

OPERATING SYSTEMS PRACTICE (COM301P)

Name: Vinayak Sethi

Roll No: COE18B061

Assignment 1

Q1. Test drive and understand the usage of all the commands given in the **50 Most Frequently Used UNIX / Linux Commands and linuxcommands.pdf**

1. tar command

The Linux “tar” stands for tape archive, which is used by large number of **Linux/Unix** system administrators to deal with tape drives backup. The tar command used to rip a collection of files and directories into highly compressed archive file commonly called **tarball** or **tar,gzip** and **bzip** in Linux.

- a.) tar -cvf file.tar directory:** This command creates a tar file called file.tar which is the Archive of all files in current directory.
- b.) tar -xvf file.tar:** This command extracts files from Archives.
- c.) tar -tvf file.tar:** This command will list the contents(**t**) of file.tar.
- d.) tar -xvf file.tar filename:** This command will extract a single file called filename from file.tar.
- e.) tar -rvf file.tar file2:** This command will add file or directory to existing tar archived file using the option **r**(append).

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ tar -cvf Asgn1.tar Asgn1/
Asgn1/
Asgn1/Intr.txt
Asgn1/Int.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1.tar Asgn2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ tar -xvf Asgn1.tar
Asgn1/
Asgn1/Intr.txt
Asgn1/Int.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ tar -tvf Asgn1.tar
drwxrwxr-x vinayak/vinayak 0 2020-08-27 17:29 Asgn1/
-rw-rw-r-- vinayak/vinayak 73 2020-08-26 19:51 Asgn1/Intr.txt
-rw-rw-r-- vinayak/vinayak 77 2020-08-26 19:53 Asgn1/Int.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ tar -xvf Asgn1.tar Asgn1/Intr.txt
Asgn1/Intr.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ tar -rvf Asgn1.tar Asgn2
Asgn2/
Asgn2/Theory/
Asgn2/Theory/src1/
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ tar -xvf Asgn1.tar
Asgn1/
Asgn1/Intr.txt
Asgn1/Int.txt
Asgn2/
Asgn2/Theory/
Asgn2/Theory/src1/
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
```

2. grep command

Grep is an acronym that stands for **G**lobal **R**egular **E**xpression **P**rint. Grep is a Linux / Unix command-line tool used to search for a string of characters in a specified file. The text search pattern is called a regular expression. When it finds a match, it prints the line with the result. The grep command is handy when searching through large log files.

Syntax:

grep [options] pattern [files]

Option Description

a.) **-i** : Ignores case for matching.

b.) **-c** : This prints only a count of the lines that match a pattern.

c.) **-l** : Displays list of filenames only.

d.) **-w** : Match whole word.

e.) **-o** : Print only the matched parts of a matching line,

f.) **-n** : Display the matched lines and their line numbers.

g.) **-v** : This prints out all the lines that do not matches the pattern.

h.) grep “^pattern” filename: The ^ regular expression pattern specifies the start of a line. This can be used in grep to match the lines which start with the given string or pattern.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UnixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -i "Unix" sample1.txt
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UnixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -c "Unix" sample1.txt
4
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -l "Unix" *
input.txt
grep: Lab Assignments: Is a directory
sample1.txt
sample2.txt
sample2.txt.bak
sample3.txt
samplemv.txt
grep: Theory: Is a directory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -w "Unix" sample1.txt
Unix is a great OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -o "Unix" sample1.txt
Unix
Unix
Unix
Unix
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -n "Unix" sample1.txt
1: Unix is a great OS.
2: Unix systems use a centralized operating system kernel which manages system and process activities.
3: Unix is a great OS.
4: A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -v "Unix" sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
UNIX is a free OS.
UNIX is a free OS.
UNIXOS systems use a centralized operating system kernel which manages system and process activities.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep '^UNIX' sample1.txt
UNIX is a free OS.
UNIX is a free OS.
UNIX is a free OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

3. find command

The Linux **Find Command** is one of the most important and frequently used command command-line utility in Unix-like operating systems. Find command is used to search and locate the list of files and directories based on conditions you specify for files that match the arguments.

find can be used in a variety of conditions like you can find files by **permissions, users, groups, file type, date, size**, and other possible criteria.

a.) find ./Directory -name filename: It will search for filename in given directoryname.

b.) find ./Directory -name *.extension: It will search all the files with given extension like txt, docx etc in the directory.

c.) find ./Directory -empty: This command find all empty folders and files in the entered directory or sub-directories.

d.) find ./Directory -perm 777: This command find all the files in the given directory or sub-directory with the given permissions(777).

e.) find ./ -type f -name "*txt" -exec grep 'pattern' {} \; : This command print lines which have 'given 'pattern' in them and '-type f' specifies the input type is a file.

f.) find ./Directory -name filename -exec rm -i {} \; : When this command is entered, a prompt will come for confirmation, if you want to delete filename or not. if we enter 'Y/y' it will delete the file.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ find ./Documents -name sample1.txt
./Documents/Test/sample1.txt
./Documents/Operating Systems/sample1.txt
vinayak@vinayak-Swift-SF315-52G:~$ find ./Documents -name *.txt
./Documents/Test/sample2.txt
./Documents/Test/sample3.txt
./Documents/Test/sample1.txt
./Documents/Test/Input.txt
./Documents/Test/Assignment1.txt
./Documents/Operating Systems/sample2.txt
./Documents/Operating Systems/sample3.txt
./Documents/Operating Systems/sample1.txt
./Documents/Operating Systems/Input.txt
./Documents/Operating Systems/samplemv.txt
./Documents/Operating Systems/Lab Assignments/samplemv.txt
./Documents/Operating Systems/Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~$ find ./Documents -empty
./Documents/Test/Assignment1.txt
./Documents/Operating Systems/Lab Assignments/Assignment
./Documents/Operating Systems/Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~$ find ./Documents -perm 777
./Documents/3rd Yr Books/John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman-Introduction to Automata Theory, Languages, and Computations-Prentice Hall (2006).pdf
vinayak@vinayak-Swift-SF315-52G:~$ find ./ -type f -name "*.txt" -exec grep 'processing' {} \;
multiprocessing
70583 ba424016db7469603d57fe135c97c70b88b91fb7 multiprocessing-autoconf
70584 792a7f939fddeac1a1cedc16eaf815a300ba3267 multiprocessing-autoconf
70585 3adb3974e77c09072449d1f36f09ea264db0af6 multiprocessing-autoconf
70586 8ad251da474274f17ab093bdf558a2a93334176c multiprocessing-autoconf
70592 41e344e571a1a16023613af10339a4ac272b045371 multiprocessing-autoconf
70593 73f4e8adfebf8d25da01dd7f3cb041444100ea multiprocessing-autoconf
    processing, or compiling of Source Code by a
    and functions for writing embedded signal processing
    processing, or compiling of Source Code by a
    and functions for writing embedded signal processing
The Commons HttpClient binary code is included with no modifications except postprocessing
    interpreting software instructions and processing data contained in
processing tools are not generally available, and the
processing tools are not generally available, and the
vinayak@vinayak-Swift-SF315-52G:~$ find ./Documents -name samplemv.txt -exec rm -i {} \;
rm: remove regular file './Documents/Operating Systems/samplemv.txt'? y
rm: remove regular file './Documents/Operating Systems/Lab Assignments/samplemv.txt'? y
vinayak@vinayak-Swift-SF315-52G:~$ find ./Documents -name *.txt
./Documents/Test/sample2.txt
./Documents/Test/sample3.txt
./Documents/Test/sample1.txt
./Documents/Test/Input.txt
./Documents/Test/Assignment1.txt
./Documents/Operating Systems/sample2.txt
./Documents/Operating Systems/sample3.txt
./Documents/Operating Systems/sample1.txt
./Documents/Operating Systems/Input.txt
./Documents/Operating Systems/Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~$ []

```

g.) find / -mtime 30: To find all the files which are modified 30 days back.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ find / -mtime 30
find: '/etc/cups/ssl': Permission denied
find: '/etc/polkit-1/localauthority': Permission denied
find: '/etc/ssl/private': Permission denied
find: '/proc/tty/driver': Permission denied
find: '/proc/1/task/1/fd': Permission denied
find: '/proc/1/task/1/fdinfo': Permission denied
find: '/proc/1/task/1/ns': Permission denied
find: '/proc/1/fd': Permission denied
find: '/proc/1/map_files': Permission denied
find: '/proc/1/fdinfo': Permission denied
find: '/proc/1/ns': Permission denied

```

h.) find / -cmin -60: To find all the files which are changed in the last 1 hour.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ find / -cmin -60  
/etc/cups  
/etc/cups/subscriptions.conf.0  
/etc/cups/subscriptions.conf  
find: '/etc/cups/ssl': Permission denied  
find: '/etc/polkit-1/localauthority': Permission denied  
find: '/etc/ssl/private': Permission denied  
/proc  
/proc/fb  
/proc/fs/ext4  
/proc/fs/ext4/sda4  
/proc/fs/ext4/sda4/options  
/proc/fs/ext4/sda4(mb)_groups  
/proc/fs/ext4/sda4/es_shrinker_info  
/proc/fs/jbd2  
/proc/fs/jbd2/sda4-8
```

i.) **find / -size 100M:** To find all **100MB** files.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ find / -size 100M  
find: '/etc/cups/ssl': Permission denied  
find: '/etc/polkit-1/localauthority': Permission denied  
find: '/etc/ssl/private': Permission denied  
find: '/proc/tty/driver': Permission denied  
find: '/proc/1/task/1/fd': Permission denied  
find: '/proc/1/task/1/fdinfo': Permission denied  
find: '/proc/1/task/1/ns': Permission denied  
find: '/proc/1/fd': Permission denied  
find: '/proc/1/map_files': Permission denied  
find: '/proc/1/fdinfo': Permission denied  
find: '/proc/1/ns': Permission denied  
find: '/proc/2/task/2/fd': Permission denied  
find: '/proc/2/task/2/fdinfo': Permission denied  
find: '/proc/2/task/2/ns': Permission denied
```

j.) **find ~ -empty:** This will list all the empty files in home directories.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ find ~ -empty  
/home/vinayak/.gconf  
/home/vinayak/Templates  
/home/vinayak/Public  
/home/vinayak/.cache/xreader/WebKitCache/Version 16/Blobs  
/home/vinayak/.cache/obexd  
/home/vinayak/.cache/evolution/addressbook/trash  
/home/vinayak/.cache/evolution/mail/trash  
/home/vinayak/.cache/evolution/tasks/trash  
/home/vinayak/.cache/evolution/calendar/trash  
/home/vinayak/.cache/evolution/memos/trash  
/home/vinayak/.cache/evolution/sources/trash  
/home/vinayak/.cache/mozilla/firefox/1x3l0ysp.default  
/home/vinayak/.cache/mozilla/firefox/6pylpsy6.default-release/cache2/entries  
/home/vinayak/Downloads/Python-3.8.5/Parser/pgen/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/PC/Layout/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/PC/Layout/support/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/tkinter/test/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/tkinter/test/test_tkinter/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/tkinter/test/test_ttk/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/urllib/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/pydoc_data/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/sqlite3/test/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data01/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data01/subdirectory/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/zipdata01/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/zipdata02/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data02/one/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data02/two/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/namespace_pkgs/not_a_namespace_pkg/foo/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/namespace_pkgs/module_and_namespace_package/a_test/empty  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data03/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data03/namespace/porton1/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data03/namespace/porton2/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data03/namespace/resource1.txt  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/nullcert.pem  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data/package2 submodule2.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/test_importlib/data/package/submodule.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/test/leakers/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/email/mime/_init_.py  
/home/vinayak/Downloads/Python-3.8.5/Lib/lib2to3/tests/data/fixers/myfixes/_init_.py
```

k.) find -iname filename: It returns the position of file from current working directory.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ find -iname Intr.txt
./Asgn1/Intr.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
```

4. ssh command

The ssh command provides a secure encrypted connection between two hosts over an insecure network. This connection can also be used for terminal access, file transfers, and for tunneling other applications. Graphical X11 applications can also be run securely over SSH from a remote location.

a.) ssh: Displays the details of ssh client installed if not then error.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ service ssh status
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2020-08-29 19:44:44 IST; 30s ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 33548 (sshd)
    Tasks: 1 (limit: 9303)
   Memory: 1.2M
  CGroup: /system.slice/ssh.service
          └─33548 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Aug 29 19:44:44 vinayak-Swift-SF315-52G systemd[1]: Starting OpenBSD Secure Shell server...
Aug 29 19:44:44 vinayak-Swift-SF315-52G sshd[33548]: Server listening on 0.0.0.0 port 22.
Aug 29 19:44:44 vinayak-Swift-SF315-52G sshd[33548]: Server listening on :: port 22.
Aug 29 19:44:44 vinayak-Swift-SF315-52G systemd[1]: Started OpenBSD Secure Shell server.
vinayak@vinayak-Swift-SF315-52G:~$ 
```

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ssh
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote tun]] destination [command]
```

b.) ssh username@ipaddress: Connects to the pc with username and ip address as given

```
vinayak@vinayak-Swift-SF315-52G:~$ ssh vinayak@localhost
vinayak@localhost's password:
Last login: Sat Aug 29 19:51:17 2020 from 127.0.0.1
```

c.) **service ssh stop:** It stops any active ssh server.

```
vinayak@vinayak-Swift-SF315-52G:~$ service ssh stop
===== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to stop 'ssh.service'.
Authenticating as: Vinayak Sethi,,, (vinayak)
Password:
===== AUTHENTICATION COMPLETE ===
```

d.) **service ssh start:** Starts the ssh server.

```
vinayak@vinayak-Swift-SF315-52G:~$ service ssh start
===== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to start 'ssh.service'.
Authenticating as: Vinayak Sethi,,, (vinayak)
Password:
===== AUTHENTICATION COMPLETE ===
```

e.) **service ssh status:** Displays the status of ssh i.e whether active or not.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ service ssh status
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2020-08-29 19:52:53 IST; 3min 16s ago
    Docs: man:sshd(8)
           man:sshd_config(5)
  Process: 35637 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 35652 (sshd)
   Tasks: 1 (limit: 9303)
  Memory: 1.3M
   CGroup: /system.slice/ssh.service
           └─35652 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Aug 29 19:52:53 vinayak-Swift-SF315-52G systemd[1]: Starting OpenBSD Secure Shell server...
Aug 29 19:52:53 vinayak-Swift-SF315-52G sshd[35652]: Server listening on 0.0.0.0 port 22.
Aug 29 19:52:53 vinayak-Swift-SF315-52G sshd[35652]: Server listening on :: port 22.
Aug 29 19:52:53 vinayak-Swift-SF315-52G systemd[1]: Started OpenBSD Secure Shell server.
vinayak@vinayak-Swift-SF315-52G:~$ 
```

5. sed command

SED command in UNIX stands for stream editor and it can perform lot's of function on file like, searching, find and replace, insertion or deletion. Though most common use of SED command in UNIX is for substitution or for find and replace. By using SED we can edit files even without opening it, which is much quicker way to find and replace something in file, than first opening that file in VI Editor and then changing it.

When we copy a DOS file to Unix, we can find \r\n in the end of each line.

a.) **sed 's/\\$/\n' Doc.txt:** This converts the DOS file format to Unix file format using sed command.

b.) **sed -n '1!G;h;\$p' Doc.txt:** Print the file content in reverse order.

c.) sed '/.=/' Doc.txt |sed 'N; s/\n/ /': It add line number for all non-empty-lines in a file.

```
vinayak@vinayak-SF315-52G:~/Documents/OS$ sed '/.=/' Doc.txt
An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources

For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware, although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers
vinayak@vinayak-SF315-52G:~/Documents/OS$ sed -n '1!G;h;$p' Doc.txt
For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware, although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers.

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.
vinayak@vinayak-SF315-52G:~/Documents/OS$ cat Doc.txt
An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware, although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers.
vinayak@vinayak-SF315-52G:~/Documents/OS$ 1
1 An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.

3
Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.
5 For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware, although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers.
vinayak@vinayak-SF315-52G:~/Documents/OS$ 
```

6. awk command

awk is abbreviated from the names of the developers – Aho, Weinberger, and Kernighan. Awk is a scripting language used for manipulating data and generating reports.

What can we do with awk?

1. AWK Operations:

- (a) Scans a file line by line
- (b) Splits each input line into fields
- (c) Compares input line/fields to pattern
- (d) Performs action(s) on matched lines

2. Useful For:

- (a) Transform data files
- (b) Produce formatted reports

a.) awk {print} filename: By default awk prints every line of data from the specified file.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Test
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ awk '{print}' sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$
```

b.) awk '/pattern/ {print}' filename: It prints the lines which matches the given pattern in input file.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Test
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ awk '/Operating system / {print}' sample1.txt
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$
```

c.) awk '{print \$1, \$n}' filename: It prints the 1st and nth column of input file.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Test$ awk '{print $1, $4}' sample1.txt
This test
An an
An a
Operating one
Operating one
Unix great
UNIX free
Unix a
Unix great
UNIX free
UNIxOS a
A also
UNIX free
Multiuser
Yet OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$
```

d.) awk '{print NR,\$0}' sample2.txt: NR command keeps a current count of the number of input records. Remember that records are usually lines. The awk command with NR prints all the lines along with the line number.

e.) awk '{print \$1,\$NF}' sample2.txt: NF command keeps a count of the number of fields within the current input record. \$1 represents the first field and \$n represents the last field of line.

f.) awk 'NR==3, NR==6 {print NR,\$0}' sample2.txt: It prints the lines from line 3 to line 6.

g.) awk '{ if (length(\$0) > max) max = length(\$0) } END { print max }' sample2.txt: To find the length of the longest line present in the file.

h.) awk 'END { print NR }' sample2.txt: To count the number of lines in file.

i.) awk -F ':' '\$3==\$4' sample2.txt: Print all the lines from sample2.txt that has the same uid and gid.

j.) awk '!(\$0 in array) { array[\$0]; print }' sample1.txt: Removes duplicate lines from file.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk '{print NR,$0}' sample2.txt
1 Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
2 There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
3 Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
4 A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
5 UNIX is a free OS.
6 Multiuser operating system.
7 Yet another powerful OS.
8
9
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk '{print $1,$NF}' sample2.txt
Unix Labs.
There available.
Several system.
A environment.
UNIX OS.
Multiuser system.
Yet OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk 'NR==3, NR==6 {print NR,$0}' sample2.txt
3 Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
4 A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
5 UNIX is a free OS.
6 Multiuser operating system.
7 Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk '{ if (length($0) > max) max = length($0) } END { print max }' sample2.txt
170
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk 'END { print NR }' sample2.txt
9
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk -F ':' '$3==$4' sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk '!($0 in array) { array[$0]; print }' sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
UNIXOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

7. vim command

It opens a file in vim editor.

a.) vim +n filename: It goes to nth line in given file.

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ vim +4 Doc.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ █

          vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
          File Edit View Search Terminal Help
An operating system (OS) is system software that manages computer hardware, soft
ware resources, and provides common services for computer programs.

Time-sharing operating systems schedule tasks for efficient use of the system an
d may also include accounting software for cost allocation of processor time, ma
ss storage, printing, and other resources.
█
For hardware functions such as input and output and memory allocation, the opera
ting system acts as an intermediary between programs and the computer hardware,
although the application code is usually executed directly by the hardware and f
requently makes system calls to an OS function or is interrupted by it. Operatin
g systems are found on many devices that contain a computer – from cellular phon
es and video game consoles to web servers and supercomputers.
~
~
~
~
~
~
~
4,0-1      All

```

b.) vim -R filename: It opens the file in read mode.

```

          vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
          File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ vim -R Doc.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ █

          vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
          File Edit View Search Terminal Help
An operating system (OS) is system software that manages computer hardware, soft
ware resources, and provides common services for computer programs.

Time-sharing operating systems schedule tasks for efficient use of the system an
d may also include accounting software for cost allocation of processor time, ma
ss storage, printing, and other resources.

For hardware functions such as input and output and memory allocation, the opera
ting system acts as an intermediary between programs and the computer hardware,
although the application code is usually executed directly by the hardware and f
requently makes system calls to an OS function or is interrupted by it. Operatin
g systems are found on many devices that contain a computer – from cellular phon
es and video game consoles to web servers and supercomputers.
~
~
~
~
~
~
~
"Doc.txt" [readonly] 5L, 817C      1,1      All

```

c.) vim +/search-term filename: It goes to the first match of the specified search-term in given file.

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ vim +/OS Doc.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
File Edit View Search Terminal Help
An operating system is system software that manages computer hardware, software resources, and provides common services for computer programs.

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between programs and the computer hardware, although the application code is usually executed directly by the hardware and frequently makes system calls to an OS function or is interrupted by it. Operating systems are found on many devices that contain a computer – from cellular phones and video game consoles to web servers and supercomputers.

~
~
~
~
~
~
~
~
"Doc.txt" 5L, 812C           5,276          All

```

8. diff command

diff stands for difference. This command is used to display the differences in the files by comparing the files line by line. Unlike its fellow members, cmp and comm, it tells us which lines in one file have to be changed to make the two files identical.

The important thing to remember is that diff uses certain special symbols and instructions that are required to make two files identical. It tells us the instructions on how to change the first file to make it match the second file.

Special symbols are:

a : add
c : change
d : delete

a.) diff file1name file2name: The normal output format consists of one or more sections that describe the differences. Each section looks like this:

```

change-command
< from-file-line...
---
> to-file-line...

```

In our example, 0a1, 2d2 and 4c4,5 are change commands. Each change command contains the following, from left to right:

- The line number or range of lines in the first file.
- A special change character.

- The line number or range of lines in the second file.

The change character can be one of the following:

- a - Add the lines.
- c - Change the lines.
- d - Delete the lines.

The change command is followed by the complete lines that are removed (<) and added to the file (>).

Let's explain the output:

- **0a1** - Add line **1** of the second file at the beginning of the file1 (after the line **0**).
 - ◆ **> Kubuntu** - The line from the second line that is added to the first file as described above.
- **2d2** - Delete line **2** in the first file. The **2** after the **d** symbol means that if the line is not deleted it would appear on line **2** in the second file.
 - ◆ **< Arch Linux** - the deleted line.
- **4c4,5** - Replace (change) line **5** in the first file with lines **4-5** from the second file.
 - ◆ **< CentOS** - The line in the first file to be replaced.
 - ◆ **---** - Separator.
 - ◆ **> Arch Linux** and **> Centos** - Lines from the second file replacing the line in the first file.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ diff file1.txt file2.txt
0a1
> Kubuntu
2d2
< Arch Linux
4c4,5
< CentOS
---
> Arch Linux
> Centos
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
```

b.) diff -c file1name file2name: To view differences in context mode, use the **-c** option.

The first file is indicated by *******, and the second file is indicated by **---**.

The line with ********* is just a separator.

The first two lines of this output show us information about file 1 and file 2. It lists the file name, modification date, and modification time of each of our files, one per line.

The next line has three asterisks *** followed by a line range from the first file (in our case lines 1 through 5, separated by a comma). Then four asterisks ****. After that it shows the contents of the first file with the following indicators:

- (i) If the line needs to be unchanged, it is prefixed by two spaces.
 - (ii) If the line needs to be changed, it is prefixed by an symbol and a space.
- The symbol means are as follows:

- (a) + : It indicates a line in the second file that needs to be added to the first file to make them identical.
- (b) - : It indicates a line in the first file that needs to be deleted to make them identical.
- (c) ! : Indicates that this line is part of a group of one or more lines that needs to change. There is a corresponding group of lines prefixed with "!" in the other file's context as well.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ diff -c file1.txt file2.txt
*** file1.txt    2020-08-29 10:22:23.410708577 +0530
--- file2.txt    2020-08-29 10:22:46.614810476 +0530
*****
*** 1,5 ****
Ubuntu
- Arch Linux
Debian
! CentOS
Fedora
--- 1,6 ----
+ Kubuntu
Ubuntu
Debian
! Arch Linux
! Centos
Fedora
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
```

c.) **diff -u file1name file2name**: To view differences in unified mode, use the -u option. It is similar to context mode but it doesn't display any **redundant information** or it shows the information in concise form.

The first file is indicated by ---, and the second file is indicated by +++.

The first two lines of this output show us information about file 1 and file 2. It lists the file name, modification date, and modification time of each of our files, one per line.

After that the next line has two at sign @ followed by a line range from the first file (in our case lines 1 through 5, separated by a comma) prefixed by – and then space and then again followed by a line range from the second file prefixed by + and at the end two at sign @. Followed by the file content in output tells us which line remain unchanged and which lines needs to added or deleted(indicated by symbols) in the file 1 to make it identical to file 2.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ diff -u file1.txt file2.txt
--- file1.txt    2020-08-29 10:22:23.410708577 +0530
+++ file2.txt    2020-08-29 10:22:46.614810476 +0530
@@ -1,5 +1,6 @@
+Kubuntu
 Ubuntu
-Arch Linux
 Debian
-CentOS
+Arch Linux
+Centos
 Fedora
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$
```

d.) **diff -ui file1name file2name:** Use the -i option to tell diff to ignores case.

e.) **diff -w file1name file2name:** It ignores white spaces while comparing.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ diff -ui file1.txt file2.txt
--- file1.txt    2020-08-29 10:22:23.410708577 +0530
+++ file2.txt    2020-08-29 10:22:46.614810476 +0530
@@ -1,5 +1,6 @@
+Kubuntu
 Ubuntu
-Arch Linux
 Debian
+Arch Linux
 CentOS
 Fedora
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ diff -w file1.txt file2.txt
0a1
> Kubuntu
2d2
< Arch Linux
4c4,5
< CentOS
---
> Arch Linux
> Centos
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$
```

9. sort command

Sort is a Linux program used for printing lines of input text files and concatenation of all files in sorted order. Sort command takes blank space as field separator and entire Input file as sort key. It is important to notice that sort command don't actually sort the files but only print the sorted output, until we redirect the output.

a.) sort filename: This command sorts the file assuming content to be in ASCII.

b.) sort sourcefile > destinationfile Or sort -o sourcefile destinationfile: Using the -o option is functionally the same as redirecting the output to a file.

c.) sort -r filename: This will do sorting in reverse order.

d.) sort -n filename: This option is used to sort the file with numeric data present inside.

e.) sort -k filename: Use the -k option to sort on a certain column.

f.) sort -c filename: This option is used to check if the file given is already sorted or not.

Note : If there is no output then the file is considered to be already sorted

g.) sort -u filename: To sort and remove duplicates pass the -u option to sort. This will write a sorted list to standard output and remove duplicates.

File Edit View Search Terminal Help

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intro.txt
HI
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ sort Intro.txt
BYE
HI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
I AM VINAYAK SETHI
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ sort Intro.txt > Intr.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn1 Asgn3 google index.html index.html.2 Intr.txt Perm
Asgn2 Asgn4 google.co.in index.html.1 Intro.txt num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intr.txt
BYE
HI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
I AM VINAYAK SETHI
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ sort -r Intro.txt
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
HI
BYE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat num.txt
2313
43635
21
42552
1241

vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ sort -n num.txt
21
1241
2313
42552
43635
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ sort -c Intro.txt
sort: Intro.txt:3: disorder: I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Int.txt
HI
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYE
BYE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ sort -u Int.txt
BYE
HI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
I AM VINAYAK SETHI
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ 
```

10. export command

export is bash shell BUILTINS commands, which means it is part of the shell. It marks an environment variables to be exported to child-processes.

export is defined in POSIX as The shell shall give the export attribute to the variables corresponding to the specified names, which shall cause them to be in the environment of subsequently executed commands. If the name of a variable is followed by = word, then the value of that variable shall be set to the word

a.) export: To view all the exported variables.

```
File Edit View Search Terminal Help
vi
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ export
declare -x CINNAMON VERSION="4.6.6"
declare -x COLORTERM="truecolor"
declare -x DBUS SESSION_BUS ADDRESS="unix:path=/run/user/1000/bus"
declare -x DEFAULTS_PATH="/usr/share/gconf/cinnamon.default.path"
declare -x DESKTOP_SESSION="cinnamon"
declare -x DISPLAY=:0"
declare -x GDMSESSION="cinnamon"
declare -x GDM_LANG="en_US"
declare -x GJS_DEBUG_OUTPUT="stderr"
declare -x GJS_DEBUG_TOPICS="JS ERROR;JS LOG"
declare -x GNOME_DESKTOP_SESSION_ID="this-is-deprecated"
declare -x GNOME_TERMINAL_SCREEN="/org/gnome/Terminal/screen/06ea5979_d45b_4816_a704_0bc12055cd15"
declare -x GNOME_TERMINAL_SERVICE=:1.118"
declare -x GPG_AGENT_INFO="/run/user/1000/gnupg/S.gpg-agent:0:1"
declare -x GTK_MODULES="gail:atk-bridge"
declare -x GTK_OVERLAY_SCROLLING="1"
declare -x HOME="/home/vinayak"
declare -x LANG="en_IN"
```

b.) export -p: To view all exported variables on current shell.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ export -p
declare -x CINNAMON VERSION="4.6.6"
declare -x COLORTERM="truecolor"
declare -x DBUS SESSION_BUS ADDRESS="unix:path=/run/user/1000/bus"
declare -x DEFAULTS_PATH="/usr/share/gconf/cinnamon.default.path"
declare -x DESKTOP_SESSION="cinnamon"
declare -x DISPLAY=:0"
declare -x GDMSESSION="cinnamon"
declare -x GDM_LANG="en_US"
declare -x GJS_DEBUG_OUTPUT="stderr"
declare -x GJS_DEBUG_TOPICS="JS ERROR;JS LOG"
declare -x GNOME_DESKTOP_SESSION_ID="this-is-deprecated"
declare -x GNOME_TERMINAL_SCREEN="/org/gnome/Terminal/screen/06ea5979_d45b_4816_a704_0bc12055cd15"
declare -x GNOME_TERMINAL_SERVICE=:1.118"
declare -x GPG_AGENT_INFO="/run/user/1000/gnupg/S.gpg-agent:0:1"
declare -x GTK_MODULES="gail:atk-bridge"
```

c.) export | grep PATH: To view PATH related environment variables.

