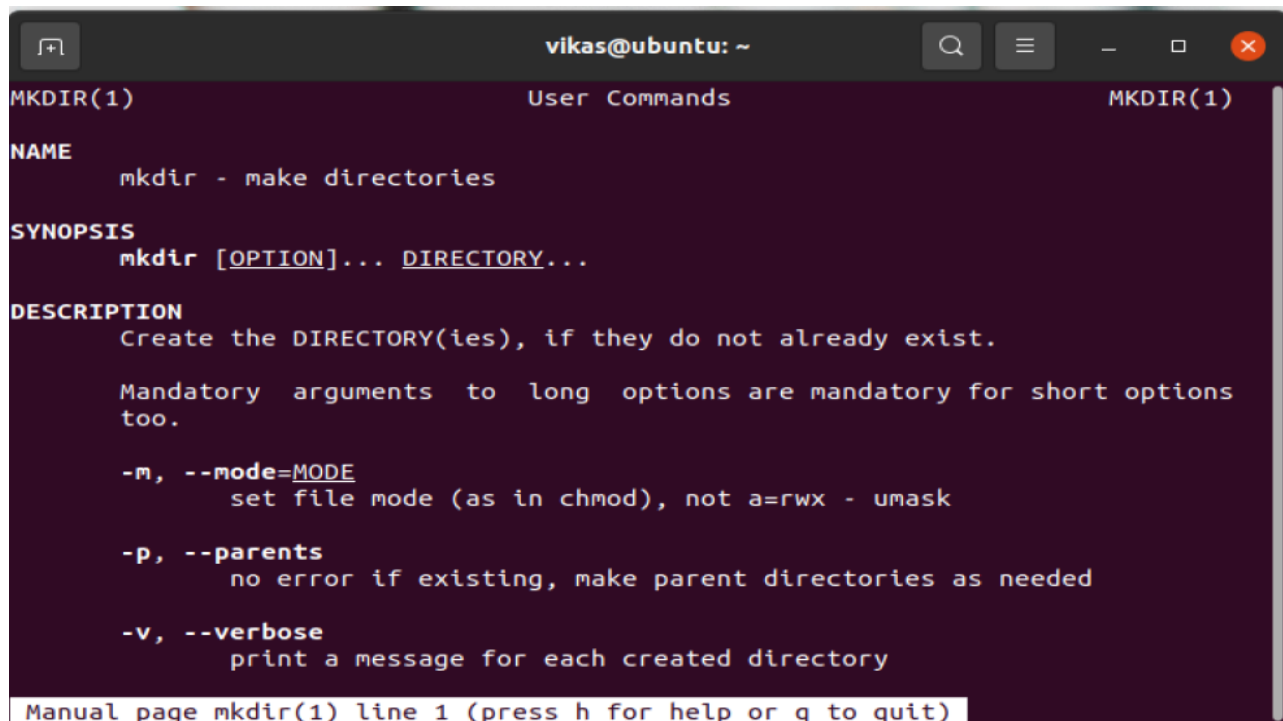
## 8) **man** :-

To know more about a command and how to use it we use the **man** command . It shows the manual pages of the command.

For example, “**man cd**”

It show the manual pages of the **cd** command.

```
vikas@ubuntu:~$ man mkdir
vikas@ubuntu:~$
```



The screenshot shows a terminal window titled "vikas@ubuntu: ~" with standard window controls. The terminal displays the manual page for the `mkdir` command. At the top, it says "MKDIR(1)" on the left and right, and "User Commands" in the center. The content is organized into sections: "NAME" with the description "mkdir - make directories", "SYNOPSIS" with the command format "mkdir [OPTION]... DIRECTORY...", and "DESCRIPTION" which explains that it creates directories if they don't exist and lists several options: `-m, --mode=MODE` (set file mode), `-p, --parents` (no error if existing, make parent directories), and `-v, --verbose` (print a message for each created directory). At the bottom, a status bar reads "Manual page mkdir(1) line 1 (press h for help or q to quit)".

```
vikas@ubuntu: ~
MKDIR(1)                                User Commands                                MKDIR(1)

NAME
    mkdir - make directories

SYNOPSIS
    mkdir [OPTION]... DIRECTORY...

DESCRIPTION
    Create the DIRECTORY(ies), if they do not already exist.

    Mandatory arguments to long options are mandatory for short options
    too.

    -m, --mode=MODE
        set file mode (as in chmod), not a=rwx - umask

    -p, --parents
        no error if existing, make parent directories as needed

    -v, --verbose
        print a message for each created directory

Manual page mkdir(1) line 1 (press h for help or q to quit)
```

