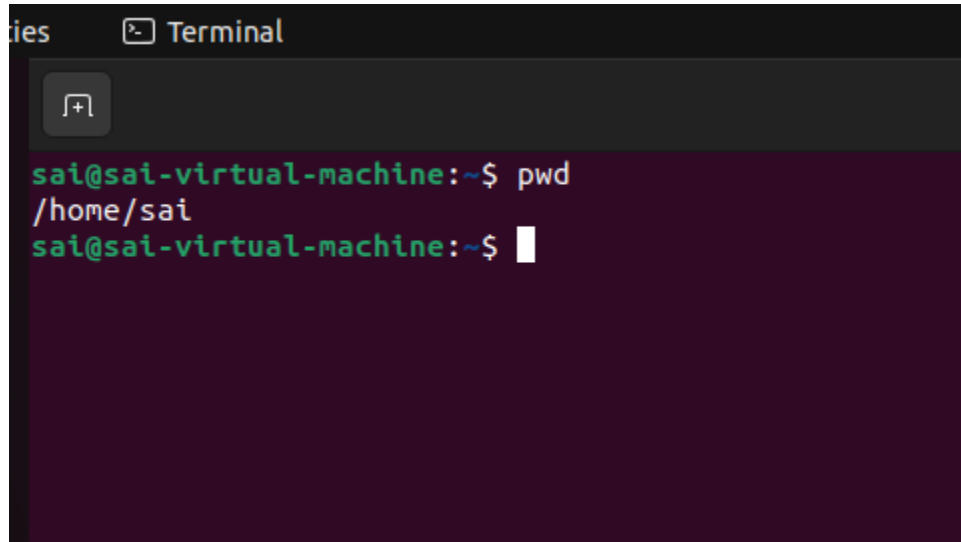


Linux commands

1. **pwd:** This command is used to show the current working directory

A screenshot of a Linux terminal window. The title bar shows a window icon and the word "Terminal". Inside the terminal, the prompt is "sai@sai-virtual-machine:~\$". The user has entered the command "pwd", and the output is "/home/sai". The prompt is now "sai@sai-virtual-machine:~\$" with a cursor at the end.

```
sai@sai-virtual-machine:~$ pwd
/home/sai
sai@sai-virtual-machine:~$
```

2. **ls:** It will show the full list or content of your directory. Just type *ls* and press the enter key.

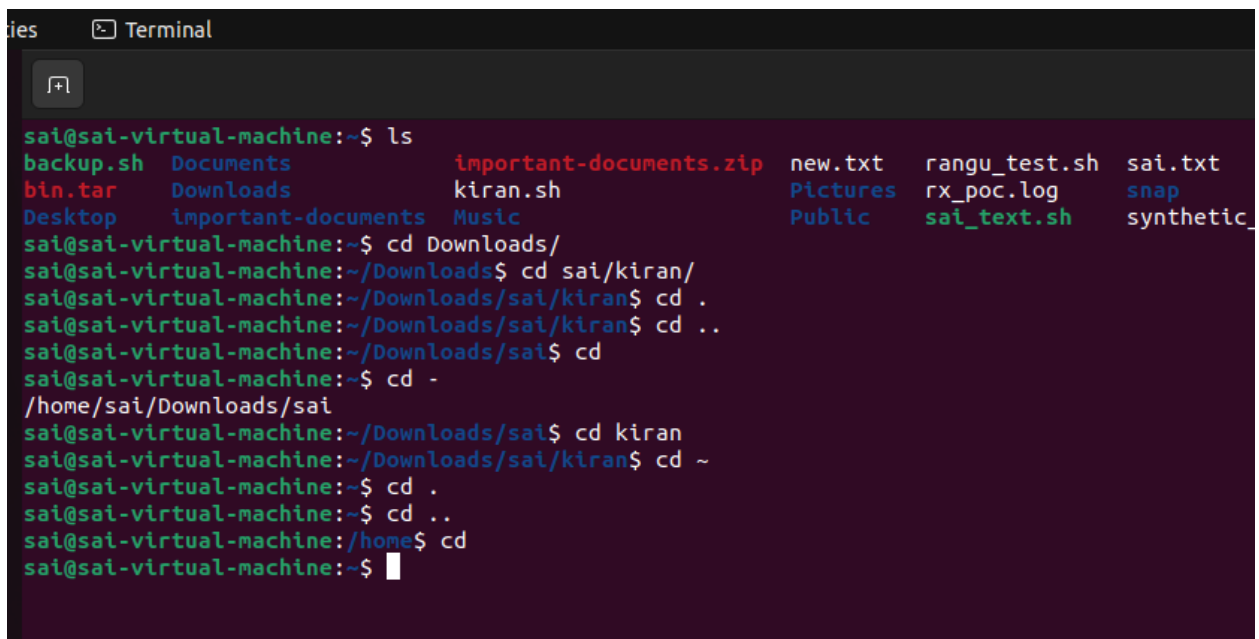
command	Options
<u>ls -a</u>	In Linux, hidden files start with . (dot) symbol and they are not visible in the regular directory. The (ls -a) command will enlist the whole list of the current directory including the hidden files.
<u>ls -l</u>	It will show the list in a long list format.(data and time)
ls -n	It is used to print group ID and owner ID instead of their names.
ls -lt	It will sort the list by displaying recently modified files at top.
<u>ls ~</u>	It gives the contents of home directory.
<u>ls ../</u>	It give the contents of parent directory.
ls --version	It checks the version of ls command.

```
ies Terminal

sai@sai-virtual-machine:~$ ls
backup.sh  Documents          important-documents.zip  new.txt  rangu_test.sh  sai.txt
bin.tar    Downloads          kiran.sh               Pictures  rx_poc.log     snap
Desktop    important-documents Music                Public    sai_text.sh    synthetic_his
sai@sai-virtual-machine:~$ ls -l
total 124
-rwxrwxr-x 1 sai sai 1417 Jan 18 17:05 backup.sh
-rw-rw-r-- 1 sai sai 10240 Jan 2 22:47 bin.tar
drwxr-xr-x 2 sai sai 4096 Mar 27 16:13 Desktop
drwxr-xr-x 2 sai sai 4096 Nov 5 2023 Documents
drwxr-xr-x 3 sai sai 4096 May 9 11:45 Downloads
drwxr-xr-x 2 sai sai 4096 Jan 18 13:05 important-documents
-rw-rw-r-- 1 sai sai 4995 Sep 28 2022 important-documents.zip
-rw-rw-r-- 1 sai sai 49 Jan 8 22:39 kiran.sh
drwxr-xr-x 2 sai sai 4096 Nov 5 2023 Music
-rw-rw-r-- 1 sai sai 30 May 9 10:29 new.txt
drwxr-xr-x 3 sai sai 4096 Jan 18 11:33 Pictures
drwxr-xr-x 2 sai sai 4096 Nov 5 2023 Public
-rw-rw-r-- 1 sai sai 101 Jan 8 22:37 rangu_test.sh
-rw-rw-r-- 1 sai sai 13 Jan 17 17:55 rx_poc.log
-rwxrwxr-x 1 sai sai 523 Jan 3 12:04 sai_text.sh
-rw-rw-r-- 1 sai sai 10 Jan 8 21:50 sai.txt
drwx----- 6 sai sai 4096 Mar 27 16:13 snap
-rw-rw-r-- 1 sai sai 263 Jun 5 2023 synthetic_historical_fc_accuracy.tsv
drwxr-xr-x 2 sai sai 4096 Nov 5 2023 Templates
drwxrwxr-x 2 sai sai 4096 Mar 27 16:17 test
-rw-rw-r-- 1 sai sai 0 Mar 27 16:13 test_f
-rw-rw-r-- 1 sai sai 147 Jan 8 22:27 test.sh
-rw-rw-r-- 1 sai sai 330 Jan 2 21:10 test.txt
drwxrwxr-x 2 sai sai 4096 Jan 2 22:43 text.txt
drwxr-xr-x 2 sai sai 4096 Nov 5 2023 Videos
-rw-rw-r-- 1 sai sai 32 Jan 17 16:08 weather_project.log
-rwxrwxr-x 1 sai sai 0 Jan 17 16:09 weather_project.sh
-rw-rw-r-- 1 sai sai 6685 Jan 17 16:12 weather_report
-rw-rw-r-- 1 sai sai 32 Jan 17 17:23 wether_project.log
sai@sai-virtual-machine:~$ ls ~
backup.sh  Documents          important-documents.zip  new.txt  rangu_test.sh  sai.txt
bin.tar    Downloads          kiran.sh               Pictures  rx_poc.log     snap
Desktop    important-documents Music                Public    sai_text.sh    synthetic_his
sai@sai-virtual-machine:~$ ls ../
sai
sai@sai-virtual-machine:~$ ls --version
ls (GNU coreutils) 8.32
Copyright (C) 2020 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.
```