```

vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ export | grep PATH
declare -x DEFAULTS_PATH="/usr/share/gconf/cinnamon.default.path"
declare -x MANDATORY_PATH="/usr/share/gconf/cinnamon.mandatory.path"
declare -x PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/usr/games:/usr/local/games"
declare -x XDG_SEAT_PATH="/org/freedesktop/DisplayManager/Seat0"
declare -x XDG_SESSION_PATH="/org/freedesktop/DisplayManager/Session0"
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 

```

d.) export -f function_name: It must be used if the names refer to functions. If -f is not used, the export will assume the names are variables.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ name() { echo "Vinayak"; }
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ export -f name
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ bash
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ name
Vinayak
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 

```

11. xargs command

xargs is a great command that reads streams of data from standard input, then generates and executes command lines; meaning it can take output of a command and passes it as argument of another command. If no command is specified, xargs executes echo by default.

a.) echo 'Test' | xargs mkdir: This command creates a directory.

b.) echo 'Test' | xargs rmdir: This command remove the directory.

c.) echo 'sample1.txt' | xargs cat: This command displays the content of the file.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Input.txt Lab 'Lab Assignment 1.odt' sample1.txt sample2.txt sample.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ echo 'Test' | xargs mkdir
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Input.txt 'Lab Assignment 1.odt' sample1.txt Test
Lab sample1.txt sample2.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ echo 'Test' | xargs rmdir
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Input.txt 'Lab Assignment 1.odt' sample2.txt Theory
Lab sample1.txt sample.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ echo 'sample1.txt' | xargs cat
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
Unix is a Free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
Unix is a Free OS.
UnixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
Unix is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 

```

12. ls command

ls is a Linux shell command that lists directory contents of files and directories.

a.) **ls -a:** list all files including hidden file starting with '.'

b.) **ls -color:** colored list [=always/never/auto]

c.) **ls -d:** list directories - with ' */'

d.) **ls -i:** list file's inode index number

e.) **ls -l:** list with long format - show permissions

f.) **ls -lh:** list long format with readable file size

g.) **ls -r:** list in reverse order

h.) **ls -R:** list recursively directory tree

i.) **ls -s:** list file size

j.) **ls -S:** sort by file size

k.) **ls -t:** sort by time & date

l.) **ls -X:** sort by extension name

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -a
.
.. 'Lab Assignment 1.odt' sample2.txt Theory
   Lab_Assignments sample2.txt.bak
Assignment1.txt '.~lock.Lab Assignment 1.odt#' sample3.txt
Input.txt sample1.txt samplemv.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls --color
Assignment1.txt Lab_Assignments sample2.txt.bak Theory
Input.txt sample1.txt sample3.txt
'Lab Assignment 1.odt' sample2.txt samplemv.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -d
.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -i
663161 Assignment1.txt 663238 sample1.txt 663226 samplemv.txt
663401 Input.txt 663392 sample2.txt 663140 Theory
663886 'Lab Assignment 1.odt' 663350 sample2.txt.bak
663061 Lab_Assignments 663381 sample3.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -l
total 3244
-rw-rw-r-- 1 vinayak vinayak 0 Aug 15 19:54 Assignment1.txt
-rw-rw-r-- 1 vinayak vinayak 1497 Aug 19 13:08 Input.txt
-rw-rw-r-- 1 vinayak vinayak 3285229 Aug 25 21:50 'Lab Assignment 1.odt'
drwxrwxr-x 2 vinayak vinayak 4096 Aug 16 13:26 Lab_Assignments
-rw-rw-r-- 1 vinayak vinayak 908 Aug 16 12:23 sample1.txt
-rw-rw-r-- 1 vinayak vinayak 593 Aug 19 16:13 sample2.txt
-rw-rw-r-- 1 vinayak vinayak 589 Aug 19 15:59 sample2.txt.bak
-rw-rw-r-- 1 vinayak vinayak 1497 Aug 19 15:15 sample3.txt
-rw-rw-r-- 1 vinayak vinayak 591 Aug 15 23:35 samplemv.txt
drwxrwxr-x 3 vinayak vinayak 4096 Aug 24 15:04 Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -lh
total 3.2M
-rw-rw-r-- 1 vinayak vinayak 0 Aug 15 19:54 Assignment1.txt
-rw-rw-r-- 1 vinayak vinayak 1.5K Aug 19 13:08 Input.txt
-rw-rw-r-- 1 vinayak vinayak 3.2M Aug 25 21:50 'Lab Assignment 1.odt'
drwxrwxr-x 2 vinayak vinayak 4.0K Aug 16 13:26 Lab_Assignments
-rw-rw-r-- 1 vinayak vinayak 908 Aug 16 12:23 sample1.txt
-rw-rw-r-- 1 vinayak vinayak 593 Aug 19 16:13 sample2.txt
-rw-rw-r-- 1 vinayak vinayak 589 Aug 19 15:59 sample2.txt.bak
-rw-rw-r-- 1 vinayak vinayak 1.5K Aug 19 15:15 sample3.txt
-rw-rw-r-- 1 vinayak vinayak 591 Aug 15 23:35 samplemv.txt
drwxrwxr-x 3 vinayak vinayak 4.0K Aug 24 15:04 Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -r
Theory           sample2.txt.bak   Lab_Assignments      Assignment1.txt
samplemv.txt    sample2.txt       'Lab Assignment 1.odt'
sample3.txt     sample1.txt       Input.txt
'Lab Assignment 1.odt'          sample2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -R
.:
Assignment1.txt      Lab_Assignments      sample2.txt.bak   Theory
Input.txt            sample1.txt          sample3.txt
'Lab Assignment 1.odt'  sample2.txt          samplemv.txt

./Lab_Assignments:
Assignment

./Theory:
allsortft.cpp
'(COE18B061) Preparatory Assignment 1'
'(COE18B061) Preparatory Assignment 1.zip'

'./Theory/(COE18B061) Preparatory Assignment 1':
allsort      mycopy.c      'Preparatory Assignment 1 (COE18B061).pdf'
allsort.cpp  mysort        'Preparatory Assignment 1.odt'
allsortft    mysort.c      remove
allsortft.cpp mysortfp     remove.c
mycopy       mysortfp.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -s
total 3244
0 Assignment1.txt          4 sample1.txt          4 samplemv.txt
4 Input.txt                4 sample2.txt          4 Theory
3212 'Lab Assignment 1.odt' 4 sample2.txt.bak
4 Lab_Assignments          4 sample3.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -S
'Lab Assignment 1.odt'  Input.txt    sample2.txt      Assignment1.txt
Lab_Assignments         sample3.txt  samplemv.txt
Theory                  sample1.txt  sample2.txt.bak
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -t
'Lab Assignment 1.odt'  sample2.txt.bak  Lab_Assignments Assignment1.txt
Theory                  sample3.txt    sample1.txt
sample2.txt              Input.txt    samplemv.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -X
Lab_Assignments  'Lab Assignment 1.odt'  sample1.txt  samplemv.txt
Theory          Assignment1.txt        sample2.txt
sample2.txt.bak  Input.txt          sample3.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ []

```

13. pwd command

pwd stands for Print Working Directory.

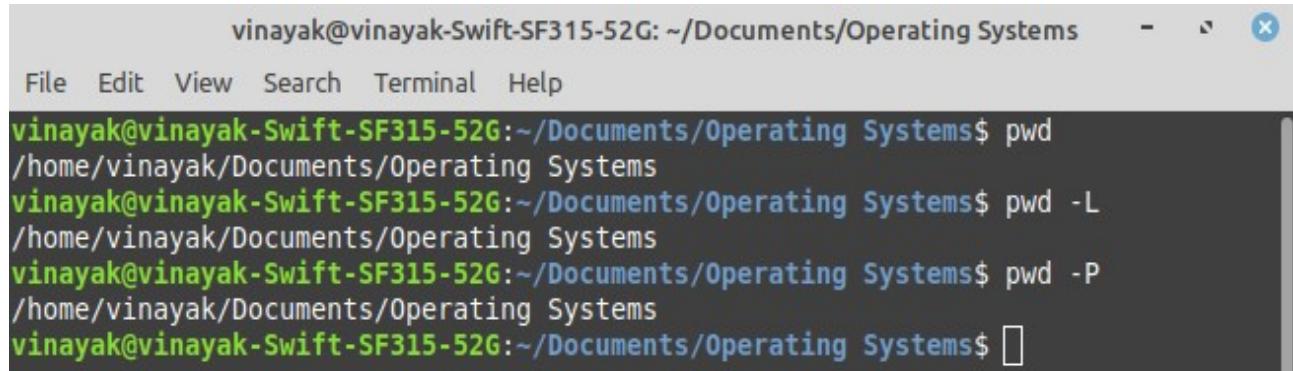
pwd is shell built-in command(pwd) or an actual binary(/bin/pwd).

\$PWD is an environment variable which stores the path of the current directory.

a.) **pwd:** It prints the path of the working directory, starting from the root.

b.) **pwd -L**: It prints the symbolic path.

c.) **pwd -P**: It prints the actual path.



```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ pwd
/home/vinayak/Documents/Operating Systems
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ pwd -L
/home/vinayak/Documents/Operating Systems
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ pwd -P
/home/vinayak/Documents/Operating Systems
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

14. cd command

The cd (“change directory”) command is used to change the current working directory in Linux and other Unix-like operating systems. The current working directory is the directory (folder) in which the user is currently working in. Each time we interact with our command prompt, we are working within a directory.

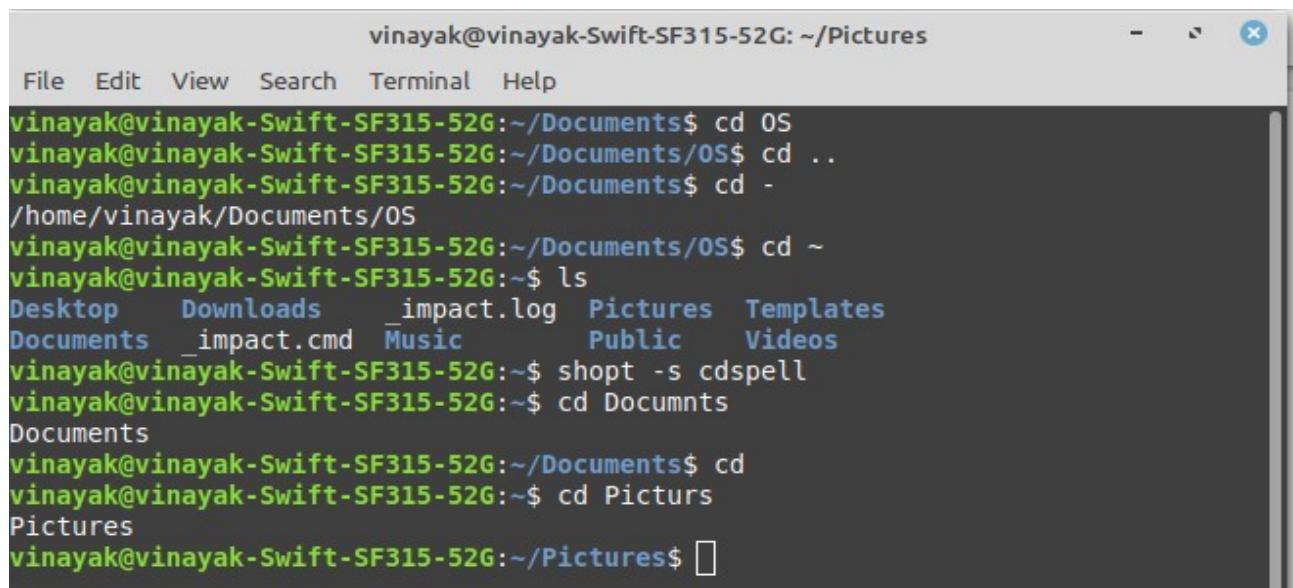
a.) **cd directortname**: It changes current working directory to directortname.

b.) **cd ..** : It moves to previous working directory.

c.) **cd -** : To change back to the previous working directory,

d.) **cd ~** : This command is used to change directory to the home directory.

e.) **shopt -s cdspell**: This command automatically correct mistyped directory names on cd.



```
vinayak@vinayak-Swift-SF315-52G: ~/Pictures
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents$ cd OS
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ cd ..
vinayak@vinayak-Swift-SF315-52G:~/Documents$ cd -
/home/vinayak/Documents/OS
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ cd ~
vinayak@vinayak-Swift-SF315-52G:~$ ls
Desktop Downloads _impact.log Pictures Templates
Documents _impact.cmd Music Public Videos
vinayak@vinayak-Swift-SF315-52G:~$ shopt -s cdspell
vinayak@vinayak-Swift-SF315-52G:~$ cd Documentns
Documents
vinayak@vinayak-Swift-SF315-52G:~/Documents$ cd
vinayak@vinayak-Swift-SF315-52G:~/$ cd Pictures
Pictures
vinayak@vinayak-Swift-SF315-52G:~/Pictures$ 
```

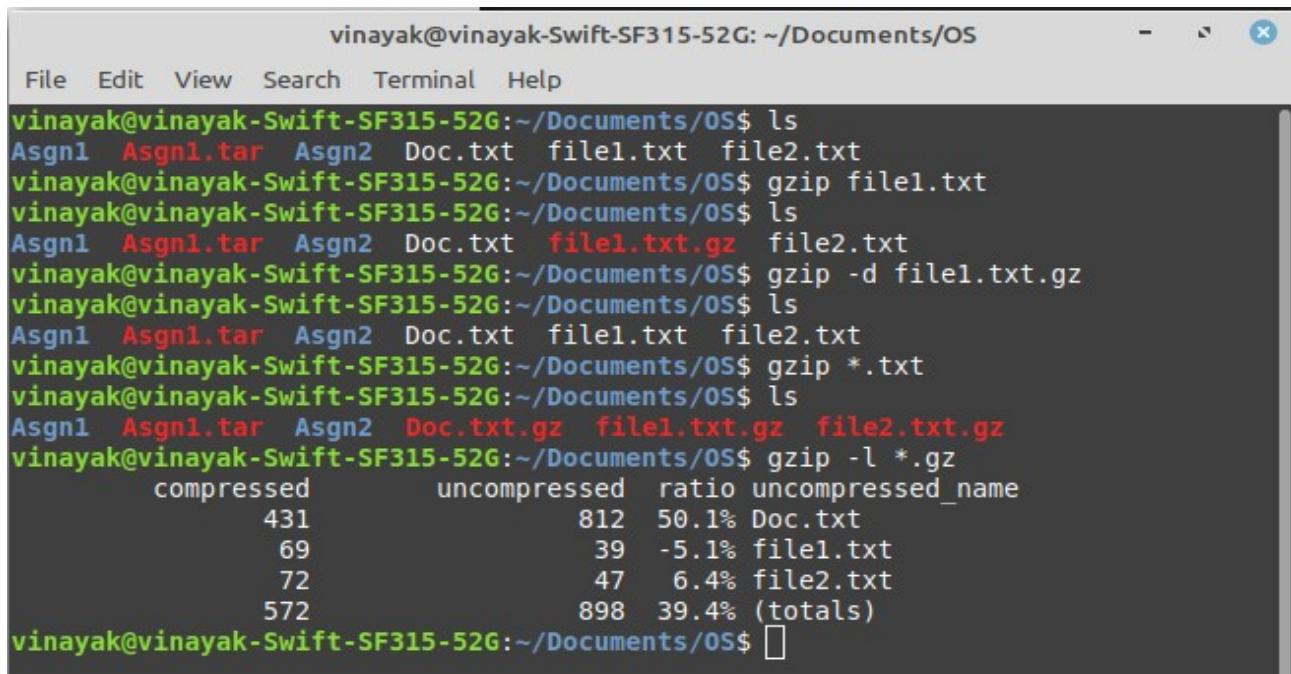
15. gzip command

gzip command compresses files. Each single file is compressed into a single file. The compressed file consists of a GNU zip header and deflated data. If given a file as an argument, gzip compresses the file, adds a “.gz” suffix, and deletes the original file. With no arguments, gzip compresses the standard input and writes the compressed file to standard output.

a.) **gzip filename:** To create a filename.gz compressed file.

b.) **gzip -d filename.gz:** To uncompress a filename.gz file.

c.) **gzip -l filename.gz:** It displays the compression ratio of compressed files using -l.



The screenshot shows a terminal window with the following session:

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/OS
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt file2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ gzip file1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt.gz file2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ gzip -d file1.txt.gz
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt file2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ gzip *.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt.gz file1.txt.gz file2.txt.gz
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ gzip -l *.gz
      compressed      uncompressed      ratio      uncompressed_name
              431                  812    50.1%  Doc.txt
                 69                   39   -5.1%  file1.txt
                 72                   47    6.4%  file2.txt
              572                  898   39.4% (totals)
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
```

16. bzip2 command

bzip2 command in Linux is used to compress and decompress the files i.e. it helps in binding the files into a single file which takes less storage space as the original file use to take. It has a slower decompression time and higher memory use. It uses Burrows-Wheeler block sorting text compression algorithm, and Huffman Coding. Each file is replaced by a compressed version of itself, with the name original name of the file followed by extension bz2.

a.) **bzip2 -z filename:** This command forces compression.

b.) **bzip2 -k filename:** This option does compression but does not delete the original file.

c.) **bzip2 -d filename.bz2:** This command is used for decompression of compressed files.

d.) **bzcat filename.bz2:** This command is used to read the compressed file without decompressing it.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt file2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ bzip2 -z file1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt.bz2 file2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ bzip2 -k file2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt.bz2 file2.txt bz2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ bzip2 -d file1.txt.bz2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt file2.txt file2.txt.bz2
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ bzcat file2.txt.bz2
Ubuntu
Ubuntu
Debian
Arch Linux
Centos
Fedora
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ bzip2 -z file1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt.bz2 file2.txt file2.txt.bz2
```

17. unzip command

The unzip command extracts all files from the specified ZIP archive to the current directory.

a.) **unzip filename.zip:** Extracts a .zip file.

b.) **unzip -l filename.zip:** It lists the content of .zip file without extracting it.

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1.txt file1.zip file2.txt.bz2 file2.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ unzip file1.zip
Archive: file1.zip
replace file1.txt? [y]es, [n]o, [A]ll, [N]one, [r]ename: r
new name: file1cpy.txt
extracting: file1cpy.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ ls
Asgn1 Asgn1.tar Asgn2 Doc.txt file1cpy.txt file1.zip file2.txt bz2 file2.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ unzip -l file2.zip
Archive: file2.zip
  Length      Date      Time     Name
----- -----
        47 2020-08-29 10:22  file2.txt
----- -----
          47                      1 file
vinayak@vinayak-Swift-SF315-52G:~/Documents/OS$ 
```

18. shutdown command

The shutdown command in Linux is used to shutdown the system in a safe way. We can shutdown the machine immediately, or schedule a shutdown using 24 hour format. It brings the system down in a secure way. When the shutdown is initiated, all logged-in users and processes are notified that the system is going down, and no further logins are allowed.

Only root user can execute shutdown command.

a.) sudo shutdown +40: This command will schedule a system shutdown in 40 minutes from now.

b.) sudo shutdown -r +50: This command will reboot the system after 50 minutes from now.

c.) sudo shutdown -h +60: This command will shutdown the system after 60 minutes from now i.e. equivalent to poweroff.

d.) sudo shutdown -c: This command will cancel the scheduled shutdown.

e.) sudo shutdown -Fr +100: It will force the filesystem check during reboot.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ sudo shutdown +40
Shutdown scheduled for Thu 2020-08-27 17:56:09 IST, use 'shutdown -c' to cancel.
vinayak@vinayak-Swift-SF315-52G:~$ sudo shutdown -r +50
Shutdown scheduled for Thu 2020-08-27 18:06:44 IST, use 'shutdown -c' to cancel.
vinayak@vinayak-Swift-SF315-52G:~$ sudo shutdown -h +60
Shutdown scheduled for Thu 2020-08-27 18:17:34 IST, use 'shutdown -c' to cancel.
vinayak@vinayak-Swift-SF315-52G:~$ sudo shutdown -Fr +100
Shutdown scheduled for Thu 2020-08-27 18:58:48 IST, use 'shutdown -c' to cancel.
vinayak@vinayak-Swift-SF315-52G:~$ sudo shutdown -c
vinayak@vinayak-Swift-SF315-52G:~$ 
```

19. ftp command

The **ftp** command runs the classical command-line file transfer client, FTP. It's an interactive text user interface for using the ARPANET standard File Transfer Protocol. It can be used to transfer files to and from a remote network.

a.) ftp ipaddress: Used to open a ftp connection to a remote system.

b.)help or ?: list all available FTP commands.

c.) cd: change directory on the remote machine.

- d.) **lcd:** change directory on the local machine.
- e.) **ls:** list the names of the files and directories in the current remote directory.
- f.) **mkdir:** create a new directory within the current remote directory.
- g.) **pwd:** print the current working directory on the remote machine.
- h.) **delete:** remove a file in the current remote directory.
- i.) **rmdir:** remove a directory in the current remote directory.
- j.) **get:** copy one file from the remote to the local machine.
- k.) **mget:** copy multiple files from the remote to the local machine.
- l.) **put:** copy one file from the local to the remote machine.
- m.) **mput:** copy one file from the local to the remote machine.

```
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~# service vsftpd start
root@vinayak-Swift-SF315-52G:~# ftp localhost
Connected to localhost.
220 (vsFTPd 3.0.3)
Name (localhost:vinayak): vinayak
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> pwd
257 "/home/vinayak" is the current directory
ftp> cd Documents
250 Directory successfully changed.
ftp> cd "Operating Systems"
250 Directory successfully changed.
ftp> mkdir FTPcheck
550 Permission denied.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r--    1 1000      1000      1497 Aug 30 14:27 Input.txt
drwxrwxr-x   2 1000      1000      4096 Aug 30 13:39 Lab
-rw-rw-r--    1 1000      1000  12839918 Aug 30 15:43 Lab Assignment 1.odt
drwxrwxr-x   2 1000      1000      4096 Aug 30 15:43 Theory
-rw-rw-r--    1 1000      1000      591  Aug 30 14:28 sample.txt
-rw-rw-r--    1 1000      1000      910  Aug 30 15:46 sample1.txt
-rw-rw-r--    1 1000      1000      591  Aug 30 13:38 sample2.txt
226 Directory send OK.
```

20. crontab command

The crontab is a list of commands that we want to run on a regular schedule, and also the name of the command used to manage that list. Crontab stands for “cron table,” because it uses the job scheduler cron to execute tasks; cron is the system process which will automatically perform tasks for us according to a set schedule. The schedule is called the crontab, which is also the name of the program used to edit that schedule.

a.) **crontab -u user -l:** View crontab user entry for specific user.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ crontab -u vinayak -l
no crontab for vinayak
vinayak@vinayak-Swift-SF315-52G:~$ □
```

b.) **crontab -e:** Schedule a crontab entry.

File Edit View Search Terminal Help

```
vinayak@vinayak-Swift-SF315-52G:~$ crontab -e
no crontab for vinayak - using an empty one
```

vinayak@vinayak-Swift-SF315-52G: ~

File Edit View Search Terminal Help

GNU nano 4.8 /tmp/crontab.wf33A0/crontab

```
#  
# To define the time you can provide concrete values for  
# minute (m), hour (h), day of month (dom), month (mon),  
# and day of week (dow) or use '*' in these fields (for 'any').  
#  
# Notice that tasks will be started based on the cron's system  
# daemon's notion of time and timezones.  
#  
# Output of the crontab jobs (including errors) is sent through  
# email to the user the crontab file belongs to (unless redirected).  
#  
# For example, you can run a backup of all your user accounts  
# at 5 a.m every week with:  
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/  
#  
# For more information see the manual pages of crontab(5) and cron(8)  
#  
# m h dom mon dow command  
*/10 * * * * /home/ramesh/check-disk-space
```

[Wrote 25 lines]

^{^G} Get Help ^{^O} Write Out ^{^W} Where Is ^{^K} Cut Text ^{^J} Justify ^{^C} Cur Pos
^{^X} Exit ^{^R} Read File ^{^V} Replace ^{^U} Paste Text ^{^T} To Spell [^] Go To Line

File Edit View Search Terminal Help

```
vinayak@vinayak-Swift-SF315-52G:~$ crontab -e
no crontab for vinayak - using an empty one
crontab: installing new crontab
vinayak@vinayak-Swift-SF315-52G:~$
```

```

crontab: installing new crontab
vinayak@vinayak-Swift-SF315-52G:~$ crontab -u vinayak -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
*/10 * * * * /home/ramesh/check-disk-space
vinayak@vinayak-Swift-SF315-52G:~$ 

```

21. service command

The service command is used to run a **System V init script**. Usually all system V init scripts are stored in **/etc/init.d** directory and service command can be used to start, stop, and restart the daemons and other services under Linux. All scripts in /etc/init.d accepts and supports at least the start, stop, and restart commands.

a.) service bluetooth status: It checks the status of a service (bluetooth).

