1. **whoami** → To get the username

```
rakshit@RG:~$ whoami
rakshit
rakshit@RG:~$
```

2. **pwd** → To see the directory you are currently working with

```
rakshit@RG:~$ pwd
/home/rakshit
rakshit@RG:~$
```

3. **ls** → to list the directories

```
rakshit@RG:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
rakshit@RG:~$
```

4. **ls -la** → List the directories including hidden and with other details

```
rakshit@RG:~$ ls -la
total 80
drwxr-x--- 16 rakshit rakshit 4096 Sep 12 17:30 .
drwxr-xr-x  3 root    root    4096 Sep 11 17:25 ..
-rw-------  1 rakshit rakshit    0 Sep 11 18:06 .bash_history
-rw-r--r--  1 rakshit rakshit  220 Sep 11 17:25 .bash_logout
-rw-r--r--  1 rakshit rakshit 3771 Sep 11 17:25 .bashrc
drwx------ 12 rakshit rakshit 4096 Sep 12 16:59 .cache
drwx------ 12 rakshit rakshit 4096 Sep 12 17:11 .config
drwxr-xr-x  2 rakshit rakshit 4096 Sep 12 16:59 Desktop
drwxr-xr-x  2 rakshit rakshit 4096 Sep 11 17:51 Documents
drwxr-xr-x  2 rakshit rakshit 4096 Sep 11 17:51 Downloads
drwx------  2 rakshit rakshit 4096 Sep 12 17:12 .gnupg
-rw-------  1 rakshit rakshit   20 Sep 12 17:25 .lesshst
drwx------  3 rakshit rakshit 4096 Sep 11 17:51 .local
drwxr-xr-x  2 rakshit rakshit 4096 Sep 11 17:51 Music
drwxr-xr-x  3 rakshit rakshit 4096 Sep 12 16:59 Pictures
-rw-r--r--  1 rakshit rakshit  807 Sep 11 17:25 .profile
drwxr-xr-x  2 rakshit rakshit 4096 Sep 11 17:51 Public
drwx------  3 rakshit rakshit 4096 Sep 11 17:51 snap
drwx------  2 rakshit rakshit 4096 Sep 11 18:03 .ssh
-rw-r--r--  1 rakshit rakshit    0 Sep 11 18:04 .sudo_as_admin_successful
drwxr-xr-x  2 rakshit rakshit 4096 Sep 11 17:51 Templates
drwxr-xr-x  2 rakshit rakshit 4096 Sep 11 17:51 Videos
```

5. **cd <directory name>** → To enter the directory

```
rakshit@RG:~$ cd Desktop
rakshit@RG:~/Desktop$
```

6. **cd ..** → To go to the previous directory

```
rakshit@RG:~/Desktop/sample$ cd ..
rakshit@RG:~/Desktop$
```

7. **cd** → To go to home directory

```
rakshit@RG:~/Desktop/sample$ cd
rakshit@RG:~$
```

8. **cd <path>** : open a particular directory using path

```
rakshit@RG:~/sample$ cd college/jammu/miet
rakshit@RG:~/sample/college/jammu/miet$
```

9. **touch <file name>** → To create a file

```
rakshit@RG:~$ touch clg.txt
rakshit@RG:~$ ls
clg.txt  Desktop  Documents  Downloads  Music  Pictures  Public  sample  snap  Templates  Videos
rakshit@RG:~$
```

