LINUX PROGRAMING

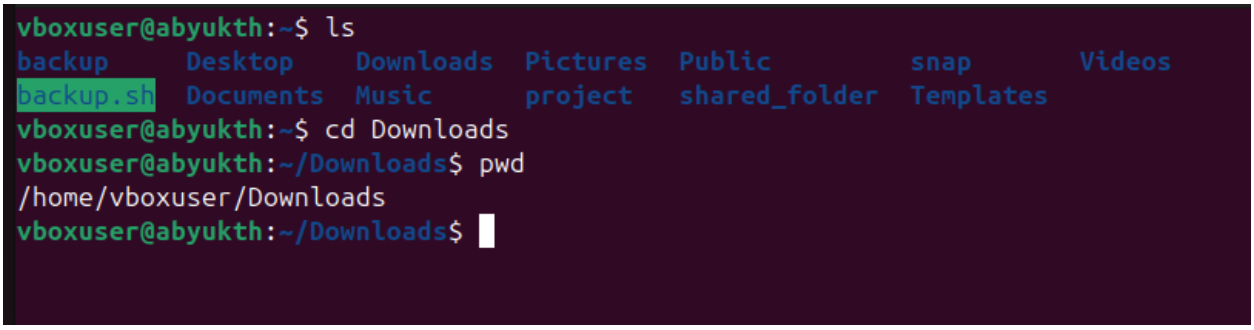
ASSIGNMENT-2

1.What does the command pwd, whoami, and hostname display?

A.

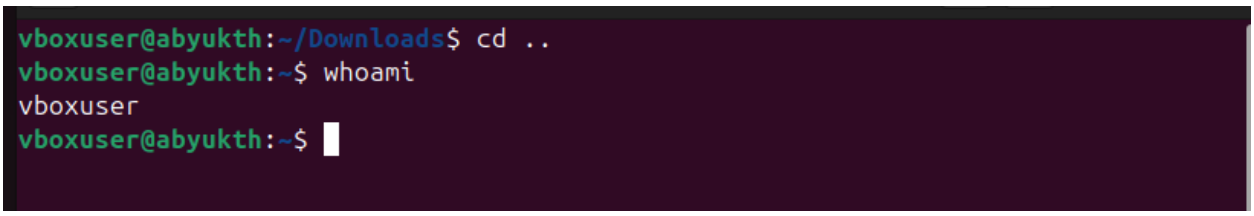
Pwd:

- Full form is Present working directory.
- Pwd displays the current directory path you are working in.



Whoami:

- Shows the username of the currently logged in user account.



Hostname:

- Hostname will display the name of the computer in the network.

```
vboxuser@abyukth:~/Downloads$ cd ..
vboxuser@abyukth:~$ whoami
vboxuser
vboxuser@abyukth:~$ hostname
abyukth
vboxuser@abyukth:~$
```

2. Write the command to create a directory named "project" inside the /home/student folder and keep three .txt file into it. Give output snapshot.

A.

Mkdir: mkdir full form is make directory. It is used to create one or more directories.

```
vboxuser@abyukth:~$ ls
backup     Desktop    Downloads  Pictures  Public         snap     Templates
backup.sh  Documents  Music      project   shared_folder  student  Videos
vboxuser@abyukth:~$ cd student
vboxuser@abyukth:~/student$ mkdir project
vboxuser@abyukth:~/student$ ls
project
vboxuser@abyukth:~/student$ cd project
vboxuser@abyukth:~/student/project$ touch file1.txt file2.txt file3.txt
vboxuser@abyukth:~/student/project$ ls
file1.txt  file2.txt  file3.txt
vboxuser@abyukth:~/student/project$ ls -l
total 0
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file1.txt
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file2.txt
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file3.txt
vboxuser@abyukth:~/student/project$
```

3.Explain the difference between absolute path and relative path with proper examples?

A.

Absolute path:

- An absolute path provides the full address or location of a file or folder from the directory.
- It always starts at the top-level directory.

- It starts from the root '/' directory.
- It does not depend upon where you currently are working from.

```
vboxuser@abyukth:~$ cd /home/student/project
```

Relative path:

- A relative path defines the position of a file or folder with reference to the working location.
- It is based not from root but from your existing location.

```
vboxuser@abyukth:~$ cd project
vboxuser@abyukth:~/project$
```

4.What command will give you the already executed command traces in the terminal. Give output snapshot?

