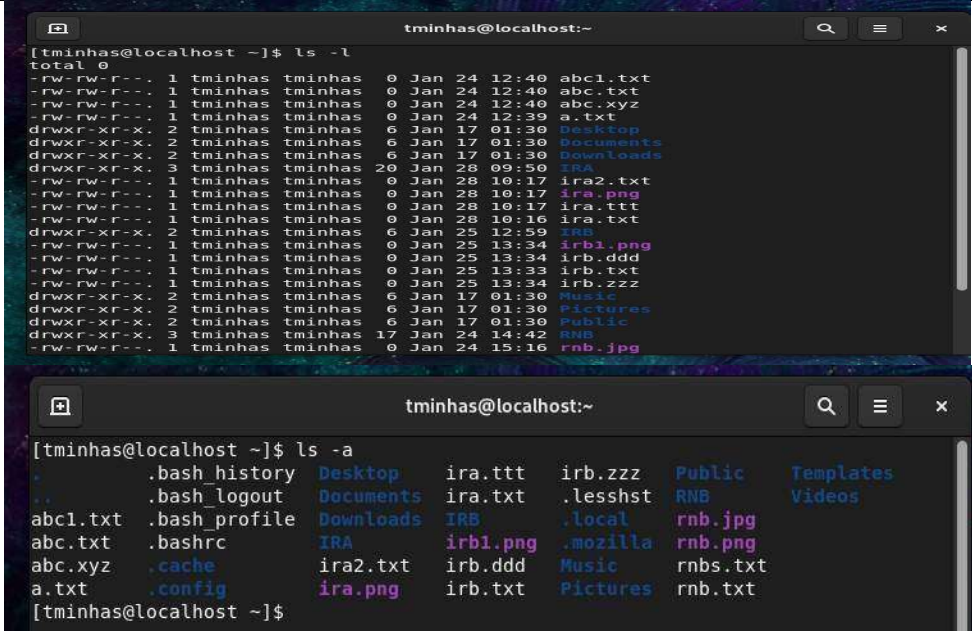# Exercise#3

1. Using *man* describe the following and explicitly describe its uses with examples.

Command	Explanation with 2 important options and their usage
ls	-a and -l, -a option is used for all files and folders, starting with dot(.). Files names beginning with . are hidden files. -l option is used for longlisting which means we can permissions, size owner, group, and time stamp of file or directory. ls -a ls -l We can combine both options as ls -la

The above screenshot ls -l shows details about files but no hidden files.
However, the screenshot for ls -a shows the hidden files for example .bash_history.

| touch | Update the file time-stamps
-a used to change only the access time
-m used change only the modification time
touch -a
touch -m
We can combine both options as
touch -am |

N01687444@osboxes:~ — man touch

```
TOUCH(1)                        User Commands                       TOUCH(1)

NAME
       touch - change file timestamps

SYNOPSIS
       touch [OPTION]... FILE...

DESCRIPTION
       Update  the  access  and modification times of each FILE to the current
       time.

       A FILE argument that does not exist is created empty, unless -c  or  -h
       is supplied.

       A  FILE  argument  string of - is handled specially and causes touch to
       change the times of the file associated with standard output.

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

       -a     change only the access time

Manual page touch(1) line 1/82 23% (press h for help or q to quit)
```

N01687444@osboxes:~

```
[N01687444@osboxes ~]$ man touch
[N01687444@osboxes ~]$ touch -a file.txt
[N01687444@osboxes ~]$ touch -m file.txt
[N01687444@osboxes ~]$ touch -am file.txt
[N01687444@osboxes ~]$ 
```

sort	Sort lines of the text files -n is used for numeric sort i.e compare according to string numerical value -r is used to reverse the result of comparison . sort -n file.txt sort -r file.txt sort -nr file.txt

```
SORT(1)                        User Commands                        SORT(1)

NAME
       sort - sort lines of text files

SYNOPSIS
       sort [OPTION]... [FILE]...
       sort [OPTION]... --files0-from=F

DESCRIPTION
       Write sorted concatenation of all FILE(s) to standard output.

       With no FILE, or when FILE is -, read standard input.

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

       -b, --ignore-leading-blanks
              ignore leading blanks

       -d, --dictionary-order
              consider only blanks and alphanumeric characters

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