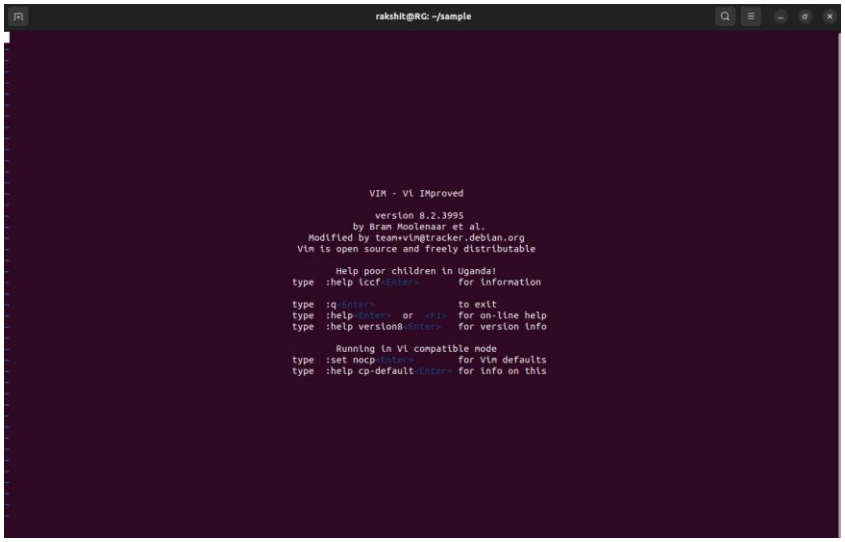
10. **touch <file name>{from..to}** → To create multiple files ending with from to to.

```
rakshit@RG:~/sample/miet$ touch clg{1..9}.txt
rakshit@RG:~/sample/miet$ ls
clg1.txt  clg2.txt  clg3.txt  clg4.txt  clg5.txt  clg6.txt  clg7.txt  clg8.txt  clg9.txt
rakshit@RG:~/sample/miet$
```

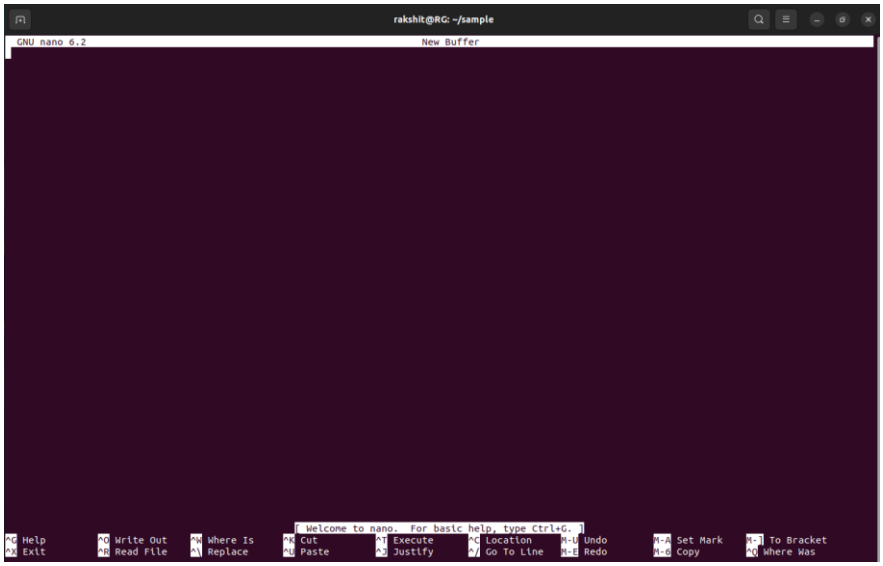
```
rakshit@RG:~/sample/miet$ ls
clgf.txt  clgg.txt  clgh.txt  clgi.txt  clgj.txt  clgk.txt
rakshit@RG:~/sample/miet$
```