3. **cd:** used to change the current working directory

- ***cd ~*** or ***cd*** by itself will always bring the user to the home directory.
- ***cd .*** will allow the user to stay in the similar directory they are in currently.
- ***cd ~username*** will allow the user to stay in the home directory of the username.
- ***cd ..*** will bring the user above one direct(one step back).
- ***cd -*** will change the user to the old directory(go back).
- ***Cd /*** - change to root node(starting)
- ***Cd Downloads/sai/kiran***-use ***"/"*** to change maltipule directories.



```

sai@sai-virtual-machine:~$ ls
backup.sh  Documents          important-documents.zip  new.txt    rangu_test.sh  sai.txt
bin.tar    Downloads          kiran.sh               Pictures   rx_poc.log     snap
Desktop    important-documents Music              Public     sai_text.sh    synthetic_
sai@sai-virtual-machine:~$ cd Downloads/
sai@sai-virtual-machine:~/Downloads$ cd sai/kiran/
sai@sai-virtual-machine:~/Downloads/sai/kiran$ cd .
sai@sai-virtual-machine:~/Downloads/sai/kiran$ cd ..
sai@sai-virtual-machine:~/Downloads/sai$ cd
sai@sai-virtual-machine:~$ cd -
/home/sai/Downloads/sai
sai@sai-virtual-machine:~/Downloads/sai$ cd kiran
sai@sai-virtual-machine:~/Downloads/sai/kiran$ cd ~
sai@sai-virtual-machine:~$ cd .
sai@sai-virtual-machine:~$ cd ..
sai@sai-virtual-machine:/home$ cd
sai@sai-virtual-machine:~$

```

4. ***mkdir***: is used to create new directory

- ***mkdir sai***
- ***mkdir saikiran rangu*** (2 directorys at a time)

```
es Terminal
sai@sai-virtual-machine:~/Downloads/sai/kiran$ mkdir sairangu rangusai
sai@sai-virtual-machine:~/Downloads/sai/kiran$ ls
rangusai  sairangu
sai@sai-virtual-machine:~/Downloads/sai/kiran$
```

5. **rmdir**: used to remove directory

- **rmdir sai**
- **rmdir saikiran sairangu**

```
es Terminal
sai@sai-virtual-machine:~/Downloads/sai$ ls
hello  sai1  sai3
sai@sai-virtual-machine:~/Downloads/sai$ rmdir hello
sai@sai-virtual-machine:~/Downloads/sai$ rmdir sai1 sai3
sai@sai-virtual-machine:~/Downloads/sai$ ls
sai@sai-virtual-machine:~/Downloads/sai$
```

6. **rm**: is used to remove

7. **mv**: used to move existing file or directory from one location to another. It is also used to rename a file or directory. If you want to rename a single directory or file then 'mv' option will be better to use.

we can use **mv** to rename

- **mv hello.txt hellomain** (renamed to hellomain and also not a txt file)
- **mv hello.txt hellomain.txt** (now renamed to same file)

```
[+]  
sai@sai-virtual-machine:~$ ls  
backup.sh  Documents  important-documents  Music  Public  saikk.txt  s  
bin.tar    Downloads  important-documents.zip  new.txt  rangu_test.sh  sai_text.sh  T  
Desktop    hello.txt  kiran.sh             Pictures  rx_poc.log     snap         t  
sai@sai-virtual-machine:~$ mv hello.txt renamed_hello.txt  
sai@sai-virtual-machine:~$ ls  
backup.sh  Documents  important-documents.zip  new.txt  rangu_test.sh  
bin.tar    Downloads  kiran.sh                Pictures  renamed_hello.txt  
Desktop    important-documents  Music                  Public    rx_poc.log  
sai@sai-virtual-machine:~$
```

- we can use rename but we have to install using command
- **sudo apt install rename**

- Converting captel to lower in all files

```
javatpoint@javatpoint-Inspiron-3542:~/Directory$ rename 'y/A-Z/a-z/' *  
javatpoint@javatpoint-Inspiron-3542:~/Directory$ ls  
new_folder1  new_folder2  new_folder3
```

→ changing txt files to pdf

```
javatpoint@javatpoint-Inspiron-3542:~/Directory$ rename 's/\.txt$/\.pdf/' *.txt  
javatpoint@javatpoint-Inspiron-3542:~/Directory$ ls  
demo1.pdf  demo2.pdf  demo.pdf  new_folder1  new_folder2  new_folder3
```

→ we can convert all txt files to zip or pdf using for loop

- **for I in *.txt;**
- **do**
- **newfile=\${I/.txt/.zip};**
- **mv \$I \$newfile;**
- **done;**

8. **man:** man is an interface to view the system's reference manual.

- **man ls** (display the ls manual)
- **man -w**

9. **whatis:** is used to get brief information about Linux commands or functions. It displays the manual page description in a single line

- **whatis cd**
- **whatis ls**
- **whatis pwd**
- **whatis chmod**
- **whatis cat**

```

sai@sai-virtual-machine:~$ whatis cat
cat (1) - concatenate files and print on the standard output
sai@sai-virtual-machine:~$ whatis cd
cd: nothing appropriate.
sai@sai-virtual-machine:~$ whatis pwd
pwd (1) - print name of current/working directory
sai@sai-virtual-machine:~$ whatis ls
ls (1) - list directory contents
sai@sai-virtual-machine:~$ whatis mkdir
mkdir (1) - make directories
sai@sai-virtual-machine:~$ whatis rm
rm (1) - remove files or directories
sai@sai-virtual-machine:~$ whatis rename
rename: nothing appropriate.
sai@sai-virtual-machine:~$ whatis whoami
whoami (1) - print effective userid
sai@sai-virtual-machine:~$ whatis chmod
chmod (1) - change file mode bits
sai@sai-virtual-machine:~$ whatis tac
tac (1) - concatenate and print files in reverse
sai@sai-virtual-machine:~$

```

→Linux user:

10.whoami: It tells you about the system's username.

11.who: who command gives the information about the users logged on to the system.

12.who am i: This command displays the information about the current user only.

13.w: This command tells about the users who are logged in and what are they doing.

14.id : This command tells about your user id, primary group id, and a list of groups that belongs to you.