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ service bluetooth status
● bluetooth.service - Bluetooth service
  Loaded: loaded (/lib/systemd/system/bluetooth.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2020-08-29 09:43:53 IST; 6h ago
    Docs: man:bluetoothd(8)
   Main PID: 915 (bluetoothd)
     Status: "Running"
      Tasks: 1 (limit: 9303)
     Memory: 2.5M
      CGroup: /system.slice/bluetooth.service
              └─915 /usr/lib/bluetooth/bluetoothd

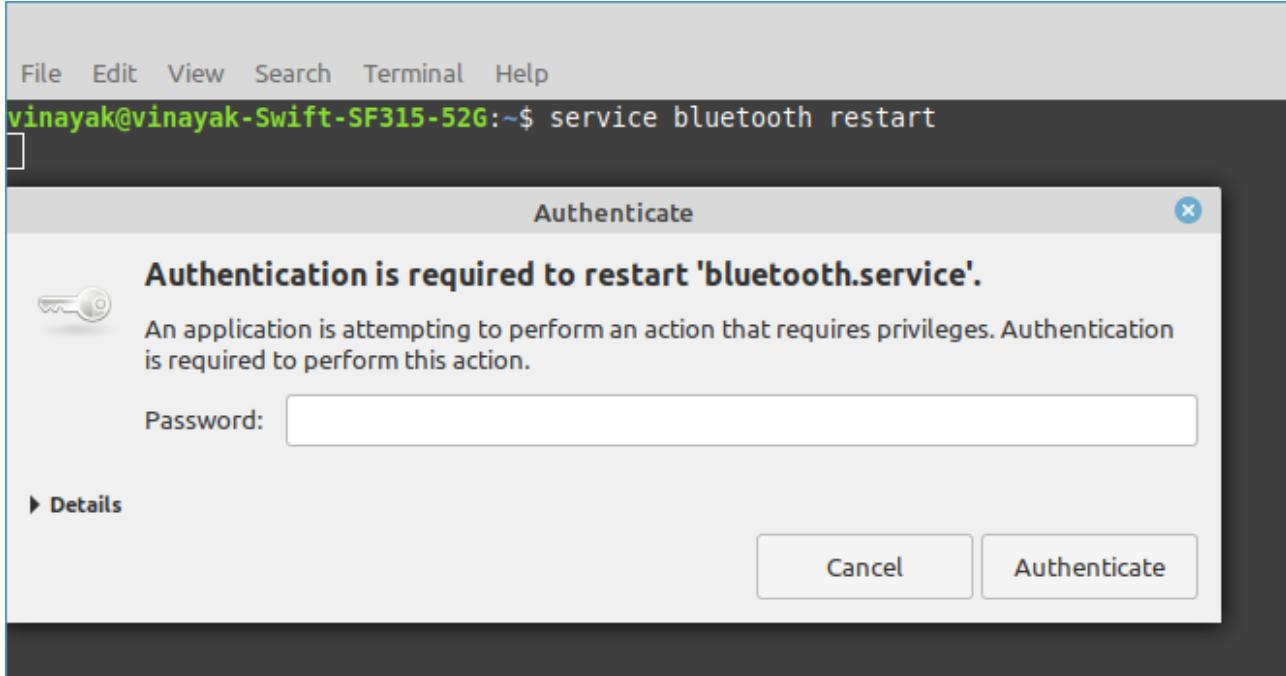
Aug 29 09:43:53 vinayak-Swift-SF315-52G bluetoothd[915]: Starting SDP server
Aug 29 09:43:53 vinayak-Swift-SF315-52G bluetoothd[915]: Bluetooth management interface 1.14 initialized
Aug 29 09:43:54 vinayak-Swift-SF315-52G bluetoothd[915]: Failed to set mode: Blocked through rfkill (0x12)
Aug 29 09:44:15 vinayak-Swift-SF315-52G bluetoothd[915]: Endpoint registered: sender=:1.50 path=/MediaEndpoint/A2DPSink/sbc
Aug 29 09:44:15 vinayak-Swift-SF315-52G bluetoothd[915]: Endpoint registered: sender=:1.50 path=/MediaEndpoint/A2DPSource/sbc
Aug 29 09:44:25 vinayak-Swift-SF315-52G bluetoothd[915]: Endpoint unregistered: sender=:1.50 path=/MediaEndpoint/A2DPSink/sbc
Aug 29 09:44:25 vinayak-Swift-SF315-52G bluetoothd[915]: Endpoint unregistered: sender=:1.50 path=/MediaEndpoint/A2DPSource/sbc
Aug 29 09:44:26 vinayak-Swift-SF315-52G bluetoothd[915]: Endpoint registered: sender=:1.79 path=/MediaEndpoint/A2DPSink/sbc
Aug 29 09:44:26 vinayak-Swift-SF315-52G bluetoothd[915]: Endpoint registered: sender=:1.79 path=/MediaEndpoint/A2DPSource/sbc
Aug 29 09:44:42 vinayak-Swift-SF315-52G bluetoothd[915]: Failed to set mode: Blocked through rfkill (0x12)
vinayak@vinayak-Swift-SF315-52G:~$ 

```

b.) service --status-all: It checks the status of all services.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ service --status-all
[ + ] acpid
[ - ] alsa-utils
[ - ] anacron
[ + ] apparmor
[ + ] avahi-daemon
[ + ] bluetooth
[ - ] console-setup.sh
[ + ] cron
[ - ] cryptdisks
[ - ] cryptdisks-early
[ + ] cups
[ + ] cups-browsed
[ + ] dbus
[ - ] dns-clean
[ + ] grub-common
[ + ] hddtemp
[ - ] hwclock.sh
[ + ] irqbalance
[ + ] kerneloops
[ - ] keyboard-setup.sh
[ + ] kmod
[ + ] lightdm
[ + ] lm-sensors
[ - ] lvm2
[ - ] lvm2-lvmpolld
[ - ] mintsystem
[ + ] network-manager
[ + ] networking
[ + ] openvpn
[ - ] plymouth
[ - ] plymouth-log
[ - ] pppd-dns
[ + ] procps
[ - ] pulseaudio-enable-autospawn
[ - ] rsync
[ + ] rsyslog
[ - ] saned
[ - ] speech-dispatcher
[ + ] udev
[ + ] ufw
[ - ] uuidd
[ - ] x11-common
vinayak@vinayak-Swift-SF315-52G:~$ 
```

c.) **service bluetooth restart:** It restarts a service (bluetooth).



22. ps command

The ps command, short for Process Status, is a command line utility that is used to display or view information related to the processes running in a Linux system.

As we all know, Linux is a multitasking and multiprocessing system. Therefore, multiple processes can run concurrently without affecting each other.

The ps command lists current running processes alongside their PIDs and other attributes.

a.) **ps:** Shows the processes for the current shell –

A screenshot of a terminal window showing the output of the 'ps' command. The terminal bar at the top shows the path 'File Edit View Search Terminal Help' and the command 'vinayak@vinayak-Swift-SF315-52G:~\$ ps'. The output of the command is:

PID	TTY	TIME	CMD
8746	pts/0	00:00:00	bash
9983	pts/0	00:00:00	ps

vinayak@vinayak-Swift-SF315-52G:~\$

Result contains four columns of information.

Where,

PID – the unique process ID

TTY – terminal type that the user is logged into

TIME – amount of CPU in minutes and seconds that the process has been running

CMD – name of the command that launched the process.

Note – Sometimes when we execute ps command, it shows TIME as 00:00:00. It is nothing but the total accumulated CPU utilization time for any process and 00:00:00 indicates no CPU time has been given by the kernel till now.

b.) ps -A or ps -e: Show all the current running process.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -A
  PID TTY      TIME CMD
    1 ?        00:00:01 systemd
    2 ?        00:00:00 kthreadd
    3 ?        00:00:00 rcu_gp
    4 ?        00:00:00 rcu_par_gp
    6 ?        00:00:00 kworker/0:0H-kblockd
    9 ?        00:00:00 mm_percpu_wq
   10 ?       00:00:00 ksoftirqd/0
   11 ?       00:00:15 rcu_sched
   12 ?       00:00:00 migration/0
   13 ?       00:00:00 idle_inject/0
   14 ?       00:00:00 cpuhp/0
   15 ?       00:00:00 cpuhp/1
   16 ?       00:00:00 idle_inject/1
   17 ?       00:00:00 migration/1
   18 ?       00:00:00 ksoftirqd/1
   20 ?       00:00:00 kworker/1:0H-kblockd
   21 ?       00:00:00 cpuhp/2
   22 ?       00:00:00 idle_inject/2
   23 ?       00:00:00 migration/2
   24 ?       00:00:00 ksoftirqd/2
   26 ?       00:00:00 kworker/2:0H-events_highpri
   27 ?       00:00:00 cpuhp/3
   28 ?       00:00:00 idle_inject/3
   29 ?       00:00:00 migration/3
   30 ?       00:00:00 ksoftirqd/3
   32 ?       00:00:00 kworker/3:0H-events_highpri
   33 ?       00:00:00 cpuhp/4
   34 ?       00:00:00 idle_inject/4
   35 ?       00:00:00 migration/4
   36 ?       00:00:00 ksoftirqd/4
   38 ?       00:00:00 kworker/4:0H-events_highpri
   39 ?       00:00:00 cpuhp/5
   40 ?       00:00:00 idle_inject/5
   41 ?       00:00:00 migration/5
```

c.) ps -u root: List processes running as root.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -u root
  PID TTY      TIME CMD
    1 ?        00:00:01 systemd
    2 ?        00:00:00 kthreadd
    3 ?        00:00:00 rcu_gp
    4 ?        00:00:00 rcu_par_gp
    6 ?        00:00:00 kworker/0:0H-kblockd
    9 ?        00:00:00 mm_percpu_wq
   10 ?       00:00:00 ksoftirqd/0
   11 ?       00:00:15 rcu_sched
   12 ?       00:00:00 migration/0
   13 ?       00:00:00 idle_inject/0
   14 ?       00:00:00 cpuhp/0
   15 ?       00:00:00 cpuhp/1
   16 ?       00:00:00 idle_inject/1
   17 ?       00:00:00 migration/1
   18 ?       00:00:00 ksoftirqd/1
   20 ?       00:00:00 kworker/1:0H-kblockd
   21 ?       00:00:00 cpuhp/2
   22 ?       00:00:00 idle_inject/2
   23 ?       00:00:00 migration/2
   24 ?       00:00:00 ksoftirqd/2
   26 ?       00:00:00 kworker/2:0H-kblockd
   27 ?       00:00:00 cpuhp/3
   28 ?       00:00:00 idle_inject/3
   29 ?       00:00:00 migration/3
   30 ?       00:00:00 ksoftirqd/3
   32 ?       00:00:00 kworker/3:0H-events_highpri
   33 ?       00:00:00 cpuhp/4
   34 ?       00:00:00 idle_inject/4
   35 ?       00:00:00 migration/4
```

d.) **ps -u username:** List all the process running under given username.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -u vinayak
  PID TTY      TIME CMD
 1385 ?    00:00:00 systemd
 1388 ?    00:00:00 (sd-pam)
 1412 ?    00:15:46 pulseaudio
 1415 ?    00:00:00 gnome-keyring-d
 1418 ?    00:00:01 cinnamon-sessio
 1426 ?    00:00:01 dbus-daemon
 1428 ?    00:00:00 gvfsd
 1433 ?    00:00:00 gvfsd-fuse
 1506 ?    00:00:00 ssh-agent
 1518 ?    00:00:00 at-spi-bus-laun
 1523 ?    00:00:00 dbus-daemon
 1526 ?    00:00:02 at-spi2-registr
 1538 ?    00:00:00 csd-sound
 1539 ?    00:00:00 csd-cursor
 1540 ?    00:00:00 csd-housekeepin
 1541 ?    00:00:00 csd-print-notif
 1543 ?    00:00:00 csd-media-keys
 1544 ?    00:00:00 csd-orientation
 1554 ?    00:00:00 csd-mouse
 1556 ?    00:00:01 csd-keyboard
 1557 ?    00:00:00 csd-automount
 1560 ?    00:00:00 csd-wacom
 1561 ?    00:00:00 csd-xrandr
 1562 ?    00:00:01 csd-power
 1563 ?    00:00:00 csd-background
 1564 ?    00:00:00 csd-a11y-settin
```

e.) **pid -L PID:** Print the thread of a process corresponding of the PID.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ pidof Xorg
1101
vinayak@vinayak-Swift-SF315-52G:~$ ps -L 1101
  PID   LWP TTY      STAT   TIME COMMAND
 1101   1101  tty7    Ssl+  11:22 /usr/lib/xorg/Xorg -core :0 -seat seat0 -au
 1101   1132  tty7    Ssl+  0:00 /usr/lib/xorg/Xorg -core :0 -seat seat0 -au
 1101   1133  tty7    Ssl+  0:00 /usr/lib/xorg/Xorg -core :0 -seat seat0 -au
 1101   1134  tty7    Ssl+  0:00 /usr/lib/xorg/Xorg -core :0 -seat seat0 -au
 1101   1135  tty7    Ssl+  0:00 /usr/lib/xorg/Xorg -core :0 -seat seat0 -au
 1101   1138  tty7    Ssl+  0:36 /usr/lib/xorg/Xorg -core :0 -seat seat0 -au
vinayak@vinayak-Swift-SF315-52G:~$ 
```

f.) **ps -C processname or pidof processname:** To find the pid of a given process.

```
vinayak@vinayak-Swift-SF31
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -C Xorg
  PID TTY      TIME CMD
 1101  tty7    00:12:48 Xorg
vinayak@vinayak-Swift-SF315-52G:~$ pidof Xorg
1101
vinayak@vinayak-Swift-SF315-52G:~$ 
```

g.) **ps -fp PID:** This is used to find process from PID.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ ps -fp 38  
UID          PID      PPID    C STIME TTY          TIME CMD  
root         38        2  0 14:55 ?          00:00:00 [kworker/4:0H-events_highpri]  
vinayak@vinayak-Swift-SF315-52G:~$ 
```

23. free command

In LINUX, there exists a command line utility for this and that is **free** command which displays the total amount of free space available along with the amount of memory used and swap memory in the system, and also the buffers used by the kernel.

a.) free: free command without any option shows the used and free space of swap and physical memory in KB.

When no option is used then free command produces the columnar output as shown where column:

1. **total** displays the total installed memory (MemTotal and SwapTotal *i.e* present in /proc/meminfo).

2. **used** displays the used memory.

3. **free** displays the unused memory.

4. **shared** displays the memory used by tmpfs(*Shmen i.e*present in /proc/meminfo and displays zero in case not available).

5. **buffers** displays the memory used by kernel buffers.

6. **cached** displays the memory used by the page cache and slabs(Cached and Slab available in /proc/meminfo).

7. **buffers/cache** displays the sum of buffers and cache.

b.) free -b: It just displays the output in unit bytes.

c.) free -k: This option displays the result in kilobytes.

d.) free -t: This option displays an additional line containing the total of the total, used and free columns.

e.) free -s 3 -c 3: This option allows us to display the output of free command after a set time gap given by the user. With -s we can only specify the time

gap but not the number of times we want the output to be displayed. For this, -c is used along with -s specifying the number of times the output will be displayed.

f.) free -h: It shows all output fields automatically scaled to shortest three digit unit and display the units of print out. Following units are used.

B = bytes
Ki = kibibyte
Mi = mebibyte
Gi = gibibyte
Ti = tebibyte
Pi = pebibyte

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ free
      total        used        free      shared  buff/cache   available
Mem:    8030048     2783860     2090532      119072      3155656     4817608
Swap:  2097148          0     2097148
vinayak@vinayak-Swift-SF315-52G:~$ free -b
      total        used        free      shared  buff/cache   available
Mem:  8222769152  2848866304  2142511104  121929728  3231391744  4935036928
Swap: 2147479552          0  2147479552
vinayak@vinayak-Swift-SF315-52G:~$ free -k
      total        used        free      shared  buff/cache   available
Mem:    8030048     2782112     2092240      119072      3155696     4819324
Swap:  2097148          0     2097148
vinayak@vinayak-Swift-SF315-52G:~$ free -t
      total        used        free      shared  buff/cache   available
Mem:    8030048     2783292     2091052      119072      3155704     4818176
Swap:  2097148          0     2097148
Total: 10127196     2783292     4188200
vinayak@vinayak-Swift-SF315-52G:~$ free -s 3 -c 3
      total        used        free      shared  buff/cache   available
Mem:    8030048     2782268     2092052      119072      3155728     4819180
Swap:  2097148          0     2097148
      total        used        free      shared  buff/cache   available
Mem:    8030048     2779124     2095196      119072      3155728     4822332
Swap:  2097148          0     2097148
      total        used        free      shared  buff/cache   available
Mem:    8030048     2779236     2095076      119072      3155736     4822212
Swap:  2097148          0     2097148
vinayak@vinayak-Swift-SF315-52G:~$ free -h
      total        used        free      shared  buff/cache   available
Mem:    7.7Gi       2.7Gi       2.0Gi      116Mi      3.0Gi       4.6Gi
Swap:   2.0Gi        0B        2.0Gi
vinayak@vinayak-Swift-SF315-52G:~$ 
```

24. top command

top command is used to show the Linux processes. It provides a dynamic real-time view of the running system. Usually, this command shows the

summary information of the system and the list of processes or threads which are currently managed by the Linux Kernel.

a.) top: Used to display the top processes running.

```
vinayak@vinayak-S:~$ top - 16:49:44 up 7:06, 1 user, load average: 0.37, 0.60, 0.55
Tasks: 326 total, 2 running, 323 sleeping, 0 stopped, 1 zombie
%Cpu(s): 4.5 us, 1.7 sy, 0.0 ni, 93.5 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem : 7841.8 total, 961.7 free, 4356.6 used, 2523.6 buff/cache
MiB Swap: 2048.0 total, 2043.7 free, 4.3 used. 2919.1 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
1631 vinayak 20 0 3490484 260092 103868 S 19.9 3.2 23:14.70 cinnamon
1094 root 20 0 957044 348032 137260 S 9.3 4.3 20:53.98 Xorg
1366 vinayak 20 0 1932720 20732 16644 S 4.3 0.3 11:57.70 pulseaudio
12118 vinayak 20 0 799580 97388 63172 S 3.3 1.2 6:50.18 cinnamon-settin
4362 vinayak 20 0 780108 76192 33092 S 2.0 0.9 0:21.85 gnome-screensho
328 root -51 0 0 0 0 S 1.7 0.0 1:57.83 irq/109-SYNA7DB
1117 root -51 0 0 0 0 S 1.3 0.0 2:31.43 irq/130-nvidia
1998 vinayak 20 0 1147876 364320 139868 R 1.0 4.5 12:42.58 chrome
2042 vinayak 20 0 381392 113424 66184 S 1.0 1.4 4:56.94 chrome
7680 vinayak 20 0 586460 129184 86760 S 1.0 1.6 2:05.00 Discord
7696 vinayak 20 0 596508 136604 90768 S 1.0 1.7 2:26.06 Discord
7848 vinayak 20 0 918248 207896 110520 S 1.0 2.6 2:11.68 Discord
7849 vinayak 20 0 997952 261212 108872 S 1.0 3.3 29:50.54 Discord
10565 vinayak 20 0 12.6g 212828 102432 S 1.0 2.7 0:44.02 chrome
21064 vinayak 20 0 462612 40080 30588 S 0.7 0.5 0:07.10 gnome-terminal-
24432 vinayak 20 0 16348 4068 3204 R 0.7 0.1 0:00.15 top
596 root -51 0 0 0 0 S 0.3 0.0 0:42.71 irq/128-iwlwifi
1119 root 20 0 0 0 0 S 0.3 0.0 0:23.39 nv_queue
3022 vinayak 20 0 9112616 344500 106916 S 0.3 4.3 3:29.03 chrome
3838 vinayak 20 0 9042880 280444 114744 S 0.3 3.5 4:09.24 chrome
1 root 20 0 169684 11616 8304 S 0.0 0.1 0:02.63 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.02 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_par_gp
6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/0:0H-kblockd
9 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 mm_percpu_wq
10 root 20 0 0 0 0 S 0.0 0.0 0:00.43 ksoftirqd/0
11 root 20 0 0 0 0 I 0.0 0.0 0:18.03 rcu_sched
12 root rt 0 0 0 0 S 0.0 0.0 0:00.10 migration/0
13 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/0
14 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/0
15 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/1
16 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/1
17 root rt 0 0 0 0 S 0.0 0.0 0:00.28 migration/1
18 root 20 0 0 0 0 S 0.0 0.0 0:00.22 ksoftirqd/1
20 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/1:0H-kblockd
21 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/2
22 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/2
23 root rt 0 0 0 0 S 0.0 0.0 0:00.29 migration/2
24 root 20 0 0 0 0 S 0.0 0.0 0:00.12 ksoftirqd/2
26 root 0 -20 0 0 0 I 0.0 0.0 0:00.08 kworker/2:0H-kblockd
27 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/3
28 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/3
29 root rt 0 0 0 0 S 0.0 0.0 0:00.29 migration/3
30 root 20 0 0 0 0 S 0.0 0.0 0:00.07 ksoftirqd/3
32 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/3:0H-events_highpri
33 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/4
34 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/4
35 root rt 0 0 0 0 S 0.0 0.0 0:00.29 migration/4
36 root 20 0 0 0 0 S 0.0 0.0 0:00.10 ksoftirqd/4
38 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/4:0H-events_highpri
```

Here,

PID: Shows task's unique process id.

USER: User name of owner of task.

PR: Stands for priority of the task.

NI: Represents a Nice Value of task. A Negative nice value implies higher priority, and positive Nice value means lower priority.

VIRT: Total virtual memory used by the task.

SHR: Represents the amount of shared memory used by a task.

%CPU: Represents the CPU usage.

%MEM: Shows the Memory usage of task.

TIME+: CPU Time, the same as ‘TIME’, but reflecting more granularity through hundredths of a second.

b.) top -u username: It will display specific user process detail.

```

File Edit View Search Terminal Help
top - 16:56:12 up 7:12, 1 user, load average: 1.74, 0.97, 0.69
Tasks: 321 total, 1 running, 319 sleeping, 0 stopped, 1 zombie
%Cpu(s): 5.7 us, 2.1 sy, 0.0 ni, 91.9 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem : 7841.8 total, 987.6 free, 4333.5 used, 2520.8 buff/cache
MiB Swap: 2048.0 total, 2043.7 free, 4.3 used. 2949.1 avail Mem

      PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 1366 vinayak    20   0 1932720 20732 16644 S  4.6  0.3 12:15.93 pulseaudio
12118 vinayak    20   0 799580 97388 63172 S  4.3  1.2  7:07.47 cinnamon-settin
23435 vinayak    20   0 4743444 187400 94600 S  4.0  2.3  0:20.07 chrome
4362 vinayak    20   0 781504 77508 33092 S  3.3  1.0  0:22.53 gnome-screensho
1998 vinayak    20   0 1145828 364488 139744 S  2.6  4.5 12:49.33 chrome
7680 vinayak    20   0 587228 128804 86760 S  1.3  1.6  2:08.62 Discord
7696 vinayak    20   0 597532 136652 90768 S  1.3  1.7  2:29.58 Discord
7848 vinayak    20   0 918248 208004 110520 S  1.3  2.6  2:15.50 Discord
7849 vinayak    20   0 997952 262164 108872 S  1.3  3.3  29:54.68 Discord
2042 vinayak    20   0 381392 113464 66180 S  1.0  1.4  5:01.58 chrome
23498 vinayak   20   0 4641976 117124 79128 S  1.0  1.5  0:03.95 chrome
3022 vinayak    20   0 4908572 336256 106916 S  0.7  4.2  3:33.11 chrome
3838 vinayak    20   0 9042600 277540 114488 S  0.7  3.5  4:12.00 chrome
18058 vinayak   20   0 4689216 132584 81644 S  0.7  1.7  0:22.77 chrome
24687 vinayak 20   0 16348 4120 3256 R  0.7  0.1 0:00.09 top
10376 vinayak   20   0 4705544 149512 91492 S  0.3  1.9  1:14.22 chrome
21064 vinayak   20   0 462612 40080 30588 S  0.3  0.5  0:08.98 gnome-terminal-
1346 vinayak    20   0 18916 10156 8160 S  0.0  0.1  0:00.21 systemd
1347 vinayak    20   0 171076 3828 0 S  0.0  0.0  0:00.00 (sd-pam)
1369 vinayak    20   0 240144 7448 6480 S  0.0  0.1  0:00.24 gnome-keyring-d
1374 vinayak    20   0 376696 26500 19652 S  0.0  0.3  0:00.91 cinnamon-sessio
1381 vinayak    20   0 8708 5660 3712 S  0.0  0.1  0:04.53 dbus-daemon
1384 vinayak    20   0 239780 7836 6736 S  0.0  0.1  0:00.12 gvfsd
1389 vinayak    20   0 382056 8440 7552 S  0.0  0.1  0:00.01 gvfsd-fuse
1462 vinayak    20   0 6032 452 0 S  0.0  0.0  0:00.11 ssh-agent
1474 vinayak    20   0 309868 9380 8444 S  0.0  0.1  0:00.01 at-spi-bus-laun
1479 vinayak    20   0 7612 4364 3632 S  0.0  0.1  0:01.12 dbus-daemon
1482 vinayak    20   0 162888 6696 6000 S  0.0  0.1  0:09.08 at-spi2-registr
1494 vinayak    20   0 609140 29712 20536 S  0.0  0.4  0:01.42 csd-sound
1495 vinayak    20   0 193560 22864 17344 S  0.0  0.3  0:00.33 csd-cursor
1496 vinayak    20   0 267704 22608 17100 S  0.0  0.3  0:01.31 csd-housekeepin
1497 vinayak    20   0 275704 26360 20028 S  0.0  0.3  0:00.33 csd-print-notif
1500 vinayak    20   0 945148 25752 19376 S  0.0  0.3  0:00.54 csd-media-keys
1503 vinayak    20   0 415152 23208 17624 S  0.0  0.3  0:00.30 csd-orientation
1513 vinayak    20   0 267680 23308 17568 S  0.0  0.3  0:00.37 csd-mouse
1514 vinayak    20   0 298968 24964 19120 S  0.0  0.3  0:07.30 csd-keyboard
1515 vinayak    20   0 341552 23392 17704 S  0.0  0.3  0:00.26 csd-automount
1516 vinayak    20   0 305892 25364 19264 S  0.0  0.3  0:00.31 csd-wacom
1517 vinayak    20   0 342380 23960 18092 S  0.0  0.3  0:00.35 csd-xrandr
1518 vinayak    20   0 456060 36952 29860 S  0.0  0.5  0:02.64 csd-power
1519 vinayak    20   0 335172 33028 27032 S  0.0  0.4  0:00.46 csd-background
1521 vinayak    20   0 267728 23028 17468 S  0.0  0.3  0:00.31 csd-ally-settin
1530 vinayak    20   0 196608 25964 17232 S  0.0  0.3  0:00.40 csd-clipboard
1532 vinayak    20   0 267360 22752 17208 S  0.0  0.3  0:00.28 csd-screensaver
1533 vinayak    20   0 269412 24920 18756 S  0.0  0.3  0:00.42 csd-xsettings
1542 vinayak    20   0 417104 24384 18576 S  0.0  0.3  0:00.37 csd-color
1556 vinayak    20   0 267376 22812 17352 S  0.0  0.3  0:00.29 csd-ally-keyboa
1561 vinayak    20   0 156412 5848 5012 S  0.0  0.1  0:00.09 dconf-service
1586 vinayak    20   0 317552 11248 9584 S  0.0  0.1  0:00.17 gvfs-udisks2-vo
1596 vinayak    20   0 235688 5832 5212 S  0.0  0.1  0:00.03 gvfs-mtp-volume
1601 vinayak    20   0 238092 7288 6456 S  0.0  0.1  0:00.03 gvfs-gphoto2-vo

```

25. df command

The df command (short for disk free), is used to display information related to file systems about total space and available space.

a.) df: It displays the space available on all currently mounted file systems.