11. **vi / nano** → Open the writing mode

vi

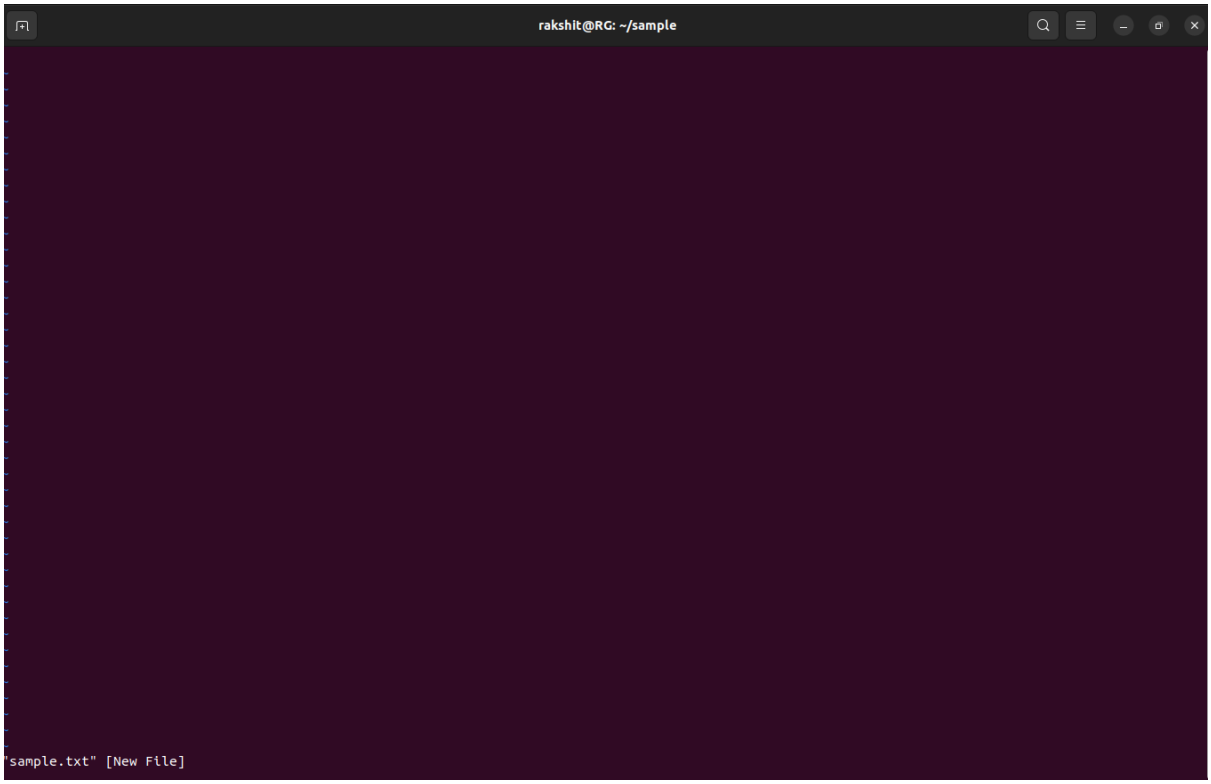


nano

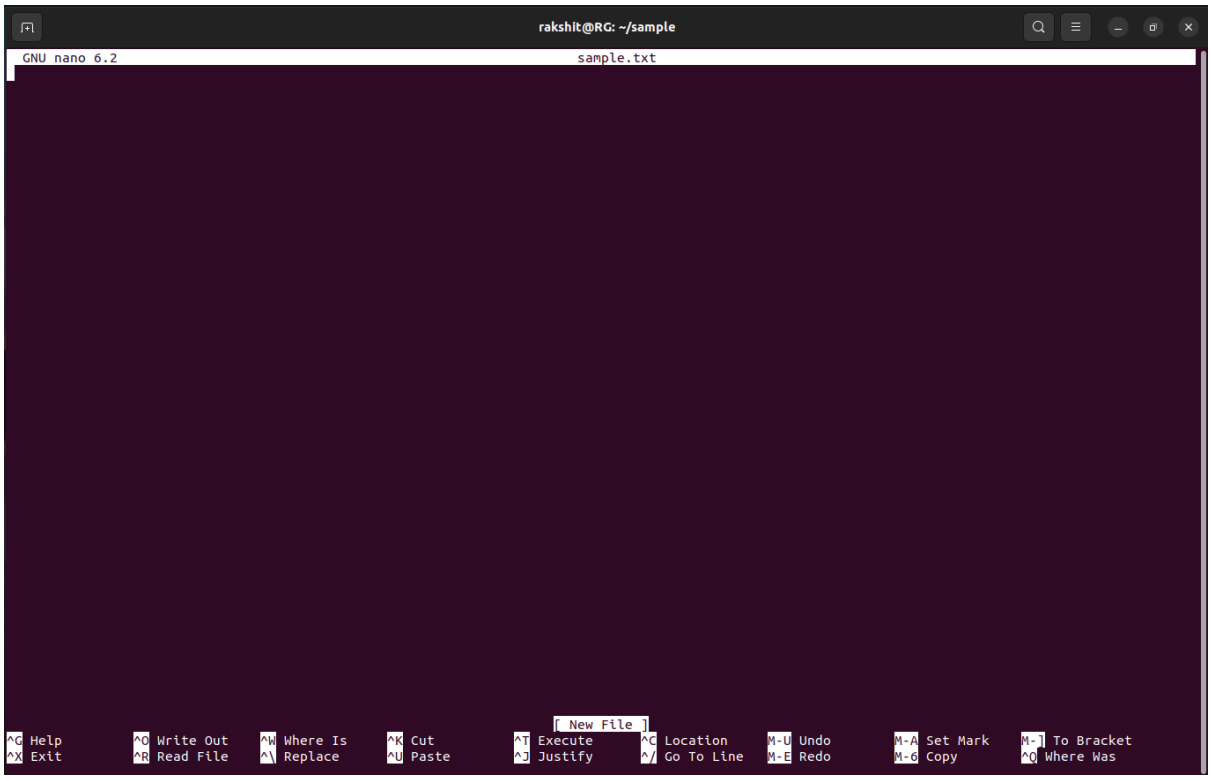


12. **vi / nano <file name>** → create a file and open in writing mode

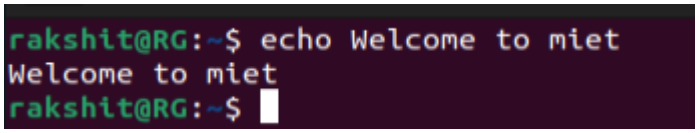
vi



nano



13. **echo<string>** → To print the string



14. **hostname** → To get the hostname

```
rakshit@RG:~$ hostname
RG
rakshit@RG:~$
```

15. **cat <file name>** → To read the file

```
rakshit@RG:~/sample$ cat sample.txt
this is a sample file!!
rakshit@RG:~/sample$
```

16. **mkdir<directory name>** → create a directory

```
rakshit@RG:~/sample$ mkdir sample
rakshit@RG:~/sample$
```

17. **mkdir -help** → help in mkdir

```
rakshit@RG:~/sample$ mkdir --help
Usage: mkdir [OPTION]... DIRECTORY...
Create the DIRECTORY(ies), if they do not already exist.

Mandatory arguments to long options are mandatory for short options too.
  -m, --mode=MODE   set file mode (as in chmod), not a=rwx - umask
  -p, --parents     no error if existing, make parent directories as needed
  -v, --verbose     print a message for each created directory
  -Z                   set SELinux security context of each created directory
                         to the default type
      --context[=CTX]  like -Z, or if CTX is specified then set the SELinux