- > **-Z, --context:** It is used to print only the security context of the process.
- > **-g, --group:** It is used to print only the effective GID.
- > **-G, --groups:** It is used to print all group Ids.
- > **-n, --name:** It is used to print a name instead of a number

```

sai@sai-virtual-machine:~$ whoami
sai
sai@sai-virtual-machine:~$ who
sai    tty2      2024-05-09 14:55 (tty2)
sai@sai-virtual-machine:~$ who am i
sai@sai-virtual-machine:~$ id
uid=1000(sai) gid=1000(sai) groups=1000(sai),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),122(lpadmin),135(lxd),136(sambashare)
sai@sai-virtual-machine:~$ w
 15:34:56 up 38 min, 1 user,  load average: 0.18, 0.11, 0.06
USER    TTY      FROM          LOGIN@   IDLE   JCPU   PCPU   WHAT
sai     tty2     tty2          14:55    43:11  0.09s  0.08s  /usr/libexec/gnome-session-binary --session=ubuntu
sai@sai-virtual-machine:~$

```

15.**adduser:** is used to add new user

- **sudo adduser saikiran** (it will ask password and add new password)
- **adduser -m** (is used to create new user and add to home page)
- **sudo adduser -m saikiran**

```
sai@sai-virtual-machine:~$ sudo adduser saikiran
[sudo] password for sai:
Adding user `saikiran' ...
Adding new group `saikiran' (1001) ...
Adding new user `saikiran' (1001) with group `saikiran' ...
Creating home directory `/home/saikiran' ...
Copying files from `/etc/skel' ...
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
passwd: password updated successfully
Changing the user information for saikiran
Enter the new value, or press ENTER for the default
    Full Name []: saikiran rangu
    Room Number []: 1
    Work Phone []: 8978553778
    Home Phone []: 8978553778
    Other []: 89778553778
Is the information correct? [Y/n] y
sai@sai-virtual-machine:~$ su
```

16.**passwd:** is used to create new password

- **sudo passwd**

```
root@sai-virtual-machine:/home/sai# sudo passwd
New password:
Retype new password:
passwd: password updated successfully
root@sai-virtual-machine:/home/sai# pwd
```

17.**su:** is used to change the user it will ask password.

- **su saikiran**

➔ **su to root:** You can change the user to root when you know the root password.

- **su root** (to logout from the root use exit)

➔ **su -** : If any user name is not mentioned then by default, it will assume root as the target user.

- **su -** (by default it will login to root)

```
sai@sai-virtual-machine:~$ su saikiran
Password:
saikiran@sai-virtual-machine:/home/sai$ ls
```

18.groupadd: The groupadd command creates or add a group in our system.

- **sudo groupadd saiki**

19.groups: get group members

- **sudo useradd sai**
- **getent:** list the groups
- **getent group** (get the info of group)

20.groupdel: used to delete group

- **sudo groupdel rsk**

```
sai@sai-virtual-machine:/home$ sudo addgroup rsk
Adding group `rsk' (GID 1001) ...
Done.
sai@sai-virtual-machine:/home$ sudo useradd sai
useradd: user 'sai' already exists
sai@sai-virtual-machine:/home$ sudo useradd saikiran
useradd: user 'saikiran' already exists
sai@sai-virtual-machine:/home$ groups
sai adm cdrom sudo dip plugdev lpadmin lxd sambashare
sai@sai-virtual-machine:/home$ getent group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,sai
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
```


Files:

21.cat: cat is used as multipurpose in linux like create file, open content of file, concatenate, display line by line.

- **Cat >newfile.txt** (created new txt file)

➔ Afer entering data **ctrl+d** to save data and exit

- **Cat newfile.txt** (opening new file)

```
sai@sai-virtual-machine:~$ cd new
sai@sai-virtual-machine:~/new$ cat > newfile.txt
saikiran age 23
sai@sai-virtual-machine:~/new$ cat newfile.txt
saikiran age 23
```

22.touch: create a new file, update the time stamp on existing files and directories. It can also create empty files in Linux.

- **touch new.txt**
- **touch sai.txt saiki.pdf** (can create differnrt type of files at a time)
- **ls -l new.txt** (get the info about file like time date)

```
sai@sai-virtual-machine:~/new$ touch new2.txt
sai@sai-virtual-machine:~/new$ ls -l new2.txt
-rw-rw-r-- 1 sai sai 0 May  9 17:46 new2.txt
sai@sai-virtual-machine:~/new$ touch s1.txt s2.pdf s3.txt
sai@sai-virtual-machine:~/new$ ls
new2.txt  newfile.txt  s1.txt  s2.pdf  s3.txt
sai@sai-virtual-machine:~/new$
```

23.->redirect(>): same as touch

- **>saikiran.txt**
- **ls -l saikieran.txt**

```
sai@sai-virtual-machine:~/new$ >saiki61.txt
sai@sai-virtual-machine:~/new$ ls -l saiki61.txt
-rw-rw-r-- 1 sai sai 0 May  9 17:53 saiki61.txt
sai@sai-virtual-machine:~/new$
```

24.echo: echo is used to to display line and also used to create a file, but we should specify the file content on the command line.

- **echo "sai"** (diaplying the message sai)

- **echo “saikiran age 23” >new.txt** (created a file and entered data)

```
sai@sai-virtual-machine:~/new$ echo "sai"
sai
sai@sai-virtual-machine:~/new$ echo "saikiran age 23" >new66.txt
sai@sai-virtual-machine:~/new$ ls
new2.txt  new66.txt  newfile.txt  s1.txt  s2.pdf  s3.txt  sai61.txt
sai@sai-virtual-machine:~/new$ ls -l new66.txt
-rw-rw-r-- 1 sai sai 16 May  9 17:59 new66.txt
sai@sai-virtual-machine:~/new$
```

25.printf: same as echo

- **printf “sai” >new.txt**

26.vim: We can create a file using the **Vim text editor**. If you do not have the vim editor installed on your machine install it

- **sudo apt install vim**
- **vim new.txt** (enter data and press **esc** press **:qw** to save and exit)

```
sai@sai-virtual-machine:~/new$ sudo apt install vim
[sudo] password for sai:
Sorry, try again.
[sudo] password for sai:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
vim is already the newest version (2:8.2.3995-1ubuntu2.16).
The following packages were automatically installed and are no longer
needed:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
sai@sai-virtual-machine:~/new$ vim new71.txt
sai@sai-virtual-machine:~/new$ cat new71.txt
hello saikiran rangu
age 23
sai@sai-virtual-machine:~/new$
```

27.nano: We can create a file using the **nano text editor**

- **nano sai.txt** (press **ctrl+x** and **y** and again **enter**)

```

sai@sai-virtual-machine:~/new$ nano new.txt81
sai@sai-virtual-machine:~/new$ cat new.txt81
hello world this is saikiran
sai@sai-virtual-machine:~/new$

```

28.cp: cp is used to copy the file

- **cp sai.txt new.txt** (used to copy all the content to new.txt file)

➔ Option 'r' with the copy command can be used to copy a directory including all its content from a source directory to the destination directory.

- **Cp -r new home/sai/Desktop** (copying directory to new location)

```

saikiran@saikiran-virtual-machine:~/Desktop$ cd rangu/
saikiran@saikiran-virtual-machine:~/Desktop/rangu$ ls
Desktop  hello.txt  sai.txt
saikiran@saikiran-virtual-machine:~/Desktop/rangu$ cp hello.txt hh.txt
saikiran@saikiran-virtual-machine:~/Desktop/rangu$ ls
Desktop  hello.txt  hh.txt  sai.txt
saikiran@saikiran-virtual-machine:~/Desktop/rangu$ cd ..
saikiran@saikiran-virtual-machine:~/Desktop$ cp -r rangu sai
saikiran@saikiran-virtual-machine:~/Desktop$ ls
rangu  sai
saikiran@saikiran-virtual-machine:~/Desktop$ cd sai/
saikiran@saikiran-virtual-machine:~/Desktop/sai$ ls
rangu  sai.txt
saikiran@saikiran-virtual-machine:~/Desktop/sai$

```

29.cut: is used to filter the data by column.