b.) df -a: It includes pseudo, duplicate and inaccessible file systems.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            3969420      0   3969420  0% /dev
tmpfs           803008     1740   801268  1% /run
/dev/sda4       116824016  41647032  69199592 38% /
tmpfs           4015024    77216  3937808  2% /dev/shm
tmpfs           5120        4    5116  1% /run/lock
tmpfs           4015024      0   4015024  0% /sys/fs/cgroup
/dev/sda2       779772     8024   771748  2% /boot/efi
tmpfs           803004      88   802916  1% /run/user/1000
vinayak@vinayak-Swift-SF315-52G:~$ df -a
Filesystem      1K-blocks    Used Available Use% Mounted on
sysfs            0          0      0      - /sys
proc             0          0      0      - /proc
udev            3969420      0   3969420  0% /dev
devpts           0          0      0      - /dev/pts
tmpfs           803008     1740   801268  1% /run
/dev/sda4       116824016  41647044  69199580 38% /
securityfs      0          0      0      - /sys/kernel/security
tmpfs           4015024    77216  3937808  2% /dev/shm
tmpfs           5120        4    5116  1% /run/lock
tmpfs           4015024      0   4015024  0% /sys/fs/cgroup
cgroup2          0          0      0      - /sys/fs/cgroup/unified
cgroup           0          0      0      - /sys/fs/cgroup/systemd
pstore           0          0      0      - /sys/fs/pstore
efivars          0          0      0      - /sys/firmware/efi/efivars
none             0          0      0      - /sys/fs/bpf
cgroup           0          0      0      - /sys/fs/cgroup/cpuset
cgroup           0          0      0      - /sys/fs/cgroup/memory
cgroup           0          0      0      - /sys/fs/cgroup/cpu,cpuacct
cgroup           0          0      0      - /sys/fs/cgroup/net_cls,net_prio
cgroup           0          0      0      - /sys/fs/cgroup/pids
cgroup           0          0      0      - /sys/fs/cgroup/freezer
cgroup           0          0      0      - /sys/fs/cgroup/rdma
cgroup           0          0      0      - /sys/fs/cgroup/perf_event
cgroup           0          0      0      - /sys/fs/cgroup/blkio
cgroup           0          0      0      - /sys/fs/cgroup/hugetlb
cgroup           0          0      0      - /sys/fs/cgroup/devices
systemd-1         -         -      -      - /proc/sys/fs/binfmt_misc
debugfs          0          0      0      - /sys/kernel/debug
hugepages         0          0      0      - /dev/hugepages
tracesfs         0          0      0      - /sys/kernel/tracing
mqueue            0          0      0      - /dev/mqueue
fusectl           0          0      0      - /sys/fs/fuse/connections
configfs          0          0      0      - /sys/kernel/config
/dev/sda2       779772     8024   771748  2% /boot/efi
tmpfs           803004      88   802916  1% /run/user/1000
gvfsd-fuse        0          0      0      - /run/user/1000/gvfs
/dev/fuse          0          0      0      - /run/user/1000/doc
binfmt_misc        0          0      0      - /proc/sys/fs/binfmt_misc
vinayak@vinayak-Swift-SF315-52G:~$
```

c.) df -h directory: This is used to make df command display the output in human-readable format.

d.) df -H directory: This option is same as -h but it use powers of 1000 instead of 1024.

e.) df -total: This option is used to produce total for a size, used and available columns in the output.

f.) df -T directory: With the help of this option, we will be able to see the corresponding type of the file system.

g.) df -m: This option is used to display information of all file system usage in MB(Mega Byte).

h.) df -i: This option is used to display inode information in the output.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ df -h /home/vinayak
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda4       112G   40G   66G  38% /
vinayak@vinayak-Swift-SF315-52G:~$ df -H /home/vinayak
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda4       120G   43G   71G  38% /
vinayak@vinayak-Swift-SF315-52G:~$ df --total
Filesystem    1K-blocks  Used Available Use% Mounted on
udev          3969420     0  3969420  0% /dev
tmpfs          803008   1740  801268  1% /run
/dev/sda4     116824016 41660084 69186540 38% /
tmpfs          4015024   77472  3937552  2% /dev/shm
tmpfs           5120      4   5116  1% /run/lock
tmpfs          4015024     0  4015024  0% /sys/fs/cgroup
/dev/sda2      779772   8024   771748  2% /boot/efi
tmpfs          803004     76   802928  1% /run/user/1000
total        131214388 41747400 83489596 34% -
vinayak@vinayak-Swift-SF315-52G:~$ df -T /home/vinayak
Filesystem  Type 1K-blocks  Used Available Use% Mounted on
/dev/sda4   ext4  116824016 41660076 69186548 38% /
vinayak@vinayak-Swift-SF315-52G:~$ df -m
Filesystem    1M-blocks  Used Available Use% Mounted on
udev          3877      0   3877  0% /dev
tmpfs          785      2   783  1% /run
/dev/sda4     114086  40661   67588  38% /
tmpfs          3921     88   3834  3% /dev/shm
tmpfs           5      1     5  1% /run/lock
tmpfs          3921      0   3921  0% /sys/fs/cgroup
/dev/sda2      762      8   754  2% /boot/efi
tmpfs          785      1   785  1% /run/user/1000
vinayak@vinayak-Swift-SF315-52G:~$ df -i
Filesystem    Inodes IUsed IFree IUse% Mounted on
udev          992355   616  991739  1% /dev
tmpfs          1003756  1030 1002726  1% /run
/dev/sda4     7454720 630256 6824464  9% /
tmpfs          1003756   97 1003659  1% /dev/shm
tmpfs          1003756    4 1003752  1% /run/lock
tmpfs          1003756   18 1003738  1% /sys/fs/cgroup
/dev/sda2      0      0     0  - /boot/efi
tmpfs          1003756   89 1003667  1% /run/user/1000
vinayak@vinayak-Swift-SF315-52G:~$ df -T
Filesystem  Type 1K-blocks  Used Available Use% Mounted on
udev        devtmpfs  3969420     0  3969420  0% /dev
tmpfs        tmpfs   803008   1740  801268  1% /run
/dev/sda4   ext4   116824016 41636876 69209748 38% /
tmpfs        tmpfs   4015024   93064  3921960  3% /dev/shm
tmpfs        tmpfs   5120      4   5116  1% /run/lock
tmpfs        tmpfs   4015024     0  4015024  0% /sys/fs/cgroup
/dev/sda2   vfat   779772   8024   771748  2% /boot/efi
tmpfs        tmpfs   803004     76   802928  1% /run/user/1000
vinayak@vinayak-Swift-SF315-52G:~$ 
```

26. kill command

kill command in Linux (located in /bin/kill), is a built-in command which is used to terminate processes manually. kill command sends a signal to a process which terminates the process. If the user doesn't specify any signal which is to be sent along with kill command then default TERM signal is sent that terminates the process.

a.) kill -l: It prints all the available signals.

```
vinayak@vinayak-Swift-SF315-52G:~$ kill -l
 1) SIGHUP      2) SIGINT      3) SIGQUIT      4) SIGILL      5) SIGTRAP
 6) SIGABRT     7) SIGBUS      8) SIGFPE       9) SIGKILL     10) SIGUSR1
11) SIGSEGV     12) SIGUSR2     13) SIGPIPE     14) SIGALRM     15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
21) SIGTTIN     22) SIGTTOU     23) SIGURG      24) SIGXCPU     25) SIGXFSZ
26) SIGVTALRM   27) SIGPROF     28) SIGWINCH    29) SIGIO       30) SIGPWR
31) SIGSYS      34) SIGRTMIN    35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
vinayak@vinayak-Swift-SF315-52G:~$
```

b.) **kill pid:** kills process with a given pid.

```
vinayak@vinayak-Swift-SF315-52G:~$ pidof chrome
16408 16402 16317 16309 16283 16281 16273 16171 16133 16027 16022 16016 16015 16
002 15987 15908 15871 15858 15846 15844 15822 15817 15816 15806
vinayak@vinayak-Swift-SF315-52G:~$ kill 16408
vinayak@vinayak-Swift-SF315-52G:~$ pidof chrome
16402 16317 16309 16283 16281 16273 16171 16133 16027 16022 16016 16015 16002 15
987 15908 15871 15858 15846 15844 15822 15817 15816 15806
vinayak@vinayak-Swift-SF315-52G:~$
```

c.) **kill -s signal pid:** It sends a kill signal to kill a process.

```
vinayak@vinayak-Swift-SF315-52G:~$ pidof chrome
16402 16317 16309 16283 16281 16273 16171 16133 16027 16022 16016 16015 16002 15
987 15908 15871 15858 15846 15844 15822 15817 15816 15806
vinayak@vinayak-Swift-SF315-52G:~$ kill -s SIGKILL 16402
vinayak@vinayak-Swift-SF315-52G:~$ pidof chrome
16317 16309 16283 16281 16273 16171 16133 16027 16022 16016 16015 16002 15987 15
908 15871 15858 15846 15844 15822 15817 15816 15806
vinayak@vinayak-Swift-SF315-52G:~$
```

d.) **Kill -s KILL -1:** Kill all processes in one go.

27. rm command

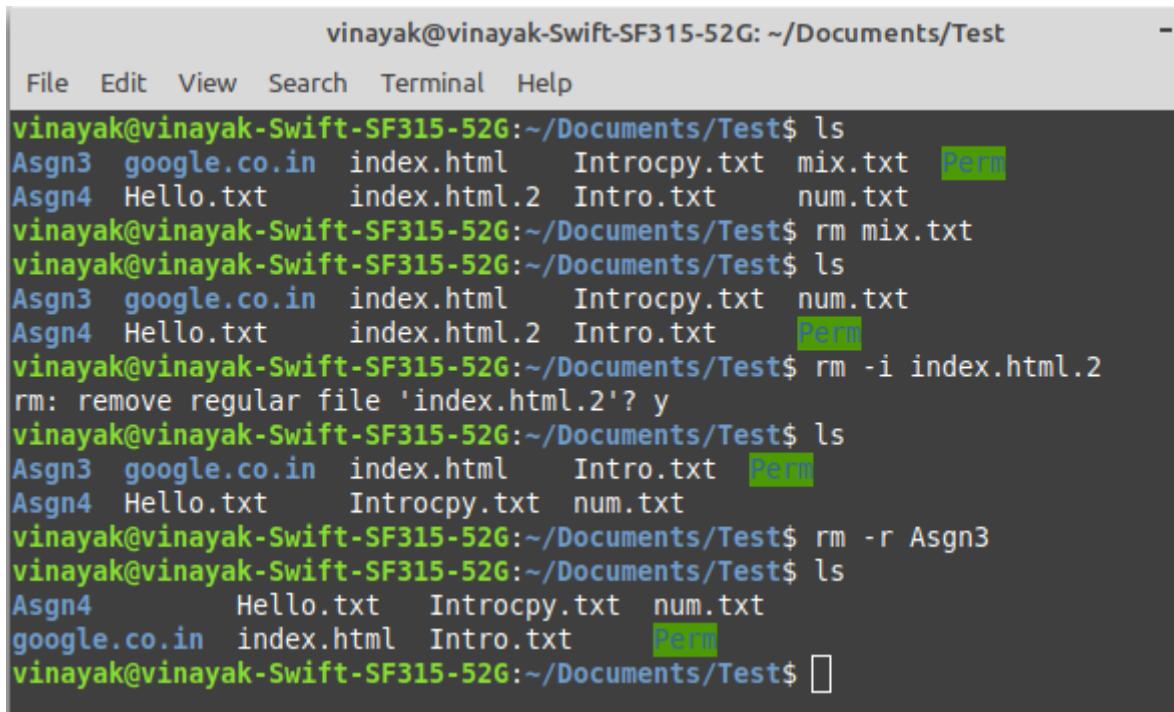
rm stands for remove here. rm command is used to remove objects such as files, directories, symbolic links and so on from the file system like UNIX. To be more precise, rm removes references to objects from the filesystem,

where those objects might have had multiple references (for example, a file with two different names).

a.) **rm filename:** Used to remove file.

b.) **rm -i filename:** get confirmation before removing a file.

c.) **rm -r directoryname:** It recursively removes all files and directories under the given directory. This also removes the given directory itself.



The screenshot shows a terminal window with the following session:

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Test
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn3 google.co.in index.html Introcpy.txt mix.txt Perm
Asgn4 Hello.txt index.html.2 Intro.txt num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ rm mix.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn3 google.co.in index.html Introcpy.txt num.txt
Asgn4 Hello.txt index.html.2 Intro.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ rm -i index.html.2
rm: remove regular file 'index.html.2'? y
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn3 google.co.in index.html Intro.txt Perm
Asgn4 Hello.txt Introcpy.txt num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ rm -r Asgn3
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Hello.txt Introcpy.txt num.txt
google.co.in index.html Intro.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$
```

28. cp command

cp stands for copy. This command is used to copy files or group of files or directory. It creates an exact image of a file on a disk with different file name. cp command require at least two filenames in its arguments.

a.) **cp file1name file2name:** If the command contains two file names, then it copy the contents of 1st file to the 2nd file. If the 2nd file doesn't exist, then first it creates one and content is copied to it. But if it existed then it is simply overwritten without any warning.

b.) **cp file1name file2name directory1:** If the command has one or more arguments, specifying file names and following those arguments, an argument specifying directory name then this command copies each source file to the destination directory with the same name, created if not existed but if already existed then it will be overwritten,

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Hello.txt Introcpy.txt num.txt
google.co.in index.html Intro.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cp num.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Hello.txt Introcpy.txt num.txt Perm
google.co.in index.html Intro.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cp num.txt Num.txt Asgn4
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Hello.txt Introcpy.txt num.txt Perm
google.co.in index.html Intro.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cd Asgn4
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test/Asgn4$ ls
num.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test/Asgn4$ 

```

c.) cp -R Source_dir Destination_dir: If the command contains two directory names, cp copies all files of the source directory to the destination directory, creating any files or directories needed. This mode of operation requires an additional option, typically R, to indicate the recursive copying of directories.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 google.co.in Hello.txt index.html Introcpy.txt Intro.txt num.txt Num.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cd Asgn4
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test/Asgn4$ ls
num.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test/Asgn4$ cd ..
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cp -R Asgn4 Asgn5
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Asgn5 google.co.in Hello.txt index.html Introcpy.txt Intro.txt num.txt Num.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cd Asgn5
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test/Asgn5$ ls
num.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test/Asgn5$ 

```

d.) cp -p file1 file2: Copy file1 to file2 preserving the mode, ownership and timestamp.

e.) cp -i file1 file2: Copy file1 to file2, if file2 exists prompt for confirmation before overwriting it.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Test
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 google.co.in index.html Intro.txt Num.txt
Asgn5 Hello.txt Introcpy.txt num.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cp -p Hello.txt Heyy.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 google.co.in Heyy.txt Introcpy.txt num.txt Perm
Asgn5 Hello.txt index.html Intro.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cp -i Hello.txt Heyy.txt
cp: overwrite 'Heyy.txt'? y
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 google.co.in Heyy.txt Introcpy.txt num.txt Perm
Asgn5 Hello.txt index.html Intro.txt Num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ 

```

29. mv command

mv stands for move. mv is used to move one or more files or directories from one place to another in file system like UNIX. It has two distinct functions:

- (i) It rename a file or folder.
- (ii) It moves group of files to different directory.

No additional space is consumed on a disk during renaming.

a.) mv src_file destn_file: It renames the src_file with destn_file. If the destination file doesn't exist, it will be created. In the above command mv simply replaces the source filename in the directory with the destination filename(new name). If the destination file exist, then it will be overwrite and the source file will be deleted.

b.) mv -i src_file destn_file: the -i option makes the command ask the user for confirmation before moving a file that would overwrite an existing file, we have to press y for confirm moving, any other key leaves the file as it is. This option doesn't work if the file doesn't exist, it simply rename it or move it to new location.

c.) mv -b src_fiel destn_file: With this option it is easier to take a backup of an existing file that will be overwritten as a result of the mv command. This will create a backup file with the tilde character(~) appended to it.

d.) mv -v src_file destn_file: mv -v will print what is happening during file rename, which is useful while specifying shell metacharacters in the file name argument.

```
vinay
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Asgn5 google.co.in Hello.txt Heyy.txt index.html Introcpy.txt Intro.txt Num.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mv Num.txt num.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Asgn5 google.co.in Hello.txt Heyy.txt index.html Introcpy.txt Intro.txt num.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mv -i num.txt Heyy.txt
mv: overwrite 'Heyy.txt'? y
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Asgn5 google.co.in Hello.txt Heyy.txt index.html Introcpy.txt Intro.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mv -b Hello.txt Heyy.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Asgn5 google.co.in Heyy.txt Heyy.txt~ index.html Introcpy.txt Intro.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mv -v Intro.txt Heyy.txt
renamed 'Intro.txt' -> 'Heyy.txt'
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn4 Asgn5 google.co.in Heyy.txt Heyy.txt~ index.html Introcpy.txt Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ 
```

30. cat command

The cat command (short for “concatenate”) is one of the most frequently used command in Linux/Unix, Apple Mac OS X operating systems. cat command allows us to create single or multiple files, view contain of file,

concatenate files and redirect output in terminal or files. It is a standard Unix program used to concatenate and display files. The cat command display file contents to a screen.

- a.) **cat filename:** It will show the content of given filename.
- b.) **cat file1 file2:** It is used to view multiple file name.
- c.) **cat -n filename:** To view contents of a file preceding with line numbers.
- d.) **cat >newfile:** It will create a file named newfile.
- e.) **cat sourcefile > destination file:** It copies the contents of one file to another file.
- f.) **cat file1 >> file2:** This command can append the contents of one file to the end of another file.
- g.) **tac filename:** It will display the content in reverse order.
- h.) **cat -E filename:** It will highlight the end of line.
- i.) **cat "filename1" "filename2" "filename3" > "merged_filename":** It will merge the contents of file in respective order and will insert that content in "merged_filename".
- j.) **cat *.txt:** It will show the content of all the text file present in the folder.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intro.txt
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intro.txt num.txt
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
2313
43635
21
42552
1241

vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat -n Intro.txt
 1 HII
 2 I AM VINAYAK SETHI
 3 I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
 4 BYEE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat > Hello.txt
Hello world
^C
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Hello.txt
Hello world
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intro.txt > Introcpy.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Introcpy.txt
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intro.txt >> Hello.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Hello.txt
Hello world
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ tac Intro.txt
BYEE
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
I AM VINAYAK SETHI
HII
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat -E Intro.txt
HII$
I AM VINAYAK SETHI$
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM$
BYEE$
```

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat Intro.txt num.txt > mix.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat mix.txt
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
2313
43635
21
42552
1241

vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ cat *.txt
Hello world
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
BYE
HI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
I AM VINAYAK SETHI
HI
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYE
BYE
HII
I AM VINAYAK SETHI
I AM 3RD YR CSE STUDENT AT IIITDM KANCHEEPURM
BYEE
2313
43635
21
42552
1241

2313
43635
21
42552
1241

vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$
```

31. mount command

mount command is used to mount the filesystem found on a device to big tree structure(Linux filesystem) rooted at '/'. Conversely, another command umount can be used to detach these devices from the Tree.

These commands tells the Kernel to attach the filesystem found at device to the dir.

a.) lsblk: lsblk lists information about all or the specified block devices. The lsblk command reads the sysfs filesystem to gather information. The command prints all block devices (except RAM disks) in a tree-like format by default.

b.) mount -l: Lists all the file systems mounted yet.

```
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~# lsblk
NAME  MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda    8:0    0 931.5G 0 disk
└─sda1  8:1    0   817G 0 part
  └─sda2  8:2    0   763M 0 part /boot/efi
  └─sda3  8:3    0   126M 0 part
  └─sda4  8:4    0 113.7G 0 part /
sdb    8:16   0 119.2G 0 disk
└─sdb1  8:17   0   100M 0 part
  └─sdb2  8:18   0   16M 0 part
  └─sdb3  8:19   0 118.1G 0 part
  └─sdb4  8:20   0   1G 0 part
root@vinayak-Swift-SF315-52G:~# mount -l
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,noexec,relatime,size=3969420k,nr_inodes=992355,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=803008k,mode=755)
/dev/sda4 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,noexec,relatime)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,mode=755)
cgroup2 on /sys/fs/cgroup/unified type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,xattr,name=systemd)
pstree on /sys/fs/pstree type pstree (rw,nosuid,nodev,noexec,relatime)
efivarfs on /sys/firmware/efi/efivars type efivarfs (rw,nosuid,nodev,noexec,relatime)
none on /sys/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
cgroup on /sys/fs/cgroup/cpu/cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,cpu,cpuacct)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,blkio)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,freezer)
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,hugetlb)
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,memory)
cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,net_cls,net_prio)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,rdma)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,cpuset)
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,devices)
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,perf_event)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,pids)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=28,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=21049)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
/dev/sda2 on /boot/efi type vfat (rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=iso8859-1,shortname=mixed,errors=remount-ro)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=803004k,mode=700,uid=1000,gid=1000)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
root@vinayak-Swift-SF315-52G:~# 
```

c.) mount /dev/sdb1 directoryname: It mounts a file system, to the given directory.

```
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~/Documents/OS# ls
Asgn1  Asgn1.tar  Asgn2  Doc.txt  file1cpy.txt  file1.txt  file1.zip  file2.txt  file2.txt.bz2  file2.zip  '~Pictures'
root@vinayak-Swift-SF315-52G:~/Documents/OS# mount /dev/sdb1 Asgn1
root@vinayak-Swift-SF315-52G:~/Documents/OS# ls
Asgn1  Asgn1.tar  Asgn2  Doc.txt  file1cpy.txt  file1.txt  file1.zip  file2.txt  file2.txt.bz2  file2.zip  '~Pictures'
root@vinayak-Swift-SF315-52G:~/Documents/OS# cd Asgn1
root@vinayak-Swift-SF315-52G:~/Documents/OS/Asgn1# ls
EFI  'System Volume Information'
root@vinayak-Swift-SF315-52G:~/Documents/OS/Asgn1# 
```

d.) umount directoryname: It unmounts the filesystem from given directory.

```
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~/Documents/OS# ls
Asgn1  Asgn1.tar  Asgn2  Doc.txt  file1cpy.txt  file1.txt  file1.zip  file2.txt  file2.txt.bz2  file2.zip  '~Pictures'
root@vinayak-Swift-SF315-52G:~/Documents/OS# umount Asgn1
root@vinayak-Swift-SF315-52G:~/Documents/OS# ls
Asgn1  Asgn1.tar  Asgn2  Doc.txt  file1cpy.txt  file1.txt  file1.zip  file2.txt  file2.txt.bz2  file2.zip  '~Pictures'
root@vinayak-Swift-SF315-52G:~/Documents/OS# cd Asgn1
root@vinayak-Swift-SF315-52G:~/Documents/OS/Asgn1# ls
Intr.txt  Int.txt
root@vinayak-Swift-SF315-52G:~/Documents/OS/Asgn1# 
```

32. chmod command

In Unix and Unix-like operating systems, chmod is the command and system call which is used to change the access permissions of file system objects (files and directories). It is also used to change special mode flags. The request is filtered by the umask. The name is an abbreviation of change mode.

a.) chmod ug+rwx file.txt: Give full access to user and group (i.e read, write and execute) on a specific file.

b.) chmod g-rwx file.txt: Revoke all access for the group (i.e read, write and execute) on a specific file.

c.) chmod -R g-rwx directory: Apply the file permissions recursively to all the files in the sub-directories.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls -l
total 44
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ chmod ug+rwx Heyy.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls -l
total 44
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rwx-rw-r-- 1 vinayak vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ chmod g-rwx Heyy.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls -l
total 44
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rwx---r-- 1 vinayak vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ chmod -R g-rwx Asgn4
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls -l
total 44
drwx---r-x 2 vinayak vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rwx---r-- 1 vinayak vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ 
```

33. chown command

The chown command allows you to change the user and/or group ownership of a given file, directory, or symbolic link.

In Linux, all files are associated with an owner and a group and assigned with permission access rights for the file owner, the group members, and others.

a.) chown otheruser file: The following command will change the ownership of a file to a new owner.

b.) chown -R directory: It will recursively change the ownership of directory.

```
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~/Documents/Test# ls -l
total 44
drwx---r-x 2 vinayak vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rwxrwxr-- 1 vinayak vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
root@vinayak-Swift-SF315-52G:~/Documents/Test# chown rahul Heyy.txt
root@vinayak-Swift-SF315-52G:~/Documents/Test# ls -l
total 44
drwx---r-x 2 vinayak vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rwxrwxr-- 1 rahul vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
root@vinayak-Swift-SF315-52G:~/Documents/Test# chown -R rahul Asgn4
root@vinayak-Swift-SF315-52G:~/Documents/Test# ls -l
total 44
drwx---r-x 2 rahul vinayak 4096 Aug 29 17:31 Asgn4
drwxrwxr-x 2 vinayak vinayak 4096 Aug 29 17:38 Asgn5
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 17:18 google.co.in
-rwxrwxr-- 1 rahul vinayak 74 Aug 26 21:48 Heyy.txt
-rw-rw-r-- 1 vinayak vinayak 26 Aug 26 19:50 Heyy.txt~
-rw-rw-r-- 1 vinayak vinayak 14229 Aug 26 17:17 index.html
-rw-rw-r-- 1 vinayak vinayak 74 Aug 26 22:06 Introcpy.txt
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
root@vinayak-Swift-SF315-52G:~/Documents/Test# 
```

34. passwd command

passwd command in Linux is used to change the user account passwords. The root user reserves the privilege to change the password for any user on the system, while a normal user can only change the account password for his or her own account.

a.) **passwd:** It changes the password of current user.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ passwd
Changing password for vinayak.
Current password:
New password:
Retype new password:
passwd: password updated successfully
vinayak@vinayak-Swift-SF315-52G:~$
```

b.) **passwd username:** It changes the password for given username.

```
root@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~# passwd rahul
New password:
Retype new password:
passwd: password updated successfully
root@vinayak-Swift-SF315-52G:~#
```

c.) **passwd -d username:** It removes the password for a specific user. Root user can disable password for a specific user. Once the password is disabled, the user can login without entering the password.

```
root@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
root@vinayak-Swift-SF315-52G:~# passwd -d rahul
passwd: password expiry information changed.
root@vinayak-Swift-SF315-52G:~#
```

35. mkdir command

The mkdir command in Linux/Unix allows users to create or make new directories. mkdir stands for “make directory.”

With mkdir, we can also set permissions, create multiple directories (folders) at once, and much more.

a.) **mkdir dirname:** This will create a new directory with given dirname.

b.) **mkdir {dir1,dir2,...}:** This will create multiple directories.

c.) **mkdir -p directories:** A flag which enables the command to create parent directories as necessary. If the directories exist, no error is specified. If we specify the -p option, the directories will be created, and no error will be reported.

d.) mkdir -m777 directoryname: The mkdir command by default gives rwx permissions for the current user only.

To add read, write, and execute permission for all users, add the -m option with the user 777 when creating a directory.

e.) mkdir -v directoryname: When executing mkdir commands, there is no feedback for successful operations. To see the details of the mkdir process, append the -v option to the terminal command.

By getting the feedback from the process, we do not have to run the ls command to verify the directory was created.

f.) mkdir –version: It displays the version number, some information regarding the license and exits.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mkdir Asgn1
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn1
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mkdir {Asgn2,Asgn3}
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls
Asgn1 Asgn2 Asgn3
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mkdir -p Asgn2/Theory/src1
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls -R
.:
Asgn1 Asgn2 Asgn3

./Asgn1:

./Asgn2:
Theory

./Asgn2/Theory:
src1

./Asgn2/Theory/src1:

./Asgn3:
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mkdir -m777 Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ ls -l
total 16
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 14:54 Asgn1
drwxrwxr-x 3 vinayak vinayak 4096 Aug 26 14:56 Asgn2
drwxrwxr-x 2 vinayak vinayak 4096 Aug 26 14:55 Asgn3
drwxrwxrwx 2 vinayak vinayak 4096 Aug 26 14:57 Perm
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mkdir -v Asgn4
mkdir: created directory 'Asgn4'
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ mkdir --version
mkdir (GNU coreutils) 8.30
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by David MacKenzie.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Test$ 
```

36. ifconfig command

Ifconfig stands for "Interface Configuration". It is a utility for Linux machines to configure, assign, add, delete, control and query network interface in Unix/Linux machine.

Common Linux users uses IFCONFIG command to assign ip address and netmask to an interface or to disable or enable a given interface.

a.) ifconfig -a: This option is used to display all the interfaces available, even if they are down.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ifconfig -a
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 4982 bytes 480646 (480.6 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 4982 bytes 480646 (480.6 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.3 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::11be:7292:aba2:a7df prefixlen 64 scopeid 0x20<link>
            ether 7c:2a:31:a0:ba:59 txqueuelen 1000 (Ethernet)
            RX packets 1127300 bytes 750177705 (750.1 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 453697 bytes 67581321 (67.5 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vinayak@vinayak-Swift-SF315-52G:~$ 
```

b.) ifconfig -s: Display a short list, instead of details

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ifconfig -s
Iface      MTU     RX-OK RX-ERR RX-DRP RX-OVR     TX-OK TX-ERR TX-DRP TX-OVR Flg
lo       65536     4990      0      0 0       4990      0      0      0 LRU
wlp2s0    1500   1127622      0      0 0      454057      0      0      0 BMRU
vinayak@vinayak-Swift-SF315-52G:~$ 
```

c.) ifconfig interface up: This option is used to activate the driver for the given interface

d.) ifconfig interface down: This option is used to deactivate the driver for the given interface

e.) ifconfig add addr/prefixlen: This option is used to add an IPv6 address to an interface

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ifconfig lo up
SIOCSIFFLAGS: Operation not permitted
vinayak@vinayak-Swift-SF315-52G:~$ ifconfig lo down
SIOCSIFFLAGS: Operation not permitted
vinayak@vinayak-Swift-SF315-52G:~$ ifconfig add addr/prefixlen
Usage:
  ifconfig [-a] [-v] [-s] <interface> [<AF>] <address>
  [add <address>[/<prefixlen>]]
  [del <address>[/<prefixlen>]]
  [[-]broadcast [<address>]]  [[-]pointopoint [<address>]]
  [netmask <address>]  [dstaddr <address>]  [tunnel <address>]
  [outfill <NN>]  [keepalive <NN>]
  [hw <HW> <address>]  [mtu <NN>]
  [[-]trailers]  [[-]arp]  [[-]allmulti]
  [multicast]  [[-]promisc]
  [mem_start <NN>]  [io_addr <NN>]  [irq <NN>]  [media <type>]
  [txqueuelen <NN>]
  [[-]dynamic]
  [up|down] ...
<HW>=Hardware Type.
List of possible hardware types:
  loop (Local Loopback) slip (Serial Line IP) cslip (VJ Serial Line IP)
  slip6 (6-bit Serial Line IP) cslip6 (VJ 6-bit Serial Line IP) adaptive (Adaptive Serial Line IP)
  ash (Ash) ether (Ethernet) ax25 (AMPR AX.25)
  netrom (AMPR NET/ROM) rose (AMPR ROSE) tunnel (IPIP Tunnel)
  ppp (Point-to-Point Protocol) hdlc ((Cisco)-HDLC) lapb (LAPB)
  arcnet (ARCnet) dlci (Frame Relay DLCI) frad (Frame Relay Access Device)
  sit (IPv6-in-IPv4) fddi (Fiber Distributed Data Interface) hippi (HIPPI)
  irda (IrLAP) ec (Econet) x25 (generic X.25)
  eui64 (Generic EUI-64)
<AF>=Address family. Default: inet
List of possible address families:
  unix (UNIX Domain) inet (DARPA Internet) inet6 (IPv6)
  ax25 (AMPR AX.25) netrom (AMPR NET/ROM) rose (AMPR ROSE)
  ipx (Novell IPX) ddp (Appletalk DDP) ec (Econet)
  ash (Ash) x25 (CCITT X.25)
vinayak@vinayak-Swift-SF315-52G:~$ 

```

37. uname command

uname is a command-line utility that prints basic information about the operating system name and system hardware.

