```
1010100000110100001000001010010011000001000010100
0010111100111110011111101111111111111100111110101100
1110110100110011001100110011111111111101001100111111
0011000100000011001 010011 011 01001 010000
0110010001011000010 000110 010 00010 000101
1   0111 101111 11 1 01 1 0 1 11 11 1 011
0    010    0  0    1 0        10   1  00
     101  1  1    0  1        01    0   1
```

400 Linux commands
with examples

```
[~] 🔥 ls -ltr /
total 52
drwx------    2 root root 16384 Aug 16 07:54 lost+found
drwxr-xr-x    4 root root  4096 Aug 16 14:54 srv
drwxr-xr-x    3 root root  4096 Aug 16 20:23 home
drwxr-xr-x    7 root root  4096 Sep 19 20:51 mnt
lrwxrwxrwx    1 root root     7 Oct 19 02:31 sbin -> usr/bin
lrwxrwxrwx    1 root root     7 Oct 19 02:31 lib64 -> usr/lib
lrwxrwxrwx    1 root root     7 Oct 19 02:31 lib -> usr/lib
lrwxrwxrwx    1 root root     7 Oct 19 02:31 bin -> usr/bin
drwxr-xr-x    3 root root  4096 Oct 26 07:22 opt
drwxr-x---    7 root root  4096 Nov 14 19:30 root
drwxr-xr-x    2 root root  4096 Nov 14 19:30 boot
drwxr-xr-x   11 root root  4096 Nov 19 12:15 usr
drwxr-xr-x   12 root root  4096 Nov 20 07:13 var
dr-xr-xr-x  203 root root     0 Nov 27 06:53 proc
dr-xr-xr-x   13 root root     0 Nov 27 06:53 sys
drwxr-xr-x   19 root root   460 Nov 27 06:53 run
drwxr-xr-x   63 root root  4096 Nov 27 06:53 etc
drwxr-xr-x   20 root root  4040 Nov 27 08:31 dev
drwxrwxrwt    9 root root   240 Nov 27 08:31 tmp
```

Indian Linux User Group Chennai (ILUGC)
Kanchipuram Linux User Group (KanchiLUG)

```
[~] 🔥 man --help
Usage: man [OPTION...] [SECTION] PAGE...

  -C, --config-file=FILE     use this user configuration file
  -d, --debug                emit debugging messages
  -D, --default              reset all options to their default
values
      --warnings[=WARNINGS]  enable warnings from groff

 Main modes of operation:
  -f, --whatis               equivalent to whatis
  -k, --apropos              equivalent to apropos
  -K, --global-apropos       search for text in all pages
  -l, --local-file           interpret PAGE argument(s) as local
filename(s)
  -w, --where, --path, --location
```

Title : 400 Linux Commands with examples

Indian Linux User Group , Chennai https://ilugc.in/

Kanchi Linux User Group , Kanchipuram https://kanchilug.wordpress.com/

Content created : Dhanasekar T

email id : tkdhanasekar@gmail.com

Cover Image : Parameshwar Arunachalam

email id : stark20236@gmail.com

Table of Contents

● <u>1 apt</u>

apt - command line interface for Ubuntu and Debian based systems
apt provides a high-level command line interface for the package
management system

To install packages
$ sudo apt install package_name

To Check All Dependencies of a Package
$ sudo apt depends bind9

To Search for a Package
$ sudo apt search apache2

To View Information About Package
$ sudo apt show apache2

To Verify a Package for any Broken Dependencies
$ sudo apt check apache2

To Update System Packages
$ sudo apt update

To Upgrade System
$ sudo apt upgrade

To Remove Unused Packages
$ sudo apt autoremove

To Clean Old Repository of Downloaded Packages
$ sudo apt autoclean

To Remove Packages with its Configuration Files

```
$ sudo apt purge apache2
```

To Install .Deb Package

```
$ sudo apt deb package-amd64.deb
```

To Find Help for apt

```
$ sudo apt help
```

To Remove Packages

```
$ sudo apt remove package_name
```

To List Packages

```
$ sudo apt list
```

● **2 apt-cache**

apt-cache - query the APT cache for debian and ubuntu based system

Examples:

To Find Out Package Name and Description of Software
```
$ sudo apt-cache search vsftpd
```

To find and list down all the packages starting with apache2
```
$ sudo apt-cache pkgnames apache2
```

To List All Available Packages
```
$ sudo apt-cache pkgnames
```

To Check Package Information
```
$ sudo apt-cache show apache2
```

To Check Dependencies for Specific Packages
```
$ sudo apt-cache showpkg vsftpd
```

To Check statistics of Cache
```
$ sudo apt-cache stats
```

To install Packages without Upgrading
```
$ sudo apt-get install packageName --no-upgrade
```

To Upgrade Only Specific Packages
```
$ sudo apt-get install packageName --only-upgrade
```

To Remove Packages Without Configuration
```
$ sudo apt-get remove package_name
```

To Completely Remove Packages

```
$ sudo apt-get purge package_name
```

To Clean Up Disk Space

```
$ sudo apt-get clean
```

To Download a Package Without Installing

```
$ sudo apt-get download apache2
```

To Check Change Log of Package

```
$ sudo apt-get changelog vsftpd
```

To Check Broken Dependencies

```
$ sudo apt-get check
```

To Auto clean Apt-Get Cache

```
$ sudo apt-get autoclean
```

● <u>3 ar</u>

ar - create, modify, and extract from archives

install binutils for ar
$ sudo apt install binutils

create 4 text files file1.txt file2.txt file3.txt file4.txt

general syntax to create new archive
$sudo ar r [archive file] [file(s)]

To create a new archive myfiles.a and place all .txt files in archive
$sudo ar r myfiles.a *.txt

To add a new file file5.txt to the archive
$sudo ar r myfiles.a file5.txt

To print the archive members
$sudo ar p myfiles.a

To print the archive contents in a list format
$sudo ar t myfiles.a

To extract a file4.txt from archive
$sudo ar x myfiles.a file4.txt

To Extract multiple files from archive
$sudo ar x myfiles.a file1.txt file2.txt file3.txt

To Extract all files
$sudo ar x myfiles.a

To delete a file from an archive

$sudo ar d myfiles.a file5.txt

To delete multiple files

$sudo ar d myfiles.a file5.txt file4.txt file3.txt

To read the contents of an archive

$sudo ar pv myfiles.a

- ## **4 add-apt-repository**

add-apt-repository - Adds a repository into the
/etc/apt/sources.list
or /etc/apt/sources.list.d or removes an existing one
example:
$ sudo add-apt-repository ppa:PPA_REPOSITORY_NAME/PPA
$ sudo add-apt-repository ppa:libreoffice/ppa

To list all repositories
$ sudo apt policy

To remove PPA repository
$ sudo add-apt-repository --remove ppa:PPA_REPOSITORY_NAME/PPA
$ sudo add-apt-repository --remove ppa:libreoffice/ppa

● **5 adduser**

adduser - add a user to the system

install the adduser package
```
$ sudo apt install adduser
```

To add a new user
```
$ adduser username
```

To add a user with a different shell.
```
$ sudo adduser username --shell /bin/sh
```

To add a new user with a different configuration file
```
$ sudo adduser username --conf custom_config.conf
```

To add a user with different home directory.
```
$ sudo adduser username --home /home/klug/
```

To get the version of the adduser command
```
$ sudo adduser --version
```

To display the help section of the adduser command
```
$ sudo adduser -h
```

● 6 useradd

useradd - create a new user or update default new user information

To add a new user klug
```
$ sudo useradd klug
```

To set a password for account klug
```
$ sudo passwd klug
```

To create a User with Different Home Directory
```
$ sudo useradd -d /data/myprojects klug
```

To view user related info
```
$ sudo cat /etc/passwd | grep klug
```

To create a User with a Specific User ID
```
$ sudo useradd -u 1007 klug
```

Create a User with a Specific Group ID
```
$ sudo useradd -u 1007 -g mygroup klug
```
To verify the user's GID
```
$ id -gn klug
```

To Add a User klug to Multiple Groups
```
$ sudo groupadd admins
$ sudo groupadd devops
$ sudo groupadd cloud
$ sudo usermod -a -G admins,devops,cloud klug
$ sudo useradd -G admins,devops,cloud ilugc
```

To verify

```
$ id klug
$ id ilugc
```

To Add a User without Home Directory

```
$ sudo useradd -M klug
```

to check

```
$ ls -l /home/klug
```

To Create a User with Account Expiry Date

```
$ sudo useradd -e 2022-08-30 klug
```

To verify the age of the account and password

```
$ chage -l klug
```

To Create a User with Password Expiry Date

```
$ sudo useradd -e 2022-04-01 -f 40 klug
```

To verify

```
$ sudo chage -l klug
```

To Add a User with Custom Comments

```
$ sudo useradd -c "Welcome to foss world  +91-9999988888" klug
```

To verify

```
$ sudo tail -1 /etc/passwd
```

To Create User Login Shell in Linux

```
$ sudo useradd -s /sbin/nologin klug
```

To check

```
$ sudo tail -1 /etc/passwd
```

To Add a User with Specific Home Directory, Default Shell, and Custom Comment

```
$ sudo useradd -m -d /var/www/klug -s /bin/bash -c "website admin" -U klug
```

To Add a User with Home Directory, Custom Shell, Custom Comment, and UID/GID

```
$ sudo useradd -m -d /var/www/klug -s /bin/sh -c "website admin" -u 1000 -g 100 klug
```

To Add a User with Home Directory, No Shell, Custom Comment, and User ID

```
$ sudo useradd -m -d /var/www/klug -s /usr/sbin/nologin -c "web admin" -u 1001 klug
```

To Add a User with Home Directory, Shell, Custom Skell/Comment, and User ID

```
$ sudo useradd -m -d /var/www/klug -k /etc/custom.skell -s /bin/sh -c "custom message" -u 1020 klug
```

To Add a User without Home Directory, No Shell, No Group, and Custom Comment

```
$ sudo useradd -M -N -r -s /bin/false -c "Disabled group Member" klug
```

● **7_groupadd**

groupadd - create a new group

To create a new Linux group
$ sudo groupadd webadmin
To check
$ sudo grep webadmin /etc/group

To Create new group with a specific groupid
$ sudo groupadd webadmin -g 1030
To check
$ sudo grep 1030 /etc/group

To create group with group id with certain range of id
$ sudo groupadd webadmin -K GID_MIN=1500 -K GID_MAX=2000

● 8 add group

addgroup - add group to the system

install addgroup package
$ sudo apt install addgroup

To add a new group ilugc
$ sudo addgroup ilugc

To add a new group with specified group id
$ sudo addgroup klug --gid 6789

To create a group with a specific shell
$ sudo addgroup klug --shell /bin/sh

To enter verbose mode
$ sudo addgroup webadmin --debug

To display help related to addgroup command.
$ addgroup --help

● <u>9 alias</u>

alias - customised shortcut for commands

```
$ sudo alias name="value"
```

create a user klug with home directory then,

```
$ sudo alias cd="cd /home/klug"
root@klug:~# cd
root@klug:/home/klug
```

```
$ sudo alias d="df -Th"
root@klug:~# d
```

To print all the defined aliases is reusable format

```
# alias -p
```

● **<u>10 unalias</u>**

unalias - this command will remove the customised shortcuts created in alias

unalias - Removing an existing alias

$ sudo unalias [alias name]

$ sudo alias d="df -Th"

to remove the alias d

To check

$ sudo alias -p

$ sudo unalias d

will remove the shortcut d for df -Th

● **11 apg**

apg - generates several random passwords

$ apg -h

display the options

$ apg -n 2 -m 8 -x 10

-n number of passwords

-m minimum password length

-x maximum password length

will give 2 passwords with min password length 8 characters and max 10 characters

● <u>12 apropos</u>

apropos - search the manual page names and descriptions

example:

$ apropos useradd

$ apropos adduser

$ apropos df

$ apropos free

$ apropos command_name

- ## 13 arch

arch - print machine hardware name

```
$ arch
```

● <u>14 badblocks</u>

badblocks - search a device for bad blocks

By default it doesn't display any output on the screen, when there are no bad blocks as shown below.
```
$ sudo badblocks /dev/sda1
```

To view the badblocks search in verbose mode i.e how much scanning it has done so far
```
$ sudo badblocks -v /dev/sda1
```

By default it uses 1024 as block size , we specify a block size using -b option
```
$ sudo badblocks -v -b 2048 /dev/sda1
```

To Specify Maximum Bad Blocks Count to 100
```
$ sudo badblocks -v -e 100 /dev/sda1
```

Write the Badblocks to a File
```
$ sudo badblocks -v -o badblocks.log /dev/sdb1
```

To Perform a Badblock Write Mode Test
```
$ sudo badblocks -vw /dev/sda1
```

To display current progress of the test
```
$ sudo badblocks -s /dev/sda1
```

To specify the number of blocks to be tested at a time ,the default is 64 blocks.
```
$sudo badblocks -sc 2000 /dev/sda1
```

To write the list of badblocks to a file rather than on standard output

```
$ sudo badblocks -o out.txt /dev/sda1
```

To provide an input file which contains a list of known bad blocks in device, it will skip the known bad blocks at the time of test

```
$ sudo badblocks -i known-badblocks.txt /dev/sda1
```

To perform a non-destructive read-write test on device,

```
$ sudo badblocks -sn /dev/sda1
```

To test blocks from the first block to the specified last block

```
$ sudo badblocks -s /dev/sda1 2000
```

It is specified by passing the starting block number to test as an option after last block.

```
$ sudo badblocks -s /dev/sda1 2000 200
```

● <u>15 bg</u>

bg command in linux is used to place foreground jobs in background.

$ ping google.com

press CTRL+Z

To view running jobs (in my environment)

$ jobs -l

[1]+ 73192 Stopped   ping google.com

To resume the job ping google.com job with job number 1

$ bg %1

To kill the job # ping google.com

$ kill -s stop 73192

or

$ kill -s stop 1

or

$ pkill -stop 73192

or

To kill the job

$ kill -9 73192

● **16 blkid**

blkid - locate/print block device attributes

To display all the block devices
$ sudo blkid

To display the I/O limits on a particular block device
$ sudo blkid -i /dev/vda1

To displays information about /dev/vda1
$ sudo blkid -p /dev/vda1
$ sudo blkid -pi /dev/vda1

To look up the devices that matches a specific search criteria
$ sudo blkid -l -t TYPE=ext4
$ sudo blkid -l -t TYPE=swap

search based on UUID
$ sudo blkid -U 02a5af55-4c2a-45b7-9876-599abc192ada

To display in list format
$ sudo blkid -o list

● <u>**17 bluetoothctl**</u>

bluetoothctl - interactive bluetooth control tool

check for bluetoothctl status
```
$ sudo systemctl status bluetooth
$ sudo systemctl start bluetooth
$ sudo systemctl enable bluetooth
```

search for Bluetooth devices
```
$ bluetoothctl scan on
```

To make your Bluetooth adapter discoverable to other devices
```
$ bluetoothctl discoverable on
```

To connect with a Bluetooth device is to pair it with your PC
using the pair command
```
$ bluetoothctl pair MAC_ID_of_Device
```

To connect with already paired device
```
$ bluetoothctl connect MAC_ID-of_Device
```

To List Paired Devices With bluetoothctl
```
$ bluetoothctl paired-devices
```

To list devices that are within the Bluetooth range of your
computer
```
$ bluetoothctl devices
```

To trust a Bluetooth device
```
$ bluetoothctl trust MAC_ID_of_Device
```

To untrust a device

```
$ bluetoothctl untrust MAC_ID_of_Device
```

To unpair a Bluetooth device

```
$ bluetoothctl remove MAC_ID_of_Device
```

To disconnect a device from system

```
$ bluetoothctl disconnect MAC_ID_of_Device
```

To block a specific device from connecting to system

```
$ bluetoothctl block MAC_ID_of_Device
```

To enter interactive mode

```
$ bluetoothctl
[bluetooth]# devices
[bluetooth]# exit
```

● <u>18 brctl</u>

brctl - ethernet bridge administration

```
$ sudo apt install bridge-utils
```

To Create New Ethernet Bridge using addbr

```
$ sudo brctl addbr dev
$ sudo brctl addbr stage
$ sudo brctl addbr prod
```

To Display Available Ethernet Bridge using show

```
$ sudo brctl show
```

To Delete Existing Ethernet Bridge using delbr

```
$ sudo brctl delbr dev
```

To Add an Interface to Existing Bridge

```
$ sudo brctl addif dev eth0
```

To Add Multiple Interfaces to Existing Bridge

```
$ sudo brctl addif dev eth0 eth1
```

To Track MAC address of a Bridge

```
$ sudo brctl showmacs dev
```

To Set Ageing Time for Mac Address on a Bridge

```
$ sudo brctl setaging dev 300
```

To Setup Spanning Tree on Ethernet Bridge

```
$ sudo brctl stp dev on
```
or
```
$ sudo brctl stp dev yes
```

To turn off spanning tree on your ethernet bridge

```
$ sudo brctl stp dev off
```

To Display STP Parameter Values of a Bridge

```
$ sudo brctl showstp dev
```

To Change Bridge Parameters Values

```
$ sudo brctl setageing dev 200
```

● 19 bunzip2

bunzip2 - a block-sorting file compressor

To compress file input.txt it deletes original

$ bzip2 -z input.txt

will give input.txt.bz2

To decompress the input.txt.bz2

$ bzip2 -d input.txt.bz2

To compress file input.txt but does not deletes the original file

$ bzip2 -k input.txt

To check the integrity of file and to check file is corrupt or not

$ bzip2 -t input.txt.bz2

To show the compression ratio for each file processed in verbose mode

$ bzip2 -v input.txt

• **20 bzcat**

bzcat - decompresses files to stdout

bzcat - decompresses files to stdout

To read the compressed file without decompressing it

example:

create a file number.txt

$ echo "for(i=1; i<=10000; i++) {i;}" | bc > number.txt

bzip the number.txt file

$ bzip2 number.txt

$ bzcat number.txt.bz2

● **21 bzip2recover**

bzip2recover - recovers data from damaged bzip2 files

example:

$ bzip2recover file_name

$ bzip2recover archive.tar.bz2

● <u>22 blkdeactivate</u>

blkdeactivate — utility to deactivate block devices

To Deactivate all supported block devices , If a device is mounted, skip its deactivation

$ sudo blkdeactivate

To Deactivate all supported block devices , If a device is mounted, unmount it

$ sudo blkdeactivate -u

● <u>23 bc</u>

bc - An arbitrary precision calculator language

```
$ echo "12+5" | bc
$ echo "10^2" | bc
```

To store the result of complete operation in variable

```
$ x=`echo "12+5" | bc`
$ echo $x
$ echo "var=10;var" | bc
$ echo "var=10;var^=2;var" | bc
```

To store the result of complete operation in variable

```
$ x=`echo "var=500;var%=7;var" | bc`
$ echo $x
```

```
$ echo "var=11;++var" | bc
```

Variable is increased first and then result of variable is stored

```
$ echo "var=20;var++" | bc
```

Result of the variable is used first and then variable is incremented

```
$ echo "var=20;--var" | bc
```

Variable is decreased first and then result of variable is stored

```
$ echo "var=10;var--" | bc
```

Result of the variable is used first and then variable is decremented.

● [24 baobab](#)

Baobab - A graphical tool to analyze disk usage

$ baobab

$ baobab /dev/

● **25 apparmor**

AppArmor is a Linux kernel security module that allows the system administrator to restrict programs capabilities with per-program profiles in ubuntu, its similar to selinux in redhat based systems

apparmor_status - display various information about the current AppArmor policy

$ sudo apparmor_status

● **26 aa-enabled**

aa-enabled - test whether AppArmor is enabled in ubuntu systems

$ aa-enabled
Yes

● **27 aa-remove-unknown**

aa-remove-unknown - remove unknown AppArmor profiles

```
$ sudo aa-remove-unknown
```

● **28 aa-status**

aa-status - display various information about the current AppArmor policy.

```
$ sudo aa-status
```

- ## **29 aa-teardown**

aa-teardown - unload all AppArmor profiles

```
$ sudo aa-teardown
```

● <u>30 bzdiff</u>

bzdiff - compare bzip2 compressed files

examples:

To output a normal diff

```
$ bzdiff --normal file1.bz2 file2.bz2
```

To output in two columns

```
$ bzdiff -y file1.bz2 file2.bz2
```

● <u>31 bzcmp</u>

bzcmp - compare two bzip2 compressed file internally it uses cmp command

example:

$ bzcmp -b file1.bz2 file2.bz2

- ## 32 bzgrep

bzgrep - search possibly bzip2 compressed files for a regular
expression

example:

$ bzgrep -i "keyword" file.txt.bz2

$ bzgrep -i "keyword" file1.bz2

● **33 bzless**

bzless - file perusal filter for crt viewing of bzip2
compressed text

example:

```
$ echo "for(i=1; i<=10000; i++) {i;}" | bc > number.txt
$ bzip2 number.txt
$ bzless number.txt.bz2
```

● **34 bzmore**

bzmore - file perusal filter for crt viewing of bzip2 compressed text

To view the content of bzip2 compressed files page by page.

example:

$ echo "for(i=1; i<=10000; i++) {i;}" | bc > number.txt

$ bzip2 number.txt

$ bzmore number.txt.bz2

● 35 chattr

chattr - change file attributes on a Linux file system

To add attributes on files and immutable to secure from deletion
create file sample.txt
$ sudo chattr +i sample.txt

To list the file attributes on a Linux second extended file system
$ lsattr sample.txt
----i---------e----- sample.txt
Now change permission , rename , remove force will not be
permitted

To unset attribute on Files
$ sudo chattr -i sample.txt
$ lsattr sample.txt
--------------e----- sample.txt
Now its possible to rename , remove , change permissions of the
file sample.txt

To open the file only in append mode and the previous data cannot
be modified
create a text file example.txt
$ sudo chattr +a example.txt
$ lsattr example.txt
-----a---------e----- example.txt

$ echo "this is line two" > example.txt
bash: sample.txt: Operation not permitted
$ echo "this is line two" >> example.txt
$ cat example.txt

this is line one
this is line two

To secure entire directory important_folder and its files
```
$ sudo chattr -R +i important_folder
```

To unset it
```
$ sudo chattr -R -i important_folder
```

● <u>36 cancel</u>

cancel - cancel jobs

examples:

To cancel the current print job

```
$ cancel
```

To cancel all jobs

```
$ cancel -a
```

To cancel job printer-1

```
$ cancel printer-1
```

To cancel with printer name laser-100

```
$ cancel laser-100
```

To cancel all the print jobs that are queued for the user klug

```
$ cancel -u klug
```

● 37 cat

cat - concatenate files and print on the standard output

example:
To display contents of file
$ cat /etc/group

To view contents of multiple files
$ cat file3.txt file4.txt

To create a file with cat command
$ cat > file5.txt
this is file 5
^D

To view cat command with large file size
$ cat file.txt | more
$ cat file.txt | less

To display $ at the end of each Line using cat
$ cat -E file1.txt

To display line numbers in file
$ cat -n number.txt

To display multiple files
$ cat file1.txt; cat file2.txt; cat file3.txt

To redirect the standard output of a file into a new file
$ cat file1 > file2

To append in existing file

```
$ cat file1 >> file2
```

To redirect all output files to a new single file

```
$ cat file3.txt file4.txt file5.txt > file6.txt
```

● <u>38 cd</u>

cd - change directory

example:

change current directory to /usr/share

$ cd /usr/share/

switch back to previous directory

$ cd -

To change current directory to parent directory

$ cd ..

To show last working directory from where we work

$ cd --

To move two directory up from where we now

$ cd ../ ../

move to users home directory from anywhere

$ cd ~

pushd saves the current location to memory and changes to the requested directory

$ pushd /etc/perl/Net/

/etc/perl/Net ~

when popd command is entered, fetch the saved directory location from memory and makes it current working directory

$ popd

- ## **39 cfdsik**

cfdisk - display or manipulate a disk partition table

example:

$ sudo cfdisk

$ sudo cfdisk /dev/sda1

● <u>40 chacl</u>

chacl - change the access control list of a file or directory

example:

To change the ACL of a file

$ chacl u::rwx,g::r-x,o::r-- file

To set default acl for a directory

$ chacl -d u::rwx,g::r-x,o::r-- file_name

To remove the ACL

$ chacl -R file

To remove the directory default ACL

$ chacl -D /directory_name

To remove all ACL

$ chacl -B file

To list the ACL for a file/directory

$ chacl -l file/directory

To set the access ACL recursively

$ chacl -r u::r-x,g::r-x,o::r-- /directory

● 41 chage

chage - change user password expiry information

example:

To view the list of options

$ chage -h

To view the account aging information

$ chage -l user_name

To set the last password change date to your specified date

$ chage -d 2022-03-01 user_name

To set the date when the account should expire

$ chage -E 2022-06-30 user_name

To specify the maximum and minimum number of days between password change

$ chage -M 90 user_name

To give prior warning 7days before the password expires

$ chage -W 7 user_name

To make the user account to be locked after X number of inactivity days

```
$ chage -I 10 user_name
```

● <u>42 check-bios-nx</u>

check-bios-nx - determine if BIOS has blocked CPU's NX capabilities

NX stands for No eXecute is a technology used in processors to prevent the execution of certain types of code

This program attempts to determine if the running x86-based CPU has NX capabilities

If the CPU is NX-capable but the nx bit is missing from flags, exit 1 otherwise exit 0 (nothing wrong with BIOS)

$ sudo check-bios-nx --verbose

ok: the NX bit is operational on this CPU.

● **43 check-language-support**

check-language-support - returns the list of missing packages in order to provide a complete language environment

To show installed packages as well as missing ones

$ check-language-support --show-installed

To check all available languages

$ check-language-support -a

● <u>44 cheese</u>

cheese - tool to take pictures and videos from your webcam

To Start in fullscreen mode

$ cheese -f

Start in wide mode, with the thumbnails to the right of the video preview

$ cheese -w

To use the supplied DEVICE as the video capture device

$ cheese --device=DEVICE

● <u>45 cal</u>

cal - displays a calendar

example:

To Show current month calendar

$ cal

To Show calendar of selected month and year

$ cal August 2002

To Show the calendar of current year with the current date highlighted

$ cal -y

To Show the whole calendar of the year

$ cal 2010

To Show calendar of previous, current and next month

$ cal -3

● <u>46 chfn</u>

chfn - change real user name and information

$ chfn

Password:

Changing the user information for klug

Enter the new value, or press ENTER for the default

Full Name: klug

Room Number [123]: 456

Work Phone [9898]: 2323

Home Phone [9999]: 4545

To change the full name on the account

$ sudo chfn -f kanchilug klug

To change the work phone number on the account

$ sudo chfn -w 9999988888 klug

To change the room number on the account

$ sudo chfn -r 8888 klug

To change the home phone number on the account

$ sudo chfn -h 7777 klug

To change any other detail on the account

```
$ sudo chfn -o "7th floor room 55555" klug
```

● **47 chgrp**

chgrp - change group ownership

To change a directory group ownership

$ sudo chgrp ilugc example

To change group ownership of a file

$ sudo chgrp ilugc abc.txt

To recursively change group ownership

$ sudo chgrp -R ilugc example

To change the group of a file to match the group of another, reference file

To change the group ownership of the file abc.file to be the same as that of the test.file

$ sudo chgrp --reference=test.file abc.file

To list the changes that happened in our example directory

$ sudo chgrp -c -R ilugc example

To describe the action or non-action taken for every File

$ sudo chgrp -v ilugc file1

To change the group name of link files

```
$ sudo chgrp --dereference ilugc symbolic_link
```

To suppress potential error messages when executing the chgrp command

```
$ sudo chgrp -f [GROUP_NAME] [DIRECTORY/FILE_NAME]
```

```
$ sudo chgrp -f ilugc no_file
```

● 48 chmem

chmem - configure memory

The chmem command sets a particular size or range of memory online or offline

To request 1024 MiB of memory to be set online

$ sudo chmem --enable 1024

2 GiB of memory to be set online

$ sudo chmem -e 2g

This command requests the memory range starting with 0x00000000e4000000 and ending with 0x00000000f3ffffff to be set offline

$ sudo chmem --disable 0x00000000e4000000-0x00000000f3ffffff

The memory block number 10 to be set off-line

$ chmem -b -d 10

● <u>49 chmod (symbolic mode)</u>

chmod - change file mode bits

Symbolic Method
u - The file owner.
g - The users who are members of the group.
o - All other users.
a - All users, equal to ugo.

r - read
w - write
x - execute

- Removes the specified permissions.
+ Adds specified permissions.
= Changes the current permissions to the specified permissions

To set group permission to read the file
$ chmod g=r file_name

To set other users permission to read the file
$ chmod o=r file_name

To set user , group and others permission to read the file
$ chmod ugo=r file_name

To set no permission to execute for all users
$ chmod a-x file_name
or
$ chmod ugo-x file_name

To set user alone full permission and no permission to group and other users

```
$ chmod og-rwx filename
```

To set user , group and others full permissions

```
$ chmod a=rwx file_name
```

or

```
$ chmod ugo=rwx file_name
```

To set read, write and execute permission to the file's owner, read permissions to the file's group and no permissions to all other users

```
$ chmod u=rwx,g=r,o= file_name
```

To set file owners permission to group and others permissions

```
$ chmod g+u,o+u file_name
```

To set sticky bit to a given directory

```
$ chmod o+t dir_name
```

To set Recursively remove the write permission for other users and group

```
$ chmod -R o-w,g-w dir_name
```

● <u>50 chmod (numeric mode)</u>

chmod - change file mode bits

numeric method
r (read) = 4
w (write) = 2
x (execute) = 1
no permissions = 0

rwx=4+2+1=7
rw= 4+2=6
rx= 4+1=5

To set read , write , execute permission to users , group and others
$ chmod 777 file_name

To set read , write , execute permission to users and read permission only for group and others
$ chmod 744 file_name

To set users read, write and execute permissions, read and execute permissions to group members and no permissions to all other users
$ chmod 750 file_name

To recursively set read, write, and execute permissions to the file owner and no permissions for group and all other users on a given directory
$ chmod -R 700 dir_name

To set the file's permissions to be same for (file2_name) as those of the specified reference file (file1_name)

```
$ chmod --reference=file1_name file2_name
```

To set the permissions of all files and subdirectories under
the /var/www to 700
```
$ chmod -R 700 /var/www
```

To set read, write, and execute permissions, and a sticky bit to a
given directory
```
$ chmod 1777 dir_name
```

● <u>51 chown</u>

chown - change file owner and group

To change the owner of a file

`$ sudo chown klug file.txt`

To change the group of a file

`$ sudo chown :developers file.txt`

To change both owner and the group

`$ sudo chown klug:developers file.txt`

To change on symbolic link file

`$ sudo chown ilugc:devops symlnk_file`

To forcefully change the owner/group of symbolic file

`$ sudo chown -h ilugc:devops symlnk_file`

To change owner only if a file is owned by a particular user

`$ sudo chown --from=klug ilugc file_name`

To change group only if a file already belongs to a certain group

`$ sudo chown --from=:developers :devops file_name`

To copy the owner/group settings from one file to another

```
$ sudo chown --reference=fileX fileY
```

To change the owner/group of the files by traveling the directories recursively

```
$ sudo chown -R ilugc:devops dir_name/
```

To forcefully change the owner/group of a symbolic link directory recursively

```
$ sudo chown -R -H klug:developers symlnk_dir
```

To list all the changes made by the chown command

```
$ sudo chown -v -R ilugc:devops file_name
```

● **52 chpasswd**

chpasswd - update passwords in batch mode

$ sudo chpasswd

klug: p@ssword1

ilugc: p@ssword2

CTRL+D

storing username and password in a file and give input to chpasswd

$ cat > password.txt

klug: p@ssword1

ilugc: p@ssword2

then,

$sudo chpasswd < password.txt

or

$ sudo cat password.txt | chpasswd

To apply encryption algorithm on password

$sudo chpasswd -c SHA512

$ sudo chpasswd -c SHA256

$ sudo chpasswd --md5

● **53 chsh**

chsh - change login shell

To set login shell for user1

$ chsh -s /bin/bash user1

$ chsh

Password: *****

Changing the login shell for klug

Enter the new value, or press ENTER for the default

Login Shell [/bin/bash]: /bin/sh

To change current login shell from sh to bash

$ echo $SHELL

/bin/sh

$ chsh -s /bin/bash

$ echo $SHELL

/bin/bash

● **54 cksum**

cksum - checksum and count the bytes in a file

cksum command in Linux is used to display a cyclic redundancy check (CRC) value

CRC is unique for each file and only changes if the file is edited

$ cksum file.txt

2410262730 15 file.txt

after transfer of file.txt to other device or location

check with cksum

$ cksum file.txt

2410262730 15 file.txt

CRC value is same hence the file is not corrupted or edited

● **55 clear**

clear - clear the terminal screen

clear the terminal

$ clear

or

CTRL+l

or

$ reset

or

$ printf "\033c"

● 56 cmp

cmp - compare two files byte by byte

cmp command reports the byte and line number if a difference is found

$ cmp file1.txt file2.txt

To display the differing bytes in the output

$ cmp -b file1.txt file2.txt

To skip a particular number of initial bytes from both the files

$ cmp -i 100 file1.txt file2.txt

To input the number of bytes we want to skip

$ cmp -i 100:120 file1.txt file2.txt

To print byte position and byte value for all differing bytes

$ cmp -l file1.txt file2.txt

To limit the number of bytes we want to compare

$ cmp -n 500 file1.txt file2.txt

57 colrm

colrm - remove columns from a file

$ cat number.txt

123456789

$ colrm 4 6 < number.txt

123789

it will remove 4 5 and 6 column in the line

$ colrm 1 3 < number.txt

456789

it will remove 1 2 and 3 column in the line

● **58 column**

column - columnate lists

To display the information of the text file in form of columns

$ column employee.txt

To List File Content in Tabular Format

$ column -t employee.txt

To convert file rows into columns

$ column -x employee.txt

● **59 comm**

comm - compare two sorted files line by line

$ cat file1.txt

D1

D2

S1

S2

X1

$ cat file2.txt

D1

D2

S1

Z1

$ comm file1.txt file2.txt

To display first column

$ comm -23 file1.txt file2.txt

To display second column

$ comm -13 file1.txt file2.txt

To display third column

```
$ comm -12 file1.txt file2.txt
```

● <u>60 compgen</u>

compgen - is a bash built-in command which is used to list all
the commands that could be executed in the Linux system

To list all commands available to be directly executed.

$ compgen -c

To search for commands having a specific keyword

$ compgen -c | grep reminna

To count total number of commands available for use

$ compgen -c | wc -l

To list all the bash alias

$ compgen -a

To list all the bash built-ins

$ compgen -b

To list all the bash keywords

$ compgen -k

To list all the bash functions

$ compgen -A function

● <u>61 convert</u>

convert - convert between image formats as well as resize an image,blur, crop, despeckle, dither, draw on, flip, join, re-sample

```
$ convert picture.jpg picture.png

$ convert picture.png picture.jpg

$ convert picture.jpg -rotate 45 picture.png

$ convert picture.png -flip picture.png

$ convert picture.jpg -font courier -fill black -pointsize 50 -annotate +50+50 'ILUGC' picture.jpg

$ convert picture.jpg picture_flip.jpg -append appended.jpg
```

● **62 cpio**

cpio - copy files to and from archives

To create a *.cpio file

```
$ ls | cpio -ov > my_files.cpio
```

To extract a *.cpio file

```
$ cpio -iv < my_files.cpio
```

To create *.tar archive file using cpio

```
$ ls | cpio -ov -H tar > myfiles.tar
```

To extract *.tar archive file using cpio

```
$ cpio -iv -F myfiles.tar
```

To create a *.cpio archive with selected files

```
$ find . -iname "*.txt" | cpio -ov > myfiles.cpio
```

To create a *.tar archive with selected files

```
$ find . -iname "*.txt" | cpio -ov -H tar > myfiles.cpio
```

To only view *.tar archive file using cpio

```
$ cpio -it -F myfiles.tar
```

● <u>63 cp</u>

cp - copy files and directories

To copy file to a directory

```
$ cp file_name /opt/
```

To copying multiple files to a directory

```
$ cp file1_name file2_name file3_name /opt
```

To copying the files interactively

```
$ cp -i file_name /opt
```

To verbose output during copy command

```
$ cp -v file_name /opt
```

To copying a directory or folder

```
$ cp -r /home/klug /opt/backup
```

To archive files and directory during copy

```
$ cp -a /home/klug /opt/backup/
```

To copy only when source file is newer than the target file

```
$ cp -v -u  file_*.txt /opt/backup/
```

To create symbolic links using cp command

```
$ cp -s /home/klug/file1.txt /opt/backup/
```

To create hard link using cp command

```
$ cp -l /home/klug/file.txt /opt/backup/
```

To copy attributes from source to destination

```
$ cp --attributes-only /home/klug/file.txt  /opt/backup/
```

To preserve mode, ownership and timestamps when copying

```
$ cp -p file.txt /opt/backup/
```

To copy the files and directory forcefully

```
$ cp -f file.txt /opt/backup
```

● **64 cracklib-check**

cracklib-check - Check passwords using libcrack2

$ echo "abcdef123456" | cracklib-check

$ echo "password" | cracklib-check

$ echo "Wsd234$#@" | cracklib-check

or

$ cracklib-check<<<"Wsd234$#@"

● <u>65 crontab</u>

crontab - maintain crontab files for individual users

To list crontab entries

$ crontab -l

To edit the crontab entry

$ crontab -e

To list scheduled cron jobs

$ crontab -u ilugc -l

To remove scheduled jobs without confirmation

$ crontab -r

To prompt before deleting crontab

$ crontab -i -r

To schedule Jobs for Specific Time

$ crontab -e

00 09 * * * /home/ilugc/mycode.sh

To disable email notification.

$ crontab -e

* * * * * >/dev/null 2>&1

● <u>66 ctrlaltdel</u>

ctrlaltdel - set the function of the Ctrl-Alt-Del combination

$ sudo ctrlaltdel

soft

To set ctrlaltdel function to hard

$ sudo ctrlaltdel hard

hard

● **67 csplit**

csplit - split a file into sections determined by context lines

$ cat file.txt

one

two

three

four

five

six

To split file.txt into two parts (second part from 4 th line)

$ csplit file.txt 4

two files named xx00 and xx01 created

To prefix in abc in place of 'xx' in output

$ csplit -f abc file.txt 4

$ ls

$ cat abc00

$ cat abc01

● <u>68 curl</u>

curl - transfer a URL

```
$ curl https://www.ilugc.in
```

To display a progress meter during use to indicate the transfer rate, amount of data transferred, time left, etc

```
$ curl -# -O ftp://ftp.mysite.com/file.zip
```

To resumes download which has been stopped when downloading large files was interrupted

```
$ curl -C - -O ftp://mysite.int/10000MB.zip
```

To limit the upper bound of the rate of data transfer and keeps it around the given value in bytes.

```
$ curl --limit-rate 500K -O ftp://mysite.in/10000MB.zip
```

To download files from user authenticated FTP servers

```
$ curl -u username:P@ssword -O ftp://mysite.in/confidential.txt
```

To upload a file to the FTP server, use the -T followed by the name of the file to upload

```
$ curl -T file.zip -u username:P@ssword ftp://ftp.example.com/
```

● **69 cut**

cut - remove sections from each line of files

```
$ cat file.txt
```

Alpha is first line

Beta is second line

Charlie is third line

Delta is fourth line

To display 2nd character from each line of a file

```
$ cut -c2 file.txt
```

To extract first 3 characters of each line from file.txt

```
$ cut -c1-3 file.txt
```

To extract 7 characters from the beginning of each line in file.txt

```
$ cut -c-7 test.txt
```

To display only first field of each lines from a file using delimiter ":"

```
$ cut -d':' -f1 file_name
```

● **70 dumpe2fs**

dumpe2fs - dump ext2/ext3/ext4 filesystem information

To dump the file system information about a device

$ sudo dumpe2fs /dev/sda1

To display superblock information

$ sudo dumpe2fs -h /dev/sda1

To display Information of block groups

$ sudo dumpe2fs /dev/sda1

To view about superblocks

$ sudo dumpe2fs /dev/sda1 | grep -i superblock

● **71 du**

du - estimate file space usage

To check the disk usage summary of a directory

$ du /etc

$ du /home

To check disk usage in a human-readable format

$ du -h /etc

$ du -h /home/ilugc

To check the total usage size of a particular directory

$ du -sh /etc

To list the disk usage of all files in human readable format including directories

$ du -ah /home/ilugc

To print the grand total for a directory

$ du -ch /home/ilugc

To change the default block size output to Kilobytes, Megabytes or Gigabytes

$ du -BK /home/klug

$ du -BM /home/klug

```
$ du -BG /home/ilugc
```

To check the size of all the sub-directories in their current location

```
$ du -h --max-depth=1 /home/ilugc
```

or

```
$ du -h -d1 /home/ilugc
```

To exclude a particular type of file ex. python files while calculating the disk size

```
$ du -h --exclude="*.py" /home/ilugc/Documents
```

To check the disk usage of the last modification time

```
$ du -ha --time log
```

To show summary of size

```
$ du -s /home/ilugc/Documents
```

● <u>72 dpkg-reconfigure</u>

dpkg-reconfigure - reconfigure an already installed package

$ sudo dpkg-reconfigure -f package_name

$ sudo dpkg-reconfigure phpmyadmin

● <u>73 dpkg-query</u>

dpkg-query - a tool to query the dpkg database

Display package status details

$ dpkg-query -s apache2

List files 'owned' by package

$ dpkg-query -L apache2

List packages concisely

$ dpkg-query -l apache2

Show information on package

$ dpkg-query -W apache2

Find package owning file

$ dpkg-query -S apache2

● <u>74 dpkg</u>

dpkg - package manager for Debian

To install a package

```
$ sudo dpkg -i package_name.deb
```

To list all the installed packages

```
$ sudo dpkg -l
```

To remove a package

```
$ sudo dpkg -r flashpluginnonfree
```

To remove the package along with configuration file

```
$ sudo dpkg -p flashpluginnonfree
```

To view the content of a package

```
$ sudo dpkg -c package_name.deb
```

To check a package is installed or not

```
$ sudo dpkg -s package_name.deb
```

check the location of packages installed

```
$ sudo dpkg -L package_name.deb
```

To display dpkg licence

```
$ sudo dpkg --licence
```

● [75 do-release-upgrade](#)

do-release-upgrade - upgrade operating system to latest release

$ sudo apt update

$ sudo apt upgrade

$ do-release-upgrade

● <u>76 domainname</u>

domainname - show or set the system's NIS/YP domain name

To show alias names

$ domainame -a

To show all long host names (FQDNs)

$ domainname -A

To print DNS domain name

$ domainame -d

To print addresses for the host name

$ domainname -i

To show all addresses for the host

$ domainname -I

To show short host name

$ domainname -s

To show NIS/YP domain name

$ domainname -y

● **77 dmsetup**

dmsetup — low level logical volume management

To list the device mapper devices:

$ sudo dmsetup ls

To get information about any DM device

$ sudo dmsetup info /dev/VG01/LV01

To list the DM device dependencies

$ sudo dmsetup deps /dev/VG01/LV01

To get the status of a DM device

$ sudo dmsetup status /dev/VG01/LV01

To destroy the inactive table for a device

$ sudo dmsetup clear /dev/VG01/LV01

To remove all the devices

$ sudo dmsetup remove_all

To rename the device

$ dmsetup /dev/VG01/LV01 /dev/VG07/LV07

To output the table for a device

```
$ sudo dmsetup table /dev/VG01/LV01
```

● <u>78 dmidecode</u>

dmidecode - is a tool for dumping a computer's DMI (some say SMBIOS) table contents in a human-readable format

To get information about Processor

`$ sudo dmidecode -t processor`

To get hardware information

`$ sudo dmidecode`

To get BIOS information

`$ sudo dmidecode -t bios`

To print less verbose output

`$ sudo dmidecode -q`

To display the value of the given DMI string

`$ sudo dmidecode -s processor-frequency`

To get information about Baseboard

`$ sudo dmidecode -t baseboard`

To get information about Chassis

`$ sudo dmidecode -t 3`

To display the version

```
$ sudo dmidecode -V
```

To get DMI types

```
$ sudo dmidecode -t 6
```

To get the cache information

```
$ sudo dmidecode -t cache
```

To get memory Information

```
$ sudo dmidecode -t 16
```

To get the manufacturer, model and serial number

```
$ sudo dmidecode -t system
```

To Display Information of about Installed Physical Memory and DIMMs

```
$ sudo dmidecode -t 17
```

To find the maximum physical memory supported by your system

```
$ sudo dmidecode -t 16
```

● <u>79 dmesg</u>

dmesg - print or control the kernel ring buffer, it display message command and to display kernel-related messages

$ sudo dmesg | less

To read dmesg output in human readable format

$ sudo dmesg -H

To monitor real-time logs

$ sudo dmesg --follow

To print last or first 15 lines

$ sudo dmesg | head -15

$ sudo dmesg | tail -15

To search for a specific string or patterns

$ sudo dmesg | grep -i usb

To check for hard disk and will display the messages wherever sda is listed

$ sudo dmesg | grep -i sda

To list all the informational messages

$ sudo dmesg -l info

To display dmesg messages for eth0 user interface

```
$ sudo dmesg | grep -i eth0
```

● <u>80 dirname</u>

dirname - strip last component from file name

```
$ dirname /home/ilugc/myscript.sh

/home/ilugc


$ dirname -z /home/klug/autoscript.sh

/home/klug
```

● <u>81 dir</u>

dir - list directory contents

To display all the hidden files

$ dir -a

$ dir -A

To Displays author of all the files

$ dir -l --author

To list in single column

$ dir -1

To list with commas

$ dir -m

● **82 dig**

dig - DNS lookup utility

To perform a DNS lookup

$ dig ilugc.in

$ dig @8.8.8.8 google.com

To display only the IP address associated with the domain name

$ dig google.com +short

$ dig ilugc.in +short

The +trace option lists each different server the query goes through to its final destination

$ dig google.com +trace

To look up a domain name by its IP address

$ dig -x yy.zz.aa.bb

yy.zz.aa.bb ip address

Batch Mode for Reading Host Names From a File

store domain names in domain.txt and give input to dig command

$ dig -f domain.txt +short

● **83 date**

date - print or set the system date and time

date command displays the current date and time

```
$ date
```

To display the time in GMT/UTC time zone

```
$ date -u
```

To display the given date string in the format of date

```
$ date --date="1/04/2020"
$ date --date="April 2 2020"
```

To display past dates

```
$ date --date="3 year ago"
$ date --date="5 hours ago"
$date --date="1 month ago"
$ date --date="2 week ago"
$date --date="10 day ago"
```

To display future date

```
$date --date="next wed"
$ date --date="next month"
$date --date="2 day"
```

$date --date="1 year"

To set the system date and time

$date --set="Wed Apr 27 14:20:55 PDT 2022"

To display the date string present at each line of file in the date and time format

$ cat >> datefile

May 07 2022

Apr 03 2022

$ date --file=datefile

%D: Display date as mm/dd/yy.

%d: Display the day of the month (01 to 31).

%a: Displays the abbreviated name for weekdays (Sun to Sat).

%A: Displays full weekdays (Sunday to Saturday).

%h: Displays abbreviated month name (Jan to Dec).

%b: Displays abbreviated month name (Jan to Dec).

%B: Displays full month name(January to December).

%m: Displays the month of year (01 to 12).

%y: Displays last two digits of the year(00 to 99).

%Y: Display four-digit year.

%T: Display the time in 24 hour format as HH:MM:SS.

%H: Display the hour.

%M: Display the minute.

%S: Display the seconds.

```
$ date +%[format-option]

$ date "+%D"

$ date "+%D %T"

$ date "+%A %B %d %T %y"

$ date "+%Y/%m/%d"

$ date "+%Y-%m-%d"
```

● <u>84 dd</u>

dd - convert and copy a file

To backup the entire harddisk

```
$ dd if=/dev/sdc of=/dev/sdd
```

To create an image of a Hard Disk

```
$ dd if=/dev/hdb of=~/hdbdisk.img
```

To restore using the Hard Disk Image

```
$ dd if=hdcdisk.img of=/dev/hdd
```

To create a compressed disk image

```
$ dd if=/dev/sdb | gzip -c >/tmp/sdbdisk.img.gz
```

Backup a partition to another

```
$ dd if=/dev/sdb1 of=/dev/sdc1 bs=4096 conv=noerror,sync
```

To restore a disk or a partition image

```
$ dd if=/tmp/sdbdisk.img of=/dev/sdb
```

To restore compressed image

```
$ gzip -dc /tmp/sdcdisk.img.gz | dd of=/dev/sdc
```

To convert case of a file

```
$ cat file1
```

abcdefgh

```
$ dd if=~/file1 of=~/file2 conv=ucase
```

```
$ cat file2
```

ABCDEFGH

```
$ dd if=~/file2 of=~/file3 conv=lcase
```

● **85 delgroup**

delgroup - remove a user or group from the system

```
$ sudo delgroup group_name

$ sudo delgroup devops_group
```

● <u>86 delpart</u>

delpart - tell the kernel to forget about a partition

```
$ sudo umount /dev/sdb2

$ sudo delpart /dev/sdb 2
```

● **87 deluser**

deluser - remove a user or group from the system

To delete an user account

```
$ sudo deluser klug
```

To delete or account including deleting home directory

```
$ sudo deluser --remove-home klug
```

To delete account even while the user logged in

```
$ sudo deluser --force klug
```

To delete user account and backup home directory

```
$ sudo deluser --backup-to /backup_dir klug
```

● <u>88 df</u>

df - report file system disk space usage

To display all the file system

$ df -a

To display size in human readable format

$ df -h /home/klug

To get complete grand total

$ df -h --total

To display file type

$ df -T /home/ilugc

● 89 diff

GNU diff - compare files line by line

```
$ cat a.txt
Apple

Banana

Grapes

Mango

Papaya
```

```
$ cat b.txt
Apple

Banana

Grapes

Mango
```

The change character can be one of the following:

a - Add the lines.

c - Change the lines.

d - Delete the lines.

```
$ diff a.txt b.txt
```

To view differences in context mode

```
$ diff -c f1.txt f2.txt
```

To view differences in unified mode

```
$ diff -u f1.txt f2.txt
```


To ignores case

```
$ diff -i f1.txt f2.txt
```

● <u>90 diff3</u>

GNU diff3 - compare three files line by line

```
$ cat f1.txt
```
Hello

This is f1 file.

```
$ cat f2.txt
```
This is f2 file.

```
$ cat f3.txt
```
This is f3 file.

==== : It means all the files are different.

====1 : File 1 is different.

====2 : File 2 is different.

====3 : File 3 is different.

```
$ diff3 f1.txt f2.txt f3.txt
```

treat all files as text

```
$ diff3 -a f1.txt f2.txt f3.txt
```

● <u>91 e2fsck</u>

e2fsck - check a Linux ext2/ext3/ext4 file system

To check a partition

$ sudo e2fsck /dev/sdc1

To perform automatic repair using e2fsck

$ sudo e2fsck -p /dev/sdc1

or

$ sudo e2fsck -y /dev/sdc1

To check only using e2fsck

$ sudo e2fsck -n /dev/sdc1

To force the filesystem check

$ sudo e2fsck -f /dev/sdc1

T0 display a progress bar during e2fsck check

$ sudo e2fsck -f -C 0 /dev/sdc1

● **92 e2label**

e2label - Change the label on an ext2/ext3/ext4 filesystem

To display or change the filesystem label on the ext2, ext3, or ext4 filesystem located on device

```
$ sudo e2label /dev/device
$ sudo e2label /dev/device new-label-name-here
```

To view the label name of partition

```
$ sudo e2label /dev/sdb1
```

To set label name of partition

```
$ sudo e2label /dev/sdb1 mypartition
```

To remove a partition label name by supplying an empty string

```
$ sudo e2label /dev/sdb1 ""
```

● <u>93 e2mmpstatus</u>

e2mmpstatus - it is used to check Multiple-Mount Protection (MMP) status of an ext4 filesystem with the mmp feature enabled. The specified filesystem can be a device name or an ext4 filesystem label or UUID

```
$ sudo e2mmpstatus /dev/sda1
```

or

```
$ sudo e2mmpstatus LABEL=label_name
```

or

```
$ sudo e2mmpstatus UUID=cccccccccc-aaaaa-zzzzzzz-yyyyyy-xxxxxxx
```

● <u>94 e4defrag</u>

e4defrag - online defragmenter for ext4 filesystem

To defragment Linux partitions

$ sudo e4defrag <location>

or

$ sudo e4defrag <device>

$ sudo e4defrag /home/klug/directory

$ sudo e4defrag /dev/sdb2

To defragment your entire system

$ sudo -v e4defrag /

● <u>95 ebook-convert</u>

ebook-convert - tool to convert ebooks format

To convert .epub format to .docx

$ ebook-convert book.epub book.docx

To convert .docx to .epub

$ ebook-convert book.docx book.epub

To convert .epub .mobi

$ ebook-convert book.epub book.mobi

● <u>96 ebook-meta</u>

ebook-meta - ebook-meta process tool

$ ebook-meta ebook_file [options]

To display the meta data of book

$ ebook-meta my_book.pdf

To change the meta data of publish date

$ ebook-meta -d 2020-04-04T01:00:00+00:00 my_book.pdf

$ ebook-meta my_book.pdf

To change the meta data of author

$ ebook-meta -a ilugc linux_book.pdf

To set publisher in meta data

$ ebook-meta -p FTE linux_book.pdf

● <ins>97 ebook-polish</ins>

ebook-polish – ebook-polish Polishing tries to minimize the changes to the internal code of your e-book

```
$ ebook-polish [options] input_file [output_file]
```

To compress the images losslessly in ebook with quality

```
$ ebook-polish -i input book.epub new_book.epub
```

Upgrade the internal structures of the book

upgrades EPUB 2 books to EPUB 3 books

```
$ ebook-polish -U input book.epub new_book.epub
```

● **<u>98 echo</u>**

echo - display a line of text

$ echo [string]

$ echo "Welcome to Linux"

To enable the interpretation of backslash escapes -e option

\b To removes all the spaces in between the text

$ echo -e "Welcome \bto \bLinux"

WelcometoLinux

\c To suppress trailing new line with backspace interpretor '-e' to

continue without emitting new line.

$ echo -e "Welcome \cto Linux"

Welcome

\n To create new line from where it is used.

$ echo -e "Welcome \nto \nLinux"

Welcome

to

Linux

\t To create horizontal tab spaces

$ echo -e "Welcome \tto \tLinux"

Welcome   to   Linux

\r To carriage return with backspace interpretor '-e' to have

specified carriage return in output

$ echo -e "Welcome \rto Linux"

to Linux

\v To create vertical tab spaces

$ echo -e "Welcome \vto \vLinux"

Welcome

        to

            Linux

To print all files/folders

$ echo *

● <u>99 ed</u>

ed - line-oriented text editor

Type ed

$ ed

To get into insert mode press "a"

$ ed

a

this is line one

this is line two

this is line three

.

when you are done writing stop it by "." (dot)

To view the last line enter "p" into the ed command prompt.

p

To print all the lines that we inserted in the buffer by using
",p"

,p

To save these lines into a file write "f [filename]".

f myfile.txt

To write the data into the file and see how many bytes are written

w

To exit to the terminal by pressing "Q"

Q

To summarize all

$ ed

a

this is line one

this is line two

this is line three

.

p

this is line three

,p

this is line one

this is line two

this is line three

f myfile.txt

myfile.txt

w

53

Q

To check

```
$ cat myfile.txt
this is line one
this is line two
this is line three
```

● **100 egrep**

grep, egrep, fgrep, rgrep - print lines that match patterns

```
$ egrep [ options ] 'PATTERN' files
```

```
$ cat myfile.txt
```

this is line one

this is line two

this is line three

```
$ egrep this myfile.txt
```

this is line two

this is line three

To count and print the number of lines that matched the pattern and not the lines

```
$ egrep -c this myfile.txt
```

3

To Ignore the case of the pattern while matching

```
$ egrep -i this myfile.txt
```

This is line one

this is line two

this is line three

To Print only the names of the files that matched.

$ egrep -l this myfile.txt

myfile.txt

To Print only the names of the files that did not have the pattern
quite opposite to -l

$ egrep -L this myfile.txt myfile

myfile

To recursively search for the pattern in all the files of the
directory

$ egrep -r -i '.conf' .

. is current directory

To print each matched line along with the respective line numbers

$ egrep -n config myprogram.py

To print only the matched parts of the line and not the entire
line for each match

$ egrep -o config myprogram.py

To search for matches till the count reaches number mentioned as
argument

$ egrep -m 3 config myprogram.py

● **101 eject**

eject - eject removable media

To eject default cdrom drive
```
$ eject -v
```

To list default device name
```
$ eject -d
```

To display available help commands
```
$ eject -h
```

To give out more information about the command's execution
```
$ eject -v
```

To display the selected device, but perform no action
```
$ eject -n
```

● <u>102 env</u>

env - run a program in a modified environment

To print out a list of all environment variables
$ env

To run a command with an empty environment
$ env -i /bin/sh
$ env
PWD=/home/klug/test
$ exit

To remove variable from the environment
$ env -u variable_name

To end each output line with NULL
$ env -0

● <u>103 evince</u>

Evince - is a document viewer capable of displaying multiple and single
page document formats like PDF and Postscript

To Run evince in fullscreen mode.
$ evince -f your_book.pdf

To run evince in presentation mode.
$ evince -s your_book.pdf

To open the document on the page with the specified page index
$ evince -i 5 your_book.pdf

To run evince as a previewer
$ evince -w your_book.pdf

Opening a document at a specific page
$ evince --page-label=3 book.pdf

open multiple files
$ evince book1.pdf book2.pdf

To open files on the web
$ evince http://url_path/file.pdf

● 104 faillog

faillog - display faillog records or set login failure limits

To display the faillog records for all the users
$ sudo faillog -a

To lock an account klug for 2 minute / 120 seconds after failed login
$ sudo faillog -l 60 -u username

To set the maximum number of login failures
$ sudo faillog -m 5 username

To reset the counters of login failures
$ sudo faillog -r username

To display faillog records more recent than days
faillog -t 5 username
faillog --time DAYS username

To display faillog record or maintains failure counters and limits
$ sudo faillog -u username

● [105 factor](#)

factor - Print the prime factors of each specified integer NUMBER

```
$ factor number
```

```
$ factor 1000
1000: 2 2 2 5 5 5
```

```
$ factor 10000
10000: 2 2 2 2 5 5 5 5
```

```
$ factor 30
30: 2 3 5
```

```
$ factor 300
300: 2 2 3 5 5
```

```
$ factor 10000000
10000000: 2 2 2 2 2 2 2 5 5 5 5 5 5 5
```

```
$ factor 17
17: 17
```

```
$ factor 19
19: 19
```

● **106 fakeroot**

fakeroot - fakeroot runs a command in an environment wherein it appears to have root privileges for file manipulation. This is useful for allowing users to create archives (tar, ar, .deb etc.) with files in them with root permissions/ownership

```
$ fakeroot

# echo "fake root access given" > root.txt

# ls -l root.txt

# ls -l /root

# exit

$ ls -l root.txt
```

● <u>107 fallocate</u>

fallocate - preallocate or deallocate space to a file

To allocate a file with a size of 2GB and 10 GB

```
$ fallocate -l 2G file1.img
```

```
$ fallocate -l 10G file2.img
```

check with

```
$ ls -lh *.img
```

● **108 fc**

fc - shell built-in command used to list, edit and re-execute the most recently entered commands

To display the last 16 commands
```
$ fc -l
```

To reverse the order of the commands
```
$ fc -r
```

To suppress the line numbers using
```
$ fc -ln
```

To list the result starting from a specific command
```
$ fc -l 2060
```

To list a commands within a specific range
```
$ fc -l 2055 2060
```

To list the commands starting from fallocate command up to the latest command
```
$ fc -l f
```

To list between falloacate to ls command (in this case)
```
$ fc -l f l
```

To edit the last command and re-run it again
```
$ fc
```

To change the default editor to edit commands
```
$ fc -e vim
```

To set "vim" as the new default editor, edit your ~/.profile

FCEDIT=vim

save and exit

```
$ source ~/.profile
```

● <u>109 fc-list</u>

fc-list command is a part of the fontconfig system. It is used to list the available fonts and font styles

To print all the file locations of the font files present in the system
$ fc-list

To print only the names of the font families
$ fc-list : family

To print only the names of the font families which support the tamil language code
$ fc-list : family lang=ta

● **110 fdisk**

fdisk is a dialog-driven program for creation and manipulation of partition tables. It understands GPT, MBR, Sun, SGI and BSD partition tables

To view all disk partitions in linux
```
$ sudo fdisk -l
```

To view specific disk partition in linux
```
$ sudo fdisk -l /dev/sda
```

To view all available fdisk commands
```
$ sudo fdisk /dev/sda
```

To print all partition table in linux
```
$ sudo fdisk /dev/sda
Command (m for help): p
```

To check Size of a partition in linux
```
$ sudo fdisk -s /dev/sda3
```

● <u>111 fgrep</u>

grep, egrep, fgrep, rgrep - print lines that match patterns

To display the count of number of matches

```
$ fgrep -c "config_value" file.txt
```

2

To display the matched lines

```
$ fgrep -h "config.py" file.txt
```

To display case insensitive search

```
$ fgrep -i "Linux" file.txt
```

To display the file names that match the pattern

```
$ fgrep -l "config.py" file1 file2
```

To show line number of file with the line matched

```
$ fgrep -n "ubuntu" file.txt
```

To display only lines matched entirely

```
$ fgrep -x "Keyword_exact_match" file.txt
```

● 112 fg

fg command in linux used to put a background job in foreground.

$ ping ilugc.in

^Z

$ jobs -l

[1]+ 25365 Stopped   ping ilugc.in

$ fg %1

● [113 file](#)

file - determine file type

$ file [option] [filename]

To display just file type in brief mode

$ file -b filename.py

$ file -b file.img

$ file -b file.txt

$ file -b file.pdf

To display all files's file type

$ file *

To display all files filetypes in particular directory

$ file /my_home_dir/*

To display the file type of files in specific range

$ file [a-d]*

$ file [e-h]*

To view mime type of file

$ file -i filename.txt

To view file type inside compressed files

```
$ file -z file.bz2
```

● <ins>114 find</ins>

find - search for files in a directory hierarchy

To find all the files whose name is ilugc.txt in current working directory

$ find . -name ilugc.txt

To find files in home directory

$ find /home -name ilugc.txt

To find all directories whose name is klug in / directory

$ find / -type d -name klug

To find all python files whose name is myprogram.py in current working directory

$ find . -type f -name myprogram.py

To find all python files in a directory

$ find . -type f -name "*.py"

To find all the files with permission 777

$ find . -type f -perm 0777 -print

To find all empty files

$ find /home -type d -empty

● <u>115 finger</u>

finger - displays the user's login name, real name, terminal name and write status , idle time, login time, office location and office phone number

$ finger user_name

$ finger klug

To get idle status and login details of a user

$ finger -s ilugc

To avoid printing PGP key, plan and project details

$ finger -p klug

● <u>116 findfs</u>

findfs - will search the block devices in the system
looking for a filesystem or partition with specified tag

$ findfs LABEL=<label>

$ findfs LABEL=klug

$ findfs UUID=<uuid>

$ findfs UUID=cbbf8b34-7bf1-4dba-9eb0-59e85ade7083

/dev/sda5

$ findfs PARTUUID=<uuid>

$ findfs PARTLABEL=<label>

● 117 findmnt

findmnt - it will list all mounted filesystems or search for a filesystem

To display a list of currently mounted file systems

$ sudo findmnt

To display the information as an ordinary list

$ sudo findmnt -l

To display only file systems of a specific type using the -t

$ sudo findmnt --fstab -t ext4

To print all /etc/fstab filesystems and convert LABEL= and UUID= tags to the real device name

$ sudo findmnt --fstab --evaluate

To monitor mount, unmount, remount and move actions on a directory

$ sudo findmnt --poll --mountpoint /mnt/my_dir

● [118 firefox](#)

firefox - a free and open source web browser from Mozilla

To find out full path to firefox

$ type -a firefox

$ firefox

or

$ /usr/bin/firefox

To open URL in a new window

$ /usr/bin/firefox --new-window https://ilugc.in/

To open Firefox options/preference

$ /usr/bin/firefox --preferences

To set the Firefox app as the default browser

$ /usr/bin/firefox --setDefaultBrowser

● <u>119 fmt</u>

fmt - simple optimal text formatter Reformat each paragraph in the files, writing to standard output

$ cat file.txt

Hai

all Welcome

to

ILUGC

$ fmt file.txt

Hai all Welcome to ILUGC

To split long lines, but don't refill them

$ fmt -s file.txt

To make one space between words and two spaces after sentences for formatting

$ fmt -u file.txt

● **120 fold**

fold - wrap each input line to fit in specified width

$ fold [OPTION] [FILE]

$ fold testfile.txt

To limit the width by number of columns

$ fold -w[n] testfile.txt

$ fold -w40 testfile.txt

To limit the width of the output by the number of bytes

$ fold -b[n] testfile.txt

$ fold -b30 testfile.txt

To break the lines on spaces so that words are not broken

$ fold -w[n] -s testfile.txt

$ fold -w30 -s testfile.txt

● <u>121 for</u>

for - command in Linux is used to repeatedly execute a set of command for every element present in the list.

```
$ for i in 0 1 2 3 4 5 6 7 8 9; do echo $i; done
0
1
2
3
4
5
6
7
8
9
```

```
$ for i in 1 2 3 4 5; do echo "welcome $i times"; done
welcome 1 times
welcome 2 times
welcome 3 times
welcome 4 times
welcome 5 times
```

● <u>122 free</u>

free - Display amount of free and used memory in the system

$ free

To display free and used memory in bytes

$free -b

To display free and used memory in kilobytes

$free -k

To display free and used memory in megabytes

$free -m

To display free and used memory in gigabyte

$free -g

To display an additional line containing the total of the total, used and free columns

$ free -t

To display the output of free command after a set time gap

$ free -s 4 -c 4

● **123 fsck**

fsck - check and repair a Linux filesystem it is used to check and optionally repair one or more Linux filesystems

fsck [OPTIONS] [FILESYSTEM]

Unmount the device first

$ sudo umount /dev/sdb1

$ sudo fsck -p /dev/sdb1

when file system is repaired, mount the partition

$ sudo mount /dev/sdb1

To do a dry run with fsck

$ sudo fsck -N /dev/sdb1

To fix detected errors automatically with fsck

$ sudo fsck -y /dev/sdb1

To skip repair but print fsck errors in the output

$ sudo fsck -n /dev/sdb1

To force fsck to do a filesystem check

$ sudo fsck -f /dev/sdb1

To run fsck on all filesystems at once

$ fsck -AR

● 124 fsck.ext4

e2fsck - it is used to check Linux ext2/ext3/ext4 file system

```
$ sudo fsck [OPTIONS] [FILESYSTEM]
```

Unmount the device

```
$ sudo umount /dev/sdaX
```

To repair the file system

```
$ sudo fsck -p /dev/sdaX
```

when the file system is repaired, mount the partition

```
$ sudo mount /dev/sdaX
```

To do a dry run with fsck

```
$ sudo fsck -N /dev/sda
```

To fix potential problems without getting any prompts

```
$ sudo fsck -y /dev/sda
```

To skip repair but print fsck errors in the output

```
$ sudo fsck -n /dev/sdc
```

To force fsck to do a filesystem check

```
$ sudo fsck -f /dev/sdc
```

● 124 fsck.ext4

To run fsck on all filesystems at once

```
$ sudo fsck -AR
```

To skip fsck on a specific filesystem

```
$ sudo fsck -AR -t noext2 -y
```

To skip fsck on mounted filesystems

```
$ sudo fsck -M /dev/sdc
```

● <u>125 ftp</u>

ftp - internet file transfer program. ftp is the user interface to
the Internet standard File Transfer Protocol

To open an ftp connection to a remote system

$ ftp xx.xx.xx.xx

xx.xx.xx.xx is the remote server ip

to change to another directory

ftp > lcd dir_name

To download a single file from the remote server

ftp > get file_name

To download multiple files at once

ftp > mget file1 file2 file3

To upload a file

ftp > put file_name

To upload multiple files

ftp > put file1 file2 file3

To close the connection

```
ftp > quit
```

or

```
ftp > bye
```

● **126 funzip**

funzip - filter for extracting from a ZIP archive in a pipe

To extract the first member file of the archive myfile.zip and to pipe it into more

```
$ funzip myfile.zip | more
```

To test the first member file of myfile.zip

```
$ funzip myfile.zip > /dev/null
```

● **127 fuser**

fuser - identify processes using files or sockets fuser displays the PIDs of processes using the specified files or filesystems.

To find process accessing a directory

```
$ fuser .
```

or

```
$ fuser /home/ilugc
```

To view more details enable verbose

```
$ fuser -v .
```

or

```
$ fuser -v /home/ilugc
```

To find process accessing file system

```
$ fsuer -v -m /etc/profile
```

To kill a processes accessing a file or socket

```
$ sudo fuser -k .
```

To interactively kill a process

```
$ sudo fuser -ki .
```

To list all the signals

```
$ sudo fuser --list-signals
```

● 128 getent

getent - The getent command displays entries from databases supported by the Name Service Switch libraries

To Fetch the list of user accounts on a Linux system

```
$ getent passwd
```

To fetch details for a particular user

```
$ getent passwd user_name
```

To fetch a list of group accounts

```
$ getent group
```

To find the service name and its protocol

```
$ getent services 20
```
```
ftp-data              20/tcp
```

```
$ getent services 53
```
```
domain                53/tcp
```

```
$ getent services 22
```
```
ssh                   22/tcp
```

```
$ getent services 3306
```
```
mysql                 3306/tcp
```

● 129 getfacl

getfacl - getfacl displays the file name, owner, the group, and the Access Control List (ACL)

example:

To get the ACL's of a file

$ getfacl file_name

To display the file access control list

$ getfacl -a file.txt

To display the default access control list

$ getfacl -d file.txt

To avoid displaying comment header

$ getfacl --omit-header file.txt

To Print all effective rights comments

$ getfacl -e file.txt

To skip files that only have the base ACL entries

$ getfacl -s file.txt

To list the ACL's recursively

$ getfacl -R /dir_name

To get the tabular output format

```
$ getfacl -t /home/ilugc/file.txt
```

To list the numeric user and group IDs

```
$ getfacl -n file.txt
```

● **130 gpasswd**

gpasswd - administer /etc/group and /etc/gshadow

To add user user1 to the group ilugc

$ sudo gpasswd -a user1 ilugc

To give user user1 administrative rights to the group ilugc

$ sudo gpasswd -A user1 ilugc

To remove user user1 from the group ilugc

$ sudo gpasswd -d user1 ilugc

● **131 groupadd**

groupadd - create a new group

To create a group ilugc

$ sudo groupadd ilugc

To create a group ilugc with specific groupid

$ sudo groupadd ilugc -g 1234

To create a system group

$ sudo groupadd -r 499 admin

To create a new group ilugc with group ID from 5000 to 7000

$ sudo groupadd ilugc -K GID_MIN=5000 -K GID_MAX=7000

To use an encrypted password for the group

$ sudo groupadd ilugc -p pa55code123!@#

- ## 132 groupdel

groupdel - delete a group

$ sudo groupdel GROUP_NAME

$ sudo groupdel webadmin

● <u>133 groupmod</u>

groupmod - The groupmod command modifies the definition of the specified GROUP by modifying the appropriate entry in the group database.

To change the group "ilugc" to "klug"

$ sudo groupmod -n klug ilugc

To change groupid of a group

$ sudo groupmod -g 1234 ilugc

To change the group ID with non-unique

$ sudo groupmod -o 0 ilugc

To change the group password

$ sudo groupmod -p pa55@123 ilugc

● <u>134 gpg</u>

gpg - gpg is the OpenPGP part of the GNU Privacy Guard (GnuPG).
It is a tool to provide digital encryption and signing services
using the OpenPGP standard

To check gpg version

```
$ gpg --version
```

To generate a new Key pair Using gpg command

```
$ gpg --gen-key
```

To list all the public keys using gpg command

```
$ gpg --list-keys
```

To export a public key

```
$ gpg --export ilugc > ilugc-pub.gpg
```

To get the key ID from a public key file

```
$ gpg --show-keys ilugc-pub.gpg
```

To simulate import of a public key

```
$ gpg --dry-run --import ilugc-pub.gpg
```

To delete private key of a public key

```
$ gpg --delete-secret-keys xxxxxyyyyyyyzzzzz53453553
```

To delete a public key

```
$ gpg --delete-key xxxxxyyyyyyyyzzzzz53453553
```

To import a public key

```
$  gpg --import ilugc-pub.gpg
```

To encrypt a file with password

```
$ gpg -c helloworld.py
```

To decrypt a file using gpg command

```
$ gpg -d helloworld.py.gpg
```

To use a user's public key to encrypt a file

```
$ gpg --recipient ilugc --encrypt hello.txt
```

To check all the options available with gpg command

```
$ gpg --dump-options
```

● <u>135 gpg-zip</u>

gpg-zip - encrypts or signs files into an archive. It is a gpg-ized tar using the same format as PGP's PGP Zip.

To encrypt the contents of directory dirX for user ilugc to file fileY

```
$ gpg-zip --encrypt --output fileY --gpg-args  -r ilugc dirX
```

To list the contents of archive fileY

```
$ gpg-zip --list-archive fileY
```

● <u>136 gzip</u>

gzip - compress or expand files reduces the size of the named files using Lempel-Ziv coding

To compress a single file

$ gzip file_name.txt

To compress multiple files at once

$ gzip a.txt b.txt c.txt

To compress a single file and keep the original

$ gzip -c a.txt > a.txt.gz

To compress all files recursively

$ gzip -r *

To decompress a gzip compressed file

$ gzip -d file_name.txt.gz

To decompress a file and keep the original .gz file

$ gunzip -c file.txt.gz > file.txt

To list compression information

$ gzip -l file.tar.gz

To adjust compression level

level of compression range from 1 to 9

using option 1 will complete faster but space saving is less

using option 9 will complete slow but space saving is high

default gzip uses a compression level of -6

$ time gzip -1 file.tar

$ gzip -l file.tar.gz

$ time gzip -9 file.tar

$ gzip -l file.tar.gz

To check the integrity of a compressed file

$ gzip -tv file.txt.gz

To view the CRC value

$ gzip -lv file.txt.gz

To concatenate multiple files

$ gzip -c a.txt > c.gz

$ gzip -c b.txt >> c.gz

To specify our own suffix instead of .gz

$ gzip -S .cz file

To display the gzip license info

```
$ gzip -L
```

To suppress all warnings

```
$  gzip -q file.txt
```

To save the original file name and time stamp

```
$ gzip -N file.txt
```

● <u>137 groupmems</u>

groupmems - administer members of a user's primary group

user : ilugc

group : foss

To make the user ilugc a member of the group foss

```
$ sudo groupmems -g foss -a ilugc
```


To add a user to a group

```
$ sudo groupmems -a ilugc -g foss
```


To delete/remove a user from a group

```
$ sudo groupmems -d ilugc foss -g foss
```


To change the group name

```
$ sudo groupmems -g linux
```


To remove the users from group

```
$ sudo groupmems -p -g ilugc
```

or

```
$ sudo groupmems --purge -g ilugc
```


To list the members of the group

```
$ sudo groupmems -l -g foss
```

● 138 grep

grep - print lines that match patterns

```
$ cat grep_example.txt
This is line number one
this is line number two
THIS is line number three
this is line 4
This is line 5
```

To search for the given string in a single file

```
$ grep "this" grep_example.txt
this is line number two
this is line 4
```

To check for the given string in multiple files

```
$ grep "this" grep_example.txt file2.txt
```

To search case insensitive using grep -i

```
$ grep -i "4" grep_example.txt
this is line 4
```

To check for full words using grep -w

$ grep -iw "is" grep_example.txt

This is line number one

this is line number two

THIS is line number three

this is line 4

This is line 5

To search in all files recursively using grep -r

$ grep -r "key_word" *

To count the number of matches using grep -c

$ grep -c this grep_example.txt

2

To find out how many lines that does not match the pattern

$ grep -v -c this grep_example.txt

3

To show line number while displaying the output using grep -n

$ grep -n "this" grep_example.txt

2:this is line number two

4:this is line 4

To display the number of MP3 files , .txt files present in a directory

```
$ ls ~/Music | grep -c .mp3
```

```
$ ls /home/ilugc | grep -c .txt
```

● <u>139 groups</u>

groups - print the groups a user is in

$ groups [username]

Provided with a username ilugc

$ groups ilugc

ilugc : ilugc adm cdrom sudo dip plugdev lpadmin lxd sambashare
libvirt docker

To display group membership for the current user

$ groups

To find groups of root

groups

root

● <u>140 gcc</u>

gcc - GNU project C and C++ compiler

To compile a C code without options

$ gcc hello.c

To specify explicitly mention the output file name

$ gcc hello.c -o output

To see the warnings when compile C program

$ gcc -wall hello.c -o output

To get preprocessed output

$ gcc -E hello.c > output.i

To get intermediate files using

$ gcc -save-temps hello.c

To see the error while compiling the C Program

$ gcc hello.c -Werror -o output

To debug C Program in Linux during compilation

$ gcc -ggdb hello.c -wall -o output

● <u>141 gawk</u>

gawk - used for pattern scanning and processing language

```
$ cat staff.txt
```

arun 0001

babu 0002

chandru 0003

dhana 0004

kiran 0005

raj 0006

sunil 0007

teja 0008

To print current count of the number of input line

```
$ gawk '{print NR "-" $1 }' staff.txt
```

gawk prints every line of data from the input line

```
$ gawk '{print}' staff.txt
```

To print the lines matching with the given pattern

```
$ gawk '/babu/ {print}' staff.txt
```

To print the second column records of the input file

```
$ gawk '{print $2}' staff.txt
```

To display count of lines

```
$ gawk '{print NR, $0}' staff.txt
```

To find the length of the longest line present in the file

```
$ gawk '{ if (length($0) > max) max = length($0) } END { print max }' staff.txt
```

To count the lines in a file

```
$ gawk 'END { print NR }' staff.txt
```

To print lines with more than 11 characters

```
$ gawk 'length($0) > 11' staff.txt
```

● **142 gunzip**

gunzip - tool for decompressing gzip files.

To decompress a .gz file

$ gunzip file.gz

gunzip will remove the compressed file , to keep the original file

$ gunzip -k file.gz

To keep the compressed file and decompress it to another location

$ gunzip -c file.gz > /path/to/file

$ gunzip -c mydoc.gz > /home/ilugc/mydoc

To decompress multiple files

$ gunzip file1.gz file2.gz file3.gz

To recursively decompresses all files in a given directory

$ gunzip -r directory

To list the compressed file contents

$ gunzip -lv file.gz

● **143 halt**

halt - used to instruct the hardware to stop all the CPU functions

To cease all CPU function on the system

```
$ sudo halt
```

To power off the system using halt command

```
$ sudo halt -p
```

To halt with -w option to write shutdown record

```
$ sudo halt -w
```

To reboot the system

```
$ sudo halt --reboot
```

● <u>144 history</u>

history - it is a built-in shell tool that displays a list of commands used in the terminal session

To display the list of commands used since the start of the terminal session

$ history

To show only the latest 10 entries from the list of commands used since the start of the

terminal session

$ history 10

To run the 100 th command again in history

$!100

To repeat the last command

$!!

To run the command count starts the from the end of the list

for example to run the 3rd command in history count starts from the end

$!-3

To search a command by string

$!sudo

To display the command without running it

$!sudo:p

To search for a command that contains a string but not start with the string

$!?firewall-cmd

sudo firewall-cmd --get-services

To use history along with grep

$ history | grep chown

To remove a command from history

$ history -d event_number

$ history -d 100

To remove whole history

$ history -c

To view the last 10 commands

$ history | tail

● **145 hash**

hash - built-in command of bash which is used to maintain a hash table of recently executed programs

To display information about the hash table

$ hash

hits command

2   /usr/bin/man

2   /usr/bin/ls

1   /usr/bin/cat

To forget the remembered location of each name

$ hash -r

To display in a format that may be reused as input

$ hash -l

To display the remembered location of each NAME

$ hash -t cat ls

cat /usr/bin/cat

ls /usr/bin/ls

● **146 hd**

hd - hd or hexdump is used to filter and display the specified files, or standard input in a human readable specified format

$ cat dummy.txt

this is ubuntu linux

this is centos linux

this is arch linux

one-byte octal display

$ hexdump -b dummy.txt

one-byte character display

$ hexdump -c dummy.txt

canonical hex + ASCII display

$ hexdump -C dummy.txt

Two-byte decimal display

$ hexdump -d dummy.txt

Two-byte octal display

$ hexdump -o dummy.txt

Two-byte hexadecimal display

```
$ hexdump -x dummy.txt
```

Hexdump had the option of deciding a specific number of bytes from a file to hexdump

```
$ hexdump -s 2 -c dummy.txt
```

hexdump to display all input data

```
$ hexdump -v -b dummy.txt
```

● **147 head**

head - output the first part of files

To display the first 10 lines default of head command

$ head file.txt

To show the first 6 lines of file.txt

$ head -n 6 file.txt

To displaying specific number of bytes ex.10 bytes

$ head -c 10 example1.txt

To displaying the file name tag

$ head -v file.txt

To display multiple files

$ head file1.txt file2.txt

To display the first 5 lines of each file

$ head -n 5 file1.txt file2.txt

To redirect output to a text file

$ head file.txt > output.txt

To display head with Pipeline

```
$ ls /etc | head
```

```
$ ls -t | head -n 4 | sort
```

● <u>148 hdparm</u>

hdparm - is used to handle disk devices and hard disks. it get statistics about the hard disk, alter writing intervals, acoustic management, and DMA settings

To display information of the hard drive

```
$ sudo hdparm -I /dev/sda
```

To display all the options

```
$ sudo hdparm -h
```

To test hard disk drive speed

```
$ sudo hdparm -t /dev/sdb
```

To measure hard disk cache read speed

```
$ sudo hdparm -T /dev/sdb
```

To get current settings

```
$ sudo hdparm -d /dev/sdb
```

To set DMA on for a device

```
$ sudo hdparm -d1 /dev/sdb
```

To print all settings

```
$ sudo hdparm -v /dev/sda
```

● <u>149 help</u>

help - displays the information about the built-in commands
present in the Linux shell

To display information about help command

$ help help

$ help cd

To display short description about commands

$ help -d help

$ help -d ls

$ help -d cd

To display usage in pseudo-manpage format

$ help -m help

$ help -m pwd

To display short usage synopsis for each topic matching PATTERN

$ help -s pwd

$ help -s cd

● **150 host**

host - DNS lookup utility used for performing DNS lookups. It is normally used to convert names to IP addresses and vice versa

To print the IP address details of the specified domain

$ host ilugc.in

To display the domain details of the specified IP Address

$ host 54.255.56.197

To specify the query type or enables the verbose output

$ host -a ilugc.in

To specify the type of query

$ host -t ns ilugc.in

To print SOA record

$ host -t SOA ilugc.in

To print txt record

$ host -t txt ilugc.in

To compare the SOA records on authoritative nameservers

$ host -t SOA ilugc.in

To specify the number of retries you can do in case one try fails

```
$ host -R 3 ilugc.in
```

- ## **151 hostid**

hostid - is used to display the host id in hexadecimal format.

$ hostid

7c787dcd

● **152 hostnamectl**

hostnamectl - control the system hostname , also used to query and change the system hostname and related settings

To check the current host name

$ hostnamectl

To change static host name to ilugc

$ hostnamectl set-hostname ilugc --static

To set transient name to klug

$ hostnamectl set-hostname klug --transient

check with

$ hostnamectl

To set pretty hostname to foss

$ hostnamectl set-hostname "foss" --pretty

To verify the change

$ hostnamectl --pretty status

To change the host names remotely

$ hostnamectl set-hostname ilugc-server -H root@server_ip

To display the help

$ hostnamectl --help

● <u>153 hostname</u>

hostname - display the system's DNS name, and to display or set its hostname or NIS domain name

To display the system hostname

$ hostname

To get alias name of the host system

$ hostname -a

To get all Fully Qualified Domain Name of the host system

$ hostname -A

To always set a hostname, default name is used nothing is specified

$ hostname -b

To get the domain name if local domains are set. It will not return anything

if no local domain is set.

$ hostname -d

To get the FQDN , It contains short hostname and DNS domain name

$ hostname -f

To set the hostname specified in a file

```
$ sudo hostname -F filename
```

To get the IP addresses , works only if hostname is resolvable

```
$ hostname -i
```

To get all IP addresses

```
$ hostname -I
```

To get the hostname in short

```
$ hostname -s
```

To set the hostname

```
$ sudo hostname new_hostname
```

To display the NIS domain name

```
$ hostname -y
```

● <u>154 hwclock</u>

hwclock - administration tool for the time clocks

To display the hardware clock date and time

$ sudo hwclock

To set the hardware clock same as system clock

$ sudo hwclock --systohc

or

$ sudo hwclock -w

To set hardware clock date manually

$ sudo hwclock --set --date 9/2/2022

To set hardware clock time manually

$ sudo hwclock --set --date "9/2/2022 13:00:00"

To get the output of the date command, and pass it to the --set and --date option

$ sudo hwclock --set --date "Friday Sep 02 13:00:00 PDT 2022"

 To copy the hardware time to system time

$ sudo hwclock -hctosys

To run hwclock test mode

```
$ sudo hwclock --systz --test
```

● 155 hwe-support-status

hwe-support-status - Check HWE support status

$ hwe-support-status

Your Hardware Enablement Stack (HWE) is supported until April 2025.

To show help message

$ hwe-support-status -h

● <u>156 id</u>

id - print real and effective user and group IDs

To print your own id without any options

```
$ id
```

To find a specific users id

```
$ id -u ilugc
```

To find a specific users GID

```
$ id -g ilugc
```

To find out UID and all groups associated with a username

```
$ id ilugc
```

To find out all the groups a user belongs

```
$ id -G ilugc
```

To display a name instead of numbers

```
$ id -nG ilugc
```

To display real id instead of effective id

```
$ id -r -g ilugc
$ id -r -G ilugc
```

● <u>157 ifconfig</u>

Ifconfig - used to configure the kernel-resident network interfaces

To display all the interfaces available

$ sudo ifconfig -a

To display a short list

$ sudo ifconfig -s

To run in verbose mode

$ sudo ifconfig -v

To activate the driver for the given interface

$ sudo ifconfig eth0 up

or

$ sudo ifup eth0

To deactivate the driver for the given interface

$ sudo ifconfig eth0 down

or

$ sudo ifdown eth0

To view network settings of Specific Interface

$ sudo ifconfig eth0

To assign an IP address to network interface

```
$ sudo ifconfig eth0 xx.xx.xx.xx
```

To assign netmask to network interface

```
$ sudo ifconfig eth0 netmask 255.255.255.224
```

To assign a broadcast to network interface

```
$ sudo ifconfig eth0 broadcast xx.xx.xx.xx
```

To assign IP, netmask, and broadcast to network interface

```
$ sudo ifconfig eth0 xx.xx.xx.xx netmask 255.255.255.224 broadcast xx.xx.xx.xx
```

To enable promiscuous mode

```
$ sudo ifconfig eth0 promisc
```

To disable promiscuous mode

```
$ sudo ifconfig eth0 -promisc
```

To add new alias to network interface

```
$ sudo ifconfig eth0:0 xx.xx.xx.xx
```

To remove alias to network interface

```
$ sudo ifconfig eth0:0 down
```

To change the MAC address of network interface

```
$ sudo ifconfig eth0 hw ether DD:DF:CW:DQ:EZ:FS
```

● **158 import**

import - used for capturing a screenshot for any of the active
pages we have and it gives the output as an image file

To join images into a single multi-image file

$ import -adjoin image.png

To include window border in the output image

$ import -border image1.png

To obtain image by descending window hierarchy

$ import -descend image2.png

To include window manager frame

$ import -frame image3.png

To identify the format and characteristics of the image

$ import -identify image.png

To suppress all warning messages

$ import -quiet image4.img

To monitor the progress

$ import -monitor image5.png

● <u>159 info</u>

info - read Info documents

To use all matching manuals and display them for a particular command

$ info -a ls

To look up STRING in all indices of all manuals and then display the same

$ info -k cat

To display DIR to INFOPATH

$ info -d ls

To go to command-line options node

$ info -O du

To print physical location of Info file

$ info -w df

To print help message

$ info --help

● <u>160 init</u>

init - to create processes from script stored in the file
/etc/inittab which is a configuration file which is to be used by
initialization system.

To restart the system

$ init 6

To shut down system

$ init 0

● **<u>161 insmod</u>**

insmod - Simple program to insert a module into the Linux Kernel

To insert the LinuxKernelModule file (.ko) into the Linux Kernel

$ sudo insmod sample.ko

to check

$ dmesg | tail -1

or

$ sudo lsmod | grep sample

To pass the string parameter "user" and prints a message which includes the passed parameter.

$ sudo insmod sample.ko user="ilugc"

$ dmesg | tail -2

or

$ sudo lsmod | grep sample

● <u>162 install</u>

install - copy files and set attributes

To move the files from one location or another location or directory

$ install sample.txt data/

to check

$ ls data/

To copy the data from one location to another location with the comparison

$ install -C /file/* data/

$ ls data/

To use install command to change the ownership of the file

$ install -D -o ilugc file.txt /data/

To change the permission mode

$ install -D -m 777 file1.txt /data/

To get help

$ install --help

● **163 ip**

ip - show / manipulate routing, network devices, interfaces and tunnels

To displays info about all network interfaces

$ sudo ip a

$ sudo ip -4 a

$ sudo ip -6 a

$ sudo ip a show eth0

$ sudo ip a list eth0

$ sudo ip a show dev eth0

To show running interfaces

$ sudo ip link ls up

To assign the IP address to the interface

$ sudo ip a add xx.xx.xx.xx/255.255.255.0 dev eth0

To remove / delete the IP address from the interface

$ sudo ip a del xx.xx.xx.xx/255.255.255.0 dev eth0

To check

$ sudo ip addr show

To enable network interface

```
$ sudo ip link set eth1 up
```

To disable network interface

```
$ sudo ip link set eth1 down
```

To check route table

```
$ sudo ip route show
```

To add default gateway

```
$ sudo ip route add default via xx.xx.xx.xx
```

● <u>164 iptables</u>

iptables - administration tool for IPv4/IPv6 packet filtering and NAT

To check all IPtables firewall rules

iptables -L -n -v

To block specific IP Address in IPtables firewall

iptables -A INPUT -s xx.xx.xx.xx -j DROP

To unblock IP address in IPtables firewall

iptables -D INPUT -s xxx.xxx.xxx.xxx -j DROP

To block outgoing connections on a specific port

iptables -A OUTPUT -p tcp --dport xxx -j DROP

To allow incoming connections

iptables -A INPUT -p tcp --dport xxx -j ACCEPT

To allow multiple ports on IPtables using multiport

iptables -A INPUT -p tcp -m multiport --dports 22,80,443 -j ACCEPT

To allow specific network range on particular Port on IPtables

iptables -A OUTPUT -p tcp -d xx.xx.xx.xx/24 --dport 22 -j ACCEPT

To block twitter on IPtables firewall

```
# host twitter.com
```

twitter.com has address 104.244.42.65

```
$ whois 104.244.42.65 | grep CIDR
```

CIDR: 104.244.40.0/21

```
# iptables -A OUTPUT -p tcp -d 104.244.40.0/21 -j DROP
```


To setup port forwarding in IPtables

```
# iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 22 -j
REDIRECT --to-port 2222
```


To block access to specific MAC address on IPtables

```
# iptables -A INPUT -m mac --mac-source aa:bb:cc:dd:ee:ff -j DROP
```


To flush IPtables firewall chains or rules

```
# iptables -F
```


To save IPtables rules to a file

```
# iptables-save > ~/iptables.rules
```


To restore IPtables rules from a file

```
# iptables-restore < ~/iptables.rules
```


To block connection on network interface

```
# iptables -A INPUT -i eth0 -s xx.xx.xx.xx -j DROP
```

● <u>165 isoinfo</u>

isoinfo - utility programs for dumping and verifying iso9660 images.

To list the content of ISO file

$ isoinfo -i ubuntu-20.04-server-amd64.iso -l

To extract a single file from an ISO image

$ isoinfo -i ubuntu-20.04-server-amd64.iso -x MD5SUM.TXT > MD5SUM.TXT

● **166 isosize**

isosize - output the length of an iso9660 filesystem

To view the length of the iso

$ isosize Centos.iso

To show sector number and sector size

$ isosize -x Centos.iso

To display the device size in a block of 1024 bytes

$ isosize -d 1024 Centos.iso

- ## **167 iwconfig**

iwconfig - configure a wireless network interface

To display all the wireless interfaces

$ iwconfig

To displays help

$ iwconfig --help

● <u>168 iwlist</u>

iwlist - Get more detailed wireless information from a wireless interface

To list options

$ iwlist

To list frequency of wireless interface

$ iwlist wlp2s0 frequency

To list the bitrate of wl interface

$ iwlist wlp2s0 bitrate

To display power mode

$ iwlist wlp2s0 power

To list authentication

$ iwlist wlp2s0 auth

- ## **169 jobs**

jobs - used to list the jobs that you are running in the background and in the foreground

$ ping google.com

CTRL+Z

To lists jobs running in background

$ jobs

[1]+ Stopped   ping google.com

To display the process ID or jobs for the job whose name begins with "p"

$ jobs %p

To display PIDs only

$ jobs -p

To display jobs with process id

$ jobs -l

To display only running jobs

$ jobs -r

To make the job to run in foreground

```
$ fg %1
```

● <u>170 journalctl</u>

journalctl - used to query the contents of the systemd

To display newest log entries first

```
$ journalctl -r
```

To display specific number of recent log entries

```
$ journalctl -n 4
```

To display log entries of specific priority

```
$ journalctl -p [ debug, info, notice, warning, err, crit, alert,
and emerg ]
$ journalctl -p debug
$ journalctl -p info
```

To display log entries only for specific systemd unit

```
$ journalctl -u ntpd
$ journalctl -u ftpd
```

To format the output

```
$ journalctl -o verbose
```

To combine all the options

```
$ journalctl -n 3 -p debug
$ journalctl -n 4 -p info
```

● <u>171 join</u>

join - join lines of two files on a common field , join combines lines of files on a common field

```
$ cat file1.txt
```

1 andhra

2 tamilnadu

3 kerala

4 karnataka

5 pondicherry

```
$ cat file2.txt
```

1 101

2 102

3 103

4 104

5 105

To join the 2 files

```
$ join file1.txt file2.txt
```

To create a new file with the joined contents

```
$ join file1.txt file2.txt > file3.txt
```

● <u>172 kill</u>

kill - used to terminate processes manually. kill command sends a
signal to a process which terminates the process

To display all the available signals

$ kill -l

To use PID with the kill command

$ kill pid

To send a kill signal to process ID 9898

$ kill 9898

To kill multiple processes at once

$ kill 8282 9898 7474

To forcefully kill single process

$ kill -9 7890

To forcefully kill multiple process

$ kill -9 6789 7890

To find signal name

```
$ kill -l 3
```

```
$ kill -l 9
```

```
$ kill -l 15
```

To specify name of signal sending to other process with kill command

```
$ kill -s KILL 6789
```

```
$ kill -s
```

To send the signal to interrupt the process 5656

```
$ Kill -2 5656
```

To send the signal to hang up the 8181 process

```
$ kill -1 8181
```

● <u>173 killall</u>

killall - kill processes by name, killall sends a signal to all
processes running any of the specified commands

To killall a program name sample

$ killall example

To killall sshd

$ killall sshd

To send kill signal instead of default term signal

$ killall -9 sshd

killall is case sensitive, To killall a program to ignore case

$ killall -I Example

To get a list of signals that killall can send

$ killall -l

To send different signals to kill processes

$ killall -s SIGINT example

To Kill multiple processes interactively

$ killall -i example1 example2

● **174 kmod**

kmod - Program to manage Linux Kernel modules

To view all the modules currently loaded in the system.

$ sudo kmod list

To list the information of static device nodes

$ sudo kmod static-nodes

● **175 last**

last - show a listing of last logged in users

To list last five users logged in

$ last -5

To display without the host-name field

$ last -R user_name

To display the login and logout time including the dates

$ last -F

To display the host-name in the last column

$ last -a

To display within a specific time period.(-s) since and (-t) until

$ last -s yesterday -t today

To display information like system down entries and run level changes

$ last -x

● **176 lastlog**

lastlog - reports the most recent login of all users or of a given user

To print the last login of all the users

$ sudo lastlog

To print the records of specified days older ex. 7days older

$ sudo lastlog -b 7

To print the last login records of specified user

$ sudo lastlog -u user_name

● **177 lastb**

lastb - is the same as last, except that by default it shows a log of the /var/log/btmp file, which contains all the bad login attempts

To show a list of all failed login attempts

$ sudo lastb

To show a list of failed login attempts since a given time

$ sudo lastb --since YYYY-MM-DD

To show a list of failed login attempts until a given time

$ sudo lastb --until YYYY-MM-DD

To show a list of all failed login attempts at a specific time

$ sudo lastb --present hh:mm

● **178 ldd**

ldd - prints the shared objects (shared libraries) required by each program or shared object specified on the command line

To display the dependencies of cp command

$ ldd /bin/cp

To display dependencies of the command with details

$ ldd -v /bin/cp

To display unused direct dependencies of the command

$ ldd -u /bin/cp

$ ldd -u /bin/grep

● <u>179 link</u>

link - call the link function to create a link to a file

$ link FILE1 FILE2

$ vim file1.txt

1 andhra

2 tamilnadu

3 kerala

4 karnataka

5 pondicherry

link file1.txt to file2.txt

$ link file1.txt file2.txt

it would create the file file2.txt linked to the file file1.txt

● <u>180 less</u>

Less - utility that can be used to read the contents of a text file one page(one screen) at a time

$ less filename

$ dmesg | less

To display the specified text file with line numbers

$ dmesg | less -N

To make less to start at first occurrence of pattern "keyword_name" in the file.

$ dmesg | less -p "KERNEL"

$ less -p ERROR /etc/init/mysql.conf

To remove multiple blank lines

$ less -s file_name

To open multiple files

$ less filea.txt fileb.txt

To keep content on screen after quitting

$ dmesg | less -X

● <u>181 ln</u>

ln - command creates the hard and symbolic links between the files.

To create hard link with the name sample_link_file.txt

$ ln sample_file.txt sample_link_file.txt

To create symbolic or soft link to a file

$ ln -s file.txt link_file.txt

To display the created soft link

$ ls -l link_file.txt

To create a symlink to a directory

$ ln -s /home/ilugc/project ~/ilugc_project

To view the created soft link

$ ls -l ~/ilugc_project

To overwrite an existing symbolic link forcefully

$ ln -sf file.txt link_file.txt

● **182 locale**

locale - displays information about the current locale, or all locales, on standard output

To view system locale in linux

$ locale

To view more information about an environmental variable which store date and time

$ locale -k LC_TIME

To display a list of all available locales

$ locale -a

● **183 localectl**

localectl - used to query and change the system locale and
keyboard layout settings

To change or set system local

$ localectl set-locale LANG=en_IN.UTF-8

To configure a specific locale parameter

$ localectl set-locale LC_TIME=en_IN.UTF-8

● 184 logger

logger - is used to log messages in the system log or syslog.

To log the message to standard error and system logs

```
$ logger -s "This is sample message"
```

To log to message to the specified file

```
$ logger -f file "This is a sample message"
```

To log the message with specified priority

```
$ logger -p 1 "This is sample message"
```

To mark every line with specified tag

```
$ logger -t TAG "This is sample message"
```

To allow the message to start with a hyphen

```
$ logger -- "This is sample message"
```

To specify log size

```
$ logger --size 10 this is a sample log message for testing
purpose……
```

view by

```
$ tail -1 /var/log/syslog
```

To ignore empty lines

```
$ logger -e -f file1.txt
```

● **185 login**

login - used when signing onto a system. It can also be used to switch from one user to another

To log in to the system

login

To log in to the system as user ilugc

login -p ilugc

To login to a domain

login ilugc.in

To skip the second login authentication

login -f -h host_name -f user_name

login -f -h ilugc -f user1

To display help

login --help

● **186 loginctl**

loginctl - The loginctl command can be used to check and control
the status of systemd, and to view the messages of logged-in users

To Show all sessions and attributes

$ loginctl -a

To display session configuration message

$ loginctl show-session

To list currently logged in users

$ loginctl list-users

To show concise runtime status information about one or more
logged in users

$ loginctl user-status USER_NAME

To show properties of one or more users

$ loginctl show-user USER_NAME

● **187 logname**

logname - print the name of the current login user

To display user's login name

$ logname

● <u>188 logout</u>

logout - it performs the task of logging out the logged-in user from the system in that session. Logout only works in logon shells, not in the non-logon shells

To logout the user from the current session from logon shell

$ logout

● **189 logrotate**

logrotate - it allows automatic rotation, compression, removal, and mailing of log files

To force the log rotation

$ sudo logrotate -f /etc/logrotate.conf

To test the log rotation

$ sudo logrotate -d /etc/logrotate.conf

To set to verbose mode

$ sudo logrotate -v /etc/logrotate.conf

To display help

$ sudo logrotate --usage

● <u>190 logsave</u>

logsave - it will execute cmd_prog with the specified argument, and save a copy of its output to logfile

```
$ sudo logsave [filename] [command]
```

To save the output of free -h command

```
$ sudo logsave log_file.txt free -h
$ cat log_file.txt
```

To append the output of the df -Th command to an already existing file log_file.txt

```
$ sudo logsave log_file.txt df -Th
$ cat log_file.txt
```

To save the output of du -hs /home/ilugc in /tmp/output.txt

```
$ sudo logsave /tmp/output.txt du -hs /home/ilugc
```

To save the output of ls in /tmp/log_output.txt

```
$ sudo logsave /tmp/log_output.txt ls
```

● <u>191 look</u>

look - display lines beginning with a given string. it also uses binary search if the file is sorted. If file is not specified, the file /usr/share/dict/words is used

```
$ cat words.txt
```

files

Files

fiction

fig

fix

find

Find

To search for the given string fil in a specified file words.txt

```
$ look fil words.txt
```

files

To search for the given string fi in a specified file words.txt

```
$ look fi words.txt
```

files

fiction

fig

fix

find

To search for the given string in a specified file

```
$ look "#include" program.c
```

include <stdio.h>

include <string.h>

include <stdlib.h>

To ignore case of alphabetic character use -f option

```
$ look -f fil words.txt
```

files

Files

To verify the spelling of the word

```
$ look apple
```

```
$ look ban
```

```
$ look cat
```


To use binary search on the given word list

```
$ look -bf fi words.txt
```

files

Files

fiction

fig

fix

find

Find

● <u>192 lsattr</u>

lsattr is used to list the attributes of a file or directory

The syntax of the lsattr command

```
$ lsattr [options] [file/Dir]
```

To display all the files and directories in the current directory along with their file attributes

```
$ lsattr
```

```
$ lsattr file.txt
```

 To Recursively list attributes of directories and their contents

```
$ lsattr -R /etc/ssh/
```

To List the file's version/generation number

```
$ lsattr -v
```

To display the program version

```
$ lsattr -V
```

To list all files in directories

```
$ lsattr -a
```

```
$ lsattr -a ~
```

To display all the contents of the directory along

with its file attributes

$ lsattr /etc/ssh/

To list directories like other files, rather than listing their
contents

$ lsattr -d /etc/ssh/

● <u>193 ls</u>

ls - list directory contents

To list files and directories

```
$ ls
```

To long listing of files

```
$ ls -l
```

To view hidden files

```
$ ls -a
```

To list files with human readable format

```
$ ls -lh
```

To add the / character at the end of each directory.

```
$ ls -F
```

To list files in reverse order

```
$ ls -r
```

recursively list Subdirectories

```
$ ls -R
```

To sort files by file size

```
$ ls -lS
```

To display Inode number of file or directory

```
$ ls -i
```

To display UID and GID of files

```
$ ls -n
```

To order files based on last modified time

```
$ ls -lt
```

To order files based on last modified time in reverse order

```
$ ls -ltr
```

To make visual classification of files with special characters

/ - directory.

nothing - normal file.

@ - link file.

* - Executable file

```
$ ls -F
```

● <u>194 lshw</u>

lshw - used to generate the detailed information of the system's hardware configuration from various files in the /proc directory

To display full hardware information

```
$ lshw
```

To list hardware in a compact format

```
$ lshw -short
```

To lists all disks and storage controllers in the system

```
$ lshw -class disk -class storage
```

To lists all network interfaces in HTML file

```
$ lshw -class network
```

To check hardware information without the serial number or any other sensitive information

```
$ lshw -sanitize
```

To check numeric IDs of class disk.

```
$ lshw -class disk -numeric
```

To print hardware information in html format

```
$ lshw -html
```

To print hardware configuration details in xml format

```
$ lshw -xml
```

To enable the speed parameter

```
$ lshw -enable spd
```

To disable the speed parameter

```
$ lshw -disable spd
```

To get help

```
$ lshw --help
```

● <u>195 lsb_release</u>

lsb_release - print distribution-specific information

lsb_release is part of a software package LSB core

which may not be installed by default

for debian/ubuntu `$ sudo apt-get install lsb-core`

for centos `$ sudo yum install redhat-lsb-core`

for fedora `$ sudo dnf install redhat-lsb-core`

for opensuse `$ sudo zypper install lsb-core`

To display all information about OS installed

`$ lsb_release -a`

To display the distributor's ID

`$ lsb_release -i`

To display description of the OS

`$ lsb_release -d`

To display the release number of the currently installed
distribution

`$ lsb_release -r`

To display the code name of the currently installed distribution

`$ lsb_release -c`

● **196 lscpu**

lscpu - is used to get CPU information of the system

To display the complete info about the processor

$ lscpu

To display in Human Readable Format

$ lscpu -e

$ lscpu -e=cpu

To display the processor information in a parsing-friendly format

$ lscpu -p

To display output in hexadecimal

$ lscpu -x

To print the CPU info in json format

$ lscpu -J

To print a help message

$ lscpu --help

To display sizes in bytes

$ lscpu --bytes

To display both online and offline CPUs

```
$ lscpu -a -e
```

```
$ lscpu -a -p
```

 To display only offline CPUs

```
$ lscpu --offline -p
```

To display only online CPUs

```
$ lscpu --online -e
```

or

```
$ lscpu --online -p
```

To display information about caches

```
$ lscpu -C
```

To print output to a text file

```
$ lscpu | tee /home/ilugc/cpu_info.txt
```

● <u>197 lsblk</u>

lsblk - to display details about block devices

To display block devices

```
$ sudo lsblk
```

To display empty block devices

```
$ sudo lsblk -a
```

To print size information in bytes

```
$ sudo lsblk -b
```

To print zone model for devices

```
$ sudo lsblk -z
```

To skip slave entries

```
$ sudo lsblk -d
```

To print information about device owner, group, and mode of block devices

```
$ sudo lsblk -m
```

To print selected columns of block-devices

```
$ sudo lsblk -o SIZE, NAME, MOUNTPOINT
```

To display help

```
$ sudo lsblk --help
```

To produce output in the form of a list

```
$ sudo lsblk -l
```

To list information about a particular block device

```
$ sudo lsblk /dev/sdb1/
```

To display SCSI devices only

```
$ sudo lsblk -S
```

● <u>198 lspci</u>

lspci - is a utility on linux systems used to find out information about the PCI busses and devices connected to the PCI subsystem

To list all PCI devices

```
$ sudo lspci
```

To dump PCI Info in different format

```
$ sudo lspci -m
```

```
$ sudo lspci -mm
```

To display the output in tree format

```
$ sudo lspci -t
```

To get detailed device Information

```
$ sudo lspci -v
```

To get info in very verbose mode

```
$ sudo lspci -vv
```

To get info in more verbose mode

```
$ sudo lspci -vvv
```

To show PCI vendor and device codes as numbers

```
$ sudo lspci -n
```

To show PCI vendor and device codes as both numbers and names

```
$ sudo lspci -nn
```

To display info of a specific device

```
$ sudo lspci -s [device_number]
$ sudo lspci -s 00:02.0
```

To show kernel drivers handling each device

```
$ sudo lspci -k
```

To get hexadecimal dump of the whole PCI configuration space

```
$ sudo lspci -xxx
```

To get bus centric view

```
$ sudo lspci -b
```

To get PCI domain numbers

```
$ sudo lspci -D
```

● **199 lsof**

lsof - it provides a list of files that are opened by which process

To list out all the files that are opened by any process in the system

$ sudo lsof

To list all files opened by a specific user

$ sudo lsof -u USER_NAME

To list all open files by a particular Process

$ sudo lsof -c mariadb

To list all open files that are opened by a particular process

$ sudo lsof -p process_ID

To find out the list of files opened by parent process Id

$ sudo lsof -R

To lists out the files which are opened by a particular directory

$ sudo lsof -D path/to/directory

To find out files opened by network connections

$ sudo lsof -i

To find out files opened by processes running on specific port

```
$ lsof -i TCP:22
```

To list only IPv4 and IPv6 open files

```
$ sudo lsof -i 4
$ sudo lsof -i 6
```

To list all the running processes of open files of TCP Port ranges from 1-1024

```
$ sudo -i TCP:1-1024
```

To find what files and commands a specific user used

```
$ sudo lsof -i -u USER_NAME
```

● **<u>200 lslocks</u>**

lslocks - lists information about all the currently held file locks in a Linux system

$ sudo lslocks [options]

To list all file locks

$ sudo lslocks

To print the SIZE column in bytes

$ sudo lslocks -b

To print the all file locks in json format

$ sudo lslocks -J

To display the PID of all file locks

$ sudo lslocks --output PID

To display the COMMAND of file locks

$ sudo lslocks --output COMMAND

To display only the locks held by the process with specific pid

$ sudo lslocks -p <PID>

● <u>201 lsmem</u>

lsmem - it lists the ranges of available memory with their online status

To list the available online memory status

$ lsmem

To List each individual memory block, instead of combining memory blocks with similar attributes

$ lsmem -a

To print the SIZE column in bytes

$ lsmem -b

To print the output in json format

$ lsmem -J

To print output without header line

$ lsmem -n

To get help options

$ lsmem -h

To print output of specific column

$ lsmem -o RANGE

$ lsmem -o SIZE

$ lsmem -o STATE

$ lsmem -o REMOVABLE

$ lsmem -o BLOCK

To print all available columns

```
$ lsmem --output-all
```

To Produce output in the form of key="value" pairs

```
$ lsmem --pairs
```

● <u>202 lsns</u>

lsns - lists information about all the currently accessible namespaces or about the given namespace

To display information about all the currently accessible namespaces

$ sudo lsns

To print info about all currently accessible namespaces in JSON output format

$ sudo lsns -J

To print the output without header line

$ sudo lsns -n

To print only namespace identifier (inode number)

$ sudo lsns -o NS

To print only kind of namespace

$ sudo lsns -o TYPE

To print only the path to the namespace

$ sudo lsns -o PATH

To print the number of processes in the namespace

$ sudo lsns -o NPROCS

To print the lowest PID in the namespace

$ sudo lsns -o PID

To print the PPID of the PID

```
$ sudo lsns -o PPID
```

To print the command line of the PID

```
$ sudo lsns -o COMMAND
```

To print the UID of the PID

```
$ sudo lsns -o UID
```

To print the username of the PID

```
$ sudo lsns -o USER
```

To print the namespace ID as used by network subsystem

```
$ sudo lsns -o NETNSID
```

To print the nsfs mountpoint

```
$ sudo lsns -o NSFS
```

To display all output columns

```
$ sudo lsns --output-all
```

To display only the namespaces held by the process with specific PID

```
$ sudo lsns -p <PID>
```

To use the raw output format

```
$ sudo lsns -r
```

To display the specified type of namespaces only

```
$ sudo lsns -t mnt
```

```
$ sudo lsns -t net
```

```
$ sudo lsns -t ipc
```

```
$ sudo lsns -t user
```

```
$ sudo lsns -t pid
```

```
$ sudo lsns -t uts
```

```
$ sudo lsns -t cgroup
```

To display output not in columns

```
$ sudo lsns -u
```

To print help options

```
$ sudo lsns --help
```

● **203 lsinitramfs**

lsinitramfs - lists the content of given initramfs images

To list initramfs content of current running kernel

$ sudo lsinitramfs /boot/initrd.img-$(uname -r)

To display long and more verbose listing of initramfs content

$ sudo lsinitramfs -l /boot/initrd.img-$(uname -r)

● **204 lsipc**

lsipc- show information on IPC facilities currently employed in the system

To show the information on IPC in the system

$ lsipc

To drite information about active shared memory segments

$ lsipc -m

To print information about active message queues

$ lsipc -q

To print information about active semaphore sets

$ lsipc -s

To print the output data in the format of NAME=VALUE

$ lsipc -e

To print the output data in the JSON format

$ lsipc -J

To list the output format

$ lsipc -l

To display each information on a separate line

$ lsipc -n

To print without header line

```
$ lsipc --noheadings
```

To print raw output

```
$ lsipc -r
```

To Print size in bytes

```
$ lsipc -b
```

To print specific output columns

```
$ lsipc -o RESOURCE
$ lsipc -o DESCRIPTION
$ lsipc -o LIMIT
$ lsipc -o USED
$ lsipc -o USE%
```

● **205 lslogins**

lslogins - display information about known users in the system

To display information about known users in the system

$ lslogins

To Display data about the date of last password change and the account expiration date

$ sudo lslogins -a

To print separate info about each user with a colon instead of a newline

$ lslogins -c

To print output data in the format of NAME=VALUE

$ lslogins -e

To display data about the users' last failed login attempts

$ lslogins -f

To show information about supplementary groups

$ lslogins -G

To print data of users belonging to groups

$ lslogins --groups=<group_name>

To display help

$ lslogins --help

To print data containing information about the users' last login sessions

```
$ lslogins -L
```

To display each piece of information on a separate line

```
$ lslogins -n
```

To print without header line

```
$ lslogins --noheadings
```

To print specific output columns

```
$ lslogins -o USER
$ lslogins -o PROC
$ lslogins -o GECOS
$ lslogins -o UID
$ lslogins -o USER
```

To print all available columns

```
$ lslogins --output-all
```

To display information related to login by password

```
$ lslogins --pwd
```

To print raw output

```
$ lslogins -r
```

To show system accounts

```
$ lslogins -s
```

To show user accounts

```
$ lslogins -u
```

To display the users' security context

```
$ lslogins -Z
```

● **206 lsmod**

lsmod - Show the status of modules in the Linux Kernel

$ lsmod
Module   Size Used by

it has three columns

1 module name

2 shows the size of the module in bytes

3 indicates how many instances of the module are currently used
   and what is using the particular module

$ lsmod | grep kvm

kvm_intel   282624 0
kvm   663552 1 kvm_intel

$ lsmod | grep realtek

realtek   24576 1

● **207 lsusb**

lsusb - utility for displaying information about USB buses in the system and the devices connected to them

To print usb devices connected
$ sudo lsusb

To display detailed information about usb devices in verbose mode
$ sudo lsusb -v

To display physical USB device hierarchy as a tree
$ sudo lsusb -t

● **208 man**

man - an interface to the system reference manuals

it shows the section numbers of the manual and types of pages they contain

1   Executable programs or shell commands

2   System calls (functions provided by the kernel)

3   Library calls (functions within program libraries)

4   Special files (usually found in /dev)

5   File formats and conventions, e.g. /etc/passwd

6   Games

7   Miscellaneous (including macro packages and conventions), e.g.

          man(7), groff(7)

8   System administration commands (usually only for root)

9   Kernel routines [Non standard]

$ man [COMMAND NAME]

To display the whole manual of the command

$ man ls

$ man df

To display only a specific section of a manual

$ man 1 ls

$ man 1 ps

$ man 8 modprobe

$ man 8 modinfo

To display the section in which the given command is present

`$ man -f modprobe`

`$ man -f modinfo`

`$ man -f ls`

`$ man -f df`

To search by Considering Input command as a Regular Expression

`$ man -k ls`

`$ man -k cd`

`$ man -k df`

To display all available intro manual pages contained in each section, one at a time

`$ man -a intro`

To display location of man pages

`$ man -w ls`

`$ man -w du`

`$ man -w df`

`$ man -w cat`

To search for manual pages using case-sensitivity

`$ man -I Ls`

`$ man -I ls`

● **209 mandb**

mandb - used to initialize or manually update/create the index
database cache that is usually maintained by man

```
$ sudo mandb
```

To print debugging information

```
$ sudo mandb -d
```
```
$ sudo mandb -d unzip
```

To do mandb without warnings in quiet mode

```
$ sudo mandb -q
```

To force mandb to delete previous databases and recreate them

from scratch, and implies --no-purge

```
$ sudo mandb -c
```

To Create user databases with write permissions to create system
db

```
$ sudo mandb -u
```

To perform correctness checks on manual pages

```
$ sudo mandb -t
```

To specify the configuration file to use

```
$ sudo mandb --config-file=file
```

To update the index cache of the top command

```
$ sudo mandb top
```

● <u>210 manpath</u>

manpath - determine search path for manual pages

To print the search path for man pages

$ sudo manpath

To Specify the configuration file to use , default is
/etc/manpath.config

$ sudo manpath -C config_file

To print debugging information

$ sudo manpath -d

To produce a manpath consisting of all paths named as
"global" within the man-db configuration file

$ sudo manpath -g

To produce a catpath as opposed to a manpath

$ sudo manpath -c

● <u>211 md5sum</u>

md5sum - it is designed to create, read, and check file integrity using MD5

$ cat example.txt

this is line one

this is line two

To display the file hash value alongside the filename

$ md5sum [filename]

$ md5sum example.txt

To read the file in binary mode

$ md5sum -b example.txt

To read the file in text mode

$ md5sum -t example.txt

To create a BSD-style checksum with -tag

$ md5sum --tag example.txt

To check a file by comparing its hash value with the value provided in a hash file

$ cat example.txt

this is line one

this is line two

To store the MD5 checksum for example.txt in file checkmd5.md5

$ md5sum example.txt > checkmd5.md5

To check the contents of file

```
$ md5sum -c checkmd5.md5
```

example.txt: OK

After changing the contents of file

```
$ echo "Hai" >> example.txt
$ md5sum -c checkmd5.md5
```

example.txt: FAILED
md5sum: WARNING: 1 computed checksum did NOT match

To create a BSD-style checksum with tag option

```
$ md5sum --tag test.txt
```

To validate multiple files

```
$ md5sum file1.txt file2.txt file3.txt > hashfile
```

To check the integrity of above multiple files

```
$ md5sum -c hashfile
```

change the content of any one above file for eg. file2.txt and check

To display only modified files

```
$ md5sum --quiet -c hashfile
```

● **212 mesg**

mesg - it allows to control write access to your terminal by other users.

To display the current write status of your terminal

$ mesg

To allow write access to your terminal

$ mesg y

To disallow write access to your terminal

$ mesg n

● <u>213 mkdir</u>

mkdir - make directories

mkdir [options...] [directories ...]

To display the version number
$ mkdir --version

To display the help options
$ mkdir --help

To display verbose message for every directory created.
$ mkdir -v directory_1 directory_2 directory_3

To create multiple directories
$ mkdir {dir1,dir2,dir3}

To create directory without verbose
$ mkdir directory_4

To create parent directories
$ mkdir -p /dir_1/dir_2/dir_3
$ mkdir -p -v /dir_1/dir_2/dir_3

To set permissions for the directories
$ mkdir -m a=rwx [directories]
$ mkdir -m777 dir_1
$ mkdir -m755 dir_2
$ mkdir -m766 dir_3

● <u>214 mkswap</u>

mkswap - set up a Linux swap area

To make the swap

$ sudo mkswap /dev/sdb

To check the device for bad blocks before creating the swap area

$ sudo mkswap -c /dev/sdb

To create swap area larger than the file or partition it resides on

$ sudo mkswap -f /dev/sdb

To specify the page size (in bytes) to use, mkswap reads the size from the kernel

$ sudo mkswap -p PAGESIZE

To specify a label for the device, to allow swapon by label

$ sudo mkswap -L LABEL

To specify the swap space version

$ sudo mkswap -v1

To specify the UUID to use. The default is to generate a UUID

$ sudo mkswap -U UUID

● **215 modinfo**

modinfo - Show information about a Linux Kernel module

To list available modules
$ less /proc/modules
$ lsmod

To show the information on a module
$ modinfo <module_name>
$ modinfo bluetooth
$ modinfo ath10k_pci
$ modinfo snd
$ modinfo thermal_sys

To print the help options
$ modinfo --help

To print version
$ modinfo -V

To information about a kernel other than the running one
$ modinfo -0 ath10k_pci

To print shortcuts used for the –field flag's author, description, license, parm

and filename arguments

```
$ modinfo ath10k_pci -a

$ modinfo bluetooth -n

$ modinfo bluetooth -d

$ modinfo ath10k_pci -l

$ modinfo ath10k_pci -p
```


To print only provided FIELD

```
$ modinfo -F parm ath10k_pci

$ sudo modinfo -F parm bluetooth

$ sudo modinfo -F parm snd
```

● **216 modprobe**

modprobe - Add and remove modules from the Linux Kernel

To find the available modules

```
$ find /lib/modules/$(unam -r) -type f -name '*.ko' | more
```

To load a Linux Kernel Module using modprobe

```
$ sudo ln -s /path/to/kernel-module /lib/modules/`uname -r`
$ sudo depmod -a
$ sudo modprobe kernel-module
```

To add a module into the kernel

```
$ sudo modprobe <module name>
$ sudo modprobe soundcore
$ sudo modprobe torture
```

To Check if module is added to the kernel

```
$ sudo modprobe soundcore --first-time
$ sudo modprobe torture --first-time
```

To Remove a module from the kernel

```
$ sudo modprobe -r soundcore
$ sudo modprobe -r torture
```

To check the module has been successfully removed

```
$ sudo modprobe -r torture --first-time
$ sudo modprobe -r soundcore --first-time
```

To make a dry run for debugging

```
$ sudo modprobe -vn module_name
$ sudo modprobe -vn soundcore
$ sudo modprobe -vn torture
```

To suppress the error information

```
$ sudo modprobe lk
$ sudo modprobe -q lk
```

To dump out the effective configuration from the config directory and exit

```
$ sudo modprobe -c
```

● **217 mke2fs**

mke2fs - create an ext2/ext3/ext4 filesystem

To list the available mkfs* commands in a system.
```
$ ls mkfs*
```

To create a filesystem in a specific device
```
$ sudo mke2fs -t ext4 /dev/sda3
```

To create a filesystem with Journal
```
$ sudo mke2fs /dev/sda3 -j
```

To create an ext4 filesystem with 7500 bytes per inode, with a volume label MYDATA
```
$ sudo mke2fs -t ext4 -L MYDATA -i 7500 /dev/sdb1
```
to check the inode
```
$ df -i /dev/sdb1
```

To check for bad blocks on a device
```
$ sudo mke2fs -c /dev/sda3
```

To force to create a filesystem on a mounted partition
```
$ sudo mke2fs -F /dev/sda3
```

To set the volume label for partition
```
$ sudo mke2fs -L MYVOL /dev/sd3
```

To view the label name
```
$ sudo e2label /dev/sda3
```

To simulate a filesystem creation

```
$ sudo mkfs -t ext4 -n /dev/sda3
```

To create a filesystem with specific number of inodes

```
$ sudo mkfs ext4 -v -N 600000 /dev/sda3
```

To check the above created filesystem inode

```
$ tune2fs -l /dev/sda3 | grep -i inode
```

● **218 mkfs.ext4**

mkfs.ext4 - is used to create filesystem (ext2, ext3, ext4, etc) on Linux system

To format the disk as a ext4 partition

```
$ sudo mkfs.ext4 /dev/sdb
```

To check the partition for bad blocks before formatting

```
$ sudo mkfs.ext4 -c /dev/sdc
```

To quietly create an ext4 partition

```
$ sudo mkfs.ext4 -q /dev/sdb
```

To create an ext4 filesystem with label backup

```
$ sudo mkfs.ext4 -L backup /dev/sdc
```

To create an ext4 filesystem with detail verbose output

```
$ sudo mkfs.ext4 -v /dev/sdb
```

● **219 mkfs.ntfs / mkfs.vfat**

mkfs.ntfs - create an NTFS file system

mkfs.vfat - create a vfat file system

To create a NTFS file system

```
$ sudo mkfs -t ntfs /dev/sdb
```


To create a vfat file system

```
$ sudo mkfs.vfat /dev/sdc
```

● <u>220 mkinitramfs</u>

mkinitramfs - low-level tool for generating an initramfs image

To create an initramfs for current running kernel

$ mkinitramfs -o ~/tmp/initramfs-$(uname -r)

To create an initramfs for specific kernel and keep builddirs

$ mkinitramfs -k -o ~/tmp/initramfs-2.6.21-686 2.6.21-686

To get help options

$ mkinitramfs --help

● <u>221 mkisofs</u>

mkisofs - is a utility that creates an ISO 9660 image from files on disk

To create an ISO that can be used to back up another ISO file

```
$ mkisofs -o [filename.iso] [directory_path]
```

```
$ mkisofs -o bootiso.iso /boot
```

To create ISO image of a folder in Linux

```
$ mkisofs -J -allow-lowercase -R -V "BootCD" -iso-level 4 -o
BootCD.iso ~/BootCD
```

To list content of ISO file

```
$ isoinfo -l -i bootiso.iso
```

● <u>222 more</u>

more - is used to view the text files in the command prompt, displaying one screen at a time in case the file is large

To help the user to navigate the long files , Press space to continue,

'q' to quit. and display Press 'h' for instructions.

$ more -d file.txt

To display as it is and not to wrap the lines

$ more -f file.txt

To clear the screen and then displays the text

$ more -p file.txt

To display the pages on the same area by overlapping the previously displayed text.

$ more -c file.txt

To compress multiple blank lines into one single blank line

$ more -s file.txt

To omit the underlines in a file

$ more -u file.txt

To search the string inside file

$ more +/<string> file.txt

$ more +/default file.txt

To display the text after the specified number of lines of the file

```
$ more +20 file.txt
```

```
$ more +50 file.txt
```

To display first N lines of a file

```
$ more -10 file.txt
```

To use pipe to see long outputs

```
$ cat file.txt | more
```

● **223 mount**

mount - is used to mount the filesystem

syntax
```
$ mount -t type device dir
```

To list mounted file systems
```
$ mount
```

To list information about specific file systems
```
$ sudo mount -l -t ext4
```

To mount file systems
```
$ sudo mount /dev/sdb4 /media/ilugc
```

To mount ISO files
```
$ sudo mount /file.iso /media/iso-file -o loop
```

To mount an NFS
```
$ sudo mkdir /media/nfs
$ sudo mount /media/nfs
```

To mount all the /etc/fstab entries
```
$ sudo mount -a
```

To mount only specific filesystem from /etc/fstab

```
$ sudo mount /backup_data
```

or

mount with device name

```
$ sudo mount /dev/sda3
```

To bind mount points to a new directory

```
$ sudo mount -B /backup_data /mnt
```

To access contents from new mount point

```
$ sudo mount -M /backup_data /mnt/
```

To mount without writing entry into /etc/mtab

```
$ sudo mount -n /dev/sda5 /backup_data
```

To mount partition as read only

```
$ sudo mount /dev/sda4 /backup_data -r
```

To remount the mounted filesystem

```
$ sudo mount -o remount,rw /backup_data
```

224 mdadm

mdadm (Multiple Disk and Device Management) - manage MD devices
aka Linux Software RAID

```
$ sudo apt-get install mdadm
```

syntax
```
$ sudo mdadm [mode] <raiddevice> [options] <component-devices>
```

To create RAID 0 array https://www.acnc.com/raid/?raid-level=0
```
$ sudo mdadm --create /dev/md0 --level=0 --raid-devices=2
/dev/sdc1 /dev/sdd1
```

To create RAID 1 array https://www.acnc.com/raid/?raid-level=1
```
$ sudo mdadm --create /dev/md1 --level=1 --raid-devices=2
/dev/sdc1 /dev/sdd1
```

To create RAID 5 array https://www.acnc.com/raid/?raid-level=5
```
$ sudo mdadm --create /dev/md5 --level=5 --raid-devices=3
/dev/sdc1 /dev/sdd1 /dev/sde1
```

To create RAID 10 array https://www.acnc.com/raid/?raid-level=10
```
$ sudo mdadm --create /dev/md2 --level=10 --raid-devices=3
/dev/sdc1 /dev/sdd1 /dev/sde1
```

To check if it is an md device or a component of an md array
```
$ sudo mdadm -Q /dev/md0
```

To print detail of md devices
```
$ sudo mdadm -D /dev/md0
```

To add the RAID arrays to the configuration file

```
$ sudo mdadm -D -s > /etc/mdadm.conf
```

To create a file system on a RAID drive

```
$ sudo mkfs.ext4 /dev/md0
```

To mount the RAID device

```
$ sudo mkdir /mnt/raid
$ sudo mount /dev/md0 /mnt/raid
```

To deactivate or delete a RAID array
first stop the RAID device

```
$ sudo mdadm -S /dev/md0
```

then

```
$ sudo mdadm --zero-superblock /dev/sdc1 /dev/sdd1
```

To add a disk to an existing array

```
$ sudo mdadm --add /dev/md0 /dev/sdc1
```

To remove a disk from an array

```
$ sudo mdadm /dev/md0 --fail /dev/sdb1 --remove /dev/sdb1
```

To assemble and start all arrays listed in the standard config file

```
$ sudo mdadm -A -s
```

To print help

```
$ sudo mdadm --help
$ sudo mdadm --create --help
$ sudo mdadm --assemble --help
$ sudo mdadm --build --help
```

```
$ sudo mdadm --manage --help
$ sudo mdadm --misc --help
$ sudo mdadm --grow --help
$ sudo mdadm --incremental --help
$ sudo mdadm --monitor --help
```

● <u>225 mv</u>

mv - move or rename files

syntax:

$ mv [Option] source destination

To rename a file1.txt to file2.txt

$ mv file1.txt file2.txt

$ mv file1.txt /home/Documents/file2.txt

To interactively rename file1.txt to file2.txt

$ mv -i file1.txt file2.txt

To forcefully rename or move the files , not prompt before overwriting

$ mv -f file1.txt file2.txt

To prevent overwrite an existing file

$ mv -n file1.txt /home/ILUGC/Documents/

To create a backup of existing destination file that will be overwritten

$ mv -b file1.txt /home/ILUGC/Documents/

To move only If source file Is newer than destination

or when the destination file is missing

$ mv -u file1.txt ~/Documents/

To move multiple directories from one location to another

$ mv dir1 dir2 dir3 /path/to/destination_directory/

To move multiple files from one location to another

```
$ mv file1 file2 file3 /path/to/destination_dir/
```

To set SELinux security context of destination file to default type

```
$ mv -Z file1.txt /path/to/destination_dir/
```

● <u>226 nice</u>

nice - run a program with modified scheduling priority

To check all nice values of all processes

$ top

To check the nice value of vlc process

$ ps -el | grep vlc

To check the nice value of terminal

$ ps -el | grep terminal

To check the nice value of top process

$ ps -el | grep top

To set the priority of a process

$ nice -n <number><process name>

$ nice -10 vlc

$ nice -10 gnome-terminal

$ nice -n 5 bash

$ nice -n 5 top

To set the negative priority for a process

$ nice --n <number><process name>

$ nice --10 vlc

$ nice --10 gnome-terminal

$ nice --10 top

● 227 nmap

nmap - tool for network exploration and security auditing

To scan a system with hostname and IP address
```
$ nmap www.ilugc.in
$ nmap 18.140.226.100
```

To get more detailed information about the remote machines
```
$ nmap -v www.ilugc.in
```

To scan multiple hosts
```
$ nmap 157.240.16.35 104.244.42.193 18.140.226.100
```

To scan whole subnet
```
$ nmap 18.140.226.*
```

To scan to detect firewall settings
```
$ nmap -sA 18.140.226.100
```

To scan from a file
```
$ cat input.txt
157.240.16.35
104.244.42.193
18.140.226.100

$ nmap -iL input.txt
```

To scan multiple servers using last octet of IP address
```
$ nmap 172.10.0.101,102,103
```

● 227 nmap

To scan IP address range

```
$ nmap 172.10.0.101-110
```

To scan network excluding remote hosts

```
$ nmap 172.10.0.* --exclude 172.10.0.100
```

To scan OS information and traceroute

```
$ nmap -A  18.140.226.100
```

To enable OS detection with nmap

```
$ sudo nmap -O ilugc.in
```

To scan a host to check its protected by firewall

```
$ nmap -PN 18.140.226.100
```

To find out Live hosts in a network

```
$ nmap -sP 18.140.226.*
```

To perform a fast scan

```
$ nmap -F 18.140.226.100
```

To print host interfaces and routes

```
$ nmap --iflist
```

To scan for specific port

```
$ nmap -p 80 www.ilugc.in
$ nmap -p 443 www.ilugc.in
```

To scan a TCP port

```
$ nmap -p T:443,80 www.ilugc.in
```

To scan a UDP Port

```
$ nmap -pU 22 www.ilugc.in
```

To scan multiple ports

```
$ nmap -p 80,443,22,53 18.140.226.100
```

To scan ports by range

```
$ nmap -p 80-450 18.140.226.100
```

To find host services version numbers

```
$ nmap -sV 18.140.226.100
```

● <u>228 networkctl</u>

networkctl - Query the status of network links

first check

$ sudo systemctl start systemd-networkd

$ sudo systemctl enable systemd-networkd

$ sudo systemctl status systemd-networkd

To get the status information about network links

$ networkctl

To display all network links and their status

$ networkctl -a

or

$ networkctl list

To display information type, state, kernel module driver, hardware
and IP address, configured DNS

$ networkctl status

$ networkctl status wlp2s0

$ networkctl status virbr0

$ networkctl status docker0

To show Link Layer Discovery Protocol (LLDP) status

$ networkctl lldp

To prevent the networkctl output piped into a pager

$ networkctl status --no-pager

To print output without headers and footers

$ networkctl --no-legend

To show detailed link statics

$ networkctl -s

To get help commands

$ networkctl --help

To show current address label entries in the kernel

$ networkctl label

To reload .network and .netdev files

$ networkctl reload

● **229 netstat**

netstat - netstat - Print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships

To show both listening and non-listening sockets.
$ netstat -a | more

To List all tcp ports
$ netstat -at

To List all udp ports
$ netstat --au

To List only listening ports
$ netstat -l

To List only listening TCP ports
$ netstat -lt

To List only listening UDP ports
$ netstat -lu

To List only the listening UNIX ports
$ netstat -lx

To List the statistics for all ports.
$ netstat -s

To List the statistics for TCP ports

```
$ netstat -st
```

To List the statistics for UDP ports

```
$ netstat -su
```

To display the PID and program names

```
$ netstat -pt
```

To print the netstat information continuously

```
$ netstat -c
```

To get the kernel routing information

```
$ netstat -r
```

To get the port on which a program is running

```
$ netstat -ap | grep <program_name>
$ netstat -ap | grep ftp
$ netstat -ap | grep ssh
```

To get the process which is using the given port

```
$ netstat -an | grep ':<port_number>'
$ netstat -an | grep ':443'
$ netstat -an | grep ':80'
$ netstat -an | grep ':53'
```

To get the list of network interfaces

```
$ netstat -i
```

To display extended information on the interfaces

```
$ netstat -ie
```

To print the selected information every second continuously

```
$ netstat -c
```

● [230 nisdomainname](#)

nisdomainname - show or set the system's NIS/YP domain name

To print alias name
$ nisdomainname -a

To print all long host names (FQDN)
$ nisdomainname -A

To set default hostname if none available
$ nisdomainname -b

To print DNS domain name
$ nisdomainname -d

To display long host name (FQDN)
$ nisdomainname -f

To read host name or NIS domain name from given file
$ nisdomainname -F

To print ip addresses for the host name
$ nisdomainname -i

To print all addresses for the host
$ nisdomainname -I

To print short host name
$ nisdomainname -s

To print NIS/YP domain name

```
$ nisdomainname -y
```

● **231 nano**

nano - is a user-friendly, simple text editor

To create and open a new file

$ nano file.txt

This is line 1

This is line 2

This is line 3

To save the above file

press Ctrl+o

it asks File Name to Write: file.txt

hit ENTER

To exit a file

press Ctrl+x

To cut a line move to the line and

Ctrl+k

To paste the above cut line hit

Ctrl+u

To cut a select word, select the word by

SHIFT + right arrow

and cut by

Ctrl+k

and paste by

Ctrl+u

To search a word in a file

Press Ctrl+w

Search: <keyword>

It will place the cursor in the first letter of the first occurrence of the word.

To do spell check

$ sudo apt install spell

press Ctrl+t

To replace a word with another word

Ctrl+\

Search (to replace): line1

Replace with: line2

Replace this instance?

It will ask to replace first instance of the word press : Y

to replace all the occurrences of the word press : A

To show the current cursor position in the text

Ctrl + c

To Justify the current paragraph

Ctr l+ J

To go to the specified line and column number in a file

Ctrl + _

Enter line number, column number:

To go to beginning of paragraph

Ctrl + W

To go to end of paragraph

Ctrl + o

To go to first line

Ctrl + y

To go to last line

Ctrl + v

To cancel

Ctrl + c

To get help

Ctrl + g

To exit from nano editor

Ctrl + x

To go to beginning of current line

Ctrl + a

To go to end of current line

Ctrl + e

To go to previous line

Ctrl + p

To go to next line

Ctrl + n

To go one screenful up

Ctrl + y

To go one screenful down

Ctrl + v

● **232 nmcli**

nmcli - command-line tool for controlling NetworkManager

To check networkmanager is running

```
$ nmcli -t -f RUNNING general
```

To get general status of networkmanager

```
$ nmcli general
```

To list all the available device

```
$ nmcli dev status
```

To list all the available connections

```
$ nmcli con show
```

To list all the configuration of interface

```
$ nmcli con show <network_interface>
$ nmcli con show eth0
$ nmcli con show docker0
$ nmcli con show virbr0
```

To check physical network device status

```
$ nmcli dev status
```

To change hostname using nmcli

check

```
$ nmcli general hostname
```

then update the hostname

```
$ nmcli general hostname server.example.com
```

To reload connection

```
$ nmcli con reload
```

To Interactively add or edit a connection

```
$ nmcli con edit  eth0
```

To display selected fields with values of connection

```
$ nmcli -g ip4.address connection show eth0
$ nmcli -g ipv4.dns connection show eth0
```

To activate a connection

```
$ nmcli con up eth0
```

To deactivate a connection

```
$ nmcli con down eth0
```

To delete connection

```
$ nmcli con del <interface_name>
```

● <u>233 nl</u>

nl - is used for numbering lines, accepting input either from a file or from STDIN

```
$ cat file.txt
```
Apache

Squid

Samba

DNS

DHCP

To display a file with line numbers
```
$ nl file.txt
```

To number all lines including empty lines
```
$ nl -b a file.txt
```

To make line number increment at each line
```
$ nl -i 2 file.txt
$ nl -i 3 file.txt
```

To make the starting line number different
```
$ nl -v 5 file.txt
```

To add a string lafter line numbers
```
$ nl -s "..." file.txt
$ nl -s "###" file.txt
```

To change column for line numbers

```
$ nl -w2 file.txt
$ nl -w3 file.txt
```

To print the lines using a different number format

```
$ nl -n ln file.txt
$ nl -n rn file.txt
$ nl -n rz file.txt
```

● 234 netcat

netcat - is a networking , security or network monitoring tool,it can perform any operation in Linux related to TCP, UDP, or UNIX-domain sockets.

To scan a single port (port no. 22)

$ nc -v -w 2 z 192.168.122.10 22

To scan multiple ports 22 , 80 , 53

$ nc -v -w 2 z 192.168.122.10 22 80 53

To scan range of ports (20-85)

$ nc -v -w 2 z 192.168.122.10 20-85

To find service running on port

$ nc -v -n 192.168.122.10 443

$ nc -v -n 192.168.122.10 8080

To transfer files using nc

on receiving server

$ nc -l -p 9899 > file.txt

on sending server

$ nc -w 2 192.168.122.10 9899 < file.txt

● **235 newgrp**

newgrp - is used to change the current group ID (GID) during a
login session for a user

$ newgrp [-] [group]

$ newgrp ilugc

Attempts to log in to the group ilugc

Attempts to log in to the group ilugc , if successful re-
initializes the user environment.

$ newgrp - ilugc

To change the real group ID back to your original login group

$ newgrp

● <u>236 newusers</u>

newusers - update and create new users in batch

create users details in a file

$ sudo vim users.txt

ilugc:123:1002:1002:Foss Admin:/home/ilugc:/bin/bash

klug:123:1003:1003:Foss:/home/klug:/bin/bash

:x save and exit

set the required permissions

$ sudo chmod 0600 users.txt

run the newusers command to add the users in the users.txt

$ sudo newusers users.txt

check for the users added

$ cat /etc/passwd

● 237 nohup

nohup - run a command immune to hangups, with output to a non-tty

create example.sh

```
$ sudo vim example.sh
#!/bin/bash
echo "hello!!!"
:x
```

To run example.sh with nohup

```
$ nohup bash example.sh
$ cat nohup.out
```

To run a process in the background with nohup

```
$ nohup [command] &
$ nohup bash example.sh &
$ nohup ping -i 10 google.com &
```

to bring the process to foreground

```
fg
```

To run multiple processes in the background with nohup

```
$ nohup bach -c '[command1] && [command2]'
$ nohup bash -c 'date && cal && ls && free'
$ cat nohup.out
```

To redirecting output to different file

```
$ nohup [command] > /path/to/output/file.txt
$ nohup bash -c 'date && cal && ls && free' > myfile.txt
```

● <u>238 nproc</u>

nproc - print the number of processing units available

To print the number of processing units available in the system or to the current process

$ nproc

To print total installed processing units

$ nproc --all

To exclude some processing units

$ nproc --ignore=4

To display the help

$ nproc --help

● <u>239 nslookup</u>

nslookup - query Internet name servers interactively

To find out "A" record (IP address) of Domain
$ nslookup ilugc.in

To find out reverse domain lookup
$ nslookup 34.87.59.92

To Query MX (Mail Exchange) records.
$ nslookup -query=mx www.ilugc.in

To query NS(Name Server) record.
$ nslookup -query=ns www.yahoo.com

To query SOA (Start of Authority) record.
$ nslookup -type=soa www.yahoo.com

To query all Available DNS records.
$ nslookup -query=any facebook.com

To enable debug mode
$ nslookup -debug facebook.com

● <u>240 nstat</u>

nstat - simple tools to monitor kernel snmp counters and network interface statistics

To Dump absolute values of counters

$ nstat -a

To format output in JSON

$ nstat -j

To make pretty print

$ nstat -p -j

To reset history

$ nstat -r

To not update the history

$ nstat -s

To show entries with zero activity

$ nstat -z

● 241 on_ac_power

on_ac_power - test whether computer is running on AC power

```
$ on_ac_power
0 (true)  System is on mains power
1 (false) System is not on mains power
255 (false)    Power status could not be determined
```

run system on mains power
```
$ on_ac_power
$ echo $?
0
```

run system on battery power
```
$ on_ac_power
$ echo $?
1
```

● **242 openssl**

openssl - is an open-source command line tool that is commonly used to generate private keys, create CSRs, install your SSL/TLS certificate, and identify certificate information.

To generate private key and certificate signing request
```
$ openssl req -out ilugc.csr -newkey rsa:2048 -nodes -keyout
ilugc.key
```

To create a self-signed certificate
```
$ openssl req -x509 -sha256 -nodes -days 365 -newkey rsa:2048 -
keyout ilugc_selfsigned.key -out ilugc_cert.pem
```

To verify CSR file
```
$ openssl req -noout -text -in ilugc.csr
```

To create RSA private key
```
$ openssl genrsa -out private.key 2048
```

To remove passphrase from key
```
$ openssl rsa -in certkey.key -out nopassphrase.key
```

To verify private key
```
$ openssl rsa -in certkey.key –check
```

To verify certificate file
```
$ openssl x509 -in certfile.pem -text –noout
```

To verify the Certificate Signer Authority

```
$ openssl x509 -in certfile.pem -noout -issuer -issuer_hash
```

To check hash value of a certificate

```
$ openssl x509 -noout -hash -in ilugc_cert.pem
```

To Convert DER (Distinguished Encoding Rules) to PEM (Privacy Enhanced Mail) format

```
$ openssl x509 –inform der –in sslcert.der –out sslcert.pem
```

To convert PEM(Privacy Enhanced Mail) to DER(Distinguished Encoding Rules) format

```
$ openssl x509 –outform der –in sslcert.pem –out sslcert.der
```

To create CSR using an existing private key

```
$ openssl req –out certificate.csr –key existing.key –new
```

To test SSL certificate of particular url

```
$ openssl s_client -connect myurl.com:443 –showcerts
```

To check PEM file certificate expiration date

```
$ openssl x509 -noout -in certificate.pem -dates
```

To check certificate expiration date of SSL url

```
$ openssl s_client -connect myurl.com:443 2>/dev/null | openssl x509 -noout –enddate
```

● **243 od**

od - dump files in octal and other formats

```
$ cat file.txt
```
This is test message1

This is test message2

This is test message3

To print file.txt file content in octal format
```
$ od -b file.txt
```

To print file.txt file content in character format
```
$ od -c file.txt
```

To display files in hexadecimal bytes format
```
$ od -t x1 file.txt
```

To print in character format but with no offset information
```
$ od -An -c file.txt
```

To print with customize the width of hexadecimal format
```
$ od -w1 -c -Ad file.txt
```

To display the result as a decimal integer
```
$ od -i file.txt
```

To display the result as octal 2-byte units
```
$ od -o file.txt
```

To print the result as hexadecimal 2-byte units

```
$ od -x file.txt
```

To print help

```
$ od --help
```

To accept input from the command line.

```
$ od -c -
```

ilugc

give the input then ENTER , then Ctrl+d gives the od output

350

● 244 banner

banner - print large banner

To install banner

$ sudo apt install sysvbanner

syntax

$ banner text

$ banner 12345

$ banner ilugc

$ banner klug

By default it prints in Upper case letters only

It will print only alphanumeric not special characters.

● <u>245 parted</u>

parted - is a program to manipulate disk partitions

To list linux disk partitions

$ sudo parted

(parted) print

or

$ sudo parted -l

To select different hard disk with parted

(parted) select /disk_name

(parted) select /dev/vda

To create a primary partition

$ sudo parted /dev/sda mkpart primary ext4 start end

To create a logical partition

(parted) mkpart

partition type: extended

start ?

end ?

or

$ sudo parted /dev/sda mkpart extended start end

To resize disk partition

(parted) resizepart partition_number end

To change the FLAG on partition

(parted) set partition_number flag state

To toggle the state of FLAG on partition

(parted) toggle

To delete the partition

(parted) rm partition_number

or

$ sudo parted /disk/name rm partition_number

To create a partition without knowing disk size

$ sudo parted /dev/sda mkpart primary 10000 100%

To set the flag on partition

$ sudo parted /dev/sda set partition_number Flag State

To rescue a lost partition

(parted) rescue

or

(parted) rescue start end

To set the name of partition

(parted) name

or

(parted) partition_number name

● **246 partprobe**

partprobe - is a program that informs the operating system kernel
of partition table changes

To reload partition table in linux

$ sudo partprobe <device_name>

$ sudo partprobe /dev/sdc

To show a summary of devices and their partitions

$ sudo partprobe -s

To make a dry run and not to update the kernel

$ sudo partprobe --dry-run

or

$ sudo partprobe -d

To show a summary of devices and their partitions but don't notify
the kernel

$ sudo partprobe -sd

● **247 partx**

partx - tell the kernel about the presence and numbering of on-disk partitions

To list the partition table of disk

$ sudo partx --show <device_name>

$ sudo partx --show /dev/vda

To lists the length in sectors and human-readable size

$ sudo partx -o SECTORS,SIZE /dev/vda

To remove the last partition on /dev/sdd

$ sudo partx -d --nr :-1 /dev/sdd

To Print the SIZE column in bytes

$ sudo partx -b /dev/vda

To add the specified partitions, 3 to 5 (inclusive) on /dev/sdc

$ sudo partx -a --nr 3:5 /dev/sdc

List the partitions using the raw output format

$ sudo partx -r /dev/vda

List supported partition types and exit

$ sudo partx --list-types /dev/vda

To update the specified partitions

$ sudo partx -u /dev/vda

● **248 passwd**

passwd - change user password

$ passwd [options] [username]

To change system user's password
$ passwd

To change password for root
$ sudo passwd root

To display user status Information
$ sudo passwd -S ilugc

To display information of all users
$ sudo passwd -Sa

To delete user's password
$ sudo -d ilugc

To force expire the password to the user , force the user to change the password in the next login
$ sudo passwd -e ilugc

To lock a user password
$ sudo passwd -l ilugc
to check
$ sudo passwd -S ilugc

To unlock user password

```
$ sudo passwd -u ilugc
```

To set Inactive days after password expiry

```
$ sudo -i 10 ilugc
```

to check

```
$ sudo passwd -S ilugc
```

To force system users to change its password in 100 number of days

```
$ sudo passwd -n 100 ilugc
```

To set warning days before password expiry

```
$ sudo passwd -w 15 ilugc
```

to check

```
$ sudo passwd -S ilugc
```

● **249 paste**

paste - used to join files horizontally (parallel merging) by outputting lines consisting of lines from each file specified, separated by tab as delimiter, to the standard output

$ cat name

apache

nginx

mysql

ftp

jenkins

$ server

webserver

webserver

db server

file server

integration server

To merge the files in parallel with default delimiter as tab

$ paste name server

To merge files with delimiter as any character

$ paste -d "|" name server

$ paste -d "\n" name server

To paste one file at a time instead of in parallel

```
$ paste -s name server
$ paste -s name
$ paste -s server
```

To merge the contents in a column

```
$ paste - - - < name
$ paste - - < server
```

To specify a delimiter for sequential merging of files

```
$ paste -s -d ":" name server
```

To merge N consecutive lines from a file into a single line
with 2 hyphens

```
$ cat name | paste - -
```

with 3 hyphens

```
$ cat name | paste - - -
```

● 250 patch

patch - is used for adding patch files to source code or text files. It takes input as a patch file and applies differences to original files

```
$ cat first.py
#!/usr/bin/python3
print ("hello ILUGC")
```

copy the content of first.py in the new_first.py
```
$ cp first.py new_first.py
```

do some changes in the newly-created file new_first.py
```
$ vim new_first.py
#!/usr/bin/python3
print ("hello ILUGC")
print ("hello KLUG")
:x
```

Check Difference
create a patch file named as myfile.patch
```
$ touch myfile.patch
```

```
$ diff -u first.py new_first.py >myfile.patch
$ cat myfile.patch
```

To apply patch
```
$ patch < myfile.patch
```

To take backup before applying patch

```
$  patch -b < myfile.patch
```

To set backup file version

```
$ patch -b -V numbered < myfile.patch
```

To make a dry run

```
$ patch --dry-run < myfile.patch
```

To reverse a patch that is already applied

```
$ patch < myfile.patch
$ ls -l first.py
$ cat first.py
$ patch -R < myfile.patch
```

to check

```
$ ls -l first.py
$ cat first.py
```

● **251 pdf2ps**

pdf2ps - Ghostscript PDF to PostScript translator

```
$ pdf2ps options input.pdf output.ps
$ pdf2ps file.pdf file.ps
```

● **252 pdffonts**

Pdffonts - lists the fonts used in a Portable Document Format (PDF) file
along with various information for each font

$ pdffonts input.pdf

$ pdffonts file.pdf

To find out which fonts are used in range of pages of a document
for example pages 10-20 pages
$ pdffonts -f 10 -l 20 file.pdf

To extract fonts from a password-protected PDF
$ pdffonts -opw password
$ pdffonts -upw password

-opw - owner password
-upw - user password

● <u>253 pdfinfo</u>

pdfinfo - Portable Document Format (PDF) document information extractor

To find info of pdf file

```
$ pdfinfo file.pdf
```

To print metadata of pdf file

```
$ pdfinfo -meta file.pdf
```

To print the page bounding boxes

```
$ pdfinfo -box file.pdf
```

To list available encodings

```
$ pdfinfo -listenc file.pdf
```

To print all JavaScript in the PDF

```
$ pdfinfo -js file.pdf
```

To print the undecoded date strings directly from the PDF file

```
$ pdfinfo -rawdates file.pdf
```

● <u>254 pdftotext</u>

pdftotext - Portable Document Format (PDF) to text converter

syntax
```
$ pdftotext PDF-file text-file
```

To convert file.pdf to file.txt
```
$ pdftotext file.pdf file.txt
```

To convert range of pages(10-20) from file.pdf to file.txt
```
$ pdftotext -f 10 -l 20 file.pdf file.txt
```

To convert a pdf file protected and encrypted by owner password
```
$ pdftotext -opw 'password' file.pdf file.txt
```

To convert a pdf file protected and encrypted by user password
```
$ pdftotext -upw 'password' file.pdf file.txt
```

To generate a simple HTML file, including the meta information
```
$ pdftotext -htmlmeta file.pdf file.html
```

● <u>255 pgrep</u>

pgrep - look up or signal processes based on name and other attributes

syntax

```
$ pgrep [OPTIONS] <PATTERN>
```

To find the PID of the SSH

```
$ pgrep sshd
```

To print the PID with delimiter for sshd process

```
$ pgrep -d "|" sshd
$ pgrep -d ":" sshd
```

To use a space as a delimiter

```
$ pgrep ssh -d' '
```

To show the process name along with its ID

```
$ pgrep ssh -l
```

To list PID and full command line

```
$ pgrep -a sshd
```

To list all ThreadID

```
$ pgrep -w sshd
```

To match case insensitively

```
$ pgrep -i sshd
```

To print count of matching processes

```
$ pgrep -c sshd
```

To display most recently started process id

```
$ pgrep -n sshd
```

To display least recently started

```
$ pgrep -o sshd
```

To match only child processes of the given parent

```
$ pgrep -P <PID_sshd>
```

To match exactly with the command name

```
$ pgrep -x sshd
```

To print help

```
$ pgrep --help
```

● <u>256 pidgin</u>

pidgin - start Instant Messaging client in command line

To start pidgin instant Messaging client
$ pidgin

To print debugging messages to stdout
$ pidgin -d

To force online, regardless of network status
$ pidgin -f

To not automatically login
$ pidgin -n

To enable specified account
$ pidgin -l

To display this help
$ pidgin -h

● <u>257 pidof</u>

pidof - find the process ID of a running program

syntax

```
$ pidof [OPTIONS] PROGRAM_NAME
```


To find the PID of the SSH

```
$ pidof sshd
```


To force pidof to display only one PID

```
$ pidof -s sshd
```


To return only the PIDs of the processes that are running with the same root directory

```
# pidof -c pid sshd
```


To print PIDs of shells running scripts with a matching name

```
$ pidof -x sshd
```


To List zombie and I/O waiting processes

```
$ pidof -z sshd
```

● **258 ping**

ping - send ICMP ECHO_REQUEST to network hosts

syntax

```
$ ping [option] [hostname] or [IP address]
```

To check whether a remote host is up

```
$ ping google.com
$ ping ilugc.in
```

To request IPv6

```
$ ping -6 hostname/IPv6
```

To request IPv4

```
$ ping -4 hostname/IPv4
```

To change time Interval between ping packets

```
$ ping -i 0.5 ilugc.in
$ ping -i 5 ilugc.in
```

To change ping packet size

```
$ ping -s 1000 google.com
$ ping -s 512 google.com
```

To flood network using ping to test performance

```
$ sudo ping -f hostname-IP
```

To limit the number of pings

```
$ ping -c 5 google.com
$ ping -c 10 ilugc.in
```

To set time limit for ping command

```
$ ping -w 10 google.com
$ ping -w 20 ilugc.in
```

To print only summary statistics

```
$ ping -c 5 -q google.com
$ ping -c 10 -q ilugc.in
```

To add timestamp before each line in ping output

```
$ ping -D google.com
```

● **259 pip , pip3**

pip - A tool for installing and managing Python packages

To print version of pip3

```
$ pip3 --version
```

To upgrade pip3

```
$ python -m pip3 install --upgrade pip
```

To downgrade pip3

```
$ python -m pip3 install pip==19.0
```

To install a Python package

```
$ pip3 install <package_name>
```

To install the package of a specific version

```
$ pip3 install package_name==version
```

To display package information

```
$ pip3 show <package_name>
$ pip3 show numpy
```

To list of locally installed Python modules

```
$ pip3 list
```

To uninstall packages

```
$ pip3 uninstall <package_name>
$ pip3 uninstall numpy
```

To search packages

```
$ pip3 search <package_name>
$ pip3 search numpy
```

Install packages from requirements.txt

```
$ vim requirements.txt
numpy
botocore
future
:x
```

```
$ pip3 install requirements.txt
```

To list packages that don't come pre-installed with Python

```
$ pip3 freeze
```

To upgrade packages

```
$ pip3 install --user --upgrade package_name
$ pip3 install --user --upgrade numpy
```

To Downgrade packages

```
$ pip3 install --user package_name==version
$ pip3 install --user pip install numpy==1.22.2
```

To check that installed packages are compatible

```
$ pip3 check
```

To manage local and global configuration

```
$ pip3 config list
```

```
$ pip3 config edit
```

```
$ pip3 config get
```

```
$ pip3 config set
```

```
$ pip3 config unset
```

To Install package from a Git repository

```
$ pip3 install git+https://github.com/psf/requests.git
```

To Install package from a directory

```
$ pip3 install /home/user/src/requests
```

To Download a package and all of its dependencies

```
$ pip3  download <package_name>
```

To debug

```
$ python3 -m pip debug
```

● **260 pkcon**

pkcon - is the command line client for PackageKit

To search for a package type

```
$ sudo pkcon search <characters to be searched for>
```

To install a package type

```
$ sudo pkcon install <package to be installed>
```

To Install a downloaded package using pkcon

```
$ sudo pkcon install-local <package to be installed>
```

To remove a package using pkcon

```
$ sudo pkcon remove <package to be removed>
```

To refresh the package cache of pkcon

```
$ sudo pkcon refresh
```

To update packages with pkcon

```
$ sudo pkcon update
```

To List all available packages

```
$ pkcon get-packages
```

To List all configured package repositories.

```
$ pkcon repo-list
```

To List available updates

```
$ pkcon get-updates
```

To List the available filters.

$ pkcon get-filters

To List the available package groups

$ pkcon get-groups

To List the roles that a transaction can have

$ pkcon get-roles

To Print information about the PackageKit backend in use

$ pkcon backend-details

● **261 pkexec**

pkexec - Execute a command as another user

$ pkexec <command>

$ pkexec pwd

$ pkexec ls

$ pkexec df

To run the command as some other user

$ pkexec --user <username> <command>

● **262 pkg-config**

pkg-config - Return metainformation about installed libraries

To Print the Link Flags

```
$ pkg-config openssl --libs
```


To Print Compile Flags

```
$ pkg-config openssl --cflags
```


To get a version of the library

```
$ pkg-config openssl --modversion
```


To print errors

```
$ pkg-config openssl --print-errors
```


To display variables in a package

```
$ pkg-config --print-variables openssl
```


To list packages

```
$ pkg-config --list-all
```


To get the value of a variable declared in a package's .pc file

```
$ pkg-config --variable=libdir openssl
```


To get help messages

```
$ pkg-config --help
```

● **263 grpck**

grpck - verify integrity of group files

$ grpck [option] [files]

To verify the group account file
grpck /etc/group

To verify the shadow file
grpck /etc/gshadow

Exit Codes
0: Success.
1: Syntax error.
2: One or more bad group entries found.
3: Could not open group files.
4: Could not lock group files.
5: Could not write group files.

● **264 pkill**

pkill - is used to kill the current or running process on the environment

syntax

```
$ pkill [OPTIONS] <PATTERN>
```

To stop the process gracefully

```
$ pkill -15 docker
$ pkill -15 firefox
```

To reload any "X" process

```
$ pkill -HUP X
```

To Kill the Process Starts and Ends With Specific Expression

```
$ pkill '^ssh$'
```

To Kill Process Based on Full Command

```
$ pkill ping
$ pkill -9 -f "ping google.com"
```

To send a different signal to kill a process

```
$ pkill --signal SIGKILL ping
```

To make the pkill case insensitive

```
$ pkill -i [process-name]
$ pkill -i PING
```

To kill match the processes being run by a specific user

```
$  pkill -u user1
```

To kill match the processes being run by multiuser

```
$ pkill -u  user1, user2, user3
```

To send KILL signals to all processes under the user1 and patterns matching X process

```
$ pkill -9 -u user1 X
```

To kill only the oldest (least recently started) of the matching processes

```
$ pkill -9 -o chrome
```

To kill only the newest (most recently started) of the matching processes

```
$ pkill -9 -n chrome
```

● <u>265 pmap</u>

pmap - report memory map of a process

syntax

$ pmap [options] pid [...]

To display the memory map of chrome process

$ pidof chrome

$ pmap <pid_chrome>

To display the memory map in an extended format

$ pmap -x <pid_chrome>

To display the full path to the files

$ pmap -p <pid_chrome>

To display the device format

$ pmap -d <pid_chrome>

To ignore the column names while displaying the report of the memory map

$ pmap -q -d <pid_chrome>

To display everything the kernel provides

$ pmap -xx <pid_chrome>

To create a new configuration

$ pmap -n

To read the default configuration

```
$ pmap -c <pid_chrome>
```

To print in quiet mode and to hide header and footer lines

```
$ pmap -q <pid_chrome>
```

To display pmap of multiple processes

```
$ pmap <PID1> <PID2> <PID3>
```

● **266 popd**

popd - is used to remove directories from the directory stack

syntax

$ popd [OPTIONS] [DIRECTORY]

$ dirs -l -v
 0 /home/ilugc/Templates
 1 /home/ilugc/Pictures
 2 /home/ilugc/Videos
 3 /home/ilugc/Music
 4 /home/ilugc/Downloads
 5 /home/ilugc/Documents
 6 /home/ilugc/Desktop
 7 /home/ilugc

To delete directories in the directory stack

$ popd

To Delete a directory from the stack without changing the current directory

$ popd -n

To remove a directory from any position

+N is used, the Nth directory is deleted from the top

$ popd +N

$ popd +1

-N is used, the Nth directory is deleted from the bottom

$ popd -N

$ popd -1

where N is numerical parameter

● 267 poweroff

poweroff - Instructs the system to power down.

To Power off the system

```
$ sudo poweroff
```

To Halt the system

```
$ sudo poweroff --halt
```

To Reboot the system

```
$ sudo poweroff --reboot
```

● **268 pr**

pr - is used to prepare a file for printing by adding suitable footers, headers, and the formatted text.

Syntax:

$ pr [options][filename]

To print k number of columns

$ pr -k file.txt

To print 2 number of columns

$ pr -2 file.txt

To print 3 number of columns

$ pr -3 file.txt

To suppress the headers and footers

$ pr -t file.txt

To Double the paces input, reduces clutter

$ pr -d file.txt

To provide number lines which helps in debugging the code

$ pr -n file.txt

To omit page headers and trailers, eliminate any pagination

 $ pr -T file.txt

To print help

```
$ pr -h
```

● **269 printf**

printf - format and print data

Syntax

```
$ printf [-v var] format [arguments]
```

To print the message

```
$ printf "%s\n" "Hello, ILUGC"
```

To print the string value

```
$ printf "%s" "Hello, ILUGC"
```

To display output with new line

```
$ printf "Welcome to ILUGC \n"
```

To Print integer values

```
$ printf "%d\n" "1234567890"
```

To Print float values

```
$ printf "%f\n" "10.25"
```

To Print environmental variable

```
$ printf "The shell environment is: ""$SHELL \n"
```

To print date and time

```
$ printf "%(%D-%m-%Y %H:%M)T" $(date +%s)
```

● **270 printenv**

printenv - print all or part of environment

To print HOME variable value

$ printenv HOME

To Display all variables

$ printenv

To display the values of SHELL environment variables

$ printenv SHELL

$ printenv LANG

- ## 271 prtstat

prtstat - print statistics of a process

To print the statistics of sshd

$ pidof sshd

$ prtstat -r <PID_sshd>

● <u>272 ps</u>

ps - report a snapshot of the current processes.

To display processes for the current shell

```
$ ps
```

To print all processes in different formats

```
$ ps -A
```

To Display processes in BSD format

```
$ ps aux
```

To display full-format listing

```
$ ps -ef
```

To print user running processes

```
$ ps -x
```

To print user processes by real user ID or name

```
$ ps -fU ilugc
```

To display user processes by effective user ID or name

```
$ ps -fu ilugc
```

To display all processes running as root

```
$ ps -U root -u root
```

To print group processes

```
$ ps -fG docker
```

To display all processes owned by effective group name

```
$ ps -fg docker
```

To print processes by PID

```
$ ps -fp <PID>
```

To list process by PPID

```
$ ps -f --ppid <PPID>
```

To list process using a PID list

```
$ ps -fp PID1, PID2, PID3
```

To display processes by TTY

```
$ ps -t pts/0
$ ps -ft tty1
```

To print process tree

```
$ ps -e --forest
```

To print a process tree for a given process

```
$ ps -f --forest -C sshd
```

To print all threads of a process

```
$ ps -fL -C sshd
```

To list all format specifiers

```
$ ps L
```

To display the PID, PPID, user name, and command of a process.

```
$ ps -eo pid,ppid,user,cmd
```

To display file system group, nice value, start time, and elapsed time of a process.

```
$ ps -p 1154 -o pid,ppid,fgroup,ni,lstart,etime
```

To find a process name using its PID.

```
$ ps -p <PID> -o comm=
```

To display parent and child processes

```
$ ps -C sshd
```

To print all PIDs of all instances of a process

```
$ ps -C sshd -o pid=
```

To check the execution time of a process

```
$ ps -eo comm,etime,user | grep sshd
```

To print top running processes by highest memory

```
$ ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head
```

To print top running processes by highest cpu

```
$ ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%cpu | head
```

To display security context

```
$ ps -eM
$ ps --context
```

To display security information in a user-defined format

```
$ ps -eo  euser,ruser,suser,fuser,f,comm,label
```

● <u>273 pstree</u>

pstree - is used to display the parent-child relationship in a hierarchical format

To print pstree without any option

$ pstree

To Display the tree hierarchy of a user processes

$ pstree -p ilugc

To display the process tree

$ pstree <PID>

To show the command line arguments

$ pstree -a

To Use ASCII characters to draw the tree

$ pstree -A

To Disable compaction of identical subtrees

$ pstree -c

To Use VT100 line drawing characters

$ pstree -G

To print the current process and its ancestors

$ pstree -h

To print the specified process instead

$ pstree -H

To Display long lines

$ pstree -l

To Sort processes with the same ancestor by PID instead of by name

$ pstree -n

To show PIDs

$ pstree -p

To show the uid transactions

$ pstree -u

To Use UTF-8 (Unicode) line drawing characters

$ pstree -U

To show the security context

$ pstree -Z

● **274 ps2pdf**

ps2pdf - Convert PostScript to PDF using ghostscript

```
$ ps2pdf sample.ps
$ ps2pdf file.ps file.pdf
```

To embed fonts
```
$ ps2pdf -dEmbedAllFonts=true sample.ps
```

To compress the PDF
```
$ ps2pdf -dUseFlateCompression=true sample.ps
```

● **275 pvck**

pvck - Check metadata on physical volumes

syntax
$ sudo pvck [options]

To scan the physical volume metadata
$ sudo pvck PhysicalVolume /dev/sda3

To specify the starting sector from where it should scan
$ sudo pvck --labelsector sector

To debug the physical volume metadata
$ sudo pvck -d

To print help
$ pvck -h

To operate in verbose mode
$ sudo pvck -v

● **276 pwconv**

pwconv - used to recover shadow file from passwd file , if shadow file is deleted by any means replace the encrypted password in /etc/shadow with an x.

from root user delete /etc/shadow

rm /etc/shadow

check with

cat /etc/shadow

To recover /etc/shadow file from the /etc/passwd file

pwconv

and check with

cat /etc/shadow

● 277 pwd

pwd - print name of current/working directory

To get working directory path

```
$ pwd
```

To print the physical working directory and avoid listing symbolic links it prints the actual path.

```
$ pwd -P
```

To print the working directory path, including any symlinks

it prints the symbolic path.

```
$ pwd -L
```

To print $PWD Variable Contents

```
$ echo $PWD
```

● **278 pwdx**

pwdx - report current working directory of a process

syntax

```
$ pwdx pid
$ pwdx pid1 pid2 pid3
```

for example change directory to /opt

```
$ cd /opt
$ sleep 100
```
check for pidof sleep
```
$ pidof sleep
```
66135

then check for dir with
```
$ pwdx <pid_sleep>
$ pwdx 66135
```
66135: /opt

● **279 qemu-img**

qemu-img - used to create, convert and modify images offline by Xen and KVM

syntax

```
$ qemu-img subcommand [options]
```

To get help

```
$ qemu-img -h
```

To create disk image

```
$ qemu-img create ubuntu.img 25G
```

To create qcow2 disk image

```
$ qemu-img create -f qcow2 -o size=25G ubuntu.img
```

To create VMDK disk image

```
$ qemu-img create -f vmdk -o size=20G debian.img
```

To get Information about disk image

```
$ qemu-img info  ubuntu.img
$ qemu-img info  debian.img
$ qemu-img info  fedora.img
```

To shrink disk image

```
$ qemu-img convert -O qcow2  centos.qcow2 centos_shrink.qcow2
```

To compress disk image

```
$ qemu-img convert -O qcow2 -c  fedora.qcow2 fedora_compress.qcow2
```

To check disk image for errors

```
$ qemu-img check ubuntu.qcow2
$ qemu-img check debian.qcow2
$ qemu-img check centos.qcow2
```

To increase disk image size

```
$ qemu-img resize ubuntu.qcow2 +5GB
$ qemu-img resize debian.qcow2 +5GB
```

To create a new disk image on the file system.

```
$ qemu-img create -f raw ubuntu.img 25G
$ qemu-img create -f vmdk ubuntu.vmdk 25G
```

To converts an existing disk image from one format to another

```
$ qemu-img convert -O qcow2 ubuntu.vmdk ubuntu.qcow2
$ qemu-img convert -O qcow2 debian.vmdk debian.qcow2
```

To manage snapshots of an existing disk image

```
$ qemu-img snapshot -c ubuntu_snap1 ubuntu.qcow2
$ qemu-img snapshot -c centos_snap1 centos.qcow2
```

To list snapshots of VM

```
$ qemu-img snapshot -l ubuntu.qcow2
$ qemu-img snapshot -l centos.qcow2
```

To restore the state of the saved snapshot

```
$ qemu-img snapshot -a 1  ubuntu.qcow2
$ qemu-img snapshot -a 1  centos.qcow2
```

To delete snapshot

```
$ qemu-img snapshot -d 1 ubuntu.qcow2
```

```
$ qemu-img snapshot -d 1 centos.qcow2
```

To create a new base image based on an existing disk image

```
$ qemu-img rebase -b ubuntu.raw ubuntu.qcow2
```

```
$ qemu-img rebase -b centos.raw centos.qcow2
```

● **280 rcp**

rcp - is used to copy files from one networked computer to another

syntax

$ rcp options source destination

To send a file from local host to remote host

$ rcp /home/ilugc/file.txt remotehost:/home/remote_home/file.txt

To receive a file from a remote host

$ rcp remote_host:/home/remote_home/file.txt .

To have the modification times, access times, modes and ACLs if applicable as the original file

$ rcp -p remote_host:/home/remote_home/file.txt

To copy directories

$ rcp -r localdir remote_host:

To copy two files from local host to remote host

$ rcp a.txt b.txt c.txt remote_host:/var/www/

● __281 renice__

renice - alter priority of running processes

To change the priority of the running process.

```
$ sudo renice -n <nice_value>  -p <pid_of_the_process>
$ renice -n 10 -p <PID>
```

To change the priority of all programs of a specific group with group id 5 to 15

```
$ renice -n 15 -g 5
```

To change the priority of all programs of a specific user 3 to 15

```
$ sudo renice -n 15 -u 3
```

● [282 reboot](#)

reboot - is used restart or reboot the system

syntax

```
$ reboot [OPTIONS...]
```

To restart system

```
$ sudo reboot
$ sudo shutdown -r now
```

To scheduled a restart after a specific time ex. 10 minutes

```
$ sudo shutdown -r +10
```

reboot system after 06:00 A.M

```
$ sudo shutdown -r 06:00
```

To cancel restart

```
$ sudo shutdown -c "message"
$ sudo shutdown -c "scheduled shutdown is cancelled"
```

To restart remote server

```
$ ssh root@remote-server /sbin/reboot
$ ssh root@192.168.122.10 /sbin/shutdown -r now
```

To print help options

```
$ reboot --help
```

To force immediate reboot

```
$ sudo reboot -f
```

To just write wtmp record and not reboot

$ sudo reboot -w

To restart with the Init command

$ sudo init 6

To don't write wtmp record

$ sudo reboot -d

To don't send wall message before reboot

$ sudo reboot --no-wall

To reboot system using systemctl

$ sudo systemctl reboot

● **283 realpath**

realpath - print the resolved path

syntax
$ realpath /path/to/file

$ ls -l /etc/os-release
$ realpath /etc/os-release

To display each output line with NUL, not newline
$ realpath -z /etc/os-release

To suppress most error messages
$ realpath -q /etc/os-release

To print the resolved path relative to DIR
$ realpath --relative-to=DIR /etc/os-release

To print absolute paths unless paths below DIR
$ realpath --relative-base=DIR /etc/os-release

To resolve symlinks as encountered
$ realpath -P /etc/os-release

To resolve '..' components before symlinks
$ realpath -L /etc/os-release

- ## **284 rev**

rev - reverse lines characterwise

$ vim file.txt

This is sample test file

:x

$ rev file.txt

$ rev

linux

foss

ilugc

$ echo This is sample file | rev

● **285 replace**

replace - makes modifications to strings of text in files or the standard input.replace command is provided by mariadb-server

```
$ cat example.txt
```
This is paragraph one

This is paragraph two

This is paragraph three

To replace string with other string
```
$ replace paragraph line -- example.txt
```
it will be converted and renamed

To print output to stdout
```
$ replace paragraph line < example.txt
```


To save output to a new file name
```
$ replace paragraph line < example.txt > new_example.txt
```


To Print more information about what the program does
```
$ replace -v line paragraph -- example.txt
$ replace -v line paragraph < example.txt > new_example.txt
```

- ## **286 reset**

reset - is used to initialize the terminal

To wipe everything that is currently in the terminal, including the scrollback buffer and initialize the terminal

$ reset

● **287 resize2fs**

resize2fs - is used to enlarge or shrink an ext2/3/4 file system on a device

syntax

$ resize2fs [options] {device/file system name} [desired size]

To resize a extended file system

$ sudo resize2fs /dev/vda1

To forcefully resize the file system

$ sudo resize2fs -f /dev/vda1

To flush the filesystem device's buffer caches

$ sudo resize2fs -F /dev/vda1

To shrink the filesystem to the minimum size

$ sudo resize2fs -M /dev/vda1

To prints out a percentage completion bars for each resize2fs operation

$ sudo resize2fs -p /dev/vda1

To print the minimum size of the filesystem

$ sudo resize2fs -P /dev/vda1

● **288 resizepart**

resizepart - tell the kernel about the new size of a partition

```
$ sudo parted
(parted) resizepart
Partition number? 2
End?  [20.0GB]? 30000
```

To check the results

```
(parted) print
```

- ## **289 return**

return - is used to exit from a shell function

syntax

$ return [N]

$ function add { add=$(($1+$2)); return $add; }

$ add 4 4

$ echo $?

● **290 rfkill**

rfkill - tool for enabling and disabling wireless devices

To list all the available wireless interfaces on a system
```
$ sudo rfkill
$ sudo rfkill list
```

To print ID, TYPE-DESC, SOFT and HARD columns in the output
```
$ sudo rfkill -o ID,TYPE-DESC,SOFT,HARD
```

To print JSON-formatted output
```
$ sudo rfkill -o ID,TYPE-DESC,SOFT,HARD -J
$ sudo rfkill -J
```

To Block an interface by ID
```
$ sudo rfkill block 0
$ sudo rfkill block 1
```

To Block interfaces by type
```
$ sudo rfkill block bluetooth
```

To unblocking interfaces
```
$ sudo rfkill unblock 0
```

To unblock all bluetooth devices
```
$ sudo rfkill unblock bluetooth
```

To toggle the status of an interface

```
$ sudo rfkill toggle <interface_id>
$ sudo rfkill toggle 0
```

● **291 rlogin**

rlogin - remotely logs in to a system.

To login remote host
$ rlogin 192.168.122.55

To Specify the user login name remote host
$ rlogin <remote_host> -l <username>
$ rlogin 192.168.122.55 -l ilugc

● <u>292 rm</u>

rm - remove files or directories

To remove or delete file
```
$ rm file.txt
```

To delete the files interactively
```
$ rm -i file.txt
```

To delete a directory recursively
```
$ rm -r old_data/
```

To delete the files and sub-directories interactively
```
$ rm -ir old_data/
```

To Delete files forcefully
```
$ rm -f file.txt
```

To prompt once before deleting more than three files or recursive delete
```
$ touch file1.txt file2.txt file3.txt file4.txt file5.txt
$ rm -I file*
```
rm: remove 5 arguments? y

To delete all the .txt files or .mp4
```
$ rm -f *.txt
$ rm -f *.mp4
```

To remove multiple files

```
$ rm file1.txt file2.txt file3.txt
```

To remove everything from current directory

```
$ rm -v *
```

● **293 rmdir**

rmdir - remove empty directories

To remove a single empty directory

$ rmdir ~/Downloads/files

To remove multiple directories using rmdir

$ rmdir ~/Downloads/old_data1 old_data2 old_data3

To print verbose output

$ rmdir -v ~/Downloads/files

To suppress fail on non-empty message

$ rmdir -v --ignore-fail-on-non-empty old_data1 old_data2 old_data3

To remove directory and its parent directories

$ rmdir -v -p files/data/project/

To remove multiple directories with dir-

$ rmdir -v dir-*

● **294 rmmod**

rmmod - Simple program to remove a module from the Linux Kernel

syntax

$ rmmod [options] module_name

To remove a module

$ rmmod bluetooth

$ rmmod ath10k_core

To delete multiple modules

$ rmmod module_1 module_2 module_3

$ rmmod bluetooth ath10k_core

To print verbose output

$ rmmod -v bluetooth

To send errors to syslog instead of standard error

$ rmmod -s bluetooth

● [295 route](#)

route - show / manipulate the IP routing table

To display the IP/kernel routing table.

`$ route`

To display routing table in full numeric form

`$ route -n`

To add a default gateway

`$ sudo route add default gw 192.168.122.1`

To list kernel's routing cache information

`$ route -Cn`

To reject routing to a particular host or network

`$ sudo route add -host 192.168.122.101 reject`

To get details of the kernel/IP routing table using ip command

`$ ip route`

To delete the default gateway

`$ route del default`

To get the details of the local table with destination addresses assigned to the localhost

`$ ip route show table local`

To get output related to IPv4

```
$ ip -4 route
```

To get output related to IPv6.

```
$ ip -6 route
```

● **296 rsh**

rsh command executes commands on a remote shell.

syntax

$ rsh remote-machinename/ip command

To run a command on remote-host

$ rsh remote-host ls

$ rsh remote-machine/ip mkdir ~/Desktop/testfolder

$ rsh remote-machine/ip mv -v ~/Desktop/*.txt
~/Desktop/tesffolder/

$ rsh 192.168.122.50 hostname

To run command on remote-host as different user

$ rsh -l user2 192.168.122.50 whoami

To run multiple commands in the remote computer

$ rsh -l user2 192.168.122.50 "pwd ; ls"

To run commands with sudo privileges on the remote system

$ rsh -l user2 192.168.122.50 sudo -S deluser user3

To run local scripts on the remote system

$ rsh -l user2 192.168.122.50 bash test-script.sh

To save the remote system's output to the local system

$ rsh -l user2 192.168.122.50 ps > running_process.txt

● 297 rsync

rsync - a fast, versatile, remote (and local) file-copying tool

syntax

$ rsync options SOURCE DESTINATION

To copy a single file locally

$ rsync -v /home/ilugc/Desktop/file.txt
/home/ilugc/Documents/backup/

To copy multiple files locally

$ rsync -v /home/ilugc/Desktop/sample.txt
/home/ilugc/Desktop/sample2.txt /home/ilugc/Documents/backup

$ rsync -v file1.txt file2.txt /home/ilugc/Documents/backup

To copy a directory and all subdirectories locally (Copy Files and
Directories Recursively)

$ rsync -av /home/ilugc/Desktop/Linux /home/ilugc/Documents/backup

To copy a file or directory from local to remote machine

$ rsync -av /home/ilugc/Desktop/test
192.168.122.50:/home/ilugc/Documents/backup

$ rsync -av /home/ilugc/Desktop/test
user1@192.168.122.50:/home/ilugc/Documents/backup

$ rsync -av /home/ilugc/Desktop/file.txt
user1@192.168.122.50:/home/ilugc/Documents/backup

To copy multiple files or directories from local to remote machine

$ rsync -av /home/ilugc/Desktop/test/ /home/ilugc/Music
192.168.122.50:/home/ilugc/Desktop/backup

To specify rsync protocol for remote transfers

```
$ rsync -e ssh /home/ilugc/Desktop/sample.txt
192.168.122.50:/home/ilugc/Desktop
```

To copy a file or directory from a remote to a local machine current directory

```
$ rsync -av 192.168.122.50:/home/ilugc/Desktop/Test_Dir .
```

To show rsync progress during data transfer

```
$ rsync -av --progress /home/ilugc/Desktop/Test_Dir
192.168.122.50:/home/ilugc/Desktop/backup
```

To delete source files after transfer

```
$ rsync -v --remove-source-files /home/ilugc/backup/monthly.zip
192.168.122.50:/home/ilugc/Desktop/backup/
```

To make rsync dry run

```
$ rsync -av --dry-run --delete /home/ilugc/Desktop/Test_Dir1
192.168.122.50:/home/ilugc/Desktop/backup
```

To set maximum file size for transfer

```
$ rsync -av --max-size=1024k /home/ilugc/Desktop/Dir1
192.168.56.100:/home/ilugc/Desktop/backup/
```

To set minimum file size for transfer

```
$ rsync -av --min-size=50k /home/ilugc/Desktop/
192.168.122.50:/home/ilugc/Desktop/backup/
```

To set the maximum transfer speed to 100KB/s

```
$ rsync -av --bwlimit=100 --progress /home/ilugc/Desktop/Test_Dir1
192.168.122.50:/home/ilugc/Desktop/backup/
```

To copy specific file type

$ rsync -v /home/ilugc/Documents/*.txt /home/ilugc/Desktop/backup/

$ rsync -v /home/ilugc/Documents/*.mp4 /home/ilugc/Desktop/backup/

$ rsync -v /home/ilugc/Documents/*.pdf /home/ilugc/Desktop/backup/

To show the difference between the source and destination files

$ rsync -avi /home/ilugc/Desktop/Test_Dir1/
/home/ilugc/Desktop/backup/

● <u>298 rsyslogd</u>

rsyslogd - rsyslogd is used to log messages and it is based on syslogd

To start the rsyslog service

$ sudo rsyslogd

To suppress the warnings

$ sudo rsyslogd -w

To Turn on Debugging

$ sudo rsyslogd -d

To disable the DNS for remote messaging

$ sudo rsyslogd -x

To send UDP messages to all the targets

$ sudo rsyslogd -A

To make rsyslogd to listen to IPv4 addresses only

$ sudo rsyslogd -4

To make rsyslogd to listen to IPv6 addresses only

$ sudo rsyslogd -6

To selects the desired backward compatibility mode

$ sudo rsyslogd -c 4.2

To specify the alternative configuration file

```
$ sudo rsyslogd -f /etc/myconfigfile
```

To specify the hostnames to be logged

```
$ sudo rsyslogd -l