

Practical No. 1

Aim:- Install Linux OS on standalone machine

Theory:-

step 1: Download Linux ISO

Download and install virtual Box.

step 2: open virtual Box and click on new symbol give the virtual as a name and allocate RAM, storage, process core.

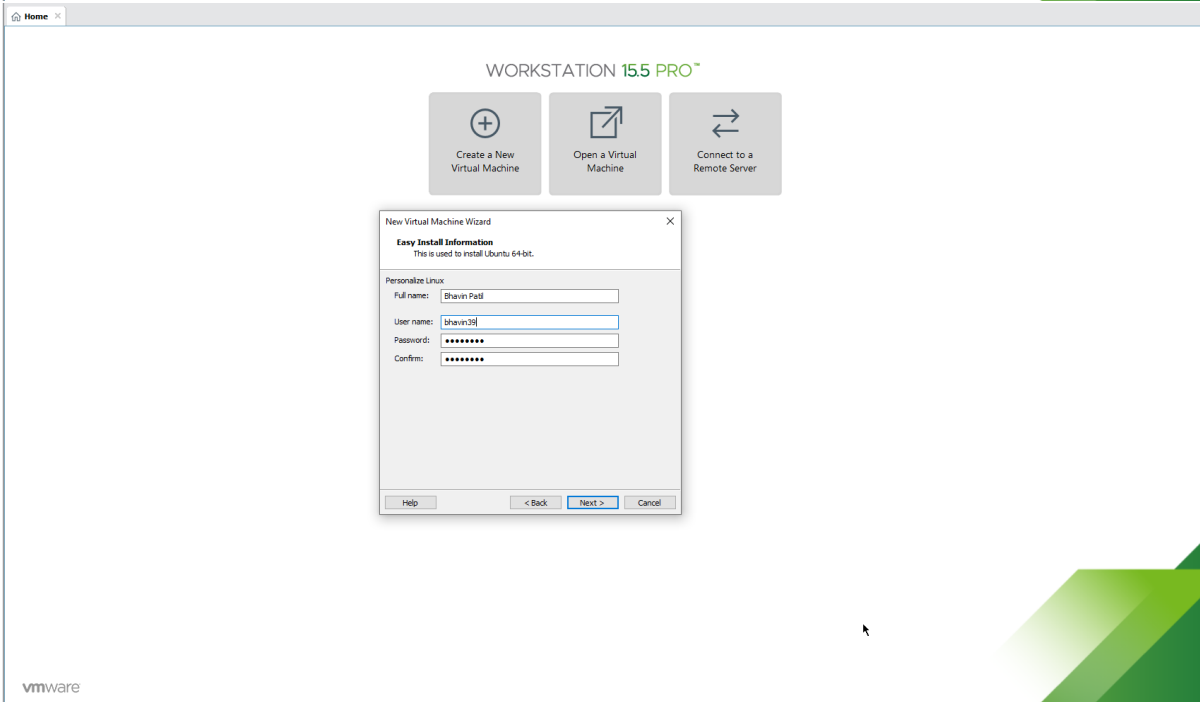
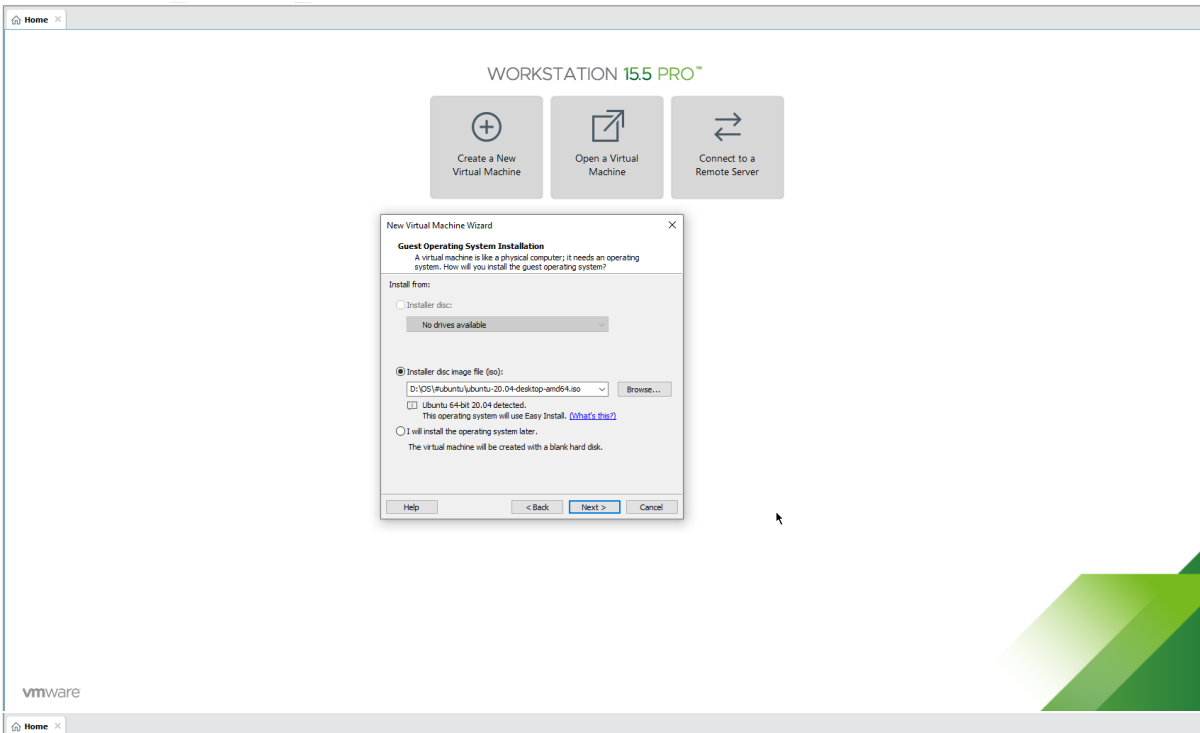
step 3: click on install and select accordingly