The uname tool is most commonly used to determine the processor architecture, the system hostname and the version of the kernel running on the system.

The [syntax](#) of the uname command takes the following form:

uname [OPTIONS]...

a.) When invoked **without any options, uname prints the kernel name.**

b.) -s, (--kernel-name) - Prints the kernel name.

c.) -n, (--nodename) - Prints the system's node name (hostname). This is the name the system uses when communicating over the network. When used

with the -n option, uname produces the same output as the hostname command.

d.) -r, (--kernel-release) - Prints the kernel release.

e.) -v, (--kernel-version) - Prints the kernel version.

f.) -m, (--machine) - Prints the name of the machine's hardware name.

g.) -p, (--processor) - Prints the architecture of the processor.

h.) -o, (--operating-system) - Print the name of the operating system. On Linux systems that is "GNU/Linux"

i.) -a, (--all) - When the -a option is used, uname behaves the same as if the -snrvmo options have been given.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ uname
Linux
vinayak@vinayak-Swift-SF315-52G:~$ uname -s
Linux
vinayak@vinayak-Swift-SF315-52G:~$ uname -n
vinayak-Swift-SF315-52G
vinayak@vinayak-Swift-SF315-52G:~$ uname -r
5.4.0-26-generic
vinayak@vinayak-Swift-SF315-52G:~$ uname -v
#30-Ubuntu SMP Mon Apr 20 16:58:30 UTC 2020
vinayak@vinayak-Swift-SF315-52G:~$ uname -m
x86_64
vinayak@vinayak-Swift-SF315-52G:~$ uname -p
x86_64
vinayak@vinayak-Swift-SF315-52G:~$ uname -o
GNU/Linux
vinayak@vinayak-Swift-SF315-52G:~$ uname -a
Linux vinayak-Swift-SF315-52G 5.4.0-26-generic #30-Ubuntu SMP Mon Apr 20 16:58:30 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
vinayak@vinayak-Swift-SF315-52G:~$ 
```

38. whereis command

The whereis command in Linux is used to locate the binary, source, and manual page files for a command. This command searches for files in a restricted set of locations (binary file directories, man page directories, and library directories). It is usually used to find executables of a program, its man pages and configuration files.

a.) whereis commandname: When we want to find out where a specific Unix command exists (for example, where does apropos command exists?), we can execute the following command.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ whereis apropos  
apropos: /usr/bin/apropos /usr/share/man/man1/apropos.1.gz  
vinayak@vinayak-Swift-SF315-52G:~$
```

b.) whereas -b command: This command is used when we only want to search for binaries of given linux command.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ whereis -b gunzip  
gunzip: /bin/gunzip  
vinayak@vinayak-Swift-SF315-52G:~$
```

c.) whereas -B: When we want to search an executable from a path other than the whereas default path, we can use -B option and give path as argument to it. For example lets search for the executable crontab.gDXDf9 in the /tmp directory, and displays it, if it is available.

```
vinayak@vinayak-  
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ whereis -u -B /tmp -f /crontab.gDXDf9  
crontab: /usr/share/man/man1/crontab.1.gz /usr/share/man/man5/crontab.5.gz /tmp/crontab.i5xYyH /tmp/crontab.gDXDf9  
vinayak@vinayak-Swift-SF315-52G:~$
```

39. whatis command

whatis command in Linux is used to get a one-line manual page descriptions. In Linux, each manual page has some sort of description within it. So this command search for the manual pages names and show the manual page description of the specified filename or argument.

a.) whatis -d commandname: This option prints the debugging information.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ whatis -d ls
From the config file /etc/manpath.config:
Mandatory mandir '/usr/man'.
Mandatory mandir '/usr/share/man'.
Mandatory mandir '/usr/local/share/man'.
Path `/bin' mapped to mandir '/usr/share/man'.
Path `/usr/bin' mapped to mandir '/usr/share/man'.
Path `/sbin' mapped to mandir '/usr/share/man'.
Path `/usr/sbin' mapped to mandir '/usr/share/man'.
Path `/usr/local/bin' mapped to mandir '/usr/local/man'.
Path `/usr/local/bin' mapped to mandir '/usr/local/share/man'.
Path `/usr/local/sbin' mapped to mandir '/usr/local/man'.
Path `/usr/local/sbin' mapped to mandir '/usr/local/share/man'.
Path `/usr/X11R6/bin' mapped to mandir '/usr/X11R6/man'.
Path `/usr/bin/X11' mapped to mandir '/usr/X11R6/man'.
Path `/usr/games' mapped to mandir '/usr/share/man'.
Path `/opt/bin' mapped to mandir '/opt/man'.
Path `/opt/sbin' mapped to mandir '/opt/man'.
Global mandir '/usr/man', catdir '/var/cache/man/fsstnd'.
Global mandir '/usr/share/man', catdir '/var/cache/man'.
Global mandir '/usr/local/man', catdir '/var/cache/man/oldlocal'.
Global mandir '/usr/local/share/man', catdir '/var/cache/man/local'.
Global mandir '/usr/X11R6/man', catdir '/var/cache/man/X11R6'.
Global mandir '/opt/man', catdir '/var/cache/man/opt'.
Global mandir '/snap/man', catdir '/var/cache/man/snap'.
Added sections: `1', `n', `l', `8', `3', `2', `3posix', `3pm', `3perl', `3am', `5', `4', `9', `6', `7'.
path directory /usr/local/sbin is in the config file
  adding /usr/local/man to manpath
  adding /usr/local/share/man to manpath
path directory /usr/local/bin is in the config file
path directory /usr/sbin is in the config file
  adding /usr/share/man to manpath
path directory /usr/bin is in the config file
path directory /sbin is in the config file
path directory /bin is in the config file
path directory /usr/games is in the config file
```

b.) whatis -v commandname: This option will prints verbose warning messages.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ whatis -v ls
ls (1)           - list directory contents
vinayak@vinayak-Swift-SF315-52G:~$ 
```

c.) whatis -r commandname: This option interprets each of the name as a regular expression. If any of the names match any part of a page name, a match will be made. This option causes the whatis command to be somewhat slower due to the nature of database searches.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ whatis -r ls
Future::Utils (3pm) - utility functions for working with "Future" objects
_llseek (2) - reposition read/write file offset
alsa-info (8) - command-line utility to gather information about the ALSA subsystem
alsabat (1) - command-line sound tester for ALSA sound card driver
alsactl (1) - advanced controls for ALSA soundcard driver
alsactl_init (7) - alsa control management - initialization
alsaloop (1) - command-line PCM loopback
alsamixer (1) - soundcard mixer for ALSA soundcard driver, with ncurses interface
alsatplg (1) - ALSA Topology Compiler
alsaucm (1) - ALSA Use Case Manager
backtrace_symbols (3) - support for application self-debugging
backtrace_symbols_fd (3) - support for application self-debugging
credentials (7) - process identifiers
deb-symbols (5) - Debian's extended shared library information file
dlsym (3) - obtain address of a symbol in a shared object or executable
dpkg-gensymbols (1) - generate symbols files (shared library dependency information)
enchant-lsmod-2 (1) - list provider and dictionary information
ExtUtils::Depends (3pm) - Easily build XS extensions that depend on XS extensions
ExtUtils::PkgConfig (3pm) - simplistic interface to pkg-config
false (1) - do nothing, unsuccessfully
flatpak-remote-ls (1) - Show available runtimes and applications
Font::TTF::AAUtils (3pm) - Utility functions for AAT tables
Font::TTF::Utils (3pm) - Utility functions to save fingers
git-ls-files (1) - Show information about files in the index and the working tree
git-ls-remote (1) - List references in a remote repository
git-ls-tree (1) - List the contents of a tree object
git-mailsplit (1) - Simple UNIX mbox splitter program
gitcredentials (7) - providing usernames and passwords to Git
Glib::Utils (3pm) - Miscellaneous utility functions
grub-menulst2cfg (1) - transform legacy menu.lst into grub.cfg
gvfs-ls (1) - (unknown subject)
hp-levels (1) - Supply Levels Utility
initramfs-tools (7) - an introduction to writing scripts for mkinitramfs
IO::Socket::SSL::Utils (3pm) - - loading, storing, creating certificates and keys

```

d.) whatis -w commandname: This option interprets each name as a pattern containing shell style wildcards. For a match about to make, an expanded name needs to match the entire page name. This option causes the whatis command to be somewhat slower due to the nature of database searches.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ whatis -w ls
ls (1) - list directory contents
vinayak@vinayak-Swift-SF315-52G:~$ 

```

e.) whatis -s section-num commandname: This option will go to search only the given manual sections. The list is a colon or a comma-separated list of sections.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ whatis -s 1 cat  
cat (1)           - concatenate files and print on the standard output  
vinayak@vinayak-Swift-SF315-52G:~$ 
```

f.) **whatis -usage:** This option will prints a short usage message and exit.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ whatis --usage  
Usage: whatis [-dvrwl?V] [-C FILE] [-L LOCALE] [-m SYSTEM] [-M PATH] [-s LIST]  
      [--debug] [--verbose] [--regex] [--wildcard] [--long]  
      [--config-file=FILE] [--locale=LOCALE] [--systems=SYSTEM]  
      [--manpath=PATH] [--sections=LIST] [--section=LIST] [--help]  
      [--usage] [--version] KEYWORD...  
vinayak@vinayak-Swift-SF315-52G:~$ 
```

g.) **whatis -V:** This option will display version information.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ whatis -V  
whatis 2.9.1  
vinayak@vinayak-Swift-SF315-52G:~$ 
```

40. locate command

locate command in Linux is used to find the files by name. There are two most widely used file searching utilities accessible to users are called find and locate. The locate utility works better and faster than find command counterpart because instead of searching the file system when a file search is initiated, it would look through a database. This database contains bits and parts of files and their corresponding paths on your system. By default, locate command does not check whether the files found in the database still exist and it never reports files created after the most recent update of the relevant database.

a.) **locate filename:** Using locate command we can quickly search for the location of a specific file (or group of files). Locate command uses the database created by updatedb.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ locate crontab  
/etc/anacrontab  
/etc/crontab  
/usr/bin/crontab  
/usr/share/bash-completion/completions/crontab  
/usr/share/doc/cron/examples/crontab2english.pl  
/usr/share/man/man1/crontab.1.gz  
/usr/share/man/man5/anacrontab.5.gz  
/usr/share/man/man5/crontab.5.gz  
/usr/share/vim/vim81/syntax/crontab.vim  
vinayak@vinayak-Swift-SF315-52G:~$
```

b.) locate “*.txt” -n 10: We can limit our search returns to a required number to avoid redundancy with our search results using the -n command. For our case we limited search results to 10.

c.) locate -c “file1”: If we want to display the count of all matching entries of file “file1”, use the locate -c command.

d.) locate -S: Used to view the locate database statistics by using the -S command.

```
File Edit View Search Terminal Help  
vinayak@vinayak-Swift-SF315-52G:~$ locate "*.txt" -n 10  
/boot/grub/gfxblacklist.txt  
/etc/X11/rgb.txt  
/etc/brltty/Input/ba/all.txt  
/etc/brltty/Input/bd/all.txt  
/etc/brltty/Input/bl/18.txt  
/etc/brltty/Input/bl/40_m20_m40.txt  
/etc/brltty/Input/ec/all.txt  
/etc/brltty/Input/ec/spanish.txt  
/etc/brltty/Input/eu/all.txt  
/etc/brltty/Input/lb/all.txt  
vinayak@vinayak-Swift-SF315-52G:~$ locate -c "/etc"*\n5013  
vinayak@vinayak-Swift-SF315-52G:~$ locate -S  
Database /var/lib/mlocate/mlocate.db:  
      52,382 directories  
     6,87,385 files  
 5,75,78,662 bytes in file names  
1,93,18,767 bytes used to store database  
vinayak@vinayak-Swift-SF315-52G:~$
```

41. man command

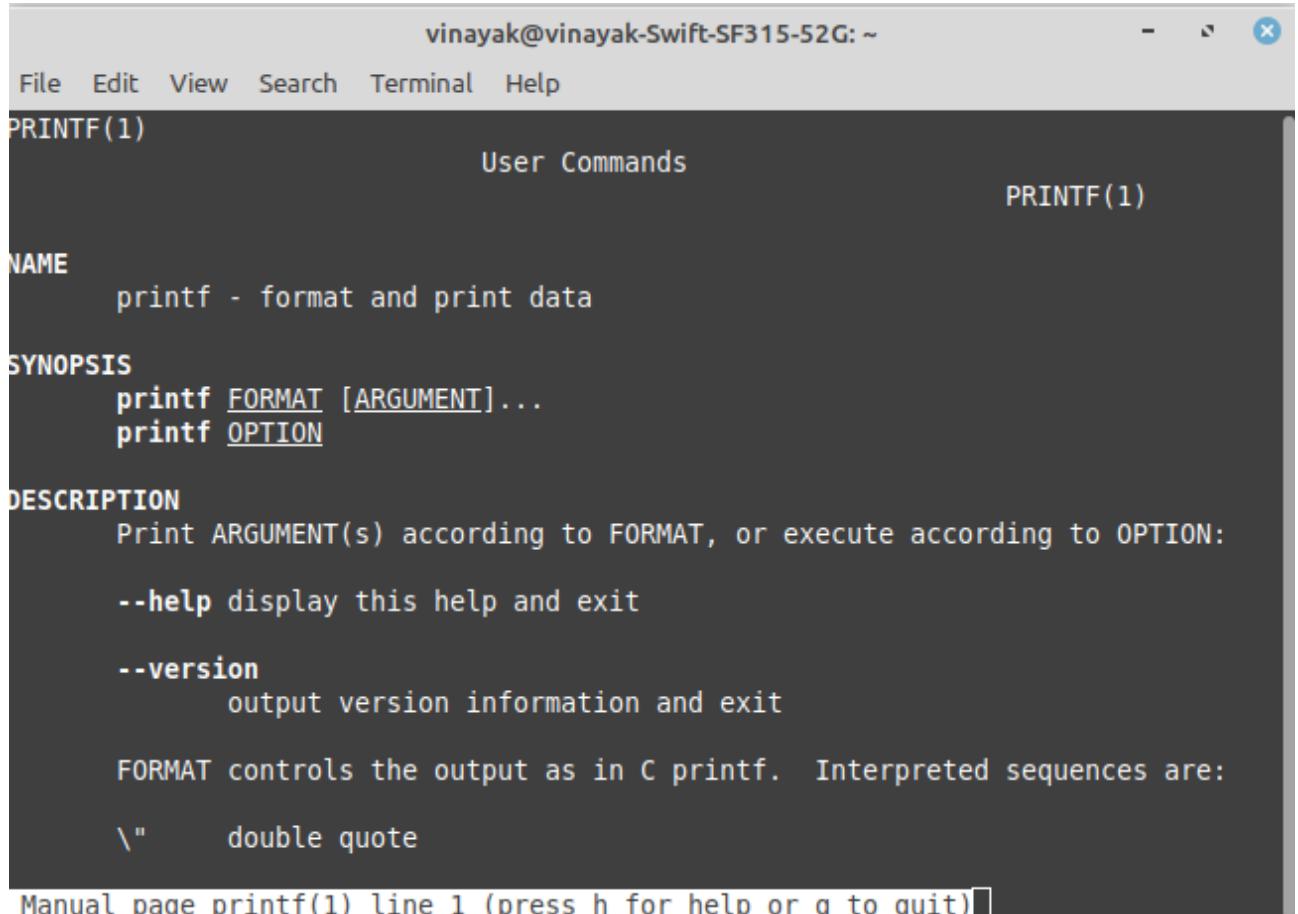
man command in Linux is used to display the user manual of any command that we can run on the terminal. It provides a detailed view of the command

which includes NAME, SYNOPSIS, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUES, ERRORS, FILES, VERSIONS, EXAMPLES, AUTHORS and SEE ALSO.

Syntax :

\$man [OPTION]... [COMMAND NAME]...

a.) man commandname: No Option means it displays the whole manual of the command.



```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
PRINTF(1)                               User Commands                               PRINTF(1)

NAME
    printf - format and print data

SYNOPSIS
    printf FORMAT [ARGUMENT]...
    printf OPTION

DESCRIPTION
    Print ARGUMENT(s) according to FORMAT, or execute according to OPTION:
        --help display this help and exit
        --version
            output version information and exit
    FORMAT controls the output as in C printf. Interpreted sequences are:
        \"      double quote

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

b.) man section-num commandname: Since a manual is divided into multiple sections so this option is used to display only a specific section of a manual.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
INTRO(2) Linux Programmer's Manual INTRO(2)

NAME
    intro - introduction to system calls

DESCRIPTION
    Section 2 of the manual describes the Linux system calls. A system call is an entry point into the Linux kernel. Usually, system calls are not invoked directly: instead, most system calls have corresponding C library wrapper functions which perform the steps required (e.g., trapping to kernel mode) in order to invoke the system call. Thus, making a system call looks the same as invoking a normal library function.

    In many cases, the C library wrapper function does nothing more than:
        * copying arguments and the unique system call number to the registers where the kernel expects them;
        * trapping to kernel mode, at which point the kernel does the real work of the system call;
        * setting errno if the system call returns an error number when the
Manual page intro(2) line 1 (press h for help or q to quit)
```

c.) **man -f commandname:** One may not be able to remember the sections in which a command is present. So this option gives the section in which the given command is present.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ man -f printf
printf (1)           - format and print data
printf (3)           - formatted output conversion
vinayak@vinayak-Swift-SF315-52G:~$ 
```

d.) **man -k commandname:** This option searches the given command as a regular expression in all the manuals and it returns the manual pages with the section number in which it is found.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ man -k cd
apt-cdrom (8)           - APT CD-ROM management utility
cd-create-profile (1)   - Color Manager Profile Creation Tool
cd-fix-profile (1)     - Color Manager Testing Tool
cd-it8 (1)              - Color Manager Testing Tool
evcd2vcd (1)            - Converts EVCD files to VCD files
fst2vcd (1)              - Converts FST files to VCD
gcov-dump (1)            - offline gcda and gcno profile dump tool
gcov-dump-9 (1)          - offline gcda and gcno profile dump tool
gcov-tool (1)             - offline gcda profile processing tool
gcov-tool-9 (1)           - offline gcda profile processing tool
gtkwave (1)              - Visualization tool for VCD, LXT, LXT2, VZT, FST, and G...
hex2hcd (1)              - Broadcom Bluetooth firmware converter
hipercdecode (1)          - Decode a HIPERC stream into human readable form.
libOpenCL (7)             - OCL-ICD implementation of OpenCL ICD loader
libOpenCL.so (7)           - OCL-ICD implementation of OpenCL ICD loader
lxt2vcd (1)              - Converts LXT2 files to VCD
mcd (1)                  - change MSDOS directory
Net::DNS::RR::CDNSKEY (3pm) - DNS CDNSKEY resource record
Net::DNS::RR::CDS (3pm)    - DNS CDS resource record
rsyncd.conf (5)            - configuration file for rsync in daemon mode
systemd-timesyncd (8)      - Network Time Synchronization
systemd-timesyncd.service (8) - Network Time Synchronization
tcdrain (3)                - get and set terminal attributes, line control, get and...
```

e.) **man -w commandname:** This option returns the location in which the manual page of a given command is present.

```
vinayak@vinayak-Swift-SF315
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ man -w ls
/usr/share/man/man1/ls.1.gz
vinayak@vinayak-Swift-SF315-52G:~$
```

f.) **man -i commandname:** It considers the command as case sensitive.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
LS(1)                               User Commands                   LS(1)
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)
```

42. tail command

The tail command is a command-line utility for outputting the last part of files given to it via standard input. It writes results to standard output. By default tail returns the last ten lines of each file that it is given. It may also be used to follow a file in real-time and watch as new lines are written to it.

a.) tail filename: It displays the last 10 lines of given file.

b.) tail -n N filename: It displays the last N lines of given file.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNIX systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
Vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tail sample1.txt
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNIX systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
Yet another powerful OS.
Vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tail -n 5 sample1.txt
Unix systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
Vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

c.) tail -f filename: Used to view the contents of file in real time.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls /var/log
alternatives.log boot.log boot.log.5 cups dmesg.3.gz gpu-manager.log kern.log mintsystem.log samba syslog.3.gz ubuntu-advantage.log Xorg.0.log
apt boot.log.1 boot.log.6 dmesg dmesg.4.gz gpu-manager-switch.log kern.log.1 mintsystem.timestamps speech-dispatcher syslog.4.gz ubuntu-system-adjustments-adjust-grub-title.log Xorg.0.log.old
auth.log boot.log.2 boot.log.7 dmesg.0 dpkg.log hp kern.log.2.gz openvpn syslog syslog.5.gz ubuntu-system-adjustments-start.log
auth.log.1 boot.log.3 bootstrap.log dmesg.1.gz faillog installer lastlog prime-supported.log syslog.1 syslog.6.gz ubuntu-system-adjustments-stop.log
auth.log.2.gz boot.log.4 btmp dmesg.2.gz fontconfig.log journal lightdm private syslog.2.gz syslog.7.gz wtmp
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tail -f /var/log/auth.log
Jan 31 12:00:03 vinayak-Swift-SF315-52G CRON[41810]: pam_unix(cron:session): session opened for user vinayak by (uid=0)
Jan 31 12:00:03 vinayak-Swift-SF315-52G CRON[41810]: pam_unix(cron:session): session closed for user vinayak
Aug 29 23:00:01 vinayak-Swift-SF315-52G CRON[42346]: pam_unix(cron:session): session opened for user vinayak by (uid=0)
Aug 29 23:00:01 vinayak-Swift-SF315-52G CRON[42346]: pam_unix(cron:session): session closed for user vinayak
Aug 29 23:03:32 vinayak-Swift-SF315-52G pkeexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Aug 29 23:03:33 vinayak-Swift-SF315-52G pkeexec[42529]: vinayak: Executing command [USER=root] [TTY=unknown] [CWD=/home/vinayak] [COMMAND=/usr/lib/x86_64-linux-gnu/cinnamon-settings-daemon/csd-backlight-helper --set-brightness 2250 -b firm ware -b platform -b raw]
Aug 29 23:03:45 vinayak-Swift-SF315-52G pkeexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Aug 29 23:03:45 vinayak-Swift-SF315-52G pkeexec[42536]: vinayak: Executing command [USER=root] [TTY=unknown] [CWD=/home/vinayak] [COMMAND=/usr/lib/x86_64-linux-gnu/cinnamon-settings-daemon/csd-backlight-helper --set-brightness 7500 -b firm ware -b platform -b raw]
Aug 29 23:10:01 vinayak-Swift-SF315-52G CRON[42762]: pam_unix(cron:session): session opened for user vinayak by (uid=0)
Aug 29 23:10:01 vinayak-Swift-SF315-52G CRON[42762]: pam_unix(cron:session): session closed for user vinayak
```

d.) tail -n N filename | sort -r: The tail command can be piped with many other commands of the unix. In the following example output of the tail command is given as input to the sort command with -r option to sort the last 5 lines coming from file sample1.txt in the reverse order.

vinayak@vi

File Edit View Search Terminal Help

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tail -n 5 sample1.txt | sort -r
Yet another powerful OS.
UNixOS systems use a centralized operating system kernel which manages system and process activities.
UNIX is a free OS.
Multiuser operating system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

43. less command

less is a command line utility that displays the contents of a file or a command output, one page at a time. It is similar to more, but has more advanced features and allows you to navigate both forward and backward through the file.

When starting less doesn't read the entire file which results in much faster load times compared to text editors like vim or nano .

The less command is mostly used for opening large files .

a.) less filename: less command don't load entire file, but load it part by part, which makes it faster to access the file.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls /var/log
alternatives.log boot.log boot.5.log cups dmesg.3.gz gpu-manager.log kern.log mintsystem.log samba syslog.3.gz ubuntu-adantage.log Xorg.0.log
apt boot.log.1 boot.log.6 dmesg dmesg.4.gz gpu-manager-switch.log kern.log.1 mintsystem.timestamps speech-dispatcher syslog.4.gz ubuntu-system-adjustments-adjust-grub-title.log Xorg.0.log.old
auth.log boot.log.2 boot.log.7 dmesg.8 dpkg.log hp kern.log.2.gz openvpn syslog syslog.5.gz ubuntu-system-adjustments-start.log
auth.log.1 boot.log.3 bootstrap.log dmesg.1.gz faillog installer lastlog prime-supported.log syslog.1 syslog.6.gz ubuntu-system-adjustments-stop.log
auth.log.2.gz boot.log.4 btmp dmesg.2.gz fontconfig.log journal lightdm private syslog.2.gz syslog.7.gz wtmp
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ less /var/log/auth.log

[1]+ Stopped less /var/log/auth.log
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
Aug 23 09:17:01 vinayak-Swift-SF315-52G CRON[14240]: pam_unix(cron:session): session opened for user root by (uid=0)
Aug 23 09:17:01 vinayak-Swift-SF315-52G CRON[14240]: pam_unix(cron:session): session closed for user root
Aug 23 09:54:45 vinayak-Swift-SF315-52G polkitd[authority=local]: Unregistered Authentication Agent for unix-session:c2 (system bus name :1.94, object path /org/gnome/PolicyKit1/AuthenticationAgent, locale en_IN) (disconnected from bus)
Aug 23 09:54:46 vinayak-Swift-SF315-52G systemd-logind[913]: System is powering down.
Aug 23 13:37:44 vinayak-Swift-SF315-52G systemd-logind[946]: New seat seat0.
Aug 23 13:37:44 vinayak-Swift-SF315-52G systemd-logind[946]: Watching system buttons on /dev/input/event0 (Power Button)
Aug 23 13:37:44 vinayak-Swift-SF315-52G systemd-logind[946]: Watching system buttons on /dev/input/event1 (LCD Switch)
Aug 23 13:37:44 vinayak-Swift-SF315-52G systemd-logind[946]: Watching system buttons on /dev/input/event2 (Sleep Button)
Aug 23 13:37:44 vinayak-Swift-SF315-52G systemd-logind[946]: Watching system buttons on /dev/input/event3 (AT Translated Set 2 keyboard)
Aug 23 13:38:02 vinayak-Swift-SF315-52G lightdm: PAM unable to dlopen(pam_kwallet.so): /lib/security/pam_kwallet.so: cannot open shared object file: No such file or directory
Aug 23 13:38:02 vinayak-Swift-SF315-52G lightdm: PAM adding faulty module: pam_kwallet.so
Aug 23 13:38:02 vinayak-Swift-SF315-52G lightdm: PAM unable to dlopen(pam_kwallet5.so): /lib/security/pam_kwallet5.so: cannot open shared object file: No such file or directory
Aug 23 13:38:02 vinayak-Swift-SF315-52G lightdm: PAM adding faulty module: pam_kwallet5.so
Aug 23 13:38:03 vinayak-Swift-SF315-52G lightdm: pam_unix(systemd-user:session): session opened for user lightdm by (uid=0)
Aug 23 13:38:03 vinayak-Swift-SF315-52G systemd: pam_unix(systemd-user:session): session opened for user lightdm by (uid=0)
Aug 23 13:38:04 vinayak-Swift-SF315-52G lightdm: gkr-pam: gnome-keyring-daemon started properly
Aug 23 13:38:10 vinayak-Swift-SF315-52G lightdm: PAM unable to dlopen(pam_kwallet.so): /lib/security/pam_kwallet.so: cannot open shared object file: No such file or directory
Aug 23 13:38:10 vinayak-Swift-SF315-52G lightdm: PAM adding faulty module: pam_kwallet.so
Aug 23 13:38:10 vinayak-Swift-SF315-52G lightdm: PAM unable to dlopen(pam_kwallet5.so): /lib/security/pam_kwallet5.so: cannot open shared object file: No such file or directory
Aug 23 13:38:10 vinayak-Swift-SF315-52G lightdm: PAM adding faulty module: pam_kwallet5.so
Aug 23 13:38:10 vinayak-Swift-SF315-52G lightdm: pam_succeed_if(lightdm:auth): requirement "user ingroup nopasswdlogin" not met by user "vinayak"
Aug 23 13:38:10 vinayak-Swift-SF315-52G lightdm: pam_unix(lightdm:auth): Couldn't open /etc/secrety: No such file or directory
Aug 23 13:38:40 vinayak-Swift-SF315-52G lightdm: pam_unix((lightdm:auth)): Couldn't open /etc/secrety: No such file or directory
Aug 23 13:38:40 vinayak-Swift-SF315-52G lightdm: gkr-pam: unable to locate damage control file
Aug 23 13:38:40 vinayak-Swift-SF315-52G lightdm: gkr-pam: stashed password to try later in open session
Aug 23 13:38:41 vinayak-Swift-SF315-52G systemd-logind[946]: Removed session c1.
Aug 23 13:38:41 vinayak-Swift-SF315-52G lightdm: pam_unix(lightdm:session): session opened for user vinayak by (uid=0)
Aug 23 13:38:41 vinayak-Swift-SF315-52G systemd-logind[946]: New session c2 opened for user vinayak.
Aug 23 13:40:23 vinayak-Swift-SF315-52G lightdm: gkr-pam: gnome-keyring-daemon started properly and unlocked secrety
Aug 23 13:40:23 vinayak-Swift-SF315-52G lightdm: gkr-pam: gnome-keyring-daemon started properly and unlocked secrety
Aug 23 13:40:27 vinayak-Swift-SF315-52G pkexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Aug 23 13:40:27 vinayak-Swift-SF315-52G pkexec[2018]: vinayak: Executing command [USER=root] [TTY=unknown] [COMMAND=/usr/lib/x86_64-linux-gnu/cinnamon-settings-daemon/csd-backlight-helper --set-brightness 2250 -b firmware -b platform -b raw]
Aug 23 13:42:13 vinayak-Swift-SF315-52G systemd-logind[946]: Lid closed.
Aug 23 14:09:23 vinayak-Swift-SF315-52G systemd-logind[946]: Lid opened.
Aug 23 14:09:23 vinayak-Swift-SF315-52G systemd-logind[946]: Operation 'sleep' finished.
Aug 23 14:09:23 vinayak-Swift-SF315-52G pkexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Aug 23 14:09:23 vinayak-Swift-SF315-52G pkexec[2259]: vinayak: Executing command [USER=root] [TTY=unknown] [COMMAND=/usr/lib/x86_64-linux-gnu/cinnamon-settings-daemon/csd-backlight-helper --set-brightness 7500 -b firmware -b platform -b raw]
Aug 23 14:09:24 vinayak-Swift-SF315-52G cinnamon-screensaver-pam-helper: pam_unix(cinnamon-screensaver:auth): Couldn't open /etc/secrety: No such file or directory
Aug 23 14:09:24 vinayak-Swift-SF315-52G cinnamon-screensaver-pam-helper: pam_unix(cinnamon-screensaver:auth): Couldn't open /etc/secrety: No such file or directory
Aug 23 14:09:24 vinayak-Swift-SF315-52G cinnamon-screensaver-pam-helper: pam_unix(cinnamon-screensaver:auth): Couldn't open /etc/secrety: No such file or directory
Aug 23 14:09:50 vinayak-Swift-SF315-52G gnome-keyring-daemon[1421]: asked to register item /org/freedesktop/secrets/collection/login/1, but it's already registered
Aug 23 14:09:56 vinayak-Swift-SF315-52G gnome-keyring-daemon[1421]: asked to register item /org/freedesktop/secrets/collection/login/1, but it's already registered
Aug 23 14:12:30 vinayak-Swift-SF315-52G pkexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Aug 23 14:12:30 vinayak-Swift-SF315-52G pkexec[2790]: vinayak: Executing command [USER=root] [TTY=unknown] [COMMAND=/usr/lib/x86_64-linux-gnu/cinnamon-settings-daemon/csd-backlight-helper --set-brightness 2250 -b firmware -b platform -b raw]
Aug 23 14:13:29 vinayak-Swift-SF315-52G pkexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Aug 23 14:13:29 vinayak-Swift-SF315-52G pkexec[2796]: vinayak: Executing command [USER=root] [TTY=unknown] [COMMAND=/usr/lib/x86_64-linux-gnu/cinnamon-settings-daemon/csd-backlight-helper --set-brightness 7500 -b firmware -b platform -b raw]
Aug 23 14:15:46 vinayak-Swift-SF315-52G sudo: vinayak : TTY=unknown : PWD=/home/vinayak : USER=root , COMMAND=/usr/lib/linuxmint/mintUpdate/dpkg_lock_check.sh
Aug 23 14:15:46 vinayak-Swift-SF315-52G sudo: pam_unix(session): session opened for user root by (uid=0)
```