## 9) **cp** :-

We use the **cp** command to copy files through the command line. It takes two arguments :- first is the

location of the file to be copied and second is where to copy.

```
vikas@ubuntu:~$ cd Desktop
vikas@ubuntu:~/Desktop$ ls
new.txt  Projects
vikas@ubuntu:~/Desktop$ cp new.txt Projects
vikas@ubuntu:~/Desktop$ ls Projects
a.out  new.txt  systemcall.c
vikas@ubuntu:~/Desktop$
```

### 10) **mv** :-

We 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**”.

```
vikas@ubuntu:~/Desktop$ ls
new.txt  Projects
vikas@ubuntu:~/Desktop$ mv new.txt newer.txt
vikas@ubuntu:~/Desktop$ ls
newer.txt  Projects
vikas@ubuntu:~/Desktop$
```

### 11) **locate** :-

The **locate** command is used to locate a file in a Linux system, just like the search command in Windows.

```
vikas@ubuntu:~/Desktop$ locate systemcall.c
/home/vikas/Desktop/Projects/systemcall.c
vikas@ubuntu:~/Desktop$
```

## 12) **sudo** :-

A widely used command in the Linux command line, **sudo** stands for "Super User Do". So, if we want any command to be done with administrative or root privileges then we can use the **sudo** command.

## 13) **apt-get** :-

We use **apt** to work with packages in the Linux command line. Use **apt-get** to install packages. This requires root privileges, so use the **sudo** command with it.

```
sudo apt install mlocate

vikas@ubuntu:~/Desktop$ sudo apt install mlocate
[sudo] password for vikas:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  ncache
The following NEW packages will be installed:
  mlocate
0 upgraded, 1 newly installed, 0 to remove and 55 not upgraded.
Need to get 50.1 kB of archives.
After this operation, 258 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 mlocate amd64 0.26-3ubuntu3 [50.1 kB]
Fetched 50.1 kB in 1s (35.8 kB/s)
Selecting previously unselected package mlocate.
(Reading database ... 187715 files and directories currently installed.)
Preparing to unpack .../mlocate_0.26-3ubuntu3_amd64.deb ...
Unpacking mlocate (0.26-3ubuntu3) ...
Setting up mlocate (0.26-3ubuntu3) ...
update-alternatives: using /usr/bin/mlocate to provide /usr/bin/locate (locate)
in auto mode
Adding group 'mlocate' (GID 133) ...
Done.
Initializing mlocate database; this may take some time... done
Processing triggers for man-db (2.9.1-1) ...
vikas@ubuntu:~/Desktop$
```

## 14) echo :-

The "**echo**" command helps us move some data, usually text into a file.

For example, if we want to create a new text file or add to an already made text file then we just need to type



“echo hello, my name is Vikas>> new.txt”.

### 15) cat :-

We use the **cat** command to display the contents of a file. It is usually used to easily view programs.

```
vikas@ubuntu:~/Desktop$ echo Hello, My name is VIKAS. >> newer.txt
vikas@ubuntu:~/Desktop$ cat newer.txt
Hello, My name is VIKAS.
vikas@ubuntu:~/Desktop$ echo I am Student of Netaji Subhas University Of Technology. >> newer.txt
vikas@ubuntu:~/Desktop$ cat newer.txt
Hello, My name is VIKAS.
I am Student of Netaji Subhas University Of Technology.
```

### 16) df :-

We use the **df** command to see the available disk space in each of the partitions in our system.

```
vikas@ubuntu:~/Desktop$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            973560         0   973560   0% /dev
tmpfs           200548      1608   198940   1% /run
/dev/sda5       19992176 7465928 11487656  40% /
tmpfs           1002728         0   1002728   0% /dev/shm
tmpfs            5120          4     5116   1% /run/lock
tmpfs           1002728         0   1002728   0% /sys/fs/cgroup
/dev/loop0       66432      66432         0 100% /snap/gtk-common-themes/1514
/dev/loop2       224256     224256         0 100% /snap/gnome-3-34-1804/66
/dev/loop4        52352     52352         0 100% /snap/snap-store/518
/dev/loop3        56832     56832         0 100% /snap/core18/1988
/dev/sda1        523248          4    523244   1% /boot/efi
/dev/loop5        56832     56832         0 100% /snap/core18/1997
/dev/loop6        33152     33152         0 100% /snap/snapd/11402
tmpfs           200544         44    200500   1% /run/user/1000
/dev/sr0         104626    104626         0 100% /media/vikas/CDROM
/dev/sr1         2809792 2809792         0 100% /media/vikas/Ubuntu 20.04.2.0 LTS amd64
/dev/loop7        33152     33152         0 100% /snap/snapd/11588
```