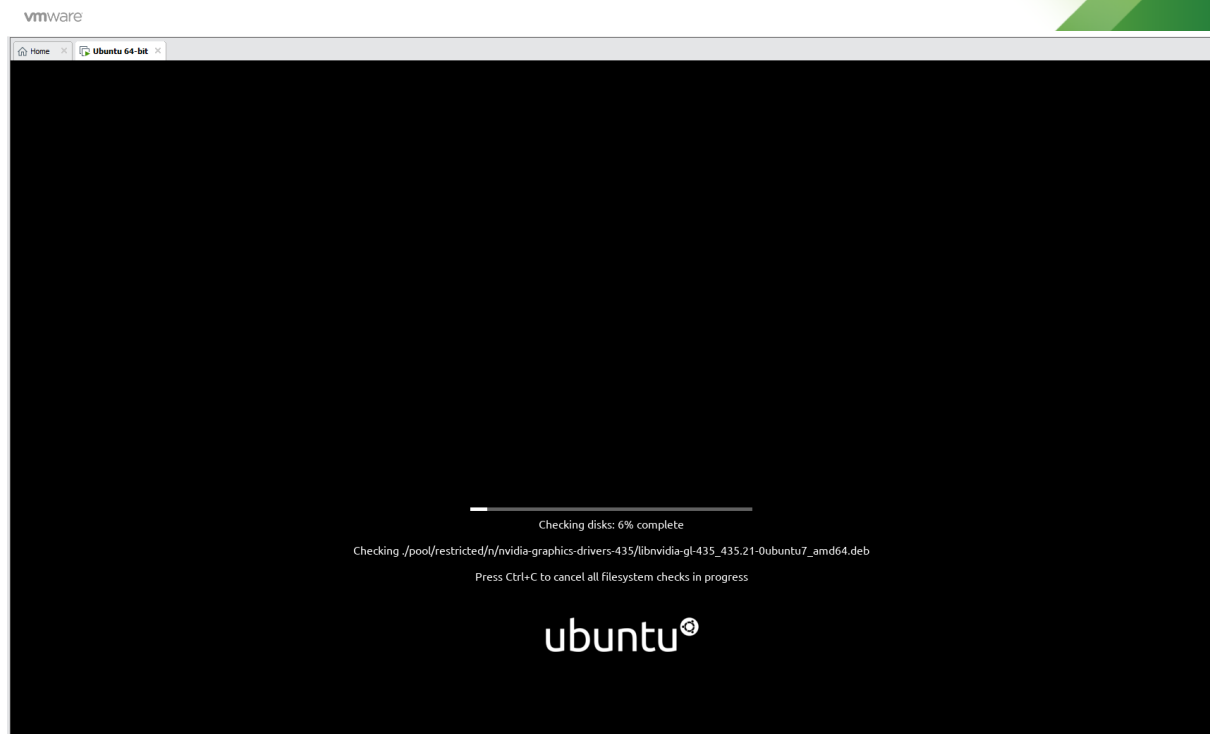
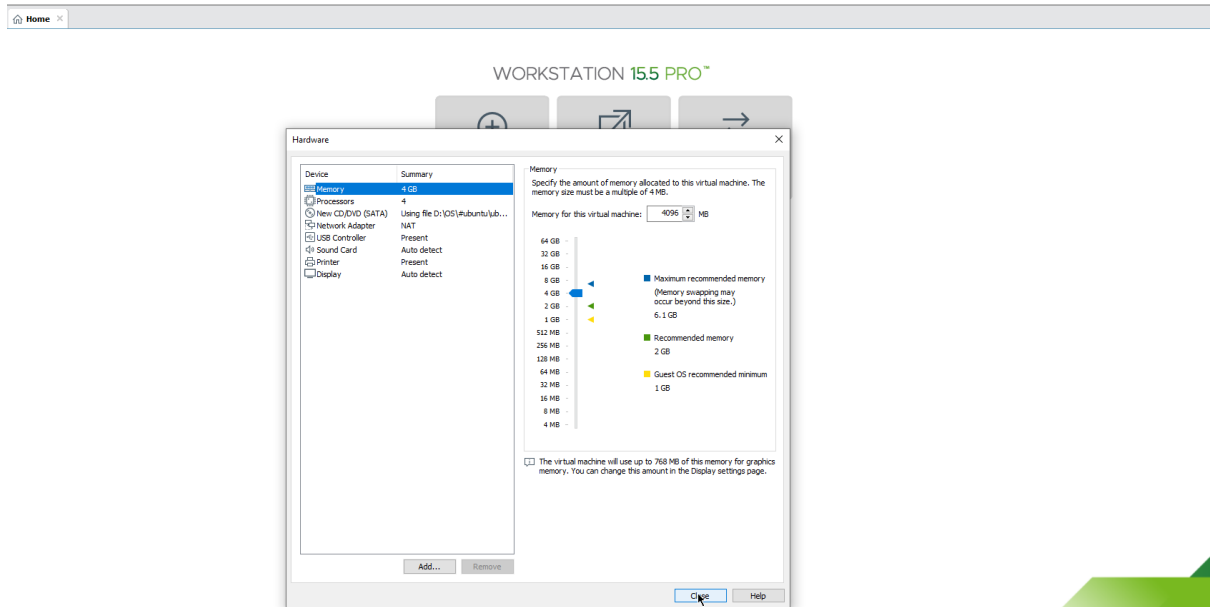
step 4: select 'Erase and Install Ubuntu?' and click continue.