b.) less -N filename: If we want **less** to shows line numbers launch the program with the -N option.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ less -N sample1.txt
[2]+ Stopped less -N sample1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems
1 This is a test document.
2 An OS is an interface between a computer user and a computer hardware.
3 An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
4 Operating system is one of the core subjects in computer science.
5 Operating system is one of the core subjects in computer science.
6 Unix is a great OS.
7 UNIX is a free OS.
8 Unix systems use a centralized operating system kernel which manages system and process activities.
9 Unix is a great OS.
10 UNIX is a free OS.
11 UnixOS systems use a centralized operating system kernel which manages system and process activities.
12 A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
13 UNIX is a free OS.
14 Multiuser operating system.
15 Yet another powerful OS.
sample1.txt (END) ]
```

c.) less -X filename: By default, when **less** exits, the file contents will be cleared from the screen. To leave file contents on screen, use the -X option:

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ less -X sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UnixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

[3]+ Stopped less -X sample1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

44. su command

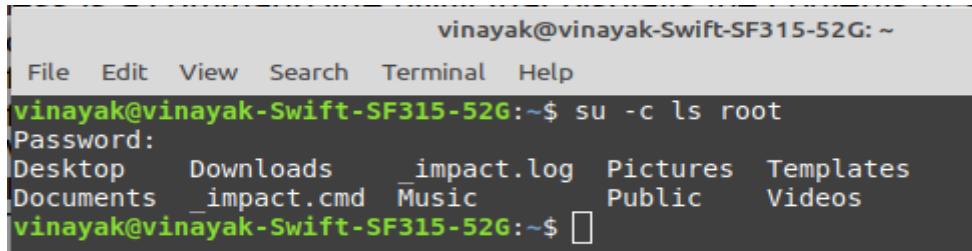
The Linux command ‘**su**’ (short for substitute or switch user) is used to switch from one account to another. User will be prompted for the password of the user switching to.

Users can also use it to switch to root account. If user types only ‘**su**’ without any option then It will be considered as root and user will be prompted to enter root user password.

a.) su -l otheruser: To switch the logged-in user in this terminal window, enter the following.

```
root@vinayak-Swift-SF3
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ su -l root
Password:
root@vinayak-Swift-SF315-52G:~# 
```

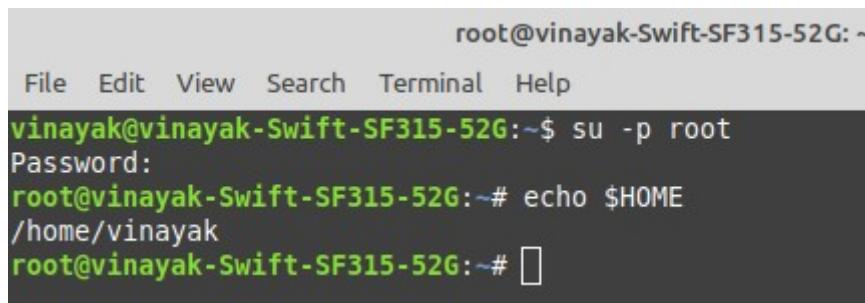
b.) su -c commandname otheruser: To run a specific command as a different user, use the –c option.



```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ su -c ls root
Password:
Desktop Downloads impact.log Pictures Templates
Documents _impact.cmd Music Public Videos
vinayak@vinayak-Swift-SF315-52G:~$
```

c.) su -p otheruser: We can keep the environment of the current user account with the –p option. Replace [other_user] with the actual username you want to switch to.

The user account will switch, but you'll keep the same home directory. This is useful if you need to run a command as a different user, but you need access to the current user's data.



```
root@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ su -p root
Password:
root@vinayak-Swift-SF315-52G:~# echo $HOME
/home/vinayak
root@vinayak-Swift-SF315-52G:~#
```

45. mysql command

mysqladmin is a command-line utility the comes with MySQL server and it is used by Database Administrators to perform some basic MySQL tasks easily such as setting root password, changing root password, monitoring mysql processes, reloading privileges, checking server status etc.

a.) mysqladmin -u root -p ping: To find out whether MySQL server is up and running.

b.) mysql -u root -p -h ipaddress: To connect to a remote mysql database. This will prompt for a password.

c.) mysql -u root -p: To connect to a local mysql database.

d.) mysqladmin -u root -p create databasename: Used to create a new database in MySQL server.

e.) mysqladmin -u root -p shutdown: Used to shutdown MySQL server safely.

46. yum command

YUM (Yellowdog Updater Modified) is an open source command-line as well as graphical based package management tool for **RPM(RedHat Package Manager)** based Linux systems. It allows users and system administrator to easily install, update, remove or search software packages on a systems. It was developed and released by **Seth Vidal** under **GPL(General Public License)** as an open source, means anyone can allowed to download and access the code to fix bugs and develop customized packages. **YUM** uses numerous third party repositories to install packages automatically by resolving their dependencies issues.

- a.) yum help:** Display yum commands and options
- b.) yum list available:** List all available packages
- c.) yum list installed:** List all installed packages
- d.) yum list all:** List installed and available packages
- e.) yum list kernel:** List installed and available kernel packages
- f.) yum search samba:** Find packages with samba in name or description
- g.) check:** Check the local RPM database for problems (runs for along time)
- h.) yum install vsftpd:** Install a package from a repository to your system
- i.) yum update:** Update all packages with available updates
- j.) yum update httpd:** Update the httpd package (if available)
- k.) yum update –security:** Apply security-related package updates
- m.) update-to:** Update one or all packages to a particular version
- n.) yum localinstall abc-1-1.i686.rpm:** Install abc package from local directory
- o.) yum localinstall <http://myrepo/abc-1-1.i686.rpm>:** Install abc from FTP site
- p.) yum remove vsftpd:** Remove the vsftpd package and dependencies
- q.) show-installed:** List installed RPM packages and statistics

47. rpm command

RPM(Red Hat Package Manager) is an default open source and most popular package management utility for **Red Hat** based systems like **(RHEL, CentOS and Fedora)**. The tool allows system administrators and users to **install, update, uninstall, query, verify and manage** system software packages in **Unix/Linux** operating systems.

Syntax	Description	Example(s)
rpm -ivh {rpm-file}	Install the package	rpm -ivh mozilla-mail-1.7.5-17.i586.rpm rpm -ivh --test mozilla-mail-1.7.5-17.i586.rpm
rpm -Uvh {rpm-file}	Upgrade package	rpm -Uvh mozilla-mail-1.7.6-12.i586.rpm rpm -Uvh --test mozilla-mail-1.7.6-12.i586.rpm
rpm -ev {package}	Erase/remove/ an installed package	rpm -ev mozilla-mail
rpm -ev --nodeps {package}	Erase/remove/ an installed package without checking for dependencies	rpm -ev --nodeps mozilla-mail
rpm -qa	Display list all installed packages	rpm -qa rpm -qa less
rpm -qi {package}	Display installed information along with package version and short description	rpm -qi mozilla-mail
rpm -qf {/path/to/file}	Find out what package a file belongs to i.e. find what package owns the file	rpm -qf /etc/passwd rpm -qf /bin/bash
rpm -qc {pacakge-name}	Display list of configuration file(s) for a package	rpm -qc httpd
rpm -qcf {/path/to/file}	Display list of configuration files for a command	rpm -qcf /usr/X11R6/bin/xeyes
rpm -qa --last	Display list of all recently installed RPMs	rpm -qa --last rpm -qa --last less
rpm -qpR {.rpm-file} rpm -qR {package}	Find out what dependencies a rpm file has	rpm -qpR mediawiki-1.4rc1-4.i586.rpm rpm -qR bash

48. ping command

The Linux **ping** command is a simple utility used to check whether a network is available and if a host is reachable. With this command, we can test if a server is up and running. It also helps with troubleshooting various connectivity issues.

The **ping** command allows us to:

- Test our internet connection.
 - Check if a remote machine is online.
 - Analyze if there are network issues, such as dropped packages or high latency.

a.) **ping hostname:** The output for the ping command contains the amount of time it takes for every packet to reach its destination and return.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ping gmail.com
PING gmail.com (172.217.160.197) 56(84) bytes of data.
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=1 ttl=119 time=54.1 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=2 ttl=119 time=250 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=3 ttl=119 time=54.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=4 ttl=119 time=54.6 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=5 ttl=119 time=54.2 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=6 ttl=119 time=54.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=7 ttl=119 time=55.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=8 ttl=119 time=54.4 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=9 ttl=119 time=57.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=10 ttl=119 time=54.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=11 ttl=119 time=54.1 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=12 ttl=119 time=54.2 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=13 ttl=119 time=54.9 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=14 ttl=119 time=54.0 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=15 ttl=119 time=54.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=16 ttl=119 time=54.9 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=17 ttl=119 time=54.7 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=18 ttl=119 time=54.3 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=19 ttl=119 time=53.9 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=20 ttl=119 time=53.9 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=21 ttl=119 time=54.2 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=22 ttl=119 time=53.6 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=23 ttl=119 time=53.8 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=24 ttl=119 time=54.0 ms
64 bytes from bom07s16-in-f5.1e100.net (172.217.160.197): icmp_seq=25 ttl=119 time=56.0 ms
```

1. from: The destination and its IP address. Note that the IP address may be different for a website depending on your geographical location.

2. icmp_seq=1: The sequence number of each ICMP packet. Increases by one for every subsequent echo request.

3. ttl=119: The Time to Live value from 1 to 255. It represents the number of network hops a packet can take before a router discards it.

4. time=54.1 ms: The time it took a packet to reach the destination and come back to the source. Expressed in milliseconds.

b.) ping localhost: We can use the name to ping localhost. The name refers to our computer, and when we use this command, we say: “ping this computer.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.042 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.062 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.069 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.068 ms
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.074 ms
64 bytes from localhost (127.0.0.1): icmp_seq=6 ttl=64 time=0.065 ms
64 bytes from localhost (127.0.0.1): icmp_seq=7 ttl=64 time=0.069 ms
64 bytes from localhost (127.0.0.1): icmp_seq=8 ttl=64 time=0.096 ms
64 bytes from localhost (127.0.0.1): icmp_seq=9 ttl=64 time=0.057 ms
64 bytes from localhost (127.0.0.1): icmp_seq=10 ttl=64 time=0.071 ms
64 bytes from localhost (127.0.0.1): icmp_seq=11 ttl=64 time=0.075 ms
64 bytes from localhost (127.0.0.1): icmp_seq=12 ttl=64 time=0.049 ms
64 bytes from localhost (127.0.0.1): icmp_seq=13 ttl=64 time=0.040 ms
64 bytes from localhost (127.0.0.1): icmp_seq=14 ttl=64 time=0.075 ms
64 bytes from localhost (127.0.0.1): icmp_seq=15 ttl=64 time=0.053 ms
64 bytes from localhost (127.0.0.1): icmp_seq=16 ttl=64 time=0.052 ms
64 bytes from localhost (127.0.0.1): icmp_seq=17 ttl=64 time=0.055 ms
64 bytes from localhost (127.0.0.1): icmp_seq=18 ttl=64 time=0.056 ms
64 bytes from localhost (127.0.0.1): icmp_seq=19 ttl=64 time=0.062 ms
64 bytes from localhost (127.0.0.1): icmp_seq=20 ttl=64 time=0.051 ms
64 bytes from localhost (127.0.0.1): icmp_seq=21 ttl=64 time=0.069 ms
64 bytes from localhost (127.0.0.1): icmp_seq=22 ttl=64 time=0.065 ms
^C
--- localhost ping statistics ---
22 packets transmitted, 22 received, 0% packet loss, time 21488ms
rtt min/avg/max/mdev = 0.040/0.062/0.096/0.012 ms
vinayak@vinayak-Swift-SF315-52G:~$ □
```

c.) ping -c 5 hostname: To make the ping command automatically stop after it sends a certain number of packets, use -c and a number. This sets the desired amount of ping requests.

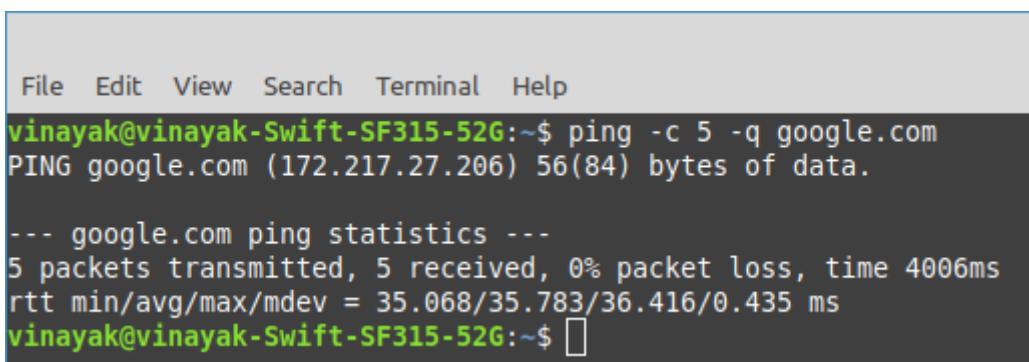
```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ping -c 5 google.com
PING google.com (142.250.67.142) 56(84) bytes of data.
64 bytes from bom12s06-in-f14.1e100.net (142.250.67.142): icmp_seq=1 ttl=119 time=37.7 ms
64 bytes from bom12s06-in-f14.1e100.net (142.250.67.142): icmp_seq=2 ttl=119 time=37.4 ms
64 bytes from bom12s06-in-f14.1e100.net (142.250.67.142): icmp_seq=3 ttl=119 time=37.6 ms
64 bytes from bom12s06-in-f14.1e100.net (142.250.67.142): icmp_seq=4 ttl=119 time=38.2 ms
64 bytes from bom12s06-in-f14.1e100.net (142.250.67.142): icmp_seq=5 ttl=119 time=37.8 ms
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 37.379/37.729/38.176/0.268 ms
vinayak@vinayak-Swift-SF315-52G:~$ □
```

d.) ping -w 5 hostname: To stop receiving a ping output after a specific amount of time, add -w and an interval in seconds to your command.

```
vinayak@vinayak-Swift-SF315-52G:~$ ping -w 5 google.com
PING google.com (172.217.27.206) 56(84) bytes of data.
64 bytes from bom07s15-in-f14.1e100.net (172.217.27.206): icmp_seq=1 ttl=119 time=36.1 ms
64 bytes from bom07s15-in-f14.1e100.net (172.217.27.206): icmp_seq=2 ttl=119 time=35.7 ms
64 bytes from bom07s15-in-f14.1e100.net (172.217.27.206): icmp_seq=3 ttl=119 time=36.2 ms
64 bytes from bom07s15-in-f14.1e100.net (172.217.27.206): icmp_seq=4 ttl=119 time=36.3 ms
64 bytes from bom07s15-in-f14.1e100.net (172.217.27.206): icmp_seq=5 ttl=119 time=36.4 ms

--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 35.724/36.158/36.409/0.239 ms
vinayak@vinayak-Swift-SF315-52G:~$ 
```

e.) ping -c 5 -q hostname: If we do not want to clog our screen with information for every packet, use the -q switch to display the summary only. The -q option prints one line with the regular ping information and then provides the statistics at the end. The letter “q” in this command stands for “quiet” output.



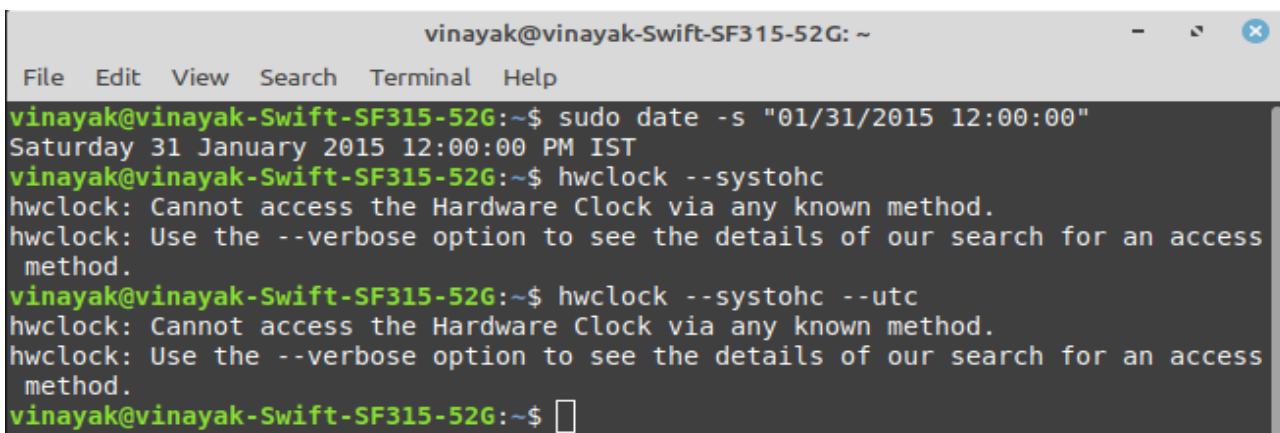
```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ping -c 5 -q google.com
PING google.com (172.217.27.206) 56(84) bytes of data.

--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 35.068/35.783/36.416/0.435 ms
vinayak@vinayak-Swift-SF315-52G:~$ 
```

49. date command

date command is used to display the system date and time. date command is also used to set date and time of the system. By default the date command displays the date in the time zone on which unix/linux operating system is configured.

a.) sudo date -s “01/31/2015 12:00:00”: Used to configure the date of the hardware.



```
vinayak@vinayak-Swift-SF315-52G:~$
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ sudo date -s "01/31/2015 12:00:00"
Saturday 31 January 2015 12:00:00 PM IST
vinayak@vinayak-Swift-SF315-52G:~$ hwclock --systohc
hwclock: Cannot access the Hardware Clock via any known method.
hwclock: Use the --verbose option to see the details of our search for an access
method.
vinayak@vinayak-Swift-SF315-52G:~$ hwclock --systohc --utc
hwclock: Cannot access the Hardware Clock via any known method.
hwclock: Use the --verbose option to see the details of our search for an access
method.
vinayak@vinayak-Swift-SF315-52G:~$ 
```

b.) date: The date command displays the current date and time, including the abbreviated day name, abbreviated month name, day of the month, the time separated by colons, the time zone name, and the year.

c.) date -u: Displays the time in GMT(Greenwich Mean Time)/UTC (Coordinated Universal Time)time zone.

d.) date --date="2 year ago": Date and time of 2 years ago.

e.) date --date="1 min ago": Date and time of 1 min ago.

f.) date --date="next sat": Date and time of next saturday.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ date
Saturday 29 August 2020 10:54:28 PM IST
vinayak@vinayak-Swift-SF315-52G:~$ date -u
Saturday 29 August 2020 05:24:31 PM UTC
vinayak@vinayak-Swift-SF315-52G:~$ date --date="2 year ago"
Wednesday 29 August 2018 10:54:48 PM IST
vinayak@vinayak-Swift-SF315-52G:~$ date --date="1 min ago"
Saturday 29 August 2020 10:54:03 PM IST
vinayak@vinayak-Swift-SF315-52G:~$ date --date="next sat"
Saturday 05 September 2020 12:00:00 AM IST
vinayak@vinayak-Swift-SF315-52G:~$ 
```

50. wget command

Wget command is a Linux command line utility that helps us to download the files from the web. We can download the files from web servers using HTTP, HTTPS and FTP protocols. We can use wget in scripts and cronjobs.

Wget is a non-interactive program so that it will run in the background. One of the good feature of wget command is mirriong using this we can download the complete website from the web.

a.) wget URL: The basic format to download a file from the web using Wget command.

b.) wget -m URL: -m option stands for turn on mirroring. We can download all the website files using this.

c.) wget -O URL: It downloads and rename the file.

d.) wget –version: It is used to check the version of wget.

File Edit View Search Terminal Help

```
vinayak@vinayak-SF315-52G:~/Documents/Test$ wget http://google.co.in/
--2020-08-26 17:17:16-- http://google.co.in/
Resolving google.co.in (google.co.in)... 216.58.203.3, 2404:6800:4009:804::2003
Connecting to google.co.in (google.co.in)|216.58.203.3|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.google.co.in/ [following]
--2020-08-26 17:17:16-- http://www.google.co.in/
Resolving www.google.co.in (www.google.co.in)... 142.250.67.163, 2404:6800:4009:812::2003
Connecting to www.google.co.in (www.google.co.in)|142.250.67.163|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html [=>] 13.90K --.-KB/s in 0s

2020-08-26 17:17:16 (77.8 MB/s) - 'index.html' saved [14229]

vinayak@vinayak-SF315-52G:~/Documents/Test$ ls
Asgn1 Asgn2 Asgn3 Asgn4 index.html Perm
vinayak@vinayak-SF315-52G:~/Documents/Test$ wget -m http://google.co.in/
--2020-08-26 17:18:06-- http://google.co.in/
Resolving google.co.in (google.co.in)... 172.217.166.35, 2404:6800:4009:804::2003
Connecting to google.co.in (google.co.in)|172.217.166.35|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.google.co.in/ [following]
--2020-08-26 17:18:06-- http://www.google.co.in/
Resolving www.google.co.in (www.google.co.in)... 142.250.67.163, 2404:6800:4009:812::2003
Connecting to www.google.co.in (www.google.co.in)|142.250.67.163|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'google.co.in/index.html'

google.co.in/index. [=>] 13.93K --.-KB/s in 0.04s

Last-modified header missing -- time-stamps turned off.
2020-08-26 17:18:06 (378 KB/s) - 'google.co.in/index.html' saved [14263]

FINISHED --2020-08-26 17:18:06--
Total wall clock time: 0.4s
Downloaded: 1 files, 14K in 0.04s (378 KB/s)
vinayak@vinayak-SF315-52G:~/Documents/Test$ ls
Asgn1 Asgn2 Asgn3 Asgn4 google.co.in index.html Perm
vinayak@vinayak-SF315-52G:~/Documents/Test$ 
```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Test

File Edit View Search Terminal Help

vinayak@vinayak-Swift-SF315-52G:~/Documents/Test\$ wget -O google http://google.co.in/

--2020-08-26 17:31:39-- http://google.co.in/

Resolving google.co.in (google.co.in)... 216.58.203.3, 2404:6800:4009:80c::2003

Connecting to google.co.in (google.co.in)|216.58.203.3|:80... connected.

HTTP request sent, awaiting response... 301 Moved Permanently

Location: http://www.google.co.in/ [following]

--2020-08-26 17:31:39-- http://www.google.co.in/

Resolving www.google.co.in (www.google.co.in)... 142.250.67.163, 2404:6800:4009:812::2003

Connecting to www.google.co.in (www.google.co.in)|142.250.67.163|:80... connected.

HTTP request sent, awaiting response... 200 OK

Length: unspecified [text/html]

Saving to: 'google'

google

[=>]

13.89K --.-KB/s in 0.04s

2020-08-26 17:31:39 (394 KB/s) - 'google' saved [14223]

vinayak@vinayak-Swift-SF315-52G:~/Documents/Test\$ ls

Asgn1 Asgn2 Asgn3 Asgn4 google google.co.in index.html index.html.1 index.html.2

[green]

Vinayak@vinayak-Swift-SF315-52G:~/Documents/Test\$ wget --version

GNU Wget 1.20.3 built on linux-gnu.