A.

- The history command lists past commands complete with line numbers.
- By default, the history shows 500–1000 past commands.
- We can also search for a command or repeat a previously entered command using history.

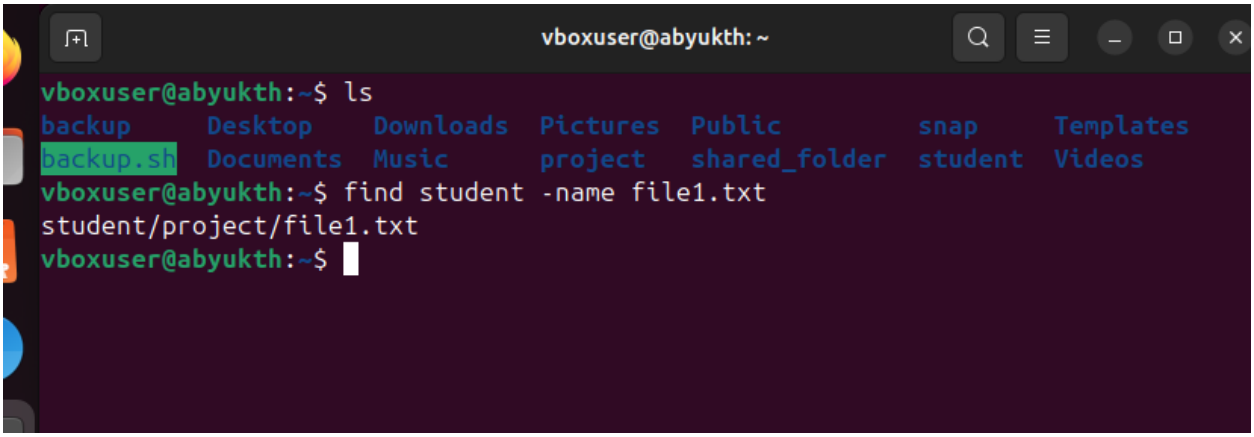
```
vboxuser@abyukth:~/project$ cd ..
vboxuser@abyukth:~$ history
    1  gedit abyukth.c
    2  # version 46.1
    3  version 46.1
    4  version 46.1-3
    5  ls
    6  desktop
    7  man ls
    8  $ls
    9  $ ls
   10  ls
   11  ls downloads
   12  ls Download
   13  ls Downloads
   14  ls Documents
   15  ls-l
   16  ls -l
   17  ls
   18  ls Documents
   19  ls -l
   20  mv hi backup
   21  mv (hi) (backup)
   22  mv (report.txt) (backup)
   23  mv [report.txt] [backup]
   24  mv [hi] [backup]
```

5. Compare the working functionality of find and locate command. Which one is faster and why?

A.

“find”:

- This option help to search for files and directories in a real-time manner by examining the directory tree.
- It can scan based on name, type, size, permissions, modification time, etc.
- Slower, because you are scanning the files in the system each time you execute the search.



“locate”:

- Searches for files based on a pre-constructed database.
- Very quick, since it doesn't look through the filesystem in real time.
- Can be used without knowing the full path.
- Database may or may not be current.
- Less flexible than find.

----image-----

6. Which command is used to modify file permissions in Linux? Give an example.

A.

“chmod” command is used to modify the file permissions.

- There are some values set for the modes.
  - Read = 4

    - Write = 2
    - Execute = 1
- Based on these values we can modify the files permissions.

Let's look up into an example:

```
vboxuser@abyukth:~/student/project$ ls
file1.txt  file2.txt  file3.txt
vboxuser@abyukth:~/student/project$ chmod 755 file1.txt
vboxuser@abyukth:~/student/project$ ls -l
total 0
-rwxr-xr-x 1 vboxuser vboxuser 0 Sep 27 12:34 file1.txt
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file2.txt
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file3.txt
vboxuser@abyukth:~/student/project$
```

From above screenshot the command (chmod 755 file.txt)

- Owner = rwx (7 = 4+2+1).
- Group = r-x (5 = 4+0+1).
- Others = r-x (5 = 4+0+1).