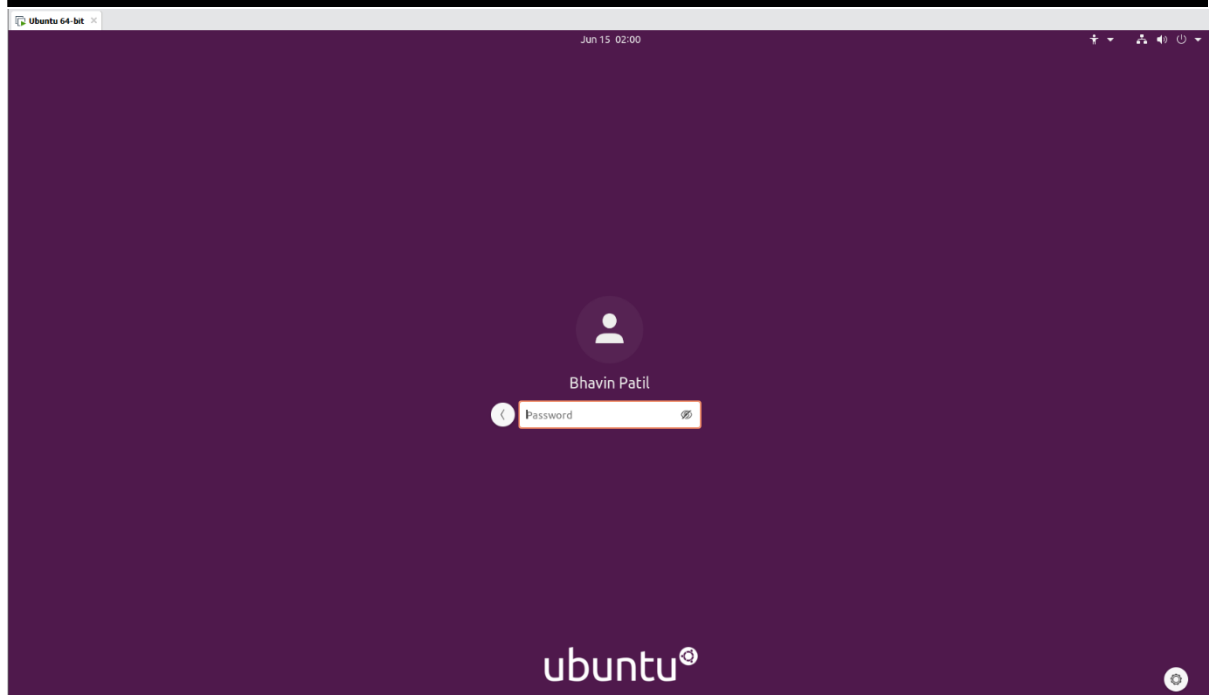
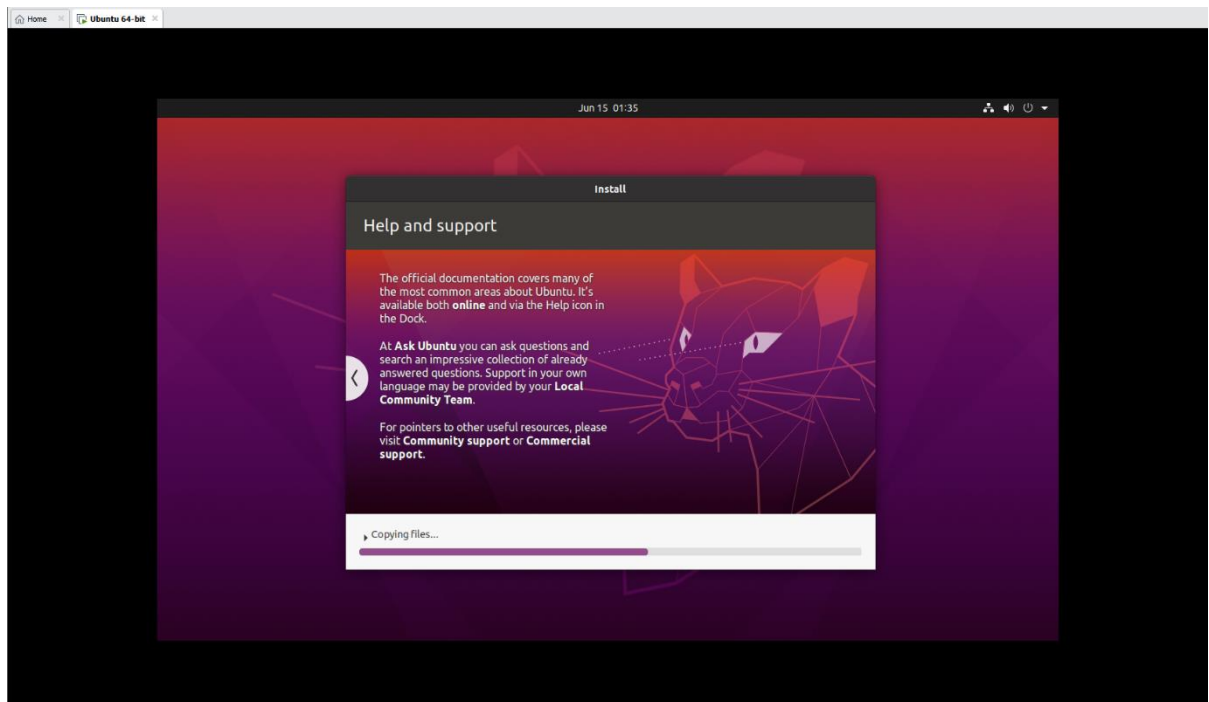
step 5: create a username and password for login and also for root user.

step 6: Once installation completes, it will restart and will show a login window

step 7: login in as a root and your Linux is ready to use.







Practical No. 2

Aim: Perform Logging/Logout via terminal and network.

Theory :-

The Login program is used to establish a new session with the system. It is normally invoked automatically by responding to the login prompt on the user terminal.

Attempting to execute login from any shell but the login shell produces an error message. The user is then prompted for a password where appropriate. If password is correct then it will be redirected to that user's current directory or if it is not then it will show error.

Syntax : `sudo Login 'username'`

If you are using Ubuntu server and you are continued to the terminal screen, you can use the `exit` command.

To logout from ubuntu desktop session, go to the top right corner and click to bring the system tray. Click on it and it shows the logout options.

When you click on log out button, it will open a dialogue box and ask for your confirmation. If it is received no input from you, you will be logged out.

```

bhavin39@ubuntu: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

bhavin39@ubuntu:~$ sudo login bhavin39
[sudo] password for bhavin39:
Password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-26-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

538 updates can be installed immediately.
237 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

bhavin39@ubuntu:~$ logout
bhavin39@ubuntu:~$
```

```

bhavin39@ubuntu: ~/Desktop/Bhavin
MAN(1) Manual pager utils MAN(1)

NAME
  man - an interface to the system reference manuals

SYNOPSIS
  man [man options] [[section] page ...] ...
  man -k [apropos options] regexp ...
  man -K [man options] [section] term ...
  man -f [whatis options] page ...
  man -l [man options] file ...
  man -W [-W [man options] page ...

DESCRIPTION
  man is the system's manual pager. Each page argument given to man is normally the name of a program, utility or function. The manual page associated with each of these arguments is then found and displayed. A section, if provided, will direct man to look only in that section of the manual. The default action is to search in all of the available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if page exists in several sections.

  The table below shows the section numbers of the manual followed by the types of pages they contain.

  1 Executable programs or shell commands
  2 System calls (functions provided by the kernel)
  3 Library calls (functions within program libraries)
  4 Special files (usually found in /dev)
  5 File formats and conventions, e.g. /etc/passwd
  6 Games
  7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7)
  8 System administration commands (usually only for root)
  9 Kernel routines [Non standard]

  A manual page consists of several sections.

  Conventional section names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS, and SEE ALSO.

  The following conventions apply to the SYNOPSIS section and can be used as a guide in other sections.

  bold text      type exactly as shown.
  italic text    replace with appropriate argument.
  [-abc]        any or all arguments within [ ] are optional.
  -a|-b         options delimited by | cannot be used together.
  argument ...  argument is repeatable.
  [expression] ... entire expression within [ ] is repeatable.

  Exact rendering may vary depending on the output device. For instance, man will usually not be able to render italics when running in a terminal, and will typically use underlined or coloured text instead.

  The command or function illustration is a pattern that should match all possible invocations. In some cases it is advisable to illustrate several exclusive invocations.

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

Practical No. 3

Aim: Perform general purpose utility commands in Linux.

Theory:-

1. Pwd command :- Use the pwd to find out the path of the current working directory you are in.
2. cd command :- cd command changes the current directory in linux.
3. ls command :- ls command used to view the contents of the directory.
4. cat command :- It is used to list the contents of a file on a standard output.
7. mkdir :- it is used to make new directories.
8. rmdir :- It is used to delete directories and content within it.
9. touch :- It will allow you to create a blank new file.
10. man :- It allows admin to display user manual that built in linux distribution.



bhavin39@ubuntu: ~/Desktop/Bhavin

```
bhavin39@ubuntu:~$ pwd
/home/bhavin39
bhavin39@ubuntu:~$ cd Desktop/
bhavin39@ubuntu:~/Desktop$ mkdir Bhavin
bhavin39@ubuntu:~/Desktop$ cd Bhavin/
bhavin39@ubuntu:~/Desktop/Bhavin$ touch Linux
bhavin39@ubuntu:~/Desktop/Bhavin$ ls
Linux
```


Practical No. 4

~~Assignment No. 2~~

Aim:- Use VI Editor.

Theory:-

Introduction -

The VI editor is a screen-based editor used by many Unix users. The VI editor has powerful features to aid programmers, but many beginning users avoid using VI because the different features overwhelm them, but also contains sections relevant to regular users of VI as well.

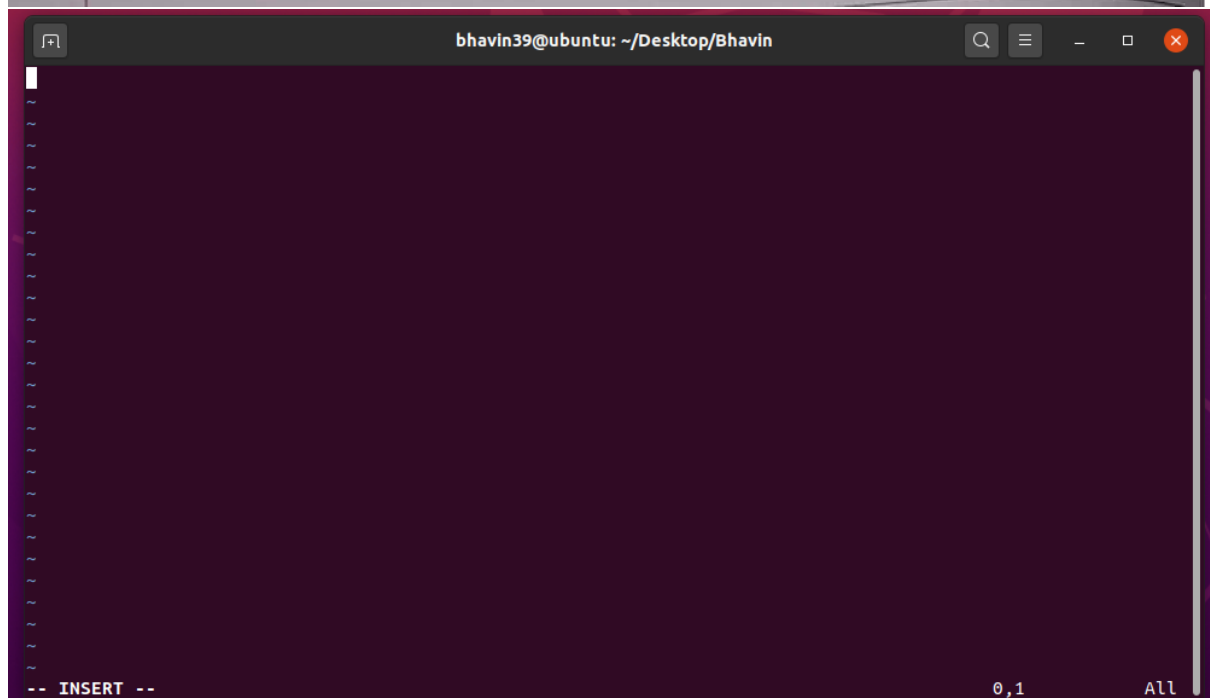
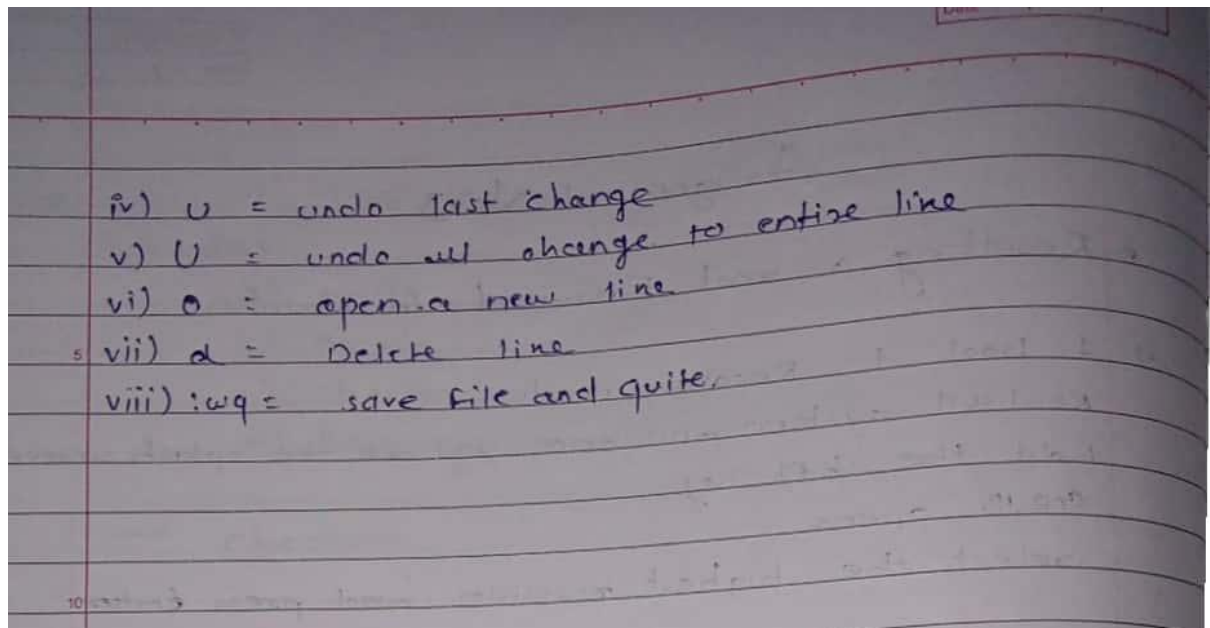
The VI editor lets a user create new files or edit existing files, the command to start the VI editor is 'vi', followed by the filename.

The first thing most users learn about the VI editor is that it has two modes i) Command and ii) Insert. The command mode allows to manipulate text which can be done with few keystrokes. The insert mode puts anything typed on keyboard into the file.

VI starts out in command mode and to put VI editor into Insert mode you need to type command which are 'a' and 'i'. Once you get in insert mode, you can get out of it by hitting escape key.

Commands:-

- i) i = insert mode
- ii) a = write after cursor
- iii) esc = terminate insert or mode



Practical No. 5

Aim:- write two shell script program in Vi editor

Theory:-

A shell program is a text file that contains standard UNIX and shell commands. Each line in shell program contains a single UNIX command exactly as it is you had typed them in your file.

The difference is that you can execute all commands in a shell program. Shell program are interpreted and not compiled program. This means when you run a shell program a child shell is started. This child shell read each line in the file.

```

bhavin39@ubuntu: ~/Desktop/Bhavin
echo -e "Enter a number: \c"
read num

i=2
rem=1

if [ $num -lt $i ];
then
    echo -e "$num is not a prime number.\n"
    exit 0
fi

while [ $i -le `expr $num / 2` -a $rem -ne 0 ];
do
    rem=`expr $num % $i`
    i=`expr $i + 1`
done

if [ $rem -ne 0 ];
then
    echo -e "$num is a prime number.\n"
else
    echo -e "$num is not a prime number.\n"
fi
~
~
~
"Prime.sh" 24L, 337C
24,2 All

bhavin39@ubuntu: ~/Desktop/Bhavin
bhavin39@ubuntu:~/Desktop/Bhavin$ ./Prime.sh
Enter a number: 45
45 is not a prime number.

bhavin39@ubuntu:~/Desktop/Bhavin$ ./Prime.sh
Enter a number: 39
39 is not a prime number.

bhavin39@ubuntu:~/Desktop/Bhavin$ ./Prime.sh
Enter a number: 29
29 is a prime number.

bhavin39@ubuntu:~/Desktop/Bhavin$
```


Practical No 6

Aim:- write any two shell script C program in vi editor.

Theory:- In order to run C program in Linux using vi editor you will need a C compiler. The most popular compiler is gcc. You can install gcc using following command
sudo apt install gcc.

Now to write a program, follow the steps:-

- i) Open terminal and vi file_name.c command
- ii) To edit the file press i key and write down the program
- iii) To save the file, press Esc button and type :wq
- iv) To compile the C file, type ~~file_name.c~~
gcc file_name.c
- v) To run the C program, type ./file_name

Program:-


```
bhavin39@ubuntu: ~/Desktop/Bhavin/C_pro  
bhavin39@ubuntu: ~/Desktop/Bhavin/C_programs  
#include <stdio.h>  
int main() {  
    int i, j, rows;  
    printf("Enter the number of rows: ");  
    scanf("%d", &rows);  
    for (i = 1; i <= rows; ++i) {  
        for (j = 1; j <= i; ++j) {  
            printf("* ");  
        }  
        printf("\n");  
    }  
    return 0;  
}  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
"Pyramid.c" 13L, 252C
```

```
bhavin39@ubuntu: ~/Desktop/Bhavin/C  
bhavin39@ubuntu: ~/Desktop/Bhavin/C_programs  
bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$ gcc Pyramid.c -o Pyramid  
bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$ ./Pyramid  
Enter the number of rows: 5  
*  
* *  
* * *  
* * * *  
* * * * *  
bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$
```



```
bhavin39@ubuntu: ~/Desktop/Bhavin/C_programs
bhavin39@ubuntu: ~/Desktop/Bhavin/C_programs
#include <stdio.h>
int main() {
    char operator;
    double first, second;
    printf("Enter an operator (+, -, *,): ");
    scanf("%c", &operator);
    printf("Enter two operands: ");
    scanf("%lf %lf", &first, &second);

    switch (operator) {
    case '+':
        printf("%.1lf + %.1lf = %.1lf", first, second, first + second);
        break;
    case '-':
        printf("%.1lf - %.1lf = %.1lf", first, second, first - second);
        break;
    case '*':
        printf("%.1lf * %.1lf = %.1lf", first, second, first * second);
        break;
    case '/':
        printf("%.1lf / %.1lf = %.1lf", first, second, first / second);
        break;
    // operator doesn't match any case constant
    default:
        printf("Error! operator is not correct");
    }

    return 0;
}
~
~
30,0-1 All
```

```
bhavin39@ubuntu: ~/Desktop/Bhavin/C_programs
bhavin39@ubuntu: ~/Desktop/Bhavin/C_programs
bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$ ls
Calculator.c  Pyramid  Pyramid.c
bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$ gcc Calculator.c -o Calculator
bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$ ./Calculator
Enter an operator (+, -, *,): *
Enter two operands: 5
5
5.0 * 5.0 = 25.0bhavin39@ubuntu:~/Desktop/Bhavin/C_programs$
```

Practical No. :- 7

Aim :- Create User

Theory :-

There are two ways to create user:
1. using graphical user management application
2. or from shell prompt we are going to create user from a shell prompt

1. Open a shell prompt

2. logged in as root if your not, type command `sudo su` and enter root password.

3. Type `useradd` and the name of user which you want to create.

e.g. `useradd user_name.`

4.

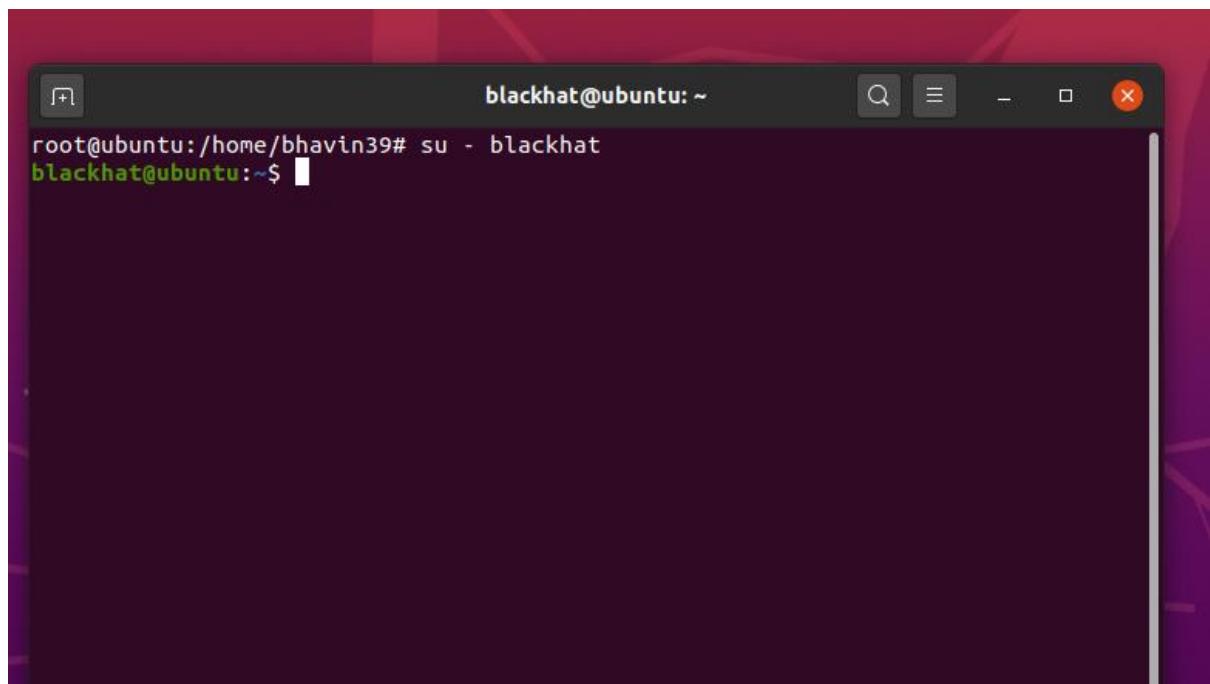
4. To create password for new user type `passwd` and username.

5. It will show to type new password, then enter a password and to confirm it you need to enter it again in the next line.

6. After that type `sudo username` to log into new user.

```
root@ubuntu: /home/bhavin39
bhavin39@ubuntu:~$ sudo su
[sudo] password for bhavin39:
root@ubuntu:/home/bhavin39#
```

```
root@ubuntu: /home/bhavin39
bhavin39@ubuntu:~$ sudo su
[sudo] password for bhavin39:
root@ubuntu:/home/bhavin39# adduser blackhat
Adding user `blackhat' ...
Adding new group `blackhat' (1001) ...
Adding new user `blackhat' (1001) with group `blackhat' ...
Creating home directory `/home/blackhat' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for blackhat
Enter the new value, or press ENTER for the default
    Full Name []: Blackhat
    Room Number []: 02
    Work Phone []: -
    Home Phone []: -
    Other []: -
Is the information correct? [Y/n] Y
root@ubuntu:/home/bhavin39# sudo blackhat
```



Practical No. 8

Aim :- Recover Root Password

~~Assignment No. 3~~

Theory :-

★ Resetting a root Password in Ubuntu.

1. Boot to Recovery mode.

Restart system and once you see the splash screen hold the shift key.

GRUB opens

Select the highest revision and press enter.

2. Drop out to root shell

Navigate to the option labeled root and press enter.

3. Remount File system with write-permissions to remount the drive with appropriate permission type:

`mount -o rw, remount /`

and press enter

4. Change Password.

type -

`passwd username`

and press enter. The system then asks to type a new password and to retype it.

Once entered and confirmed, reboot by entering:

`shutdown -r`

hit enter.

GNU GRUB version 2.04

```
*Ubuntu
Advanced options for Ubuntu
Memory test (memtest86+)
Memory test (memtest86+, serial console 115200)
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the commands
before booting or 'c' for a command-line.

GNU GRUB version 2.04

```
*Ubuntu, with Linux 5.4.0-26-generic
Ubuntu, with Linux 5.4.0-26-generic (recovery mode)
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the commands
before booting or 'c' for a command-line. ESC to return previous
menu.

Recovery Menu (filesystem state: read-only)

resume	Resume normal boot
clean	Try to make free space
dpkg	Repair broken packages
fsck	Check all file systems
grub	Update grub bootloader
network	Enable networking
root	Drop to root shell prompt
system-summary	System summary

<Ok>

```
root@ubuntu:~# mount -rw -o remount /
```

```
root@ubuntu:~# mount -rw -o remount /
root@ubuntu:~# passwd blackhat
New password: _
```

```
root@ubuntu:~# mount -rw -o remount /
root@ubuntu:~# passwd blackhat
New password:
Retype new password:
passwd: password updated successfully
root@ubuntu:~# _
```



```
root@ubuntu:~# mount -rw -o remount /
root@ubuntu:~# passwd blackhat
New password:
Retype new password:
passwd: password updated successfully
root@ubuntu:~# shutdown -r
```

Practical No. 9

Aim:- Run SUDO commands to access system privileges.

Theory:-

Sudo is a linux command used to execute programs as the root or super user. Sudo is shortening a command or program of superuser. Sometime we need to use sudo to run a command or program as root but do not want to logout or switch entire shell to root privilege.

*The user need to have a entry in sudo users to execute with sudo command.

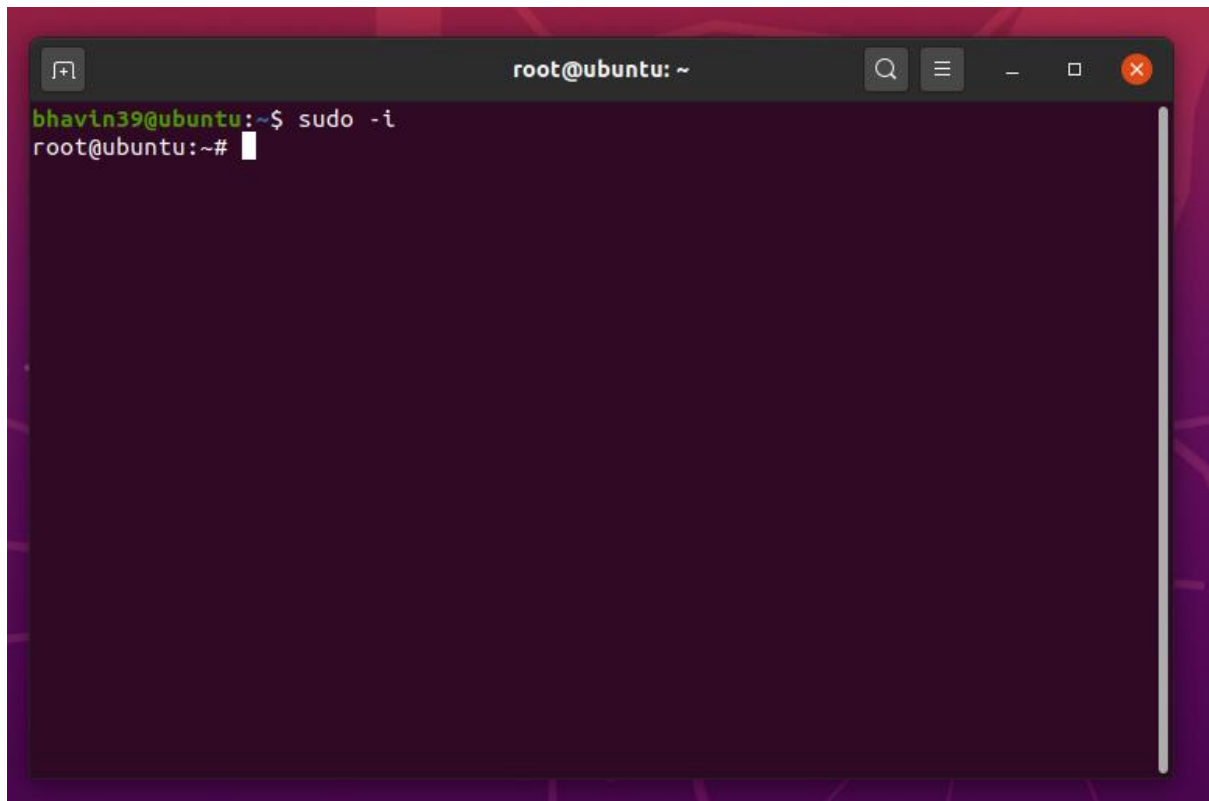
`sudo -v / -h / -l / -v / -K / -S / [-c class] / [-p prompt].`

`sudo -i` :- to change your shell to root shell, you can use this command.

`sudo -s` :- switch to super user or in root without changing your directory path.

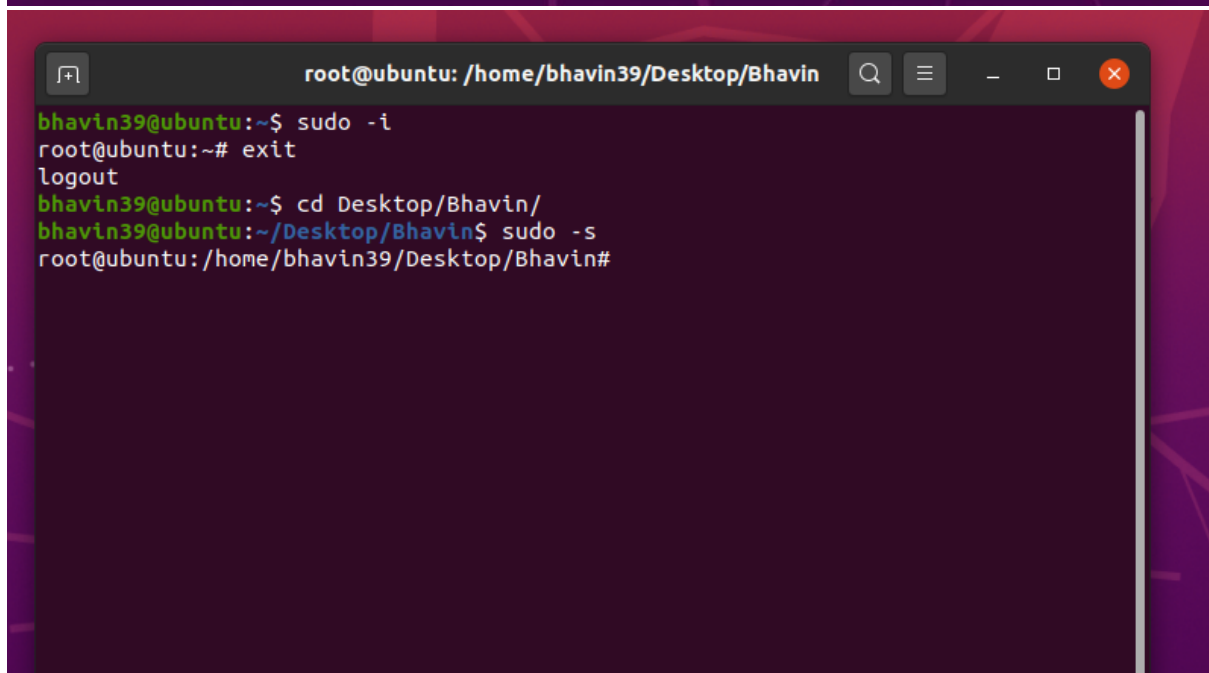
`sudo ls -lat &` to run any simple command like `ls - lat` with sudo to access system privilege.

`sudo --help` :- check other options of sudo command.



A terminal window titled "root@ubuntu: ~" with standard Ubuntu window controls. The prompt is "bhavin39@ubuntu:~\$". The user enters "sudo -i". The prompt changes to "root@ubuntu:~#".

```
root@ubuntu: ~
bhavin39@ubuntu:~$ sudo -i
root@ubuntu:~#
```



A terminal window titled "root@ubuntu: /home/bhavin39/Desktop/Bhavin" with standard Ubuntu window controls. The prompt is "bhavin39@ubuntu:~\$". The user enters "sudo -i". The prompt changes to "root@ubuntu:~#". The user enters "exit". The prompt changes to "logout". The user enters "cd Desktop/Bhavin/". The prompt changes to "bhavin39@ubuntu:~/Desktop/Bhavin\$". The user enters "sudo -s". The prompt changes to "root@ubuntu:/home/bhavin39/Desktop/Bhavin#".

```
root@ubuntu: /home/bhavin39/Desktop/Bhavin
bhavin39@ubuntu:~$ sudo -i
root@ubuntu:~# exit
logout
bhavin39@ubuntu:~$ cd Desktop/Bhavin/
bhavin39@ubuntu:~/Desktop/Bhavin$ sudo -s
root@ubuntu:/home/bhavin39/Desktop/Bhavin#
```

```
bhavin39@ubuntu: ~/Desktop/Bhavin
bhavin39@ubuntu:~$ sudo -i
root@ubuntu:~# exit
logout
bhavin39@ubuntu:~$ cd Desktop/Bhavin/
bhavin39@ubuntu:~/Desktop/Bhavin$ sudo -s
root@ubuntu:/home/bhavin39/Desktop/Bhavin# exit
exit
bhavin39@ubuntu:~/Desktop/Bhavin$ sudo ls -lrt
total 8
drwxrwxr-x 2 bhavin39 bhavin39 4096 Jun 15 03:47 Bash_programs
drwxrwxr-x 2 bhavin39 bhavin39 4096 Jun 15 03:58 C_programs
bhavin39@ubuntu:~/Desktop/Bhavin$
```