-cares +digest -gpgme +https +ipv6 +iri +large-file -metalink +nls

+ntlm +opie +psl +ssl/openssl

Wgetrc:

/etc/wgetrc (system)

Locale:

/usr/share/locale

Compile:

gcc -DHAVE_CONFIG_H -DSYSTEM_WGETRC="/etc/wgetrc"

-DLOCALEDIR="/usr/share/locale" -I . -I ../../src -I ./lib

-I ../../lib -Ddate-time -D FORTIFY_SOURCE=2 -DHAVE_LIBSSL -DNDEBUG

-g -O2 -fdebug-prefix-map=/build/wget-0YIfr9/wget-1.20.3=.

-fstack-protector-strong -Wformat -Werror=format-security

-DN0_SSLV2 -D FILE_OFFSET_BITS=64 -g -Wall

Link:

gcc -DHAVE_LIBSSL -DNDEBUG -g -O2

-fdebug-prefix-map=/build/wget-0YIfr9/wget-1.20.3=.

-fstack-protector-strong -Wformat -Werror=format-security

-DN0_SSLV2 -D FILE_OFFSET_BITS=64 -g -Wall -Wl,-Bsymbolic-functions

-Wl,-z,relro -Wl,-z,now -lpcre2-8 -luuid -ldn2 -lssl -lcrypto -lz

-lpsl ftp-opie.o openssl.o http-ntlm.o .. /lib/libgnu.a

Copyright (C) 2015 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later

<http://www.gnu.org/licenses/gpl.html>.

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

Originally written by Hrvoje Niksic <niksic@xemacs.org>.

Please send bug reports and questions to <bug-wget@gnu.org>.

Vinayak@vinayak-Swift-SF315-52G:~/Documents/Test\$

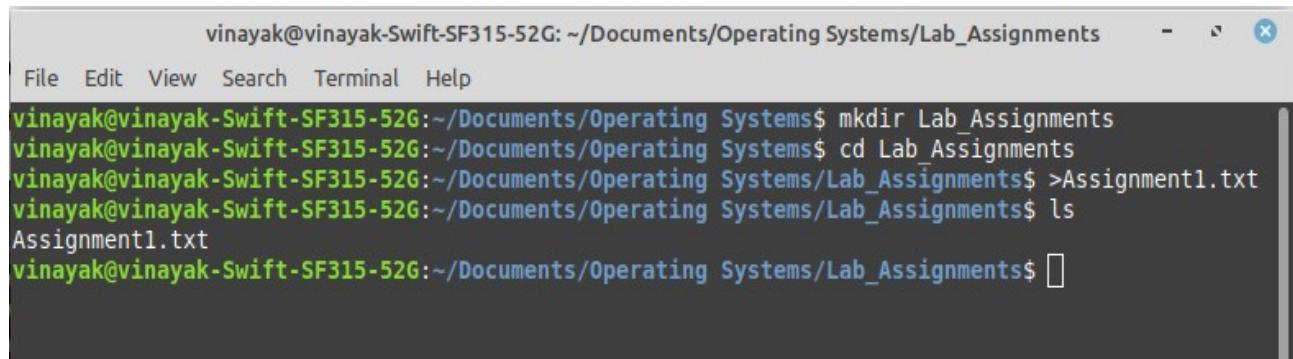
2. Create a directory and create a file inside that directory.

mkdir directory name: This command creates a new directory.

cd directory name: This command is used to go inside current directory(i.e. directory name).

>filename: Right angle bracket > symbol tells the system to output results into whatever we specified next. The target is usually a filename. We can use this symbol by itself to create a new file.

ls: This command is used to list all the contents of current directory .

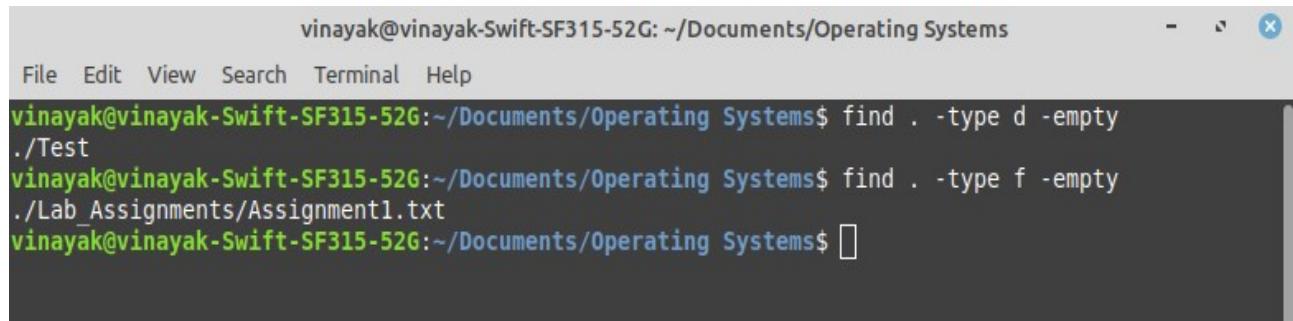


A screenshot of a terminal window titled "vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems/Lab_Assignments". The window has standard OS X-style controls at the top right. The terminal menu bar shows "File Edit View Search Terminal Help". The terminal session shows the following commands and output:
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems\$ mkdir Lab_Assignments
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems\$ cd Lab_Assignments
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments\$ >Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments\$ ls
Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments\$

3. List the files and directories that are empty in a working directory.

find . -type d -empty: This command is used to list empty directories in current directory.

find . -type f -empty: This command is used to list empty files in the current directory.



A screenshot of a terminal window titled "vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems". The window has standard OS X-style controls at the top right. The terminal menu bar shows "File Edit View Search Terminal Help". The terminal session shows the following commands and output:
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems\$ find . -type d -empty
. ./Test
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems\$ find . -type f -empty
. ./Lab_Assignments/Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems\$

4. Show commands to delete empty and non-empty directory.

rm -d directoryname: This command is used to delete the empty directory.

rm -r(--recursive or -R) directoryname: This command is used to delete a non-empty directory, and all of its contents.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt 'Lab Assignment 1.odt' sample2.txt Theory
Input.txt Lab_Assignments sample.txt
Lab sample1.txt Test
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cd Test
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Test$ ls
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Test$ cd ..
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ rm -d Test
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt 'Lab Assignment 1.odt' sample2.txt
Input.txt Lab_Assignments sample.txt
Lab sample1.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cd Lab
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab$ ls
Assignment Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab$ cd ..
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ rm -r Lab
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt 'Lab Assignment 1.odt' sample1.txt sample.txt
Input.txt Lab_Assignments sample2.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

5. Find the location of the input files using locate and find command.

locate file1name file2name: This command is used to locate the file in the system and it starts searching for home directory.

find -name filename: This command is used to find the file from the current directory. We used the -name option, and searched for a file called filename. Keep in mind that the -name argument looks for case-sensitive terms in Linux.

```
vinayak@vinayak-Swift-SF315-52G: ~
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ locate sample1.txt sample2.txt
/home/vinayak/Documents/Operating Systems/sample1.txt
/home/vinayak/Documents/Operating Systems/sample2.txt
vinayak@vinayak-Swift-SF315-52G:~$ find -name sample1.txt
./Documents/Operating Systems/sample1.txt
vinayak@vinayak-Swift-SF315-52G:~$ find -name sample2.txt
./Documents/Operating Systems/sample2.txt
vinayak@vinayak-Swift-SF315-52G:~$ 
```

6. View the user permissions and ownership of the files in the current directory and change the ownership of some selected files to another user.

sudo chown username filename: The chown command allows us to change the user and/or group ownership of a given file, directory, or symbolic link. We have to be root to change the owner of files/directories that's why **sudo** is used.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -l
total 8936
-rw-rw-r-- 1 vinayak vinayak 2413600 Aug 26 12:32 'Assignment 1.odt'
-rw-rw-r-- 1 vinayak vinayak      0 Aug 15 19:54 Assignment1.txt
-rw-rw-r-- 1 vinayak vinayak   1497 Aug 19 13:08 Input.txt
-rw-rw-r-- 1 vinayak vinayak 6695588 Aug 26 22:30 'Lab Assignment 1.odt'
drwxrwxr-x 2 vinayak vinayak   4096 Aug 26 12:50 Lab_Assignments
-rw-rw-r-- 1 vinayak vinayak    908 Aug 16 12:23 sample1.txt
-rw-rw-r-- 1 vinayak vinayak    908 Aug 16 12:23 SAMPLE1.txt
-rw-rw-r-- 1 vinayak vinayak    593 Aug 19 16:13 sample2.txt
-rw-rw-r-- 1 vinayak vinayak    589 Aug 19 15:59 sample2.txt.bak
-rw-rw-r-- 1 vinayak vinayak   1497 Aug 19 15:15 sample3.txt
-rw-rw-r-- 1 vinayak vinayak    593 Aug 26 23:17 sample.txt
drwxrwxr-x 3 vinayak vinayak   4096 Aug 25 23:28 Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ sudo chown rahul sample3.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -l
total 8936
-rw-rw-r-- 1 vinayak vinayak 2413600 Aug 26 12:32 'Assignment 1.odt'
-rw-rw-r-- 1 vinayak vinayak      0 Aug 15 19:54 Assignment1.txt
-rw-rw-r-- 1 vinayak vinayak   1497 Aug 19 13:08 Input.txt
-rw-rw-r-- 1 vinayak vinayak 6695588 Aug 26 22:30 'Lab Assignment 1.odt'
drwxrwxr-x 2 vinayak vinayak   4096 Aug 26 12:50 Lab_Assignments
-rw-rw-r-- 1 vinayak vinayak    908 Aug 16 12:23 sample1.txt
-rw-rw-r-- 1 vinayak vinayak    908 Aug 16 12:23 SAMPLE1.txt
-rw-rw-r-- 1 vinayak vinayak    593 Aug 19 16:13 sample2.txt
-rw-rw-r-- 1 vinayak vinayak    589 Aug 19 15:59 sample2.txt.bak
-rw-rw-r-- 1 rahul   vinayak   1497 Aug 19 15:15 sample3.txt
-rw-rw-r-- 1 vinayak vinayak    593 Aug 26 23:17 sample.txt
drwxrwxr-x 3 vinayak vinayak   4096 Aug 25 23:28 Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

7. List all the files in the current directory and subdirectories.

ls -R: This command is used to list all the files in the current directory and subdirectories.

```

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -R
.:
Assignment1.txt  'Lab Assignment 1.odt'  sample1.txt  sample.txt
Input.txt        Lab_Assignments          sample2.txt  Theory

./Lab_Assignments:
Assignment  Assignment1.txt

./Theory:
mycopy mycopy.c mysort mysort.c mysortfp.c remove remove.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

8. Concatenate the two input files: “sample1.txt” and “sample2.txt” and save it to a new file named “Input”.

cat file1name file2name > mergefilename: This command will merge the contents of file in respective order and will insert that content in "mergefilename". The file listed after the output redirection symbol will be overwritten, if it already exists.

cat filename: This command will display the content on the stdout.

```
File Edit View Search Terminal Help vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNIXOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt sample2.txt > Input.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt 'Lab Assignment 1.odt' sample1.txt sample.txt
Input.txt Lab Assignments sample2.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat Input.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNIXOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ]
```

9. Copy the contents of file ‘sample2.txt’ to ‘sample.txt’

cp sourcefile destinationfile: This command is used to copy content of source file to destination file.

cat filename: This command will display the content on the stdout.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cp sample2.txt sample.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment          Input.txt           Lab_Assignments   sample2.txt  Test
Assignment1.txt     'Lab Assignment 1.odt' sample1.txt    sample.txt  Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas M cIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

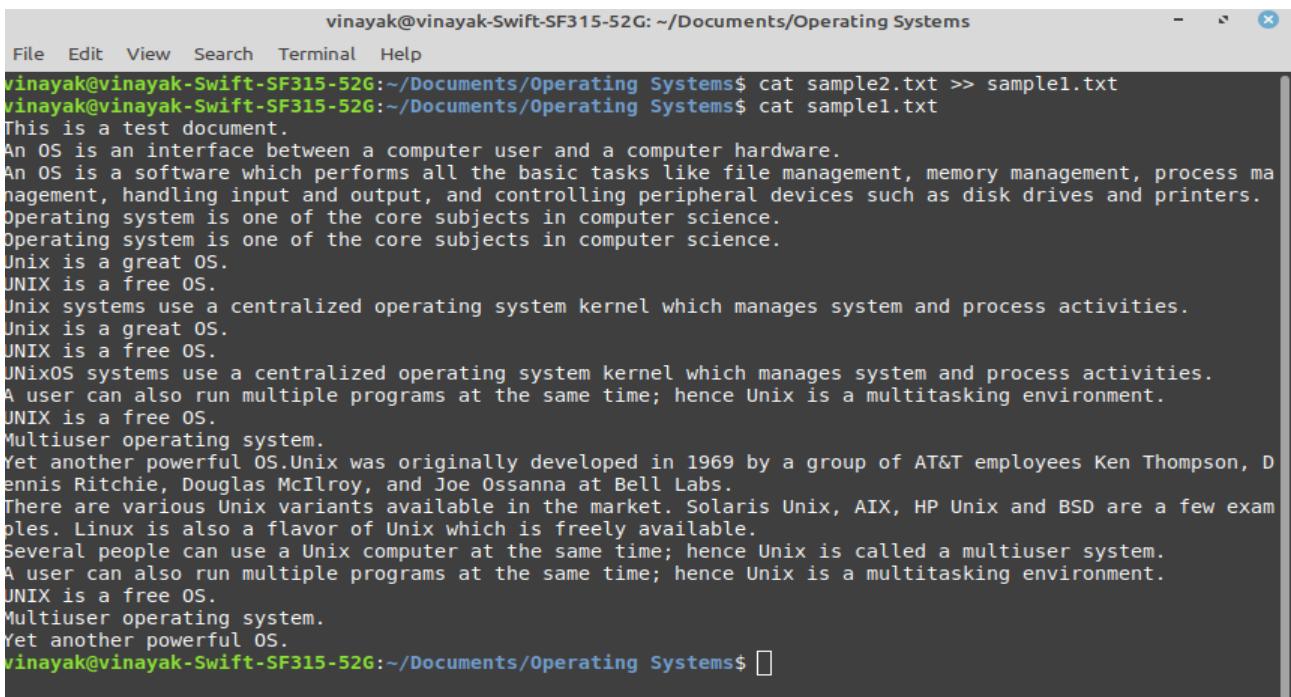
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas M cIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

10. Append the file contents of input file ‘sample2.txt’ to the end of the first input file ‘sample1.txt’.

cat sourcefile >> destinationfile: This command appends the content of source file at the end of destination file content. The >> appends to a file or creates the file if it doesn't exist.

cat filename: This command will display the content on the stdout.



```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample2.txt >> sample1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNIxOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

11. Remove the permission for the users to read, write and execute the file ‘sample.txt’.

ls -l: This command is used to list all the file/directory permissions of current working directory.

The chmod command is used to change access permissions for files which we own. The syntax is

```
chmod      permission triads      filename
[who][action][permissions]
```

where:

who	action	permissions
u = user	+ = add	r = read
g = group	- = remove	w = write
o = other		x = execute
a = all		

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -l
total 1476
-rw-rw-r-- 1 vinayak vinayak 0 Aug 15 19:54 Assignment1.txt
-rw-rw-r-- 1 vinayak vinayak 1497 Aug 15 23:24 Input.txt
-rw-rw-r-- 1 vinayak vinayak 1484992 Aug 16 18:33 'Lab Assignment 1.odt'
drwxrwxr-x 2 vinayak vinayak 4096 Aug 16 13:26 Lab_Assignments
-rw-rw-r-- 1 vinayak vinayak 908 Aug 16 12:23 sample1.txt
-rw-rw-r-- 1 vinayak vinayak 589 Aug 15 23:47 sample2.txt
-rw-rw-r-- 1 vinayak vinayak 591 Aug 15 23:35 sample.txt
drwxrwxr-x 2 vinayak vinayak 4096 Aug 14 21:23 Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ chmod u-r-w sample.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls -l
total 1476
-rw-rw-r-- 1 vinayak vinayak 0 Aug 15 19:54 Assignment1.txt
-rw-rw-r-- 1 vinayak vinayak 1497 Aug 15 23:24 Input.txt
-rw-rw-r-- 1 vinayak vinayak 1484992 Aug 16 18:33 'Lab Assignment 1.odt'
drwxrwxr-x 2 vinayak vinayak 4096 Aug 16 13:26 Lab_Assignments
-rw-rw-r-- 1 vinayak vinayak 908 Aug 16 12:23 sample1.txt
-rw-rw-r-- 1 vinayak vinayak 589 Aug 15 23:47 sample2.txt
-rw-rw-r-- 1 vinayak vinayak 591 Aug 15 23:35 sample.txt
drwxrwxr-x 2 vinayak vinayak 4096 Aug 14 21:23 Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample.txt
cat: sample.txt: Permission denied

```

12. Display the current date with the day of week, month, time and the year.

date: This command displays the current date and time, including the abbreviated day name, abbreviated month name, day of the month, the time separated by colons, the time zone name, and the year.

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ date
Saturday 15 August 2020 11:59:14 PM IST
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 

```

13. Show the calendar of previous, current and next month.

cal -3: This command shows calender of previous,current and next month calendar on the terminal

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cal -3
2020
    July           August          September
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
      1   2   3   4       2   3   4   5   6   7   8       1   2   3   4   5
      5   6   7   8   9   10  11     9   10  11  12  13  14  15     6   7   8   9   10  11  12
     12  13  14  15  16  17  18     17  18  19  20  21  22     13  14  15  16  17  18  19
     19  20  21  22  23  24  25     26  27  28  29  30           20  21  22  23  24  25  26
     26  27  28  29  30  31           30  31           27  28  29  30
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 

```

14. Sort the contents of the file ‘sample1.txt’ in alphabetical order.

sort filename: This command is used without any option, it sorts content alphabetically based on the first letter of each line.

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ sort sample1.txt
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
Multiuser operating system.
Multiuser operating system.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
This is a test document.
UNIX is a free OS.
Unix is a great OS.
Unix is a great OS.
UNIxOS systems use a centralized operating system kernel which manages system and process activities.
Unix systems use a centralized operating system kernel which manages system and process activities.
Yet another powerful OS.
Yet another powerful OS.Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

15. Erase duplicate records in the file ‘sample1.txt’ and display only the unique records

awk '!a[\$0]++' filename: This is very tricky. awk uses associative arrays to remove duplicates here. When a pattern appears for the 1st time, count for the pattern is incremented. This will still make the count as 0 since it is a postfix, and the negation of 0 which is 'True' makes the pattern printed. When the same pattern appears again, the count is now 1 and hence the inverse is 'False' and hence the pattern does not get printed.

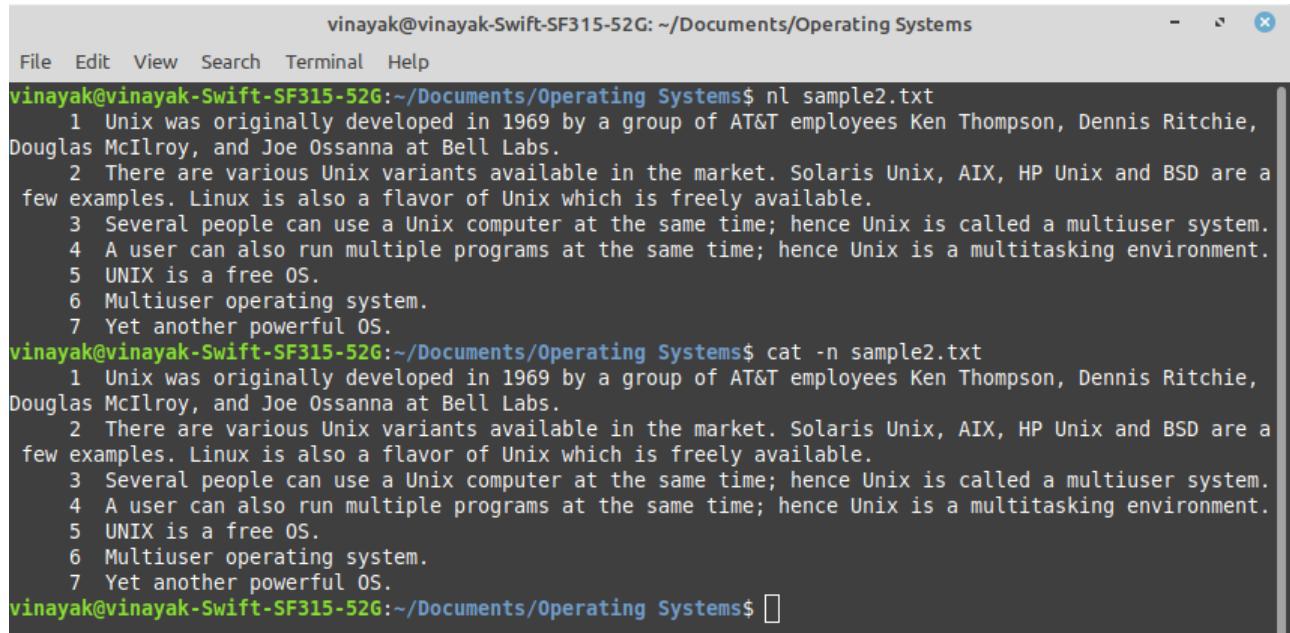
```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ awk '!a[$0]++' sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
UNIxOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

16. Add line numbers to the file ‘sample2.txt’

nl filename: This command is dedicated for adding line numbers to a file. It writes the given file to standard output, with line numbers added.

Or

cat -n filename: If we want to add numbers to the output of a file, use -n flag like below.



vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ nl sample2.txt
1 Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
2 There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
3 Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
4 A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
5 UNIX is a free OS.
6 Multiuser operating system.
7 Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat -n sample2.txt
1 Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
2 There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
3 Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
4 A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
5 UNIX is a free OS.
6 Multiuser operating system.
7 Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

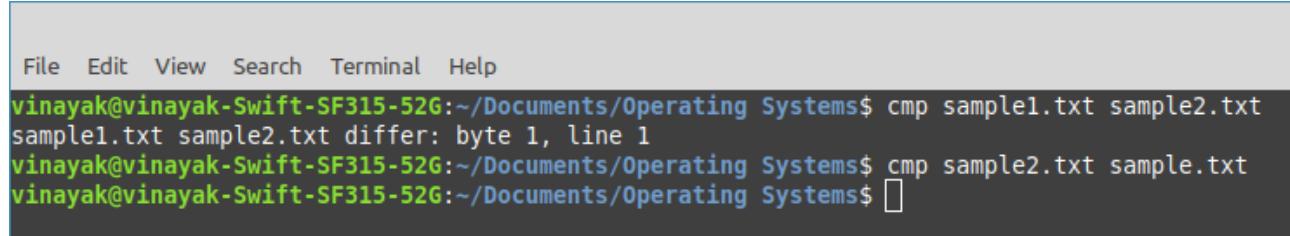
17. Find out whether the two pairs of input files are identical or not.

Compare sample1.txt and sample2.txt

Compare sample2.txt and sample.txt

cmp file1name file2name: cmp command reports the byte and line number if a difference is found. Byte 1, line 1 indicating that the first mismatch found in two files at byte 1 in first line.

If there is no output shown it means both files are identical which means sample2.txt and sample.txt are identical.



```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cmp sample1.txt sample2.txt
sample1.txt sample2.txt differ: byte 1, line 1
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cmp sample2.txt sample.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

18. Show how the input file “sample1.txt” differs line by line from “sample2.txt” in context and unified mode.

diff -c file1name file2name: To view differences in context mode, use the -c option.

The first file is indicated by ***, and the second file is indicated by ---.

The line with ***** is just a separator.

The first two lines of this output show us information about file 1 and file 2. It lists the file name, modification date, and modification time of each of our files, one per line.

The next line has three asterisks *** followed by a line range from the first file (in our case lines 1 through 15, separated by a comma). Then four asterisks ****. After that it shows the contents of the first file with the following indicators:

- (i) If the line needs to be unchanged, it is prefixed by two spaces.
 - (ii) If the line needs to be changed, it is prefixed by an symbol and a space.
- The symbol means are as follows:

- (a) + : It indicates a line in the second file that needs to be added to the first file to make them identical.
- (b) - : It indicates a line in the first file that needs to be deleted to make them identical.
- (c) ! : Indicates that this line is part of a group of one or more lines that needs to change. There is a corresponding group of lines prefixed with "!" in the other file's context as well.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ diff -c sample1.txt sample2.txt
*** sample1.txt 2020-08-16 12:23:46.073997942 +0530
--- sample2.txt 2020-08-19 16:13:41.812197351 +0530
*****
*** 1,15 ****
! This is a test document.
! An OS is an interface between a computer user and a computer hardware.
! An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
! Operating system is one of the core subjects in computer science.
! Operating system is one of the core subjects in computer science.
! Unix is a great OS.
! UNIX is a free OS.
! Unix systems use a centralized operating system kernel which manages system and process activities.
! Unix is a great OS.
! UNIX is a free OS.
! UNIXOS systems use a centralized operating system kernel which manages system and process activities.
! A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
! UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
--- 1,9 ----
! Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
! There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
! Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
! A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
! UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
+
+
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

diff -u file1name file2name: To view differences in unified mode, use the -u option. It is similar to context mode but it doesn't display any **redundant information** or it shows the information in concise form.

The first file is indicated by ---, and the second file is indicated by +++.

The first two lines of this output show us information about file 1 and file 2. It lists the file name, modification date, and modification time of each of our files, one per line.

After that the next line has two at sign @ followed by a line range from the first file (in our case lines 1 through 15, separated by a comma) prefixed by – and then space and then again followed by a line range from the second file

prefixed by + and at the end two at sign @. Followed by the file content in output tells us which line remain unchanged and which lines needs to added or deleted(indicated by symbols) in the file 1 to make it identical to file 2.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ diff -u sample1.txt sample2.txt
--- sample1.txt 2020-08-16 12:23:46.673997942 +0530
+++ sample2.txt 2020-08-19 16:13:41.812197351 +0530
@@ -1,15 +1,9 @@
-This is a test document.
-An OS is an interface between a computer user and a computer hardware.
-An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
-Operating system is one of the core subjects in computer science.
-Operating system is one of the core subjects in computer science.
-Unix is a great OS.
-UNIX is a free OS.
-Unix systems use a centralized operating system kernel which manages system and process activities.
-Unix is a great OS.
-UNIX is a free OS.
-UNIXOS systems use a centralized operating system kernel which manages system and process activities.
+Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
+There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
+Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
+A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
+
+
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

19. Solve the arithmetic expression: $((8+12)*(5-3))/2$ using linux commands.

awk ‘BEGIN {“variable assignment”; print “ “, expression}’: This command evaluates the given expression and outputs the result to standard output.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ awk 'BEGIN {a=8; b=12; c=5; d=3; e=2; print ((a+b)*(c-d))/e}'
20
vinayak@vinayak-Swift-SF315-52G:~$
```

20. Cut and display the first 10 characters of every line of the file “Input.txt”.

cut -c -n filename: To cut by character use the -c option. This selects the characters given to the -c option. This command prints starting position to the nth character.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cut -c -10 Input.txt
This is a
An OS is a
An OS is a
Operating
Operating
Unix is a
UNIX is a
Unix syste
Unix is a
UNIX is a
UNixOS sys
A user can
UNIX is a
Multiuser
Yet anothe
There are
Several pe
A user can
UNIX is a
Multiuser
Yet anothe
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

21. Print the name of the current working directory.

pwd: This command prints the current working directory.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ pwd
/home/vinayak/Documents/Operating Systems
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

22. Process Status

a. List all the running processes with their corresponding PIDs.

ps -e: This command can list all the processes running on a Linux system with the -e option.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -e
  PID TTY      TIME CMD
    1 ?        00:00:01 systemd
    2 ?        00:00:00 kthread
    3 ?        00:00:00 rcu_gp
    4 ?        00:00:00 rcu_par_gp
    6 ?        00:00:00 kworker/0:0H-kblockd
    9 ?        00:00:00 mm_percpu_wq
   10 ?       00:00:00 ksoftirqd/0
   11 ?       00:00:03 rcu_sched
   12 ?       00:00:00 migration/0
   13 ?       00:00:00 idle_inject/0
   14 ?       00:00:00 cpuhp/0
   15 ?       00:00:00 cpuhp/1
   16 ?       00:00:00 idle_inject/1
   17 ?       00:00:00 migration/1
   18 ?       00:00:00 ksoftirqd/1
   20 ?       00:00:00 kworker/1:0H-events_highpri
   21 ?       00:00:00 cpuhp/2
   22 ?       00:00:00 idle_inject/2
   23 ?       00:00:00 migration/2
   24 ?       00:00:00 ksoftirqd/2
   26 ?       00:00:00 kworker/2:0H-kblockd
   27 ?       00:00:00 cpuhp/3
   28 ?       00:00:00 idle_inject/3
   29 ?       00:00:00 migration/3
   30 ?       00:00:00 ksoftirqd/3
   32 ?       00:00:00 kworker/3:0H-kblockd
   33 ?       00:00:00 cpuhp/4
   34 ?       00:00:00 idle_inject/4
   35 ?       00:00:00 migration/4
   36 ?       00:00:00 ksoftirqd/4
   38 ?       00:00:00 kworker/4:0H-events_highpri
   39 ?       00:00:00 cpuhp/5
   40 ?       00:00:00 idle_inject/5
   41 ?       00:00:00 migration/5
   42 ?       00:00:00 ksoftirqd/5
   44 ?       00:00:00 kworker/5:0H-kblockd
   45 ?       00:00:00 cpuhp/6
   46 ?       00:00:00 idle_inject/6
   47 ?       00:00:00 migration/6
   48 ?       00:00:00 ksoftirqd/6
   50 ?       00:00:00 kworker/6:0H-kblockd
   51 ?       00:00:00 cpuhp/7
   52 ?       00:00:00 idle_inject/7
   53 ?       00:00:00 migration/7
   54 ?       00:00:00 ksoftirqd/7
   56 ?       00:00:00 kworker/7:0H-kblockd
   57 ?       00:00:00 kdevtmpfs
   58 ?       00:00:00 netns
   59 ?       00:00:00 rcu_tasks_kthre
   60 ?       00:00:00 kauditd
   63 ?       00:00:00 khungtaskd
   64 ?       00:00:00 oom_reaper
   65 ?       00:00:00 writeback
   66 ?       00:00:00 kcompactd0
   67 ?       00:00:00 ksmd
   68 ?       00:00:00 khugepaged

```

b. List the processes that are not associated with the terminal.

ps -a: This command is used to view all processes with the exception of processes associated with the terminal and session leaders. A session leader is a process that starts other processes

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -a
  PID TTY      TIME CMD
 15252 pts/1    00:00:00 ps
vinayak@vinayak-Swift-SF315-52G:~$ 

```

c. List the processes that are associated with the terminal.

ps -T: This command is used to view processes associated with the terminal.