So that 755 is code value to give permission to the file. There are different code values to modify file permissions.

7. A file has permissions -rw-r--r--. What does this mean?

A.

The file has permissions. To access a file there are some specific permission protocols should be followed. In same way “-rw-r--r--” is one of the permissions protocols. The permission can be modified So, let's break down this “-rw-r--r--”

- every permission command consists of 10 characters.
- First character:
  - - = regular file.
  - “d" = directory.
  - “l” = symbolic link.
  - “c” = character device.
  - “b” = block device.
- Permissions for next 9 characters that is divided into 3-groups:
  - User = first 3.
  - Group = next 3.
  - Others = last 3.
- Each permission is:
  - r = read.
  - w = write.
  - x = execute.
  - - = no permission.

Let’s take look “-rw-r--r--” for this:

- First character “-” = regular file
- Next 3 characters:
  - User can read and write a file, but it can’t be executed.
- Middle 3 characters:
  - Group member can only read but can’t execute and write.
- Last 3 characters:
  - Public can only read.

Example for file permissions

```
vboxuser@abyukth:~/student/project$ ls -l
total 0
-rwxr-xr-x 1 vboxuser vboxuser 0 Sep 27 12:34 file1.txt
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file2.txt
-rw-rw-r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file3.txt
vboxuser@abyukth:~/student/project$ echo 'V S Saidatta'
V S Saidatta
vboxuser@abyukth:~/student/project$
```

8. Explain the difference between chown and chgrp? with an example.

A.

Chown:

- “chown” full form is change owner.
- This chown command is used to change the owner of the file or directory.

Example:

- Syntex: chown new_owner filename.
- - chown student file1.txt

Chgrp:

- “chgrp” full form is change group.
- This chgrp command is used to change the group ownership of the file or directory.

Example:

- Syntex: chgrp new_group file name.
- - chgrp techbugs file1.text

9. A file needs to be accessible by multiple users but only writable by the owner. How will you set permissions?

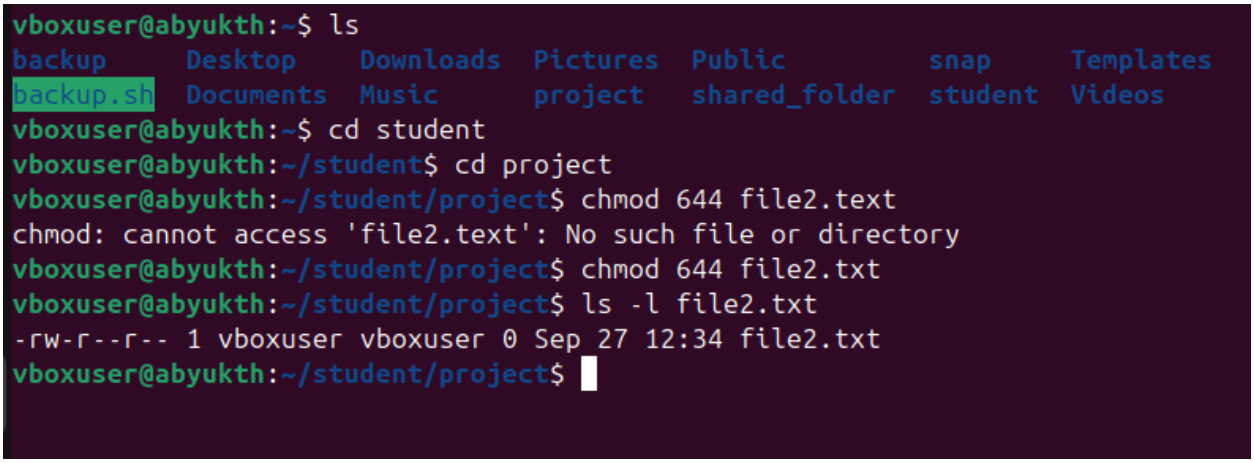
A.

We are allowed to make a file available to multiple users, but allow the owner to modify the file. We grant the owner read and write permissions, while allowing the group and others read access only.