- **cut -d- -f2 newh.txt** (2nd column will print separation of (-))
- **cut -d " " -f2 newh.txt** (2nd column separation of space will print)

```
saikiran@saikiran-virtual-machine:~$ cat main.txt
sai 23
rangu 25
sai-23
kiran-24
rangu-51
helo man
saikiran eangu
hei thai is saikiran rangu
23 45
23
45 2
23
    55
34 55
saikiran@saikiran-virtual-machine:~$ cut -d" " -f2 main.txt
23
25
sai-23
kiran-24
rangu-51
man
eangu
thai
45

2
23
55
55
saikiran@saikiran-virtual-machine:~$ cut -d- -f2 main.txt
sai 23
rangu 25
23
24
51
helo man
saikiran eangu
hei thai is saikiran rangu
23 45
23
45 2
23
    55
34 55
saikiran@saikiran-virtual-machine:~$
```

30.grep(global regular expression print): filters the content of a file which makes our search easy. Used with pipe(|).

- **sudo apt-get install grep** (install grep)
 - **cat newh.txt | grep "sai"**
 - **grep "sai" newh.txt**
- ➔ use (**grep -v**) to not display specific data
 - **grep -v "sai" newh.txt**
- ➔ Use (**grep -i**) to display data which is in lower and upper case.
 - **grep -i "sai" newh.txt**
- ➔ grep -A command is used to display the **line after the result**.
 - **grep -A2 "sai" new.txt** (print line which have sai then next line)
- ➔ grep -B command is used to display the **line before the result**.
 - **grep -B1 "sai" newh.txt** (print data then sai line data)
- ➔ grep -C command is used to display the **line after and line before** the result.
 - **grep -C2 "sai" newh.txt** (print data then sai line data)

```

saikiran@saikiran-virtual-machine:~$ cat main.txt | grep sai
sai 23
sai-23
saikiran eangu
hei thai is sainkiran rangu
saikiran@saikiran-virtual-machine:~$ grep hai main.txt
hei thai is sainkiran rangu
saikiran@saikiran-virtual-machine:~$ grep -v hai main.txt
sai 23
rangu 25
sai-23
kiran-24
rangu-51
helo man
saikiran eangu
23 45
23
45 2
23
55
34 55
saikiran@saikiran-virtual-machine:~$ grep -i hai main.txt
hei thai is sainkiran rangu
saikiran@saikiran-virtual-machine:~$ grep -A1 hai main.txt
hei thai is sainkiran rangu
23 45
saikiran@saikiran-virtual-machine:~$ grep -B2 hai main.txt
helo man
saikiran eangu
hei thai is sainkiran rangu
saikiran@saikiran-virtual-machine:~$ grep -C2 hai main.txt
helo man
saikiran eangu
hei thai is sainkiran rangu
23 45
23
saikiran@saikiran-virtual-machine:~$ cat main.txt | grep sai --colour
sai 23
sai-23
saikiran eangu
hei thai is sainkiran rangu
saikiran@saikiran-virtual-machine:~$

```

31.comm: compares two files. By default, 'comm' will always display **three columns**. First column indicates non-matching items of first file, second column indicates non-matching items of second file, and third column indicates matching items of both the files. Both the files has to be in sorted order for 'comm' command to be executed.

- **comm sai.txt kiran.txt**

```

saikiran@saikiran-virtual-machine:~$ ls
Desktop Documents Downloads main2.txt main.txt Music Pictures
saikiran@saikiran-virtual-machine:~$ comm main.txt main2.txt
      sai 23
      kiran 24
rangu 25
      sai-23
comm: file 1 is not in sorted order
comm: file 2 is not in sorted order
      kiran-24
      hei saikiran rangu
      45
rangu-51
helo man
saikiran eangu
hei thai is sainkiran rangu
23 45
23
45 2
23
55
34 55
comm: input is not in sorted order
saikiran@saikiran-virtual-machine:~$

```

32.sed: stream editor. It is used to edit streams (files) using regular expressions. But this editing is not permanent. It remains only in display, but in actual, file content remains the same.

- **echo "sai" | sed 's/sai/kiran/'** (changing sai to kiran)
- ➔ **'s** used for substitute
- ➔ **'y** is used for replace
 - **echo sai7 |sed 's/7/12/'** (op: sai12)
 - **cat sai.txt |sed 's/sai/rangu/'** (changing all sai data to rangu but only one data in single line)
- ➔ Use **g** at last to changing all the data in line and file
 - **cat sai.txt | sed 's/sai/kiran/g'** (now all the data will change even if multipule data at same line)
- ➔ Use **d** to remove complete line which have data
 - **cat sai.txt | sed '/hmm/d'**
- ➔ Use **-e** to execute multipule sed commands in single line
 - **sed -e 's/sai/kiran/; s/7/12/ sai.txt**
- ➔ Use **"="** to specify line number
 - **sed "=" sai.txt**

```

saikiran@saikiran-virtual-machine:~$ echo sai kiran rangu sai | sed 's/sai/kiran/'
kiran kiran rangu sai
saikiran@saikiran-virtual-machine:~$ echo sai kiran rangu sai | sed 's/sai/kiran/g'
kiran kiran rangu kiran
saikiran@saikiran-virtual-machine:~$ cat main.txt | sed '/hai/d'
kiran 23
rannngggguuu 25
kiran-23
kiran-24
rannngggguuu-51
helo man
kirankiran eangu
23 45
23
45 2
23
55
34 55
saikiran@saikiran-virtual-machine:~$ sed -e 's/sai/kiran/; s/rangu/rsk/' main.txt
kiran 23
rannngggguuu 25
kiran-23
kiran-24
rannngggguuu-51
helo man
kirankiran eangu
hei thai is kirannkiran rannngggguuu

```

- ➔ use 'c to modify the specific line
 - sed '3c/hey saikiran rangu cool/' sai.txt (modify the 3rd line)
- ➔ use number at '2s to change specif line like '2s it will change 2ns occurrence
 - cat sai.txt | sed '2s/sai/kiran/' (2nd sai will change to kiran)
 - cat sai.txt | sed '1-3/sai/kiran/' (1-3 sai will change to kiran not all sai)
- ➔ remove numbers from file
 - cat main.txt | sed -r 's/[0-9]//g'
- ➔ remove empty line from file
 - cat main.txt | sed '/^\$/d'
- ➔ **tr:** used to translate like from lowercase to uppercase and vice versa or new lines into spaces.
- ➔ cat sai.txt | tr 'sai' 'SAI' (conver all s,a,i data as captal)
- ➔ use -s to multipule data into single data
 - Cat sai.txt | tr -s " " (remove all spaces)
 - Cat sai.txt | tr -s "I"
 - Car sai.txt | tr '\n' " " (all data in same line sparate by space)


```

saikiran@saikiran-virtual-machine:~$ cat main.txt
hwlllo sai
sai kiran
kiran sai
saikiran@saikiran-virtual-machine:~$ sed '2c/hey cool /' main.txt
hwlllo sai
/hey cool /
kiran sai
saikiran@saikiran-virtual-machine:~$
saikiran@saikiran-virtual-machine:~$ cat main.txt | sed '2s/sai/kiran/'
hwlllo sai
kiran kiran
kiran sai
saikiran@saikiran-virtual-machine:~$ cat main.txt | sed '1,3s/sai/kiran/'
hwlllo kiran
kiran kiran
kiran kiran
saikiran@saikiran-virtual-machine:~$ cat main.txt | tr "sai" "SAI"
hwlllo SAI
SAI kIrAn
kIrAn SAI
saikiran@saikiran-virtual-machine:~$ cat main.txt | tr -s "l"
hwlo sai
sai kiran
kiran sai
saikiran@saikiran-virtual-machine:~$ cat main.txt | tr '\n' " "
hwlllo sai sai kiran kiran sai saikiran@saikiran-virtual-machine:~$