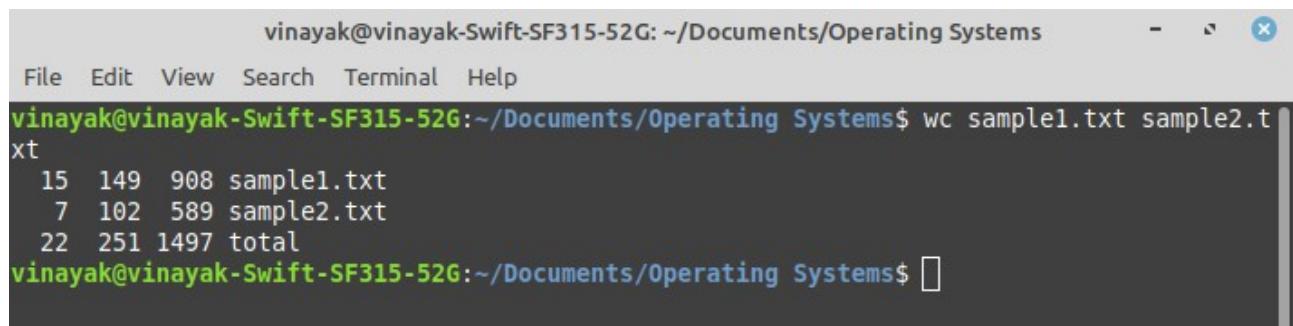
```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ ps -T
  PID  SPID TTY      TIME CMD
 14180  14180 pts/1    00:00:00 bash
 16289  16289 pts/1    00:00:00 ps
vinayak@vinayak-Swift-SF315-52G:~$ 

```

23. Print the number of characters, number of lines and number of words all the given input files.

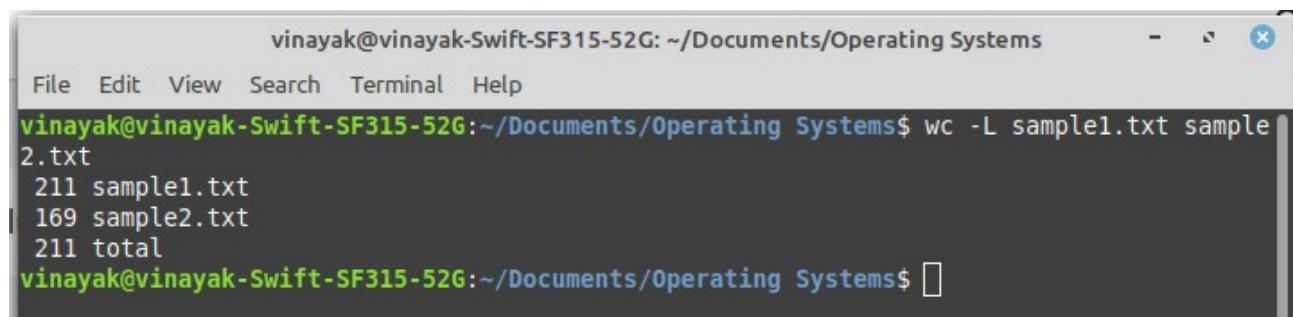
wc file1name file2name: The ‘wc’ command without passing any parameter will display a basic result of file. For ‘sample1.txt’, the three numbers shown below are 15(number of lines), 149(number of words) and 908(number of bytes) of the file. Similar is for ‘sample2.txt’. It can take multiple files as input.



```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ wc sample1.txt sample2.txt
 15 149 908 sample1.txt
    7 102 589 sample2.txt
   22 251 1497 total
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

24. Print the length of the longest line from all the input files.

wc -L file1name file2name: The ‘wc’ command allow an argument ‘-L’, it can be used to print out the length of longest (number of characters) line in a file. It can take multiple file names as Input.



```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ wc -L sample1.txt sample2.txt
211 sample1.txt
169 sample2.txt
211 total
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

25. Move the contents of the input file sample.txt to a new file.

mv -i sourcefile destinationfile: mv stands for move. mv is used to move one or more files or directories from one place to another in file system like UNIX. **-i (Interactive):** Like in cp, the -i option makes the command ask the user for confirmation before moving a file that would overwrite an existing file, you have to press **y** for confirm moving, any other key leaves the file as it is. This option doesn't work if the file doesn't exist, it simply rename it or move it to new location.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt Input.txt 'Lab Assignment 1.odt' Lab_Assignments sample1.txt sample2.txt sample.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ mv -i sample.txt samplemv.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt Input.txt 'Lab Assignment 1.odt' Lab_Assignments sample1.txt sample2.txt samplemv.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat samplemv.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 

```

26. Copy the contents of one directory to another directory.

cd -r directory1 directory2: If we want to copy a directory recursively from one location to another using the cp command, use the **-r/R** option with the cp command. It copies the folder including the subdirectories and their files to the target directory. If we do not have destination directory it also creates it.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cd Lab_Assignments
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments$ ls
Assignment Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments$ cd ..
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt 'Lab Assignment 1.odt' sample1.txt sample.txt Theory
Input.txt Lab_Assignments sample2.txt Test
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cp -r Lab_Assignments Lab
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt Lab_Assignments sample2.txt Test
Input.txt 'Lab Assignment 1.odt' sample1.txt sample.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cd Lab
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab$ ls
Assignment Assignment1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab$ 

```

27. Reverse the lines of the two input files and concatenate the file contents using a single command.

tac file1name file2name > mergefilename: tac command in Linux is used to concatenate and print files in reverse. This command will merge the contents of file in respective order and will insert that content in "mergefilename". The file listed after the output redirection symbol will be overwritten, if it already exists.

cat filename: This command will display the content on the stdout.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tac sample1.txt sample2.txt > sample3.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt Input.txt 'Lab Assignment 1.odt' Lab_Assignments sample1.txt sample2.txt sample3.txt samplemv.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample3.txt
Yet another powerful OS.
Multiuser operating system.
UNIX is a free OS.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIXxOS systems use a centralized operating system kernel which manages system and process activities.
UNIX is a free OS.
Unix is a great OS.
UNIX systems use a centralized operating system kernel which manages system and process activities.
UNIX is a free OS.
Unix is a great OS.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
An OS is an interface between a computer user and a computer hardware.
This is a test document.
Yet another powerful OS.
Multiuser operating system.
UNIX is a free OS.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
UNIX systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UNIXxOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

28. Delete all the files with *.txt extension from the working directory using yes command.

yes |rm -i *.txt: The **yes** command is used to output "y", continually, until it is aborted. This command removes all files with the extension **.txt** from the current directory. Here, **yes** outputs a constant stream of "y" characters and that output is piped to the '**rm -i**' command, which prompts for confirmation before deleting each file. The 'y' character from the **yes** command will respond "yes" to each prompt automatically.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments$ ls
Assignment Assignment1.txt Assignment2.txt Assignment3.txt Assignment4.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments$ yes |rm -i *.txt
rm: remove regular empty file 'Assignment1.txt'? rm: remove regular empty file 'Assignment2.txt'? rm: remove regular empty file 'Assignment3.txt'? rm: remove regular empty file 'Assignment4.txt'?
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments$ ls
Assignment
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems/Lab_Assignments$ 
```

29. Given the input file “sample1.txt”, print the number of the lines that match the pattern “system”.

grep -c “pattern” filename: The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern. The pattern that is searched in the file is referred to as the regular expression (grep stands for globally search for regular expression and print out). grep allows an argument “-c”, it prints only a count of the lines that match a pattern.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -c "system" sample1.txt
5
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

30. Having sample1 file as input, print the matched lines that contain the pattern “Unix” as whole words.

grep -w “pattern” filename: By default, grep matches the given string/pattern even if it found as a substring in a file. The -w option to grep makes it match only the whole words.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -w "Unix" sample1.txt
Unix is a great OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

31. Print the lines from “sample1.txt” that do not match the pattern “OS”.

grep -v “pattern” filename: We can display the lines that are not matched with the specified search sting pattern using the -v option.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -v "OS" sample1.txt
This is a test document.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
Multiuser operating system.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

32. Fetch the files that contain the word “OS”, “Operating System”, “Operating Systems” with its respective line number. (Ignore the case).

grep -n -i “pattern 1|pattern2” *: This command will display the lines with their line number(due to -n) which matches the given patterns. -i is used to ignore the case. * will search all the files in the directory.

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -n -i "Operating System\\OS" *
Binary file Assignment 1.odt matches
Input #1:2:An OS is an interface between a computer user and a computer hardware.
Input #1:3:An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Input #1:4:Operating system is one of the core subjects in computer science.
Input #1:5:Operating system is one of the core subjects in computer science.
Input #1:6:Unix is a great OS.
Input #1:7:UNIX is a free OS.
Input #1:8:Unix systems use a centralized operating system kernel which manages system and process activities.
Input #1:9:Unix is a great OS.
Input #1:10:UNIX is a free OS.
Input #1:11:UnixOS systems use a centralized operating system kernel which manages system and process activities.
Input #1:12:Unix is a free OS.
Input #1:13:Multiuser operating system.
Input #1:14:Yet another powerful OS.
Input #1:15:Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Osanna at Bell Labs.
Input #1:16:UNIX is a free OS.
Input #1:17:Multiuser operating system.
Input #1:18:Yet another powerful OS.
Binary file Lab Assignment 1.odt matches
grep: Lab Assignments: Is a directory
Input #2:1:2:An OS is an interface between a computer user and a computer hardware.
Input #2:2:An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Input #2:3:Operating system is one of the core subjects in computer science.
Input #2:4:Operating system is one of the core subjects in computer science.
Input #2:5:Unix is a great OS.
Input #2:6:UNIX is a free OS.
Input #2:7:Unix systems use a centralized operating system kernel which manages system and process activities.
Input #2:8:Unix is a great OS.
Input #2:9:UNIX is a free OS.
Input #2:10:Multiuser operating system.
Input #2:11:Yet another powerful OS.
Input #2:12:An OS is an interface between a computer user and a computer hardware.
Input #2:13:An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Input #2:14:Operating system is one of the core subjects in computer science.
Input #2:15:Operating system is one of the core subjects in computer science.
Input #2:16:Unix is a great OS.
Input #2:17:UNIX is a free OS.
Input #2:18:Unix systems use a centralized operating system kernel which manages system and process activities.
Input #2:19:Unix is a great OS.
Input #2:20:UNIX is a free OS.
Input #2:21:Multiuser operating system.
Input #2:22:Yet another powerful OS.
SAMPLE:1:2:An OS is an interface between a computer user and a computer hardware.
SAMPLE:1:3:An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
SAMPLE:1:4:Operating system is one of the core subjects in computer science.
SAMPLE:1:5:Operating system is one of the core subjects in computer science.
SAMPLE:1:6:Unix is a great OS.
SAMPLE:1:7:UNIX is a free OS.
SAMPLE:1:8:Unix systems use a centralized operating system kernel which manages system and process activities.
SAMPLE:1:9:Unix is a great OS.
SAMPLE:1:10:UNIX is a free OS.
SAMPLE:1:11:UnixOS systems use a centralized operating system kernel which manages system and process activities.
SAMPLE:1:12:Unix is a free OS.
SAMPLE:1:13:Multiuser operating system.
SAMPLE:1:14:Yet another powerful OS.
SAMPLE:1:15:Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Osanna at Bell Labs.
SAMPLE:1:16:UNIX is a free OS.
SAMPLE:1:17:Multiuser operating system.
SAMPLE:1:18:Yet another powerful OS.
SAMPLE:1:19:bak:1:Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Osanna at Bell Labs.
SAMPLE:1:20:bak:2:UNIX is a free OS.
SAMPLE:1:21:bak:3:Multiuser operating system.
SAMPLE:1:22:bak:4:Yet another powerful OS.
SAMPLE:1:23:bak:5:Multiuser operating system.
SAMPLE:1:24:bak:6:Multiuser operating system.
SAMPLE:1:25:bak:7:Yet another powerful OS.
SAMPLE:1:26:bak:8:Multiuser operating system.
SAMPLE:1:27:bak:9:Yet another powerful OS.
SAMPLE:1:28:bak:10:Multiuser operating system.
SAMPLE:1:29:bak:11:UNIX is a free OS.
```

33. Having “sample1.txt” and “core” as the input and pattern respectively, along with the matched line print three lines before and after the pattern match.

grep -B3 -A3 “pattern” filename: This command is used to print 3 lines below and after the given pattern along with the matched line. Here -B represents below the matched pattern, -A represents after the matched pattern and 3 is the number of lines to print.

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ grep -B3 -A3 "core" sample.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 

```

34. Find and replace the string “OS” with “Operating System”.

sed 's/searchpattern/replacementstring/g' filename: sed command is mostly used to replace the text in a file. Here the “s” specifies the substitution operation. The “/” are delimiters. The substitute flag /g (global replacement) specifies the sed command to replace all the occurrences of the string in the line.

```

File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt sample2.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ sed 's/OS/Operating System/g' sample1.txt sample2.txt
This is a test document.
An Operating System is an interface between a computer user and a computer hardware.
An Operating System is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great Operating System.
UNIX is a free Operating System.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great Operating System.
UNIX is a free Operating System.
Unix operating systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free Operating System.
Multiuser operating system.
Yet another powerful Operating System.
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free Operating System.
Multiuser operating system.
Yet another powerful Operating System.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 

```

35. List only the text files in the current working directory with its corresponding disk space occupied.

du -h *.txt: du command-line utility helps us to find out the disk usage of set of files or a directory. -h option is used to produce the output in human readable format. *.txt is used so that only the text files will be taken into consideration for disk usage details. Here K is Kilobytes.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt 'Lab Assignment 1.odt' sample1.txt sample.txt
Input.txt Lab_Assignments sample2.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ du -h *.txt
0 Assignment1.txt
4.0K Input.txt
4.0K sample1.txt
4.0K sample2.txt
4.0K sample.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

36. Show the last modification time of all the input text files.

date -r filename: date command with **-r** option followed by the name of file will display the last modified date and time of the file, where **-r** option is used to display the last modification date of the file.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ date -r sample1.txt
Sunday 16 August 2020 12:23:46 PM IST
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ date -r sample2.txt
Saturday 15 August 2020 11:47:16 PM IST
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

37. Delete the line that has the word “Powerful” from text file “sample2.txt”.

sed i.bak '/pattern/d' filename: SED command in UNIX is stands for stream editor. We will be using the **d** command to delete specific pattern space with sed. If we need to delete the lines but create a backup of the original file, then use option **-i** in the format, **-i.bak** This will create filename.bak of the original file name.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
Assignment1.txt Input.txt 'Lab Assignment 1.odt' Lab_Assignments sample1.txt sample2.txt sample3.txt samplemv.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ sed -i.bak '/powerful/d' sample2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample2.txt
Unix was originally developed in 1969 by a group of AT&T employees Ken Thompson, Dennis Ritchie, Douglas McIlroy, and Joe Ossanna at Bell Labs.
There are various Unix variants available in the market. Solaris Unix, AIX, HP Unix and BSD are a few examples. Linux is also a flavor of Unix which is freely available.
Several people can use a Unix computer at the same time; hence Unix is called a multiuser system.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

38. Print the roll numbers that end with even numbers in the format (COE18B002) up to COE18B050.

```
for i in {2..50..2} //i moves from 2 to 50 in steps of 2
$i -lt 10 // checks if i is less than 10
echo -n "pattern"; //prints the pattern on the stdout
```

The screenshot shows a terminal window titled "vinayak@vinayak-Swift-SF315-52G: ~". The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". The title bar says "GNU nano 4.8" and "RollCount.sh". The main area contains the following shell script:

```
for i in {2..50..2}
do
    if [ $i -lt 10 ]
    then
        echo -n "COE18B00$i";
    else
        echo -n "COE18B0$i";
    fi
    echo;
done
```

At the bottom, there is a status bar with the message "[Read 11 lines]". Below the status bar are several keyboard shortcuts:

- ^G Get Help
- ^O Write Out
- ^W Where Is
- ^K Cut Text
- ^J Justify
- ^C Cur Pos
- ^X Exit
- ^R Read File
- ^V Replace
- ^U Paste Text
- ^T To Spell
- ^L Go To Line

The screenshot shows a terminal window with the following command history and output:

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~$ nano RollCount.sh
vinayak@vinayak-Swift-SF315-52G:~$ bash RollCount.sh
COE18B002
COE18B004
COE18B006
COE18B008
COE18B010
COE18B012
COE18B014
COE18B016
COE18B018
COE18B020
COE18B022
COE18B024
COE18B026
COE18B028
COE18B030
COE18B032
COE18B034
COE18B036
COE18B038
COE18B040
COE18B042
COE18B044
COE18B046
COE18B048
COE18B050
```

The terminal prompt at the bottom is "vinayak@vinayak-Swift-SF315-52G:~\$".

39. Use filter commands like head, tail, more to view the file contents page by page.

head filename | more -1: head command is used to print from the starting line of the file. more -{n} prints n lines at a time.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ head sample1.txt | more -1
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
--More--
```

tail filename | more -1: tail command is used to print from the last line of the file. more -{n} prints n lines at a time.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ cat sample1.txt
This is a test document.
An OS is an interface between a computer user and a computer hardware.
An OS is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
Operating system is one of the core subjects in computer science.
Operating system is one of the core subjects in computer science.
Unix is a great OS.
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UnixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
Yet another powerful OS.

vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tail sample1.txt | more -1
UNIX is a free OS.
Unix systems use a centralized operating system kernel which manages system and process activities.
Unix is a great OS.
UNIX is a free OS.
UnixOS systems use a centralized operating system kernel which manages system and process activities.
A user can also run multiple programs at the same time; hence Unix is a multitasking environment.
UNIX is a free OS.
Multiuser operating system.
--More--
```

40. Compress the current working directory contents to a tar file and extract those files from the compressed tar file.

a.) tar -cvf file.tar * --remove-files: This command creates a tar file called file.tar which is the Archive of all files in current directory and –remove-files remove the individual files in the directory but not the current working directory.

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tar -cvf files.tar * --remove-files
Input.txt
Lab/
Lab/sample2.txt
Lab/sample1.txt
Lab Assignment 1.odt
sample1.txt
sample2.txt
sample.txt
Theory/
Theory/(COE18B061) Preparatory Assignment 1/
Theory/(COE18B061) Preparatory Assignment 1/mysort
Theory/(COE18B061) Preparatory Assignment 1/mysort.c
Theory/(COE18B061) Preparatory Assignment 1/mycopy.c
Theory/(COE18B061) Preparatory Assignment 1/allsort.cpp
Theory/(COE18B061) Preparatory Assignment 1/allsort
Theory/(COE18B061) Preparatory Assignment 1/remove
Theory/(COE18B061) Preparatory Assignment 1/allsortft
Theory/(COE18B061) Preparatory Assignment 1/allsortft.cpp
Theory/(COE18B061) Preparatory Assignment 1/remove.c
Theory/(COE18B061) Preparatory Assignment 1/Preparatory Assignment 1 (COE18B061).pdf
Theory/(COE18B061) Preparatory Assignment 1/mysortfp.c
Theory/(COE18B061) Preparatory Assignment 1/mysortfp
Theory/(COE18B061) Preparatory Assignment 1/mycopy
Theory/(COE18B061) Preparatory Assignment 1.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

b.) tar -xvf file.tar: This command extracts files from Archives.

```
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ tar -xvf files.tar
Input.txt
Lab/
Lab/sample2.txt
Lab/sample1.txt
Lab Assignment 1.odt
sample1.txt
sample2.txt
sample.txt
Theory/
Theory/(COE18B061) Preparatory Assignment 1/
Theory/(COE18B061) Preparatory Assignment 1/mysort
Theory/(COE18B061) Preparatory Assignment 1/mysort.c
Theory/(COE18B061) Preparatory Assignment 1/mycopy.c
Theory/(COE18B061) Preparatory Assignment 1/allsort.cpp
Theory/(COE18B061) Preparatory Assignment 1/allsort
Theory/(COE18B061) Preparatory Assignment 1/remove
Theory/(COE18B061) Preparatory Assignment 1/allsortft
Theory/(COE18B061) Preparatory Assignment 1/allsortft.cpp
Theory/(COE18B061) Preparatory Assignment 1/remove.c
Theory/(COE18B061) Preparatory Assignment 1/Preparatory Assignment 1 (COE18B061).pdf
Theory/(COE18B061) Preparatory Assignment 1/mysortfp.c
Theory/(COE18B061) Preparatory Assignment 1/mysortfp
Theory/(COE18B061) Preparatory Assignment 1/mycopy
Theory/(COE18B061) Preparatory Assignment 1.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar  Lab      sample1.txt  sample.txt
Input.txt  'Lab Assignment 1.odt'  sample2.txt  Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

41. Compress the files using zip command.

a. Zip the input file “sample1.txt” as samplezip.zip and remove the file from the current directory after zipping.

zip -rm samplezip.zip sample1.txt: This command adds the sample1.txt to samplezip.zip and -rm removes the sample1.txt from the current directory after zipping.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar Lab sample1.txt sample.txt
Input.txt 'Lab Assignment 1.odt' sample2.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -rm samplezip.zip sample1.txt
adding: sample1.txt (deflated 58%)
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar Lab sample2.txt samplezip.zip
Input.txt 'Lab Assignment 1.odt' sample.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

b. Add “sample2.txt” and update the zip archive.

zip -u samplezip.zip sample2.txt: It updates the zip archive by adding new files in it.

zip -sf samplezip.zip: It shows the content of zip archive.

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar Lab sample2.txt samplezip.zip
Input.txt 'Lab Assignment 1.odt' sample.txt Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -u samplezip.zip sample2.txt
adding: sample2.txt (deflated 40%)
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -sf samplezip.zip
Archive contains:
 sample1.txt
 sample2.txt
Total 2 entries (1497 bytes)
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$
```

c. Zip a directory with all its contents.

zip -r sampledir.zip directoryname: This will recursively zips the files in a directory.

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar      Lab           sample2.txt    samplezip.zip
Input.txt 'Lab Assignment 1.odt' sample.txt     Theory
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -r sampledir.zip Theory
adding: Theory/ (stored 0%)
adding: Theory//COE18B061 Preparatory Assignment 1/ (stored 0%)
adding: Theory//COE18B061 Preparatory Assignment 1/mysort (deflated 79%)
adding: Theory//COE18B061 Preparatory Assignment 1/mysort.c (deflated 61%)
adding: Theory//COE18B061 Preparatory Assignment 1/mycopy.c (deflated 50%)
adding: Theory//COE18B061 Preparatory Assignment 1/allsort.cpp (deflated 78%)
adding: Theory//COE18B061 Preparatory Assignment 1/allsort (deflated 73%)
adding: Theory//COE18B061 Preparatory Assignment 1/remove (deflated 83%)
adding: Theory//COE18B061 Preparatory Assignment 1/allsortft (deflated 73%)
adding: Theory//COE18B061 Preparatory Assignment 1/allsortft.cpp (deflated 73%)
adding: Theory//COE18B061 Preparatory Assignment 1/remove.c (deflated 50%)
adding: Theory//COE18B061 Preparatory Assignment 1/Preparatory Assignment 1 (COE18B061).pdf (deflated 10%)
adding: Theory//COE18B061 Preparatory Assignment 1/mysortfp.c (deflated 61%)
adding: Theory//COE18B061 Preparatory Assignment 1/mysortfp (deflated 79%)
adding: Theory//COE18B061 Preparatory Assignment 1/mycopy (deflated 83%)
adding: Theory//COE18B061 Preparatory Assignment 1.zip (stored 0%)
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar      Lab           sample2.txt    sample.txt     Theory
Input.txt 'Lab Assignment 1.odt' sampledir.zip  samplezip.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

d. Remove a file from the zip archive

zip -d samplezip.zip sample2.txt: It removes the given file from the zip archive.

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar      Lab           sample2.txt    sample.txt     Theory
Input.txt 'Lab Assignment 1.odt' sampledir.zip  samplezip.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -sf samplezip.zip
Archive contains:
  sample1.txt
  sample2.txt
Total 2 entries (1497 bytes)
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -d samplezip.zip sample2.txt
deleting: sample2.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ zip -sf samplezip.zip
Archive contains:
  sample1.txt
Total 1 entries (906 bytes)
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```

e. Unzip the contents from samplezip.zip

unzip samplezip.zip: This command extracts all files from the specified ZIP archive to the current directory.

vinayak@

File Edit View Search Terminal Help

```
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar          Lab           sample2.txt    sample.txt     Theory
Input.txt  'Lab Assignment 1.odt'  sampledir.zip  samplezip.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ unzip samplezip.zip
Archive: samplezip.zip
  inflating: sample1.txt
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ ls
files.tar  'Lab Assignment 1.odt'  sampledir.zip  Theory
Input.txt   sample1.txt          sample.txt
Lab        sample2.txt          samplezip.zip
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ unzip sampledir.zip
Archive: sampledir.zip
replace Theory/(COE18B061) Preparatory Assignment 1.zip? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
extracting: Theory/(COE18B061) Preparatory Assignment 1.zip
  inflating: Theory/(COE18B061) Preparatory Assignment 1/Preparatory Assignment 1 (COE18B061).pdf
  inflating: Theory/(COE18B061) Preparatory Assignment 1/allsort
  inflating: Theory/(COE18B061) Preparatory Assignment 1/allsort.cpp
  inflating: Theory/(COE18B061) Preparatory Assignment 1/allsortft
  inflating: Theory/(COE18B061) Preparatory Assignment 1/allsortft.cpp
  inflating: Theory/(COE18B061) Preparatory Assignment 1/mycopy
  inflating: Theory/(COE18B061) Preparatory Assignment 1/mycopy.c
  inflating: Theory/(COE18B061) Preparatory Assignment 1/mysort
  inflating: Theory/(COE18B061) Preparatory Assignment 1/mysort.c
  inflating: Theory/(COE18B061) Preparatory Assignment 1/mysortfp
  inflating: Theory/(COE18B061) Preparatory Assignment 1/mysortfp.c
  inflating: Theory/(COE18B061) Preparatory Assignment 1/remove
  inflating: Theory/(COE18B061) Preparatory Assignment 1/remove.c
  inflating: mycopy.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Operating Systems$ 
```