```
[N01687444@osboxes ~]$ nano file.txt
[N01687444@osboxes ~]$ sort -n file.txt
i love linux and support the ope-source software
krishna
linux
os
qwe2
rao
sanirbhai
sdvjbi
1abc
2isdbv
[N01687444@osboxes ~]$ sort -r file.txt
sdvjbi
sanirbhai
rao
qwe2
os
linux
krishna
i love linux and support the ope-source software
2isdbv
1abc
[N01687444@osboxes ~]$
```

```
[N01687444@osboxes ~]$ sort -nr file.txt
2isdbv
1abc
sdvjbi
sanirbhai
rao
qwe2
os
linux
krishna
i love linux and support the ope-source software
[N01687444@osboxes ~]$
```

mv	mv move(rename) files -f, --force is used for not prompting before overwriting. -i—interactive is used for prompting before overwrite. mv -f mv-i

DESCRIPTION
       Rename SOURCE to DEST, or move SOURCE(s) to DIRECTORY.

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

       --backup[=CONTROL]
              make a backup of each existing destination file

       -b   like --backup but does not accept an argument

       -f, --force
              do not prompt before overwriting

       -i, --interactive
              prompt before overwrite

       -n, --no-clobber
              do not overwrite an existing file

       If you specify more than one of -i, -f, -n, only the final one takes
       effect.
 Manual page mv(1) line 10 (press h for help or q to quit)

```
[N01687444@osboxes ~]$ touch fileabc.txt
[N01687444@osboxes ~]$ mv fileabc.txt linux
[N01687444@osboxes ~]$ tree linux
linux
├── Assignment
│   ├── excersise
│   ├── Krishna
│   └── Rao
└── fileabc.txt

4 directories, 1 file
[N01687444@osboxes ~]$ mv -f  fileabc.txt linux
mv: cannot stat 'fileabc.txt': No such file or directory
[N01687444@osboxes ~]$ mv -f linux/fileabc.txt linux/Assignment
[N01687444@osboxes ~]$ tree linux
linux
└── Assignment
    ├── excersise
    ├── fileabc.txt
    ├── Krishna
    └── Rao

4 directories, 1 file
[N01687444@osboxes ~]$ mv -i Assignment/fileabc.txt linux/Assignment/excersise
```

cp	Copy files and directories
	-attributes-only is used for copying attributes and not file data
	-v explains what is being done

	Cp -a file1.txt file2.txt Cp -v file.txt file2.txt

```
CP(1)                          User Commands                          CP(1)

NAME
       cp - copy files and directories

SYNOPSIS
       cp [OPTION]... [-T] SOURCE DEST
       cp [OPTION]... SOURCE... DIRECTORY
       cp [OPTION]... -t DIRECTORY SOURCE...

DESCRIPTION
       Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.

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

       -a, --archive
              same as -dR --preserve=all

       --attributes-only
              don't copy the file data, just the attributes

       --backup[=CONTROL]
 Manual page cp(1) line 1 (press h for help or q to quit)
```

```
[N01687444@osboxes ~]$ man cp
[N01687444@osboxes ~]$ cp -a a.txt b.txt
[N01687444@osboxes ~]$ cat b.txt
hey my name is krishna rao

[N01687444@osboxes ~]$ cp -v a.txt b.txt
'a.txt' -> 'b.txt'
[N01687444@osboxes ~]$
```

less	Less is opposite of more. It displays all the files of a directory. J Like J it continues to scroll till end of line. R repaints the screen. Less I Less R

NAME
        less - opposite of more

SYNOPSIS
        less -?
        less --help
        less -V
        less --version
        less [-[+]aABcCdeEfFgGiIJKLmMnNqQrRsSuUVwWX~]
             [-b space] [-h lines] [-j line] [-k keyfile]
             [-{oO} logfile] [-p pattern] [-P prompt] [-t tag]
             [-T tagsfile] [-x tab,...] [-y lines] [-[z] lines]
             [-# shift] [+[+]cmd] [--] [filename]...
        (See the OPTIONS section for alternate option syntax with long option
        names.)

DESCRIPTION
        Less is a program similar to more(1), but which allows backward move-
        ment in the file as well as forward movement. Also, less does not have
        to read the entire input file before starting, so with large input
        files it starts up faster than text editors like vi(1). Less uses
Manual page less(1) line 1 (press h for help or q to quit)

[N01687444@osboxes ~]$ man less
[N01687444@osboxes ~]$ less j CPAN133Assignments
j: No such file or directory
Press RETURN to continue
[N01687444@osboxes ~]$ less J CPAN133Assignments
J: No such file or directory
Press RETURN to continue
[N01687444@osboxes ~]$ less R CPAN133Assignments
R: No such file or directory
Press RETURN to continue
[N01687444@osboxes ~]$