```

- ➔ Convert lower to upper
 - Cat main2.txt | sed 'y/abcdefghijklmnopqrstuvwxyz/ABCDEFGHIJKLMNOPQRSTUVWXYZ/'
- ➔ Add some text at starting
 - Cat main2.txt | sed 's/^\$/https:\\\\/'
- ➔ remove numbers
 - cat main.txt | sed -r 's/[0-9]//g'

```

common - but not really, but there are 25 different ones!
saikiran@saikiran-virtual-machine:~$ cat main2.txt | sed 'y/abcdefghijklmnopqrstuvwxyz/ABCDEFGHIJKLMNOPQRSTUVWXYZ/'
SAI.COM
KIRAN.COM
RANGU.COM
SAI.XYZ
RANGU.XYZ
RSK.XYZ
saikiran@saikiran-virtual-machine:~$ cat main2.txt | sed 's/^/https:\/\/\\/'
https://sai.com
https://kiran.com
https://rangu.com
https://sai.xyz
https://rangu.xyz
https://rsk.xyz
saikiran@saikiran-virtual-machine:~$ cat main.txt
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu
45 56
hh 89
89 99
saikiran@saikiran-virtual-machine:~$ cat main.txt | sed -r 's/[0-9]/g'
sed: -e expression #1, char 9: unterminated 's' command
saikiran@saikiran-virtual-machine:~$ cat main.txt | sed -r 's/[0-9]//g'
sai
kiran
sai
kiran
hei saikiran rangu
hh

```

➔ remove empty space

- `cat main.txt | sed 's/^$/d/'`

```

saikiran@saikiran-virtual-machine:~$ cat main.txt
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu

45 56
hh 89
89 99
saikiran@saikiran-virtual-machine:~$ cat main.txt | sed '/^$/d'
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu
45 56
hh 89
89 99
saikiran@saikiran-virtual-machine:~$ █

```

33.uniq: used to remove all the repeated lines from a file. Also, it can be used to display a count of any word

- ➔ Uniq is used with |
 - `cat sai.txt | uniq`
- ➔ Use `-c` to count the occurence
 - `sort sai.txt | uniq -c`

```

saikiran@saikiran-virtual-machine:~$ cat sai.txt
sai 23
sai 23 sai
kiran
kiran

saikiran@saikiran-virtual-machine:~$ cat sai.txt | uniq
sai 23
sai 23 sai
kiran

saikiran@saikiran-virtual-machine:~$ cat sai.txt | uniq -c
      1 sai 23
      1 sai 23 sai
      2 kiran
      1
saikiran@saikiran-virtual-machine:~$

```

34.wc: Is used for word count by default it shows 3 numbers 1. Line, 2. Words, 3 char

- **wc sai.txt**
- ➔ use **-l** for get only line count
 - **wc -l sai.txt**
- ➔ use **-m** for count words
 - **wc -w sai.txt**
- ➔ use **-c** for count char
 - **wc -c sai.txt**

```

saikiran@saikiran-virtual-machine:~$ wc sai.txt
  5  7 32 sai.txt
saikiran@saikiran-virtual-machine:~$ wc -l sai.txt
 5 sai.txt
saikiran@saikiran-virtual-machine:~$ wc -w sai.txt
 7 sai.txt
saikiran@saikiran-virtual-machine:~$ wc -c sai.txt
32 sai.txt
saikiran@saikiran-virtual-machine:~$

```

35.od: 'od' term stands for octal dump. It displays content of a file in different human-readable formats like hexadecimal, octal and ASCII characters.

- **od -b <fileName>** (display files in octal format)
- **od -t x1 <fileName>** (display files in hexadecimal bytes format)

- `od c <fileName>` (display files in ASCII (backslashed) character format)

```
saikiran@saikiran-virtual-machine:~$ od sai.txt
00000000 060563 020151 031462 071412 064541 031040 020063 060563
00000020 020151 065412 071151 067141 065412 071151 067141 005012
00000040
saikiran@saikiran-virtual-machine:~$ od -b sai.txt
00000000 163 141 151 040 062 063 012 163 141 151 040 062 063 040 163 141
00000020 151 040 012 153 151 162 141 156 012 153 151 162 141 156 012 012
00000040
saikiran@saikiran-virtual-machine:~$ od c sai.txt
od: c: No such file or directory
00000000 060563 020151 031462 071412 064541 031040 020063 060563
00000020 020151 065412 071151 067141 065412 071151 067141 005012
```

36.sort: is used to sort in assending

- **sort sai.txt**
- ➔ for numeric use **-n**
- **sort -n sai.txt**
 - **sort -n -k2 sai.txt** (soring numbers by column 2)
- ➔ use **-k** for sort by column.
- **Sort -k2 sai.txt** (sorting by column 2)

```

saikiran@saikiran-virtual-machine:~$ cat sai.txt
sai 23
sai 24 sai
kiran
kiran 1

saikiran@saikiran-virtual-machine:~$ cat sai.txt | sort

kiran
kiran 1
sai 23
sai 24 sai
saikiran@saikiran-virtual-machine:~$ sort -n sai.txt

kiran
kiran 1
sai 23
sai 24 sai
saikiran@saikiran-virtual-machine:~$ sort sai.txt

kiran
kiran 1
sai 23
sai 24 sai
saikiran@saikiran-virtual-machine:~$ sort -k2 sai.txt

kiran
kiran 1
sai 23
sai 24 sai
saikiran@saikiran-virtual-machine:~$ █

```

37.find : find a particular file within a directory. It is used to find the list of files for the various conditions like permission, user ownership, modification, date/time, size, and more.

- ➔ (.) : For current directory name
- ➔ (/) : For the root directory
 - **Find . -name *.txt** (find all txt files)
 - **find ~ -name "*sai.txt"** (get file path)
 - **find *.txt**
 - **find .*txt**

```
saikiran@saikiran-virtual-machine:~$ find *.txt
main2.txt
sai.txt
saikiran@saikiran-virtual-machine:~$ find . *.txt
.
./.profile
./Videos
./snap
./snap/snapd-desktop-integration
./snap/snapd-desktop-integration/common
./snap/snapd-desktop-integration/common/.cache
```

➔ d: directory

- **find . -type d -name "*.sai"**

➔ searching the files which are newer than the mentioned file

- **find . newer sai.txt**

➔ -delete use for delete file but we can't undo

- **find . -name sai.txt -delete**

```
saikiran@saikiran-virtual-machine:~$ ls
Desktop Documents Downloads main2.txt Music Pictures Public sai sai.txt snap Templates Videos
saikiran@saikiran-virtual-machine:~$ find . -type d -name "*.sai"
saikiran@saikiran-virtual-machine:~$ find .newer main2.txt
find: '.newer': No such file or directory
main2.txt
saikiran@saikiran-virtual-machine:~$ find . -name "*.txt"
./.cache/tracker3/files/last-crawl.txt
./.cache/tracker3/files/first-index.txt
./.cache/tracker3/files/locale-for-miner-apps.txt
./sai.txt
./main2.txt
./Desktop/rangu/sai.txt
./Desktop/rangu/hh.txt
./Desktop/rangu/hello.txt
./Desktop/sai/sai.txt
./Desktop/sai/rangu/sai.txt
./Desktop/sai/rangu/hh.txt
./Desktop/sai/rangu/hello.txt
saikiran@saikiran-virtual-machine:~$ find *.txt
main2.txt
sai.txt
saikiran@saikiran-virtual-machine:~$ find ~ -name sai.txt
/home/saikiran/sai.txt
/home/saikiran/Desktop/rangu/sai.txt
/home/saikiran/Desktop/sai/sai.txt
/home/saikiran/Desktop/sai/rangu/sai.txt
saikiran@saikiran-virtual-machine:~$ ^C
saikiran@saikiran-virtual-machine:~$ find . -name sai.txt --delete
find: unknown predicate '--delete'
saikiran@saikiran-virtual-machine:~$ find .-name sai.txt -delete
find: '.-name': No such file or directory
saikiran@saikiran-virtual-machine:~$ find . -name sai.txt -delete
saikiran@saikiran-virtual-machine:~$ ls
Desktop Documents Downloads main2.txt Music Pictures Public sai snap Templates Videos
saikiran@saikiran-virtual-machine:~$
```

38.locate : locate command is a **background process and searches the file in the database** whereas, find command searches in the filesystem. The locate command is much faster than find command.

- **locate sai.txt**

```
saikiran@saikiran-virtual-machine:~$ ls
Desktop  Documents  Downloads  main2.txt  Music  Pictures  Public  sai  snap  Templates  Videos
saikiran@saikiran-virtual-machine:~$ locate main2.txt
Command 'locate' not found, but can be installed with:
sudo apt install plocate
saikiran@saikiran-virtual-machine:~$ sudo apt install plocate
[sudo] password for saikiran:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  liburing2
The following NEW packages will be installed:
  liburing2 plocate
0 upgraded, 2 newly installed, 0 to remove and 3 not upgraded.
Need to get 140 kB of archives.
After this operation, 555 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu jammy/main amd64 liburing2 amd64 2.1-2build1 [10.3 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy/main amd64 plocate amd64 1.1.15-1ubuntu2 [129 kB]
Fetched 140 kB in 2s (60.9 kB/s)
Selecting previously unselected package liburing2:amd64.
(Reading database ... 200563 files and directories currently installed.)
Preparing to unpack .../liburing2_2.1-2build1_amd64.deb ...
Unpacking liburing2:amd64 (2.1-2build1) ...
Selecting previously unselected package plocate.
Preparing to unpack .../plocate_1.1.15-1ubuntu2_amd64.deb ...
Unpacking plocate (1.1.15-1ubuntu2) ...
Setting up liburing2:amd64 (2.1-2build1) ...
Setting up plocate (1.1.15-1ubuntu2) ...
update-alternatives: using /usr/bin/plocate to provide /usr/bin/locate (locate) in auto mode
Adding group 'plocate' (GID 137) ...
Done.
Initializing plocate database; this may take some time... done
Created symlink /etc/systemd/system/timers.target.wants/plocate-updatedb.timer → /lib/systemd/system/plocate-upd
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.7) ...
saikiran@saikiran-virtual-machine:~$ locate main2.txt
/home/saikiran/main2.txt
```

- use -n 5 for set limit
 - **locate -n 5 *.txt**
- Use -i for ignore case-sentence
 - **locate -i *.txt**


```
saikiran@saikiran-virtual-machine:~$ locate -n3 "*.txt"
/boot/grub/gfxblacklist.txt
/etc/X11/rgb.txt
/etc/brltty/Input/ba/all.txt
saikiran@saikiran-virtual-machine:~$ locate -i -n5 '*.txt'
/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
saikiran@saikiran-virtual-machine:~$
```

39.date : is used to get date

- **date**
- **date -d now**
- **date -d yesterday**
- **date -d tomorrow**
- **date -d "nextday"**
- **date -d "lastday"**
- use **TZ** to get time zone
 - **TZ='Asia/Kolkata'** date

```
saikiran@saikiran-virtual-machine:~$ date
Sunday 12 May 2024 03:19:40 PM IST
saikiran@saikiran-virtual-machine:~$ date -d now
Sunday 12 May 2024 03:19:44 PM IST
saikiran@saikiran-virtual-machine:~$ date -d tomorrow
date: invalid date 'tomorrow'
saikiran@saikiran-virtual-machine:~$ date -d yesterday
Saturday 11 May 2024 03:20:05 PM IST
saikiran@saikiran-virtual-machine:~$ date -d "last monday"
Monday 06 May 2024 12:00:00 AM IST
saikiran@saikiran-virtual-machine:~$ date -d "next monday"
Monday 13 May 2024 12:00:00 AM IST
```

40.cal : is used to get Callander.

- **Cal**
- **Cal march 2023**

```

Processing triggers for Man-db (2.10.2-1) ...
saikiran@saikiran-virtual-machine:~$ cal
      May 2024
Su Mo Tu We Th Fr Sa
                1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

saikiran@saikiran-virtual-machine:~$ cal march 2023
      March 2023
Su Mo Tu We Th Fr Sa
                1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

saikiran@saikiran-virtual-machine:~$

```

41.sleep : is used to sleep by default it will sleep by seconds

- **sleep**
- **sleep 5**
- **sleep 1m**
- **sleep 1h**
- **sleep 1d**
- Sleep and play sound like alarm after time
 - **sleep 5; rhythmbox sound.mp3**

42.time : is used to say how long it took for execution

- **time ls**

43.df : df command is used to display the **disk space used in the file system**. The 'df' stands for "**disk filesystem**." It defines the number of blocks used, the number of blocks available, and the directory where the file system is mounted

- **df**
- **df -h** (get in human redable)

```
saikiran@saikiran-virtual-machine:~$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
tmpfs            396356        1956    394400    1% /run
/dev/sda3       25106692 13383248  10422760   57% /
tmpfs           1981772         0    1981772    0% /dev/shm
tmpfs            5120          4      5116    1% /run/lock
/dev/sda2       524252        6220    518032    2% /boot/efi
tmpfs           396352        124    396228    1% /run/user/1000
/dev/sr0        159072    159072         0 100% /media/saikiran/CDROM
saikiran@saikiran-virtual-machine:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            388M  2.0M   386M   1% /run
/dev/sda3       24G   13G   10G   57% /
tmpfs            1.9G     0   1.9G   0% /dev/shm
tmpfs            5.0M  4.0K   5.0M   1% /run/lock
/dev/sda2       512M  6.1M  506M   2% /boot/efi
tmpfs            388M  124K   387M   1% /run/user/1000
/dev/sr0        156M  156M     0 100% /media/saikiran/CDROM
```

44.awk : The Awk is a powerful scripting language used for **text scripting**. It searches and replaces the texts and sorts, validates, and indexes the database.

- Awk stands for three names who wrote of found awk script.
 - **awk '/sai/ {print}' sai.txt** (get all data which line have sai)
 - **awk '{print}' sai.txt** (display the data)
 - **awk '{print \$1,\$5}' sai.txt** (will print 1 and 5th column if 5th Is not available the 1st column will only print)

```

saikiran@saikiran-virtual-machine:~$ cat main2.txt
sai 23
kiran 24
sai-23
kiran-24
hei saikiran rangu
45saikiran@saikiran-virtual-machine:~$ awk '/sai/{print}' main2.txt
sai 23
sai-23
hei saikiran rangu
saikiran@saikiran-virtual-machine:~$ awk '{print}' main2.txt
sai 23
kiran 24
sai-23
kiran-24
hei saikiran rangu
45
saikiran@saikiran-virtual-machine:~$ awk '{print $1,$3}' main2.txt
sai
kiran
sai-23
kiran-24
hei rangu
45

```

- Cat main.txt | awk '{print \$2}'

```
saikiran@saikiran-virtual-machine:~$ cat main.txt
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu
45 56
hh 89
89 99
saikiran@saikiran-virtual-machine:~$ cat main.txt | awk '{print $1}'
sai
kiran
sai
kiran
hei
45
hh
89
saikiran@saikiran-virtual-machine:~$ cat main.txt | awk '{print $2}'
23
24
23
24
saikiran
56
89
99
saikiran@saikiran-virtual-machine:~$
```

- **awk '{print NR \$0} sai.txt** (will print the line numbers)
- **awk '{print \$NF} sai.txt** (display the last field)
- **awk -F"," '{x+=\$3}END{print x}' sai.txt** (calculate the third column)

```

saikiran@saikiran-virtual-machine:~$ cat main.txt | awk '/sai/{print $2}'
23
23
saikiran
saikiran@saikiran-virtual-machine:~$ cat main.txt | awk '/kiran/{print $1}'
kiran
kiran
hei
saikiran@saikiran-virtual-machine:~$ awk -f '.' '{print $1}' main2.txt
awk: read error (Is a directory)
saikiran@saikiran-virtual-machine:~$ awk -F '.' '{print $1}' main2.txt
sai
kiran
rangu
sai
rangu
rsk
saikiran@saikiran-virtual-machine:~$ awk '{s+=$1} END {print}' main.txt
89 99
saikiran@saikiran-virtual-machine:~$ awk '{s+=$2} END {print}' main.txt
89 99
saikiran@saikiran-virtual-machine:~$ awk '{s+=$2} END {print s}' main.txt
338
saikiran@saikiran-virtual-machine:~$ awk '{s+=$1} END {print s}' main.txt
134
saikiran@saikiran-virtual-machine:~$

```

45.zip: The zip command in Linux is used to compress files and directories into a single zip archive.

- **Zip sai.zip main.txt main2.txt** (converting main, main2 txt files to zip)
- **Zip -r sai.zip sai** (converting all files in directorus to zip)

```

saikiran@saikiran-virtual-machine:~$ ls
Desktop Documents Downloads main2.txt main.txt main.txtu Music Pictures Public rangu.txt sai sai.txt snap Templates Vid
saikiran@saikiran-virtual-machine:~$ zip sai.zip main.txt main2.txt main.txtu
  adding: main.txt (deflated 31%)
  adding: main2.txt (deflated 35%)
  adding: main.txtu (deflated 29%)
saikiran@saikiran-virtual-machine:~$ ls
Desktop Documents Downloads main2.txt main.txt main.txtu Music Pictures Public rangu.txt sai sai.txt sai.zip snap Templ
saikiran@saikiran-virtual-machine:~$ unzip sai.zip
Archive:  sai.zip
replace main.txt? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
  inflating: main.txt
replace main2.txt? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
  inflating: main2.txt
replace main.txtu? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
  inflating: main.txtu
saikiran@saikiran-virtual-machine:~$ ls
Desktop Documents Downloads main2.txt main.txt main.txtu Music Pictures Public rangu.txt sai sai.txt sai.zip snap Templ

```

46.Unzip: convert zip files to unzip

- Unzip sai.zip

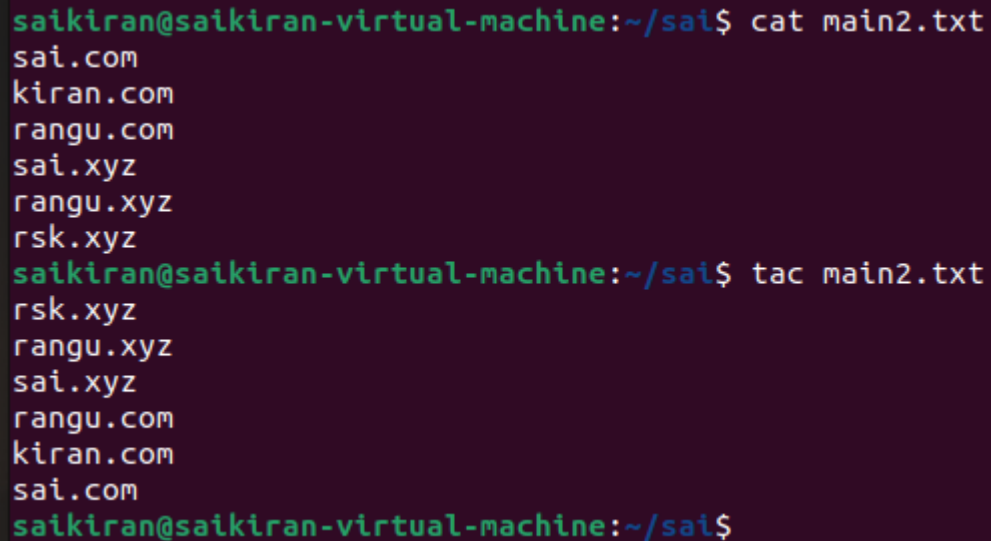
```
saikiran@saikiran-virtual-machine:~$ ls
Desktop  Downloads  Pictures  rangu.txt  sai.txt  Templates
Documents Music      Public   sai        snap    Videos
saikiran@saikiran-virtual-machine:~$ zip -r sai.zip sai
adding: sai/ (stored 0%)
adding: sai/main2.txt (deflated 35%)
adding: sai/main.txt (deflated 31%)
adding: sai/main.txtu (deflated 29%)
saikiran@saikiran-virtual-machine:~$ ls
Desktop  Downloads  Pictures  rangu.txt  sai.txt  snap
Documents Music      Public   sai        sai.zip  Templates
saikiran@saikiran-virtual-machine:~$ cd sai
saikiran@saikiran-virtual-machine:~/sai$ ls
main2.txt  main.txt  main.txtu
saikiran@saikiran-virtual-machine:~/sai$ ccat main.txt
Command 'ccat' not found, but can be installed with:
sudo apt install ccrypt
saikiran@saikiran-virtual-machine:~/sai$ cd
saikiran@saikiran-virtual-machine:~$ unzip sai.zip
Archive:  sai.zip
replace sai/main2.txt? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
inflating: sai/main2.txt
inflating: sai/main.txt
inflating: sai/main.txtu
saikiran@saikiran-virtual-machine:~$ cd sai
saikiran@saikiran-virtual-machine:~/sai$ ls
main2.txt  main.txt  main.txtu
saikiran@saikiran-virtual-machine:~/sai$ cat main.txt
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu
```

47.sudo: super user do it is mainly used **run as administrator** like install ans useradd ect.

- sudo apt install mail
- sudo adduser saikiram
- sudo groupadd sai

48.tac: The 'tac' command is the reverse of the 'cat' command. It is also known as 'cat' backward. It will display the file content in reverse order. It prints the last line first, then second last and so on.

- **Tac main.txt**



```
saikiran@saikiran-virtual-machine:~/sai$ cat main2.txt
sai.com
kiran.com
rangu.com
sai.xyz
rangu.xyz
rsk.xyz
saikiran@saikiran-virtual-machine:~/sai$ tac main2.txt
rsk.xyz
rangu.xyz
sai.xyz
rangu.com
kiran.com
sai.com
saikiran@saikiran-virtual-machine:~/sai$
```

49.more: As 'cat' command displays the file content. Same way 'more' command also displays the content of a file. Only difference is that, in case of larger files, 'cat' command output will scroll off your screen while 'more' command displays output one screenful at a time.

- Enter key: To scroll down page line by line.
- Space bar: To go to next page.
- b key: To go to the backward page.
- / key: Lets you search the string.

- **More main.txt**


```

saikiran@saikiran-virtual-machine:~/sa$ more main.txt
sa$ 23
kiran 24
sa$ 23
kiran 24
he! saikiran rangu
The 'tac' command is the reverse of the 'cat' command. It is also known as 'cat' backward. It will display the file content in reverse order. It prints the last line first, then s
o Tac main.txt
49.more: As 'cat' command displays the file content. Same way 'more' command also displays the content of a file. Only difference is that, in case of larger files, 'cat' command c
r screen while 'more' command displays output one screenful at a time.
• Enter key: To scroll down page line by line.
• Space bar: To go to next page.
• b key: To go to the backward page.
• / key: Lets you search the string.
o

50.less: similar to more it has adjustment like hight and weidth
51.head: display starting 10 line by default
o head main.txt (display 1st 10 lines by default)
o head main.txt main2.txt (display 1st 5 lines in both files by default)
The 'tac' command is the reverse of the 'cat' command. It is also known as 'cat' backward. It will display the file content in reverse order. It prints the last line first, then s
o Tac main.txt
49.more: As 'cat' command displays the file content. Same way 'more' command also displays the content of a file. Only difference is that, in case of larger files, 'cat' command c
r screen while 'more' command displays output one screenful at a time.
• Enter key: To scroll down page line by line.
• Space bar: To go to next page.
• b key: To go to the backward page.
• / key: Lets you search the string.
o

50.less: similar to more it has adjustment like hight and weidth
51.head: display starting 10 line by default
o head main.txt (display 1st 10 lines by default)
o head main.txt main2.txt (display 1st 5 lines in both files by default)
o head -3 main.txt (display only starting 3 lines)
52.tail: display last 10 line by default
o tail main.txt (display last 10 lines by default)
o tail main.txt main2.txt (display last 5 lines in both files by default)
o tail -3 main.txt (display only last 3 lines)

53.clear: used to clear the terminal
54.exit: Exit the terminal
55.mail: Used to send the mails
o mail -s "Subject" <recipient address>
o mail -s "hello sa!" mrsaikiranrangug@gmail.com<<hello sa!
o mail -s "hello sa!" mrsaikiranrangug@gmail.com
cc hello sa!
# coding: utf-8

```

50.less: similar to more it has adjustment like hight and weidth

o Less main.txt

```

sa$ 23
kiran 24
sa$ 23
kiran 24
he! saikiran rangu
The 'tac' command is the reverse of the 'cat' command. It is also known as 'cat' backward. It will display the file content in reverse order. It prints the last line first, then s
o Tac main.txt
49.more: As 'cat' command displays the file content. Same way 'more' command also displays the content of a file. Only difference is that, in case of larger files, 'cat' command c
r screen while 'more' command displays output one screenful at a time.
• Enter key: To scroll down page line by line.
• Space bar: To go to next page.
• b key: To go to the backward page.
• / key: Lets you search the string.
o

50.less: similar to more it has adjustment like hight and weidth
51.head: display starting 10 line by default
o head main.txt (display 1st 10 lines by default)
o head main.txt main2.txt (display 1st 5 lines in both files by default)
The 'tac' command is the reverse of the 'cat' command. It is also known as 'cat' backward. It will display the file content in reverse order. It prints the last line first, then s
o Tac main.txt
49.more: As 'cat' command displays the file content. Same way 'more' command also displays the content of a file. Only difference is that, in case of larger files, 'cat' command c
r screen while 'more' command displays output one screenful at a time.
• Enter key: To scroll down page line by line.
• Space bar: To go to next page.
• b key: To go to the backward page.
• / key: Lets you search the string.
o

50.less: similar to more it has adjustment like hight and weidth
51.head: display starting 10 line by default
o head main.txt (display 1st 10 lines by default)
o head main.txt main2.txt (display 1st 5 lines in both files by default)
o head -3 main.txt (display only starting 3 lines)
52.tail: display last 10 line by default
o tail main.txt (display last 10 lines by default)
o tail main.txt main2.txt (display last 5 lines in both files by default)
o tail -3 main.txt (display only last 3 lines)

53.clear: used to clear the terminal
54.exit: Exit the terminal
55.mail: Used to send the mails
o mail -s "Subject" <recipient address>
o mail -s "hello sa!" mrsaikiranrangug@gmail.com<<hello sa!
o mail -s "hello sa!" mrsaikiranrangug@gmail.com
cc hello sa!
# coding: utf-8

```

51.head: display starting 10 line by default

- **head main.txt** (display 1st 10 lines by default)
- **head main.txt main2.txt** (display 1st 5 lines in both files by default)
- **head -3 main.txt** (display only starting 3 lines)

```
saikiran@saikiran-virtual-machine:~/sai$ head main.txt
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu

saikiran@saikiran-virtual-machine:~/sai$ head main.txt main2.txt
==> main.txt <==
sai 23
kiran 24
sai 23
kiran 24
hei saikiran rangu

==> main2.txt <==
sai.com
kiran.com
rangu.com
sai.xyz
rangu.xyz
rsk.xyz
saikiran@saikiran-virtual-machine:~/sai$ head -3 main.txt
sai 23
kiran 24
sai 23
saikiran@saikiran-virtual-machine:~/sai$
```

52.tail: display last 10 line by default

- **tail main.txt** (display last 10 lines by default)
- **tail main.txt main2.txt** (display last 5 lines in both files by default)

- **tail -3 main.txt** (display only last 3 lines)

```
saikiran@saikiran-virtual-machine:~/sai$ tail main.txt

45 56
hh 89
89 99
saikiran@saikiran-virtual-machine:~/sai$ tail main2.txt main.txt
==> main2.txt <==
sai.com
kiran.com
rangu.com
sai.xyz
rangu.xyz
rsk.xyz

==> main.txt <==

45 56
hh 89
89 99
saikiran@saikiran-virtual-machine:~/sai$ tail -3 main2.txt
sai.xyz
rangu.xyz
rsk.xyz
saikiran@saikiran-virtual-machine:~/sai$
```

53.clear: used to clear the terminal

54.exit: Exit the terminal

55.mail: Used to send the mails

- **mail -s "Subject" <recipient address>**
- **mail -s "hello sai" mrsaikiranrangu@gmail.com<<<hello sai**
- **mail -s "hello sai" mrsaikiranrangu@gmail.com**

cc hello sai

ctrl+D

```
saikiran@saikiran-virtual-machine:~$ mail -s "hello sai" saikiranrangu6@gmail.cocm
Cc:
close
saikiran@saikiran-virtual-machine:~$ mail -s "hello sai" saikiranrangu6@gmail.com <<< "This is the body of the email"
saikiran@saikiran-virtual-machine:~$ mail -s "hello sai" mrsaikiranrangu@gmail.com <<< "This is the body of the email"
saikiran@saikiran-virtual-machine:~$ mail -s "hello sai" mrsaikiranrangu@gmail.com
Cc: hello sai kiran
```