Linux Tutorial

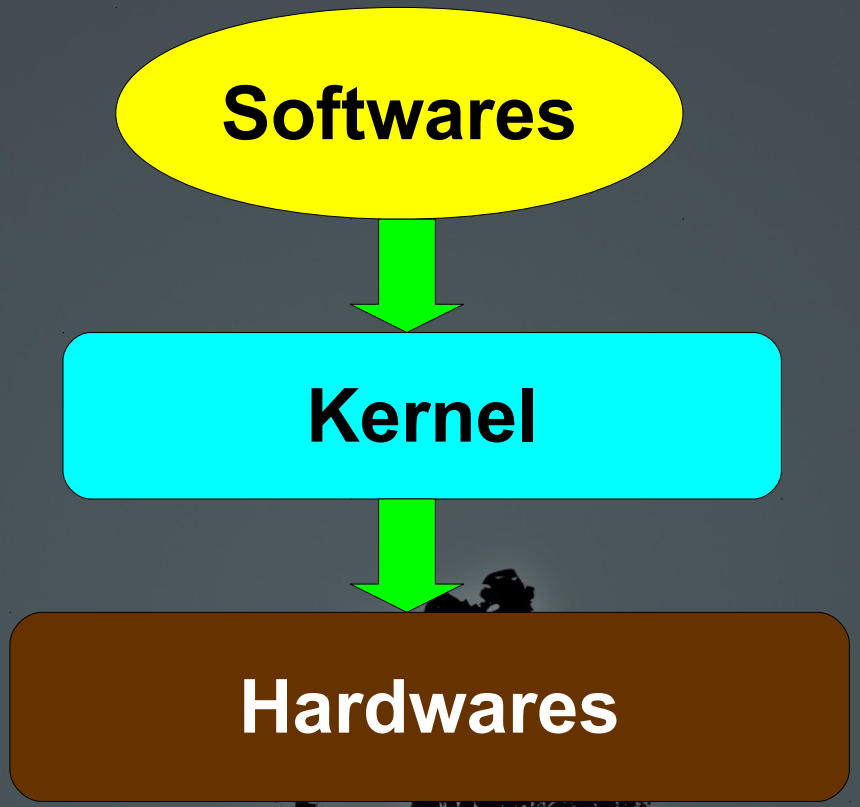
IUCAA summer school 2012

Dipanjan Mukherjee

What is Linux?

- Linux is a UNIX like operating system developed by Linus Torvalds in early 90s.
- The disinguishing feature is the **LINUX kernel**.

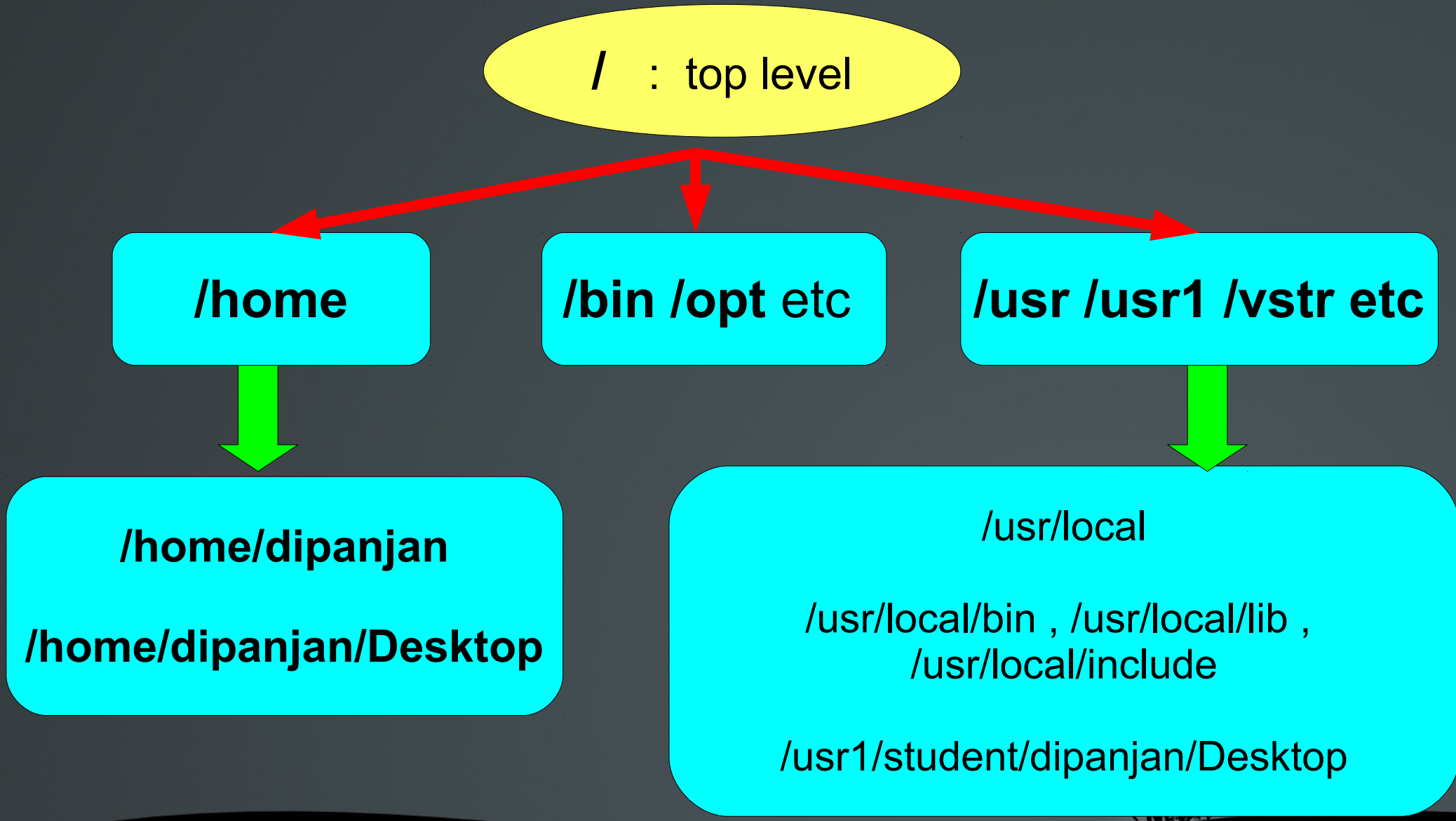
- There are several types of LINUX :
  Debian based distros like Ubuntu, Mint.
  Fedora, Red-hat types.
  Open Suse type etc



Some Jargons

- KDE and GNOME Desktop environments.
- GNU GRUB : Grand Unified Bootloader.
- EXT (Extended File system) as opposed to FAT 32 or NTFS. Currently used : ext4.
- SHELL : An user interface to the OS. e.g BASH, CSH, TCSH.

The Directory structure



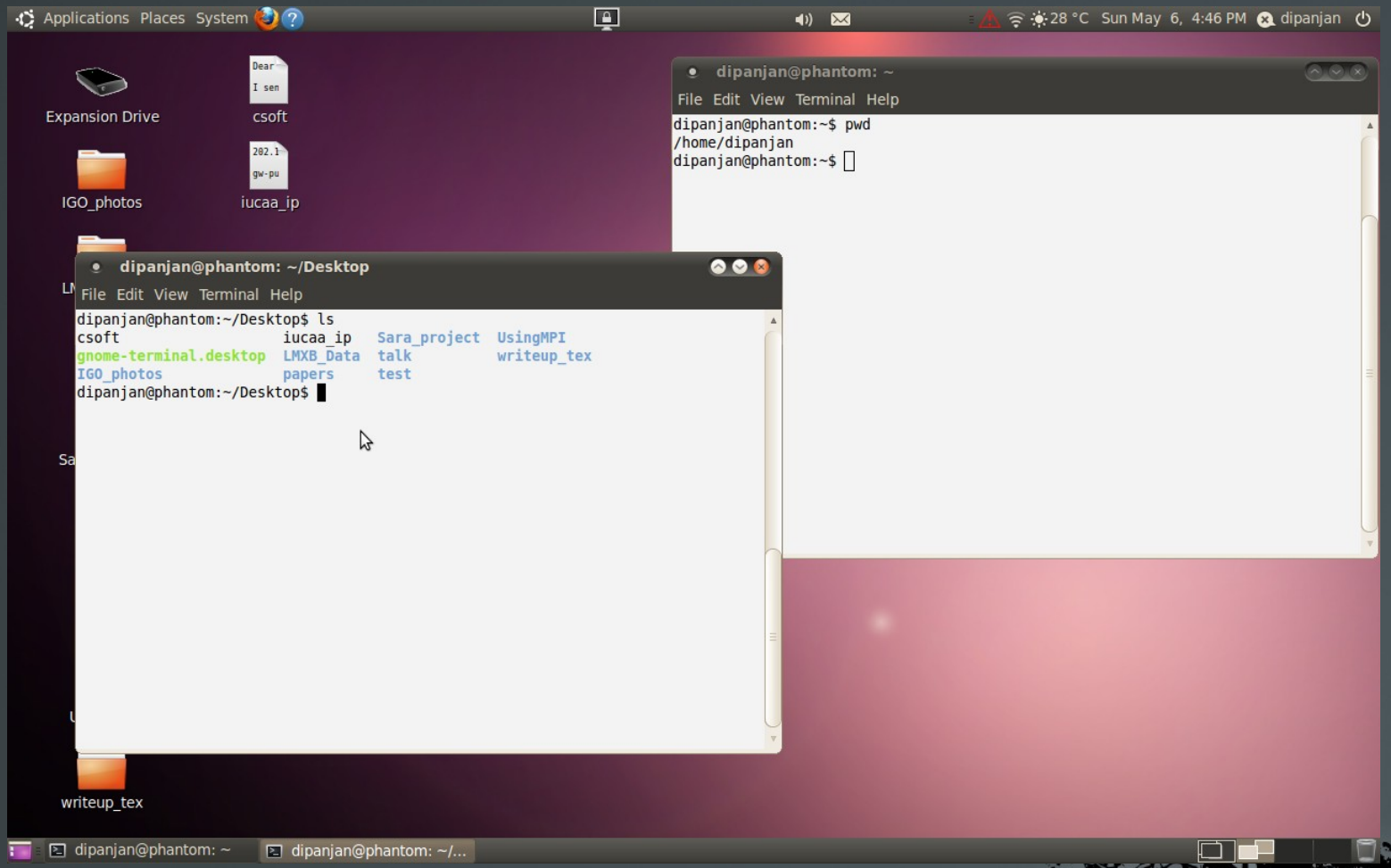
After Logging in

- **Nautilus** : graphical file manager, similar to windows. Folders in windows are called directories in LINUX world!
- **Workspace** : control which window is on your screen. Switch between workspaces by either mouse or :

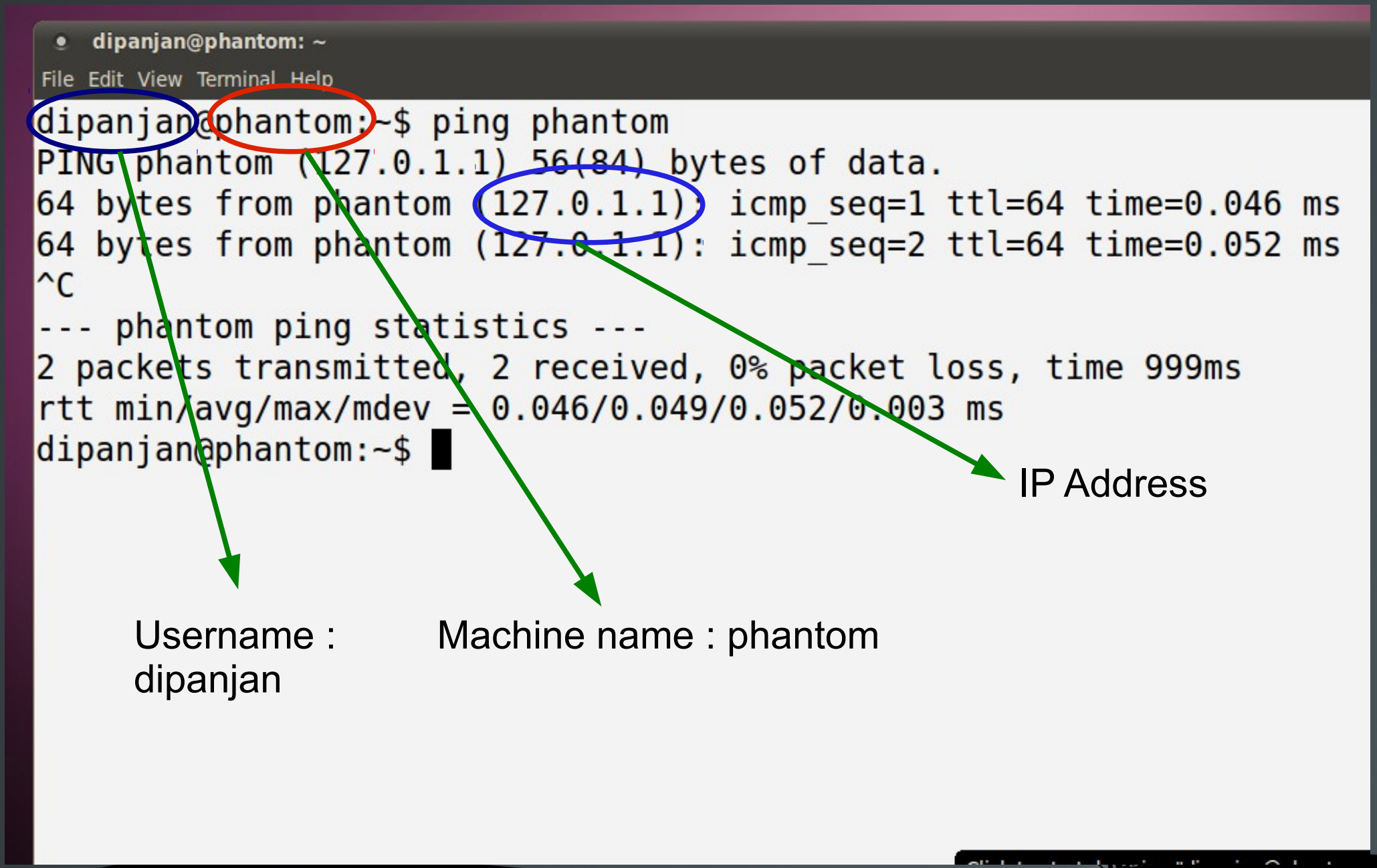
  ctrl+alt+left/right arrow key
- **Mozilla** web-browser.
- Switch between multiple objects in same workspace :

  alt+tab
- Show Desktop : ctrl+alt+d
- Recycle bin is called Trash

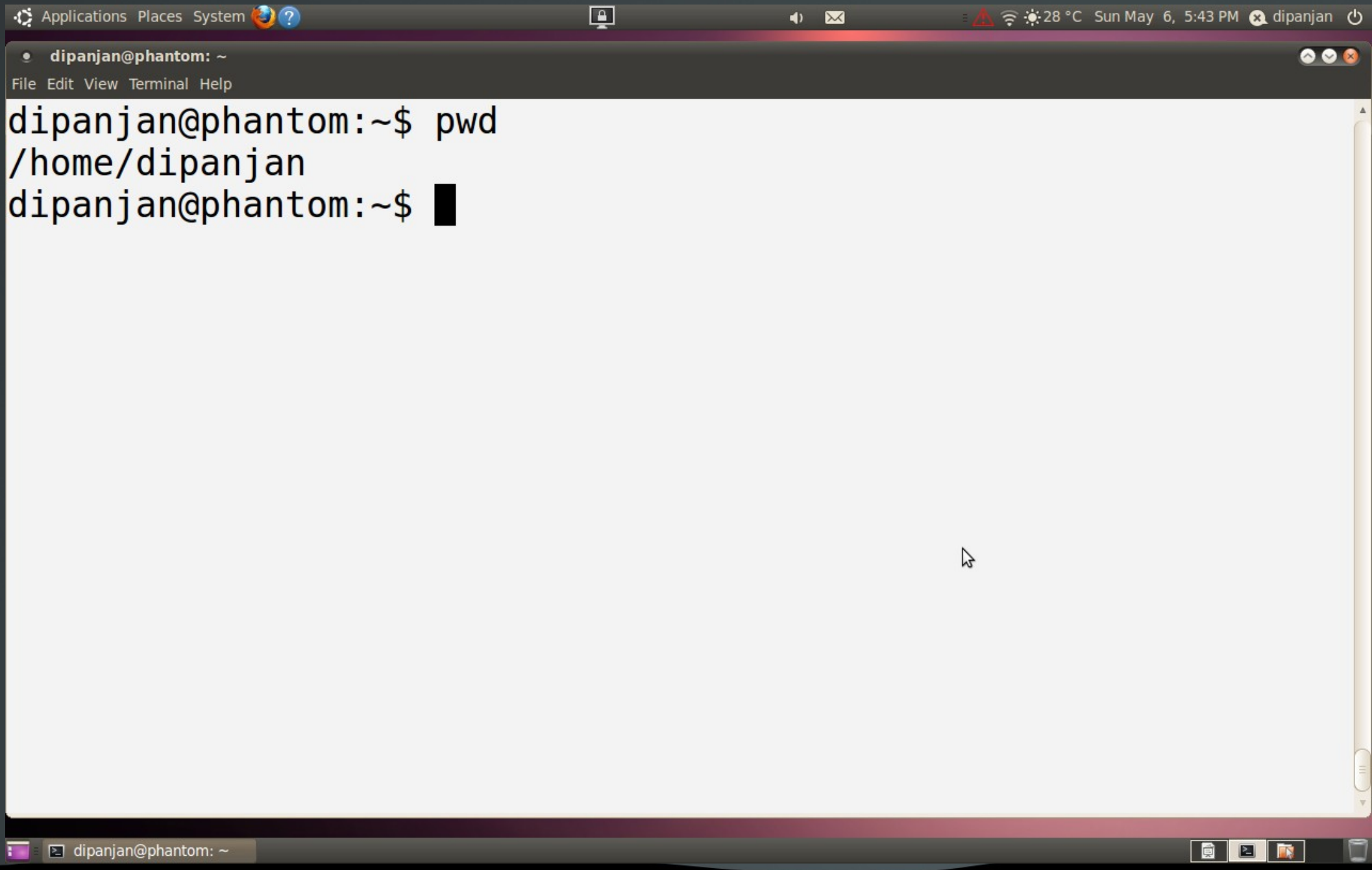
- **The terminal** : used to enter shell commands and interact with the OS.



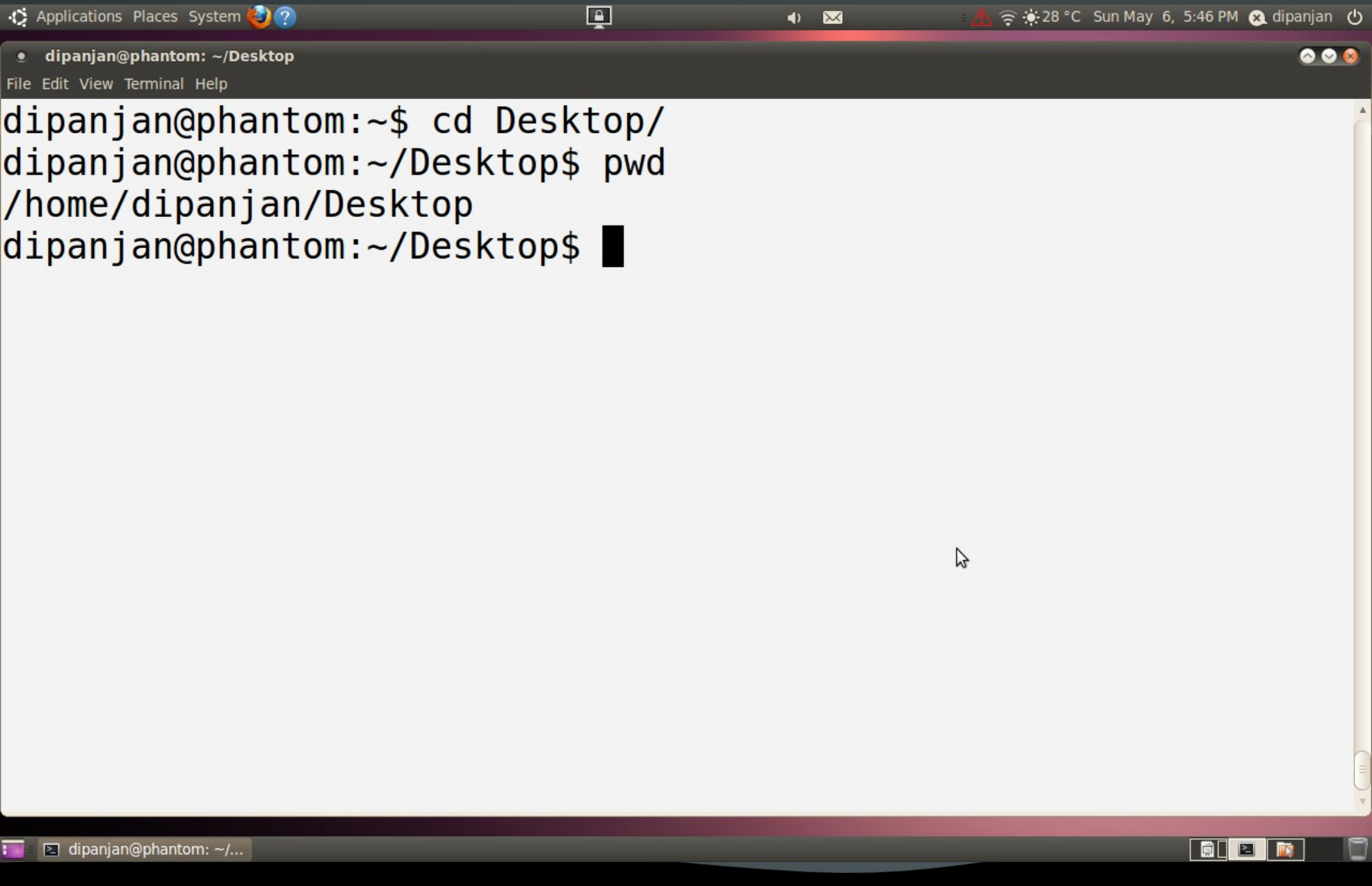
Log in id and machine name



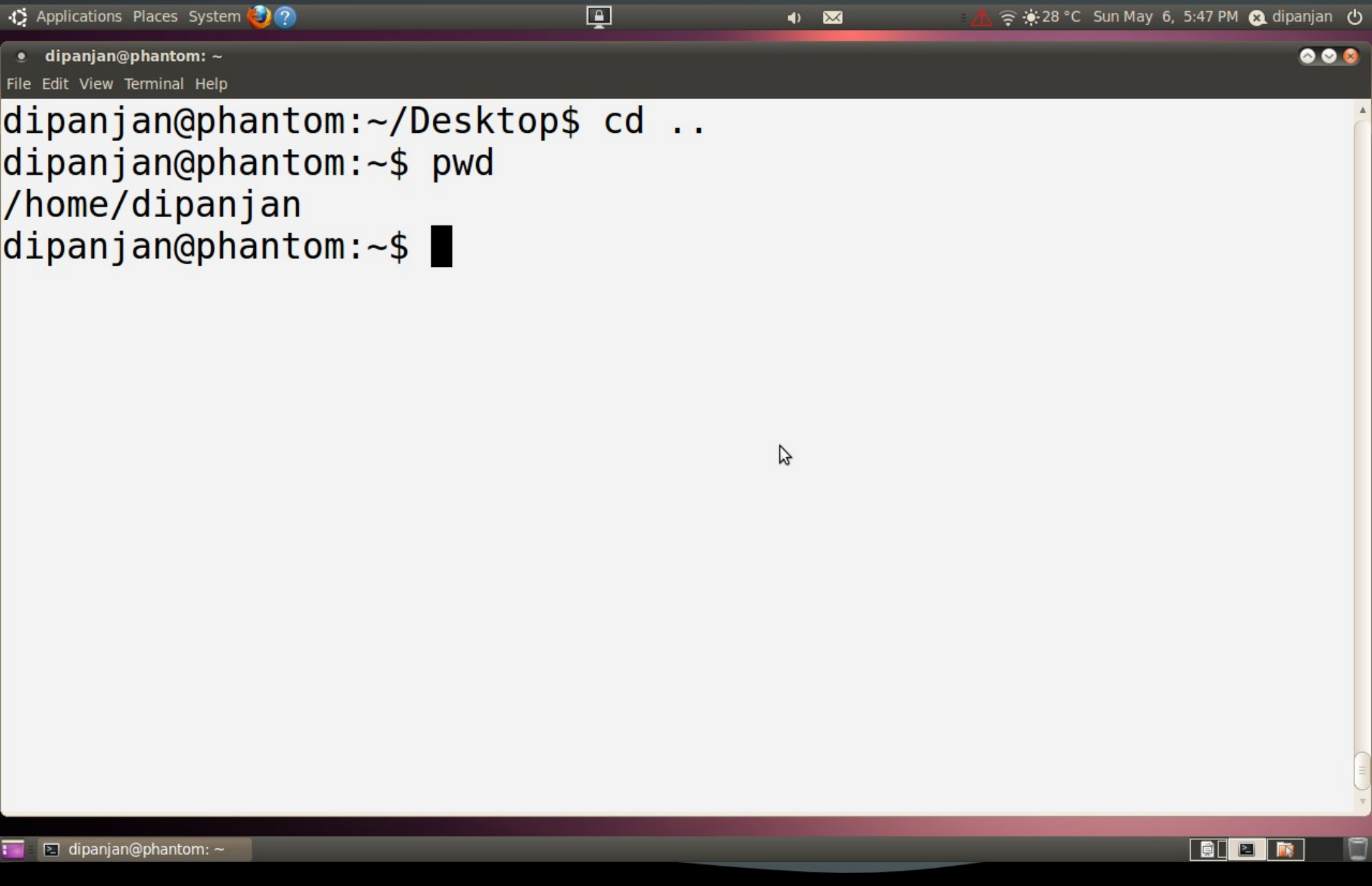
pwd : check current location

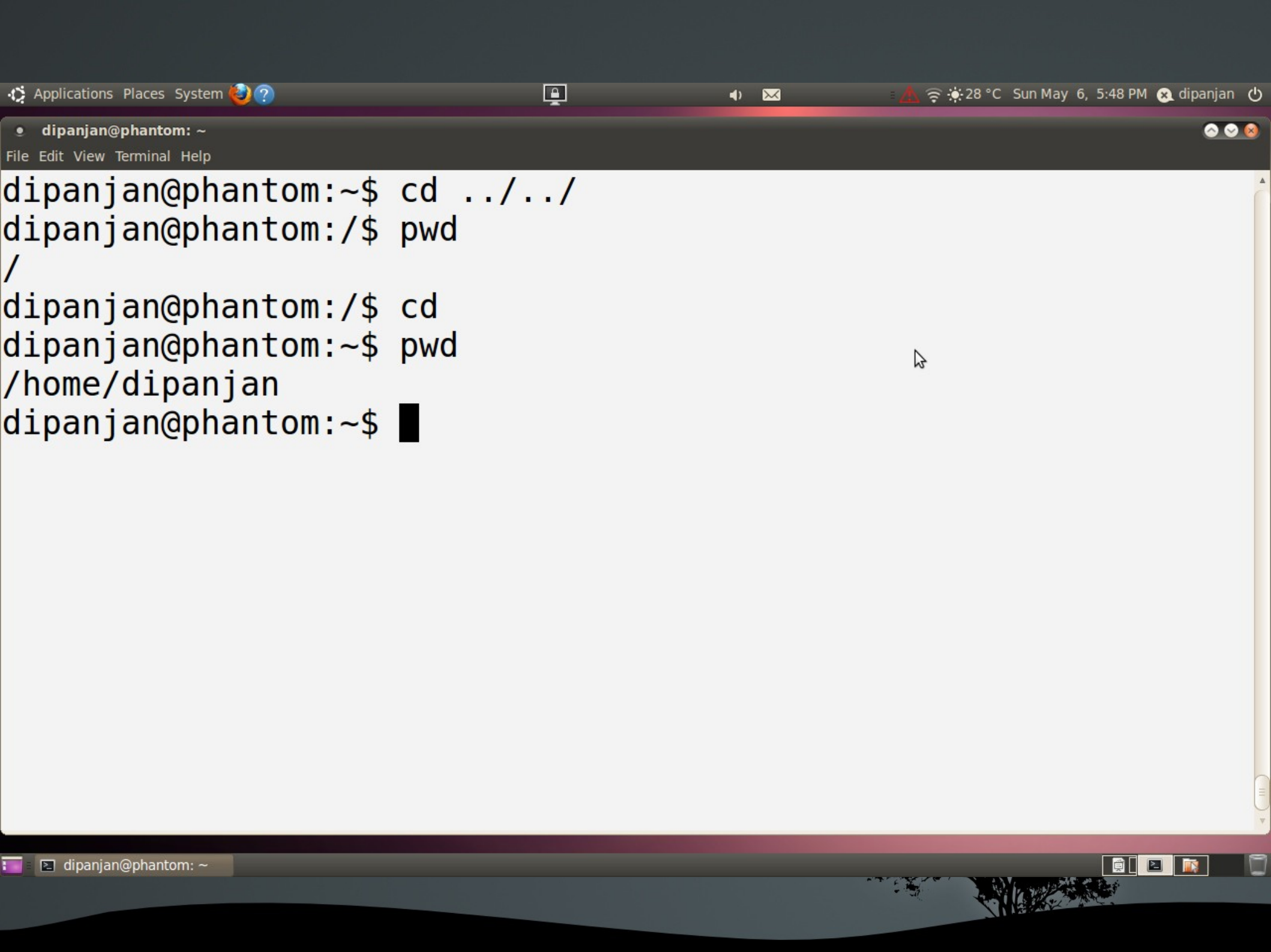


cd : change directory . Use tab to autofill commands

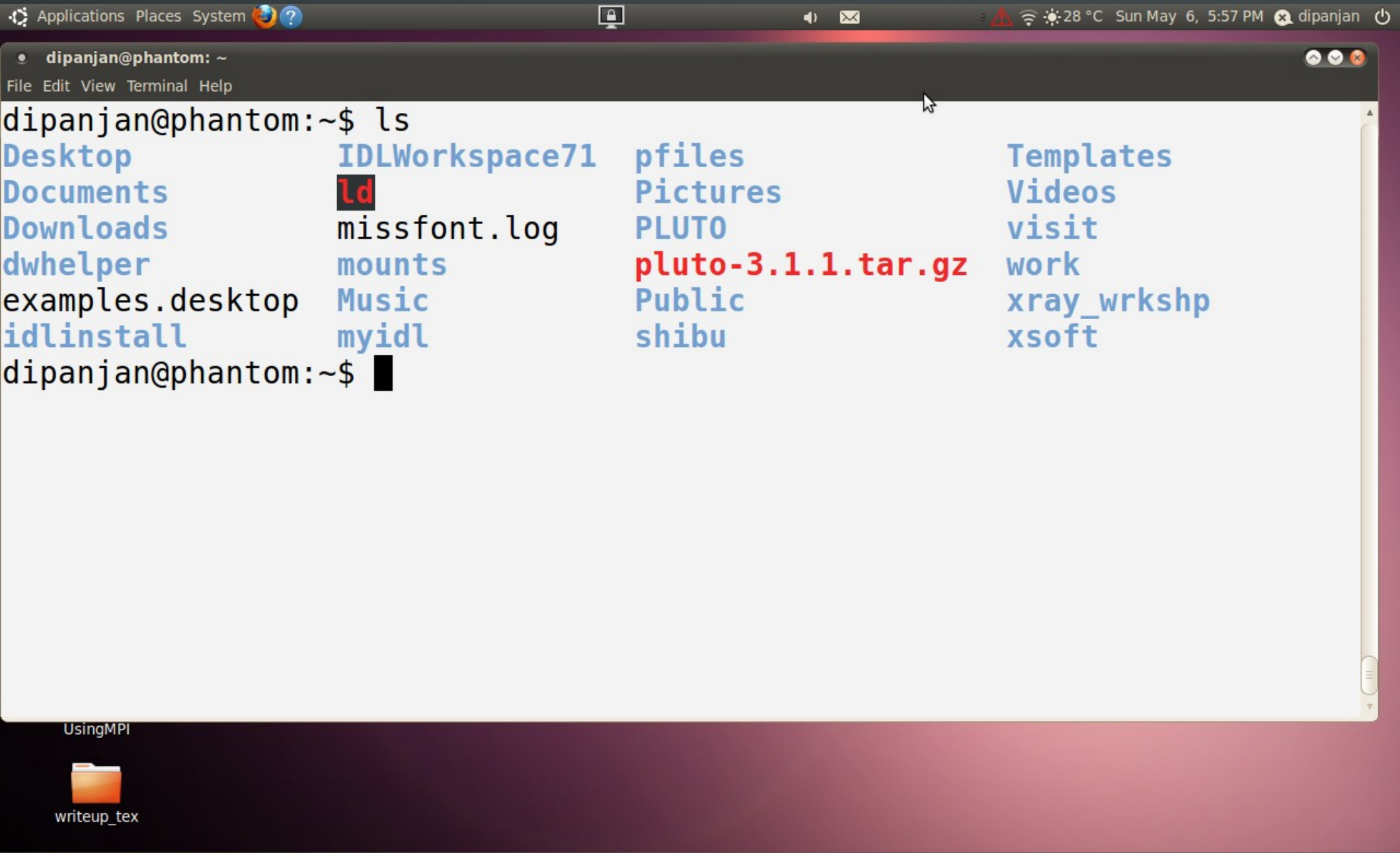


cd .. : go one level backwards




Applications Places System
28 °C Sun May 6, 5:48 PM dipanjan
dipanjan@phantom: ~
File Edit View Terminal Help
dipanjan@phantom:~$ cd ../../
dipanjan@phantom:/$ pwd
/
dipanjan@phantom:/$ cd
dipanjan@phantom:~$ pwd
/home/dipanjan
dipanjan@phantom:~$
dipanjan@phantom: ~

ls : list files.



ls -ltr .

Permission settings :

r – read

w – write

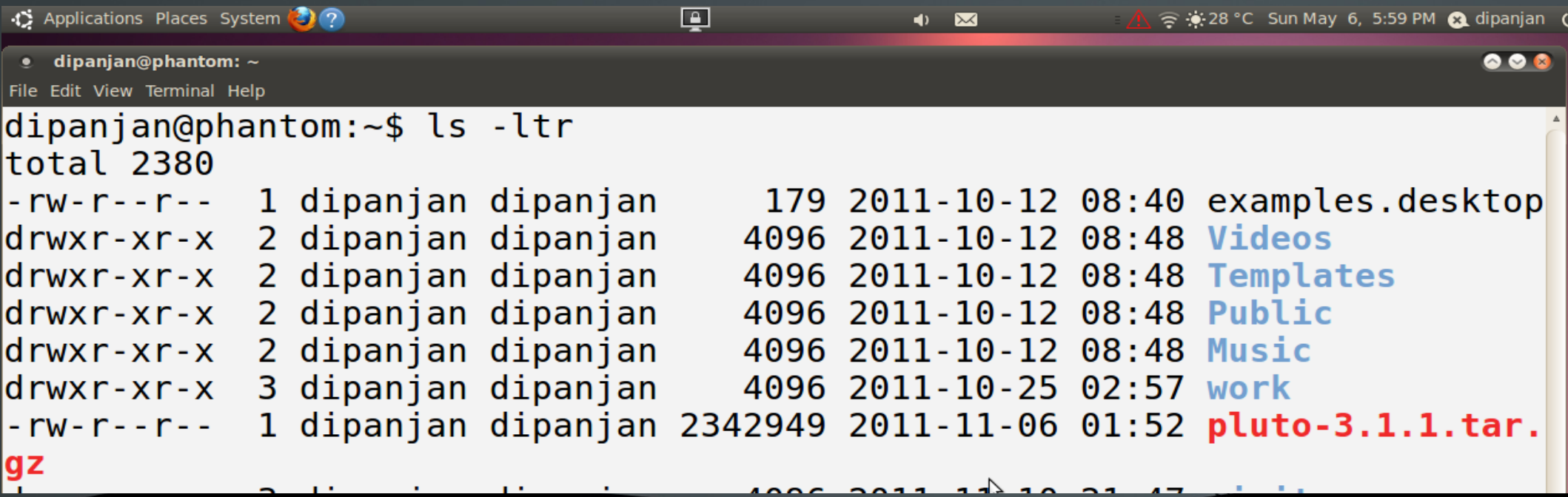
x – execute

e.g. xr – read and execute only. rw - read and write only. rwxr – read, write and execute.

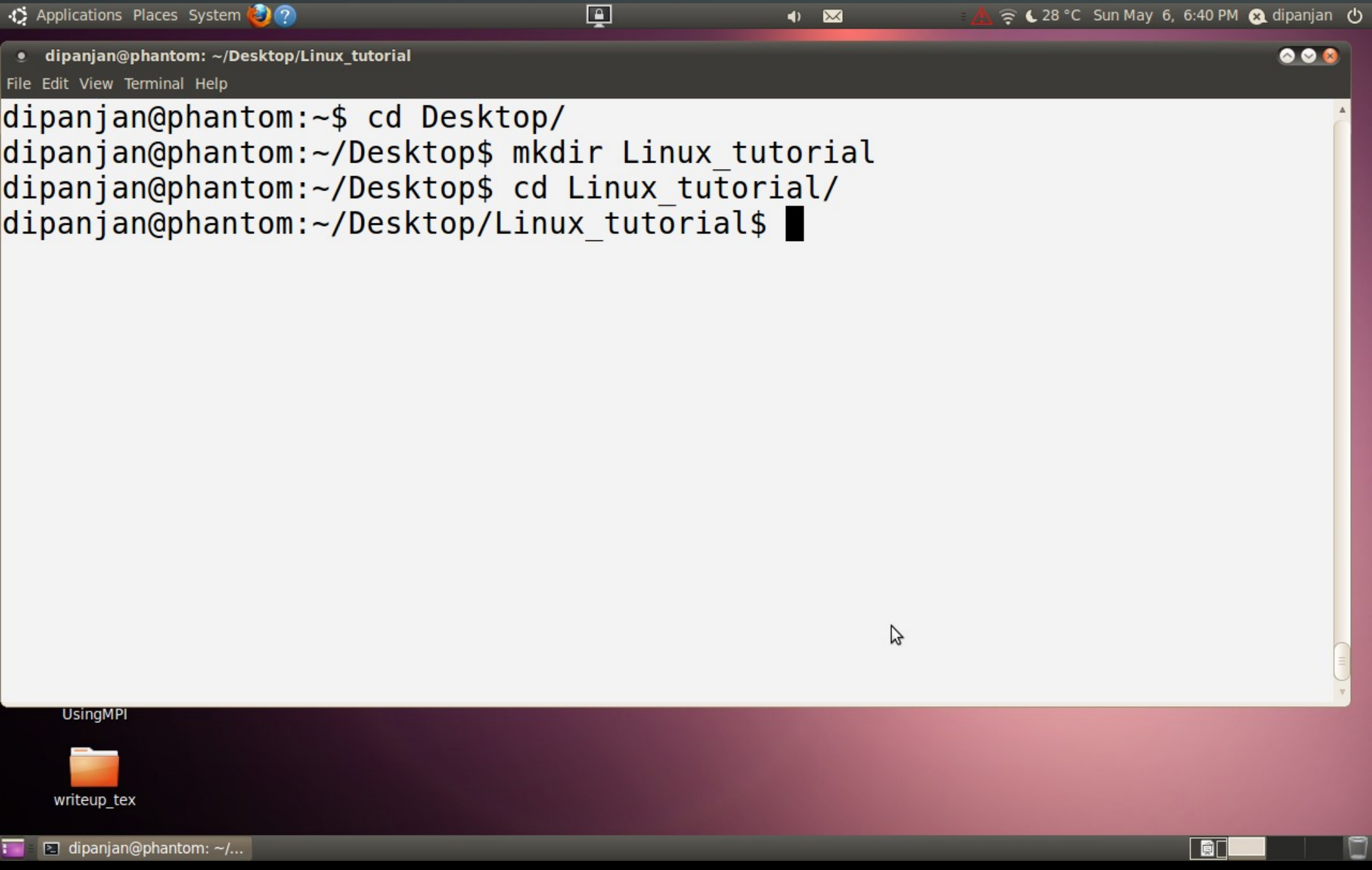
ls -al : lists all files, including hidden files : .bashrc

$clear : to clean screen.

$man ls : manual page for commands. Exit by pressing q



mkdir : make directory



Handling text files. Editors : gedit, vi, gvim, emacs.

Open a text file with gedit : **$gedit testfile.txt**

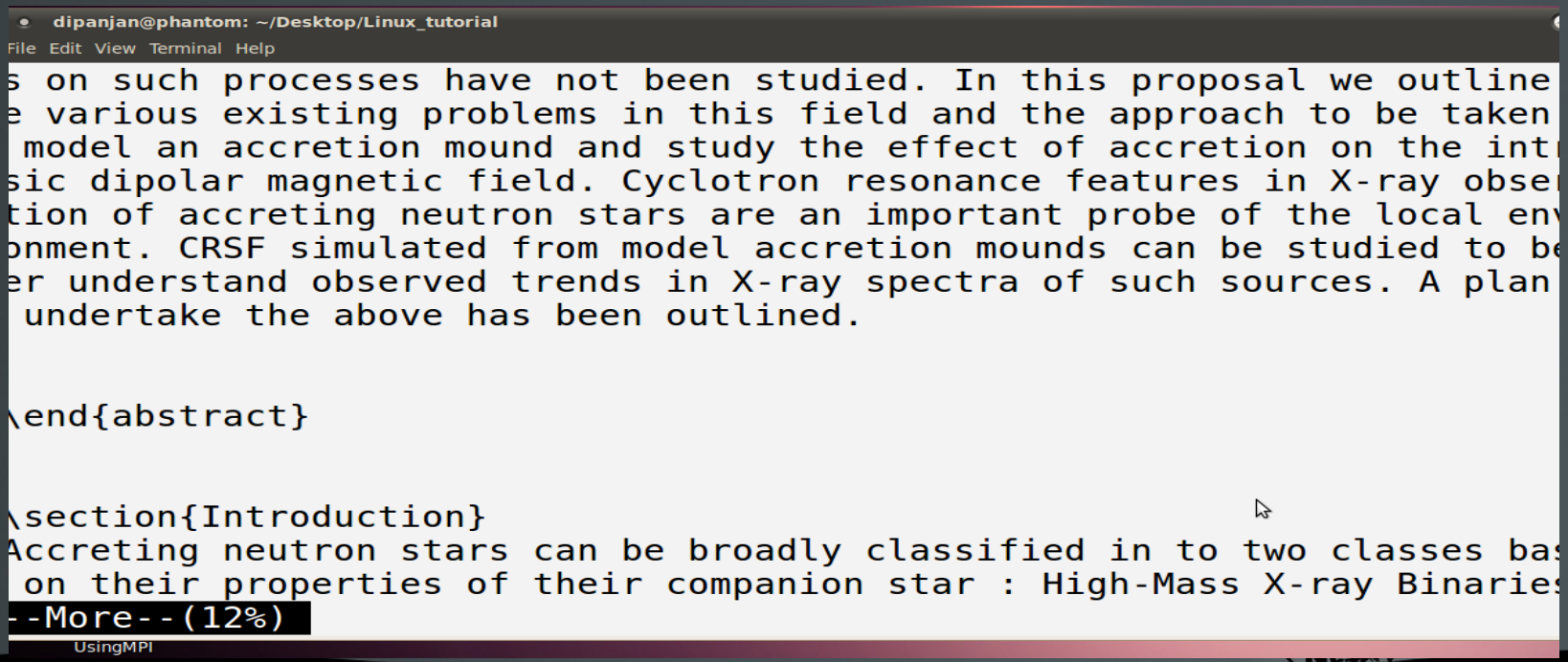
Write some text. Close the text file. You can also close by : **ctrl+c** from the terminal. ctrl+c terminates a process opened through the terminal.

To check contents of a file :

$more testfile.txt

$cat testfile.txt

$less testfile.txt



Copying files : cp filename destination

```
dipanjan@phantom: ~/Desktop/Linux_tutorial
File  Edit  View  Terminal  Help
dipanjan@phantom:~/Desktop/Linux_tutorial$ mkdir dir_test
dipanjan@phantom:~/Desktop/Linux_tutorial$ cp testfile.txt copy_test.txt
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
copy_test.txt  dir_test  testfile.txt
dipanjan@phantom:~/Desktop/Linux_tutorial$ cp testfile.txt dir_test/
```

Copying directories : cp -r directory destination

```
dipanjan@phantom:~/Desktop/Linux_tutorial$ cp -r dir_test another_dir
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
another dir  copy test.txt  dir test  testfile.txt
```

Renaming files/directories : mv filename newname

```
dipanjan@phantom:~/Desktop/Linux_tutorial$ mv copy_test.txt renamed.txt
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
another_dir  dir_test  renamed.txt  testfile.txt
```

Moving files/directories : mv directory or file destination

```
dipanjan@phantom:~/Desktop/Linux_tutorial$ mv renamed.txt another_dir/
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
another_dir  dir_test  testfile.txt
dipanjan@phantom:~/Desktop/Linux_tutorial$
```

Removing files : rm filename

```
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
another_dir  dir_test  testfile.txt
dipanjan@phantom:~/Desktop/Linux_tutorial$ rm testfile.txt
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
another_dir  dir_test
```

Removing directories : rm -r directories

```
dipanjan@phantom:~/Desktop/Linux_tutorial$ rm -r another_dir/
dipanjan@phantom:~/Desktop/Linux_tutorial$ ls
dir_test
```

Removing Everything : rm -rf * -> one of the most

Dangerous commands. Take care not to delete important files.

Copying/moving all files : cp * destination

Copying/moving same type of files : cp *.txt* destination

Editing with vi

- Open file with vi
  $vi testfile.txt
- To enter text press i
- Write text.
- Press escape to exit write mode.
- To save file enter :w (type the colon then w)
- To undo what you did last enter :u
- To exit :q
- To save and exit :wq
- To exit without saving :q!

Changing file permissions

Chmod 755 filename

7 -> full permission : read + write + execute
5 -> read+write
4 -> read only

1st place : owner's permission (user)
2nd place : group
3rd place : others

Chmod 777 -> full permission to all
Chmod 755 -> full permission to user read/write to others
Chmod 744 -> full permission to user, read only to others

Miscellaneous commands

- $ top : lists current processes
- $ kill process id
- Alternatively kill process from System-> Administration ->System Monitor

- Check file size $ du -h testfile.txt
  du -sh gives total size of current directory without listing contents.
- Find file type with $file name
- To search for a file $ find . *.txt searches all text file in current directory. To search in a known directory give $ find path *.txt

- Open pdf/ps with $ gv file.pdf or $ evince file.pdf or $ acroread file.pdf
- Open jpg or png image files with $ display file.jpg or $ eog file.jpg

- To make a tarball (similar to zip) :
  $ tar cvfz name.tar.gz file1 file2 file3 dir1 dir2 ....
- To untar : $ tar xvfz name.tar.gz

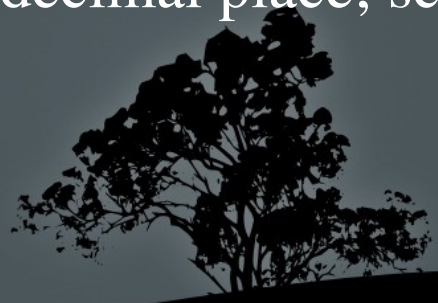
Miscellaneous commands

- Redirect output to a text file :
  $ du -ah > size.txt or $ du -ah >> size.txt (appends)
- Check end of text file $ tail filename or $ tail -f filename

- Count number of words/lines ; $ wc -w file or $ wc -l file
- Pipe output of one as input of another
  $ du -h testfile.txt | wc -w

- $ grep text * searches the pattern text inside all files in current directory.
- $ diff file1 file2 : lists differences in two files line by line. $gvimdiff also works.

- $ date : prints system time and date
- Calculator on terminal!! : $ bc set scale variable for decimal place; scale=3 gives results till 3rd decimal place.



Shell

- A program that talks to the Kernel/OS.
- Types : Bourne shell (sh), Korn shell (ksh), C shell (csh), tcsh shell and Bourne Again shell (bash).
- csh and tcsh are synctactically similar. bash and sh are again similar to each other.

- Find your shell type using :

  $ echo $SHELL

```
dipanjan@phantom:~$ echo $SHELL
/bin/bash
dipanjan@phantom:~$
```

- For tcsh or csh shell, a startup script file is run called cshrc. For bash it is bashrc. For some others it is .profile. Open cshrc using :

  $ gvim ~/.cshrc . If you edit .cshrc, run .cshrc for that session once :

  $ source ~/.cshrc

Shell

- **Setting variables** in C shell :

  $ set var1=2.0 For bash shells it is : $ var1=2.0 without using set.

```
File  Edit  View  Terminal  Help
phantom:~> set string=/home/dipanjan/Desktop
phantom:~> cd $string
phantom:~/Desktop>
```

- **Setting environment variables** in C shell :

```
phantom:~/Desktop/Linux_tutorial> pwd
/home/dipanjan/Desktop/Linux_tutorial
phantom:~/Desktop/Linux_tutorial> setenv workdir /home/dipanjan/Desktop/Linux_tutorial
phantom:~/Desktop/Linux_tutorial> echo $workdir
/home/dipanjan/Desktop/Linux_tutorial
phantom:~/Desktop/Linux_tutorial>
```

- **Export variables** in bash shell :

```
phantom:~/Desktop/Linux_tutorial> bash
dipanjan@phantom:~/Desktop/Linux_tutorial$ export workdir="/home/dipanjan/Desktop/Linux_tutorial"
dipanjan@phantom:~/Desktop/Linux_tutorial$ echo $workdir
/home/dipanjan/Desktop/Linux_tutorial
```

Shell

- **Aliasing commands** in C shell

```
phantom:~> cd
phantom:~> alias work 'cd $workdir'
phantom:~> work
phantom:~/Desktop/Linux_tutorial> pwd
/home/dipanjan/Desktop/Linux_tutorial
phantom:~/Desktop/Linux_tutorial>
```

- **Aliasing** in Bash

```
dipanjan@phantom:~$ alias work='cd $workdir'
dipanjan@phantom:~$ work
```

- **Arithmetic** in C shell :

```
phantom:~/Desktop/Linux_tutorial> set var1=2
phantom:~/Desktop/Linux_tutorial> set var2=3
phantom:~/Desktop/Linux_tutorial> @ a = $var1 * $var2
phantom:~/Desktop/Linux_tutorial> @ a = $var1 * $var2; echo $a
6
phantom:~/Desktop/Linux_tutorial> @ a = $var1 + $var2 ; echo $a
5
```

A simple SHELL script

```
dipanjan@phantom:~/Desktop/Linux_tutorial$ more test.sh
#/bin/bash
count=0
theta=0
rm -f outfile.dat
touch outfile.dat

while (( $count <= 200)); do

# ----Fermat spiral-----
#theta=$(echo "scale=10;$count/200*8*6.28318" | bc)
#r=$(echo "sale=10;0.1*$theta*$theta" | bc -l)

#-----Hyperbolic spiral-----
theta=$(echo "scale=15;1.0+$count/200*8.0*6.28318" | bc)
r=$(echo "sale=15;2.0/$theta" | bc -l)
echo $theta $r >>outfile.dat
count=$(( $count + 1 ))

done
dipanjan@phantom:~/Desktop/Linux_tutorial$ chmod 755 test.sh
dipanjan@phantom:~/Desktop/Linux_tutorial$ ./test.sh
```

Plot using GNUPLOT :
$ gnuplot

```
gnuplot> set polar

        dummy variable is t for curves
gnuplot> plot "outfile.dat" with lines
```


Fim
Thats all Folks!