                         or SMACK security context to CTX
      --help     display this help and exit
      --version  output version information and exit

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Full documentation <https://www.gnu.org/software/coreutils/mkdir>
or available locally via: info '(coreutils) mkdir invocation'
rakshit@RG:~/sample$
```

18. **mkdir -v <directory name>** → create a directory and give output for status

```
rakshit@RG:~/sample$ mkdir -v sample
mkdir: created directory 'sample'
rakshit@RG:~/sample$
```

19. **mkdir -p <directory name/next directory>** → Create a directory path

```
rakshit@RG:~/sample$ mkdir -p college/jammu/miet
rakshit@RG:~/sample$ cd college
rakshit@RG:~/sample/college$ ls
jammu
rakshit@RG:~/sample/college$ cd jammu
rakshit@RG:~/sample/college/jammu$ ls
miet
rakshit@RG:~/sample/college/jammu$ cd miet
rakshit@RG:~/sample/college/jammu/miet$ ls
rakshit@RG:~/sample/college/jammu/miet$
```

20. **rm <file name>** → to delete a file

```
rakshit@RG:~/sample$ ls
example.txt  sample.txt
rakshit@RG:~/sample$ rm sample.txt
rakshit@RG:~/sample$ ls
example.txt
rakshit@RG:~/sample$
```

21. **rmdir <directory name>** →delete directory

```
rakshit@RG:~/sample$ ls
mier  miet
rakshit@RG:~/sample$ rmdir mier
rakshit@RG:~/sample$ ls
miet
rakshit@RG:~/sample$
```