If we want it shown in megabytes then we can use the command “**df -m**”.

```
vikas@ubuntu:~/Desktop$ df -m
Filesystem      1M-blocks  Used Available Use% Mounted on
udev            951      0      951    0% /dev
tmpfs           196      2      195    1% /run
/dev/sda5       19524   7291    11219   40% /
tmpfs           980      0      980    0% /dev/shm
tmpfs            5      1       5     1% /run/lock
tmpfs           980      0      980    0% /sys/fs/cgroup
/dev/loop0       65     65       0 100% /snap/gtk-common-themes/1514
/dev/loop2      219    219       0 100% /snap/gnome-3-34-1804/66
/dev/loop4       52     52       0 100% /snap/snap-store/518
/dev/loop3       56     56       0 100% /snap/core18/1988
/dev/sda1        511      1      511    1% /boot/efi
/dev/loop5       56     56       0 100% /snap/core18/1997
/dev/loop6       33     33       0 100% /snap/snapd/11402
tmpfs           196      1      196    1% /run/user/1000
/dev/sr0         103    103       0 100% /media/vikas/CDROM
/dev/sr1        2744   2744       0 100% /media/vikas/Ubuntu 20.04.2.0 LTS amd64
/dev/loop7       33     33       0 100% /snap/snapd/11588
```

## 17) **du** :-

We use **du** to know the disk usage of a file in our system. If we want to know the disk usage for a particular folder or file in Linux then we type the command **df** and the name of the folder or file.

```
vikas@ubuntu:~$ du Documents
4      Documents
vikas@ubuntu:~$
```

## 18) ls -lah :-

We use the command “**ls -lah**” to view the file sizes of all the files in a folder.

```
vikas@ubuntu:~$ ls -lah
total 88K
drwxr-xr-x 18 vikas vikas 4.0K Apr 11 22:16 .
drwxr-xr-x  3 root  root  4.0K Apr  4 12:13 ..
-rw-r--r--  1 vikas vikas  513 Apr 11 22:28 .bash_history
-rw-r--r--  1 vikas vikas  220 Apr  4 12:13 .bash_logout
-rw-r--r--  1 vikas vikas 3.7K Apr  4 12:13 .bashrc
drwxrwxr-x 15 vikas vikas 4.0K Apr  4 11:24 .cache
drwxr-xr-x 13 vikas vikas 4.0K Apr  4 11:24 .config
drwxr-xr-x  3 vikas vikas 4.0K Apr 11 22:57 Desktop
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Documents
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Downloads
drwx----- 3 vikas vikas 4.0K Apr  5 06:35 .gnupg
drwxr-xr-x  3 vikas vikas 4.0K Apr  4 06:51 .local
drwx----- 5 vikas vikas 4.0K Apr  4 06:58 .mozilla
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Music
drwxrwxr-x  2 vikas vikas 4.0K Apr 11 22:12 OS
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Pictures
-rw-r--r--  1 vikas vikas  807 Apr  4 12:13 .profile
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Public
drwxr-xr-x  3 vikas vikas 4.0K Apr  4 06:55 snap
drwx----- 2 vikas vikas 4.0K Apr  4 10:31 .ssh
-rw-r--r--  1 vikas vikas    0 Apr  5 06:40 .sudo_as_admin_successful
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Templates
drwxr-xr-x  2 vikas vikas 4.0K Apr  4 06:51 Videos
```

## 19) uname :-

We use **uname** command to show the information about the system our Linux distro is running.

```
vikas@ubuntu:~$ uname
Linux
```

Using the command “**uname -a**” prints most of the information about the system. This prints the kernel release date, version, processor type, etc.

```
vikas@ubuntu:~$ uname -a
Linux ubuntu 5.8.0-43-generic #49~20.04.1-Ubuntu SMP Fri Feb 5 09:57:56 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
```

## 20) **hostname** :-

We use **hostname** to know your name in your host or network. It Basically displays our hostname and IP address. Just typing “**hostname**” gives the output.

“**hostname -I**” command gives our IP address in your network.

```
vikas@ubuntu:~$ hostname
ubuntu
vikas@ubuntu:~$ hostname -I
192.168.192.128
```

## 21) **clear** :-

We can use the **clear** command to clear the terminal if it gets filled up with too many commands.

## **22) exit :-**

we can exit from the terminal by using the **exit** command.

## **23) sudo halt & sudo reboot :-**

We can power off or reboot the computer by using the command **sudo halt** and **sudo reboot**.