For that we use:

- Owner = read & write
- Group = read only
- Others = read only

We can use “chmod” to modify the file permissions. For that we need to use 644 code.

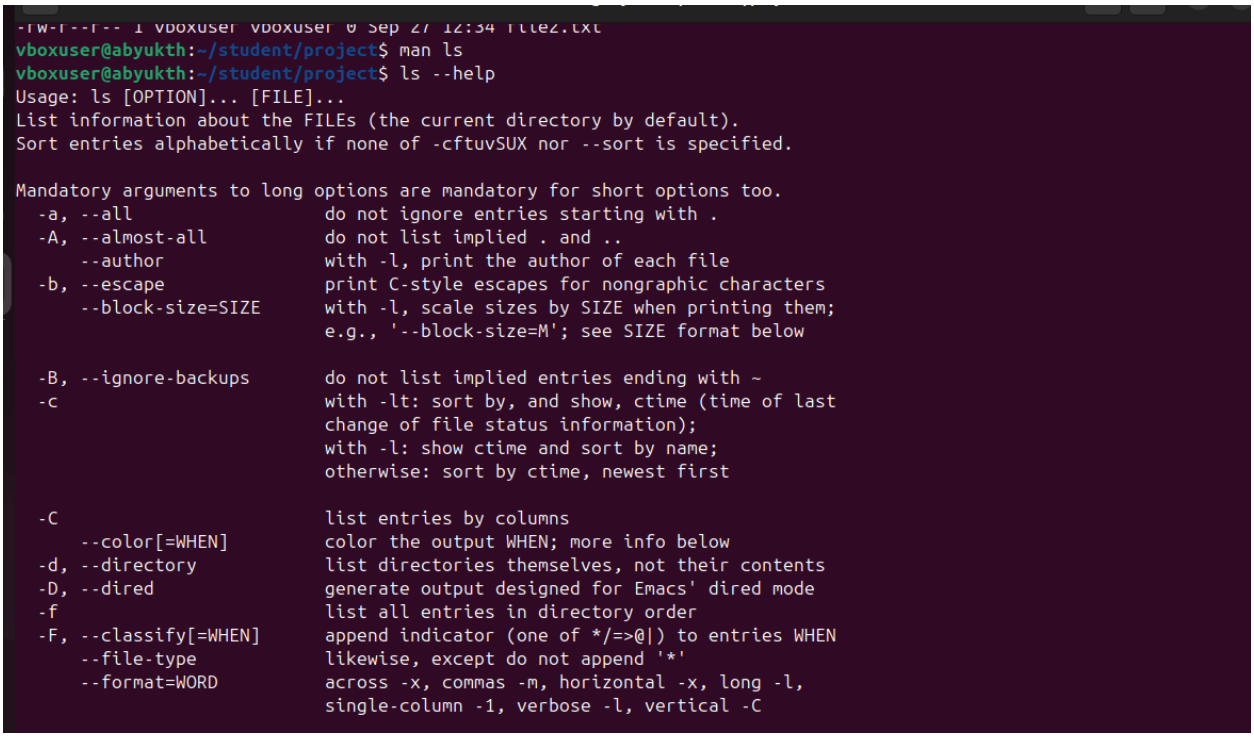


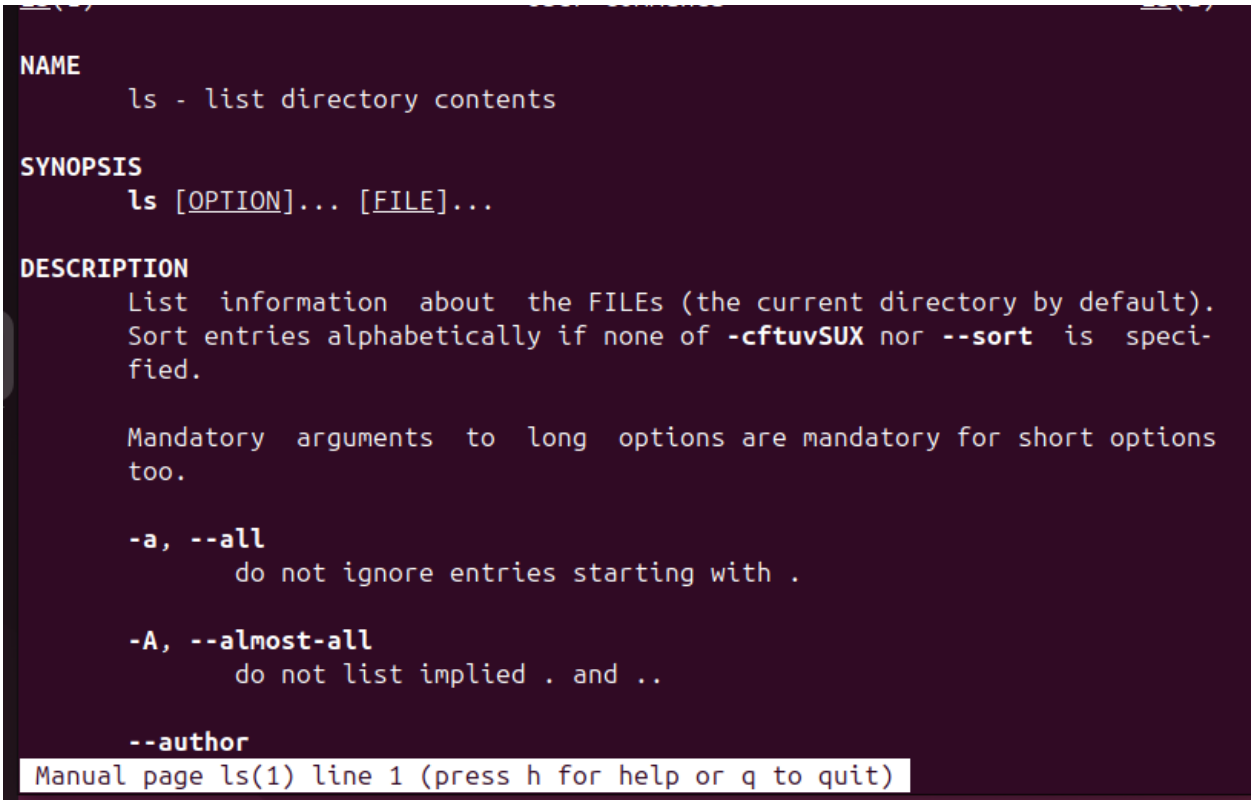
10. How do you check the manual page for any Linux commands?

A.

all the commands you can run are having a manual page that will explain how they work, what options they have, and some examples. You can see the manual for all commands by using the command man and the name of the <command>.

- Example: “man ls” will output the manual page for ls.
- Typing q will exit the man page.
- If you want to see a summary of the command for that command --help, and which will again summaries the options and usage in the terminal.


-rw-r--r-- 1 vboxuser vboxuser 0 Sep 27 12:34 file2.txt
vboxuser@abyukth:~/student/project$ man ls
vboxuser@abyukth:~/student/project$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
Mandatory arguments to long options are mandatory for short options too.
-a, --all do not ignore entries starting with .
-A, --almost-all do not list implied . and ..
--author with -l, print the author of each file
-b, --escape print C-style escapes for nongraphic characters
--block-size=SIZE with -l, scale sizes by SIZE when printing them;
e.g., '--block-size=M'; see SIZE format below
-B, --ignore-backups do not list implied entries ending with ~
-c with -lt: sort by, and show, ctime (time of last
change of file status information);
with -l: show ctime and sort by name;
otherwise: sort by ctime, newest first
-C list entries by columns
--color[=WHEN] color the output WHEN; more info below
-d, --directory list directories themselves, not their contents
-D, --dired generate output designed for Emacs' dired mode
-f list all entries in directory order
-F, --classify[=WHEN] append indicator (one of */=>@|) to entries WHEN
--file-type likewise, except do not append '*'
--format=WORD across -x, commas -m, horizontal -x, long -l,
single-column -1, verbose -l, vertical -C


NAME
ls - list directory contents
SYNOPSIS
ls [OPTION]... [FILE]...
DESCRIPTION
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
Mandatory arguments to long options are mandatory for short options too.
-a, --all
do not ignore entries starting with .
-A, --almost-all
do not list implied . and ..
--author
Manual page ls(1) line 1 (press h for help or q to quit)