22. **rmdir -p <directory name/next directory>** → Delete a directory path

```
rakshit@RG:~/sample$ ls
jammu
rakshit@RG:~/sample$ rmdir -p jammu/college/miet
rakshit@RG:~/sample$
```

23. **mv <old file name> <new file name>** → to renaming a file

```
rakshit@RG:~/sample$ ls
abc.txt
rakshit@RG:~/sample$ mv abc.txt xyz.txt
rakshit@RG:~/sample$ ls
xyz.txt
rakshit@RG:~/sample$
```

24. **mv <file name> <path where to move>** →to move a file

```
rakshit@RG:~/sample$ ls
directory  xyz.txt
rakshit@RG:~/sample$ mv xyz.txt directory
rakshit@RG:~/sample$ ls
directory
rakshit@RG:~/sample$ cd directory
rakshit@RG:~/sample/directory$ ls
xyz.txt
rakshit@RG:~/sample/directory$
```

```
rakshit@RG:~/sample$ tree
.
├── directory
│   └── xyz.txt
└── directory_2

2 directories, 1 file
rakshit@RG:~/sample$ cd directory
rakshit@RG:~/sample/directory$ mv xyz.txt ~/sample/directory_2
rakshit@RG:~/sample/directory$ cd ..
rakshit@RG:~/sample$ tree
.
├── directory
└── directory_2
    └── xyz.txt

2 directories, 1 file
rakshit@RG:~/sample$
```

25. **mv -i <file name> <path where to move>** →to move a file and take permission to overwrite if file exists

```
rakshit@RG:~/sample$ tree
.
├── directory
│   └── xyz.txt
└── directory_2
    └── xyz.txt

2 directories, 2 files
rakshit@RG:~/sample$ cd directory
rakshit@RG:~/sample/directory$ mv -i xyz.txt ~/sample/directory_2
mv: overwrite '/home/rakshit/sample/directory_2/xyz.txt'? n
rakshit@RG:~/sample/directory$
```

```
.
├── directory
│   └── xyz.txt
└── directory_2
    └── xyz.txt
```

26. **mv -n <file name> <path where to move>** →to move a file but reject for overwriting if file already exists

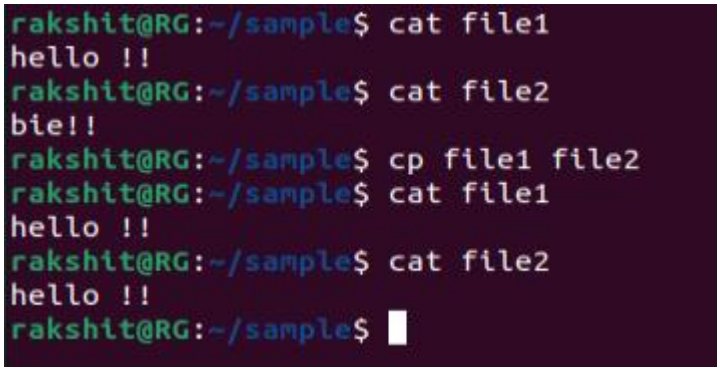
```
rakshit@RG:~/sample/directory$ mv -n xyz.txt ~/sample/directory_2
rakshit@RG:~/sample/directory$
```

```
.
├── directory
│   └── xyz.txt
└── directory_2
    └── xyz.txt
```

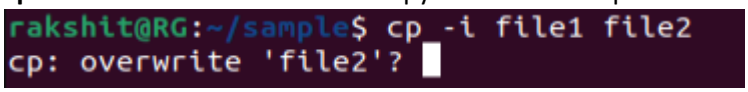
27. **mv -b <file name> <path where to move>** →to move a file and getting the backup of the file

```
rakshit@RG:~/sample/directory$ mv -b xyz.txt ~/sample/directory_2
rakshit@RG:~/sample/directory$ cd ..
rakshit@RG:~/sample$ tree
.
├── directory
└── directory_2
    ├── xyz.txt
    └── xyz.txt~
```

28. **cp<file1> <file2>** → to copy file1 in file2



29. **cp -i <file name> < file2>** →to copy a file and take permission to overwrite if file exists



30. **cp -n <file1> < file2 >** →to copy a file but reject for overwriting if file already exists

31.