```
 ESC[Atotal 4
drwxrwxr-x.  3 N01687444 N01687444   25 Oct  1 10:44 ./
drwx------. 25 N01687444 N01687444 4096 Oct  1 11:20 ../
drwxrwxr-x.  4 N01687444 N01687444  106 Sep 30 19:38 Assignment1/
CPAN133Assignments (END)
```

more   File persual filter for CRT viewing.
            -f count logical lines rather than screen lines.
            -s sqeee multiple blank line into one.   f

File Machine View Input Devices Help

Activities   Terminal   Oct 1 11:44

N01687444@osboxes:~

```
abc2           cpfd1              file008    hisfile       permfile1   test
Assignment1    Desktop            file009    krishna.txt   permfile2   Videos
a.txt          dir05              file1.txt  linux         Pictures    X
[N01687444@osboxes ~]$ sudo chown permfile osboxes
chown: invalid user: 'permfile'
[N01687444@osboxes ~]$ touch pemfile.txt
[N01687444@osboxes ~]$ sudo chown permfile.txt osboxes
chown: invalid user: 'permfile.txt'
[N01687444@osboxes ~]$ sudo chown permfile1 osboxes
chown: invalid user: 'permfile1'
[N01687444@osboxes ~]$ sudo chown permfile1 osboxes.org
chown: invalid user: 'permfile1'
[N01687444@osboxes ~]$ chmod 1644 permfile
[N01687444@osboxes ~]$ ls -l permfile
-rw-r--r-T. 1 N01687444 N01687444 0 Oct  1 11:39 permfile
[N01687444@osboxes ~]$ sudo chown  osboxes.org permfile.txt
chown: invalid user: 'osboxes.org'
[N01687444@osboxes ~]$ sudo chown osboxes permfile.txt
chown: cannot access 'permfile.txt': No such file or directory
[N01687444@osboxes ~]$ chmod 1644 permfile
[N01687444@osboxes ~]$ ls -l permfile
-rw-r--r-T. 1 N01687444 N01687444 0 Oct  1 11:39 permfile
[N01687444@osboxes ~]$ rm permfile
[N01687444@osboxes ~]$
```

N01687444@osboxes:~ — man more   Q   ≡   ✕

```
MORE(1)                          User Commands                          MORE(1)

NAME
       more - file perusal filter for crt viewing

SYNOPSIS
       more [options] file ...

DESCRIPTION
       more is a filter for paging through text one screenful at a time. This
       version is especially primitive. Users should realize that less(1)
       provides more(1) emulation plus extensive enhancements.

OPTIONS
       Options are also taken from the environment variable MORE (make sure to
       precede them with a dash (-)) but command-line options will override
       those.

       -d, --silent
           Prompt with "[Press space to continue, 'q' to quit.]", and display
           "[Press 'h' for instructions.]" instead of ringing the bell when an
           illegal key is pressed.

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

File Machine View Input Devices Help

Activities   ⌨ Terminal   Oct 1 11:30

N01687444@osboxes:~   Q   ≡   ✕

```
[N01687444@osboxes ~]$ man more
[N01687444@osboxes ~]$ more -f CPAN133Assignments

*** CPAN133Assignments: directory ***

[N01687444@osboxes ~]$ more -s
more: bad usage
Try 'more --help' for more information.
[N01687444@osboxes ~]$ more -s CPAN133Assignments

*** CPAN133Assignments: directory ***

[N01687444@osboxes ~]$ 
```

	More -f More -s
head	

tail	

2. Create folders cpfd/cpfd1/cpdf2 and then create a file <yourfirstname> inside the cpfd2 (without changing current working directory). Verify using listing command.
3. Copy this file to your home folder (using relative path). Verify using listing command.
4. Move file from cpfd2 to cpfd using absolute path. Verify using listing command.
5. What is SUID and how does it impact the usage of passwd?
6. Describe the sticky bit and its usage.
7. You have a classuser in virtualbox. Change the owner of file permfile to classuer.
8. Use the command *chmod 1644 permfile*
9. Show and explain the new permission of permfile.
10. Try to delete the permfile. What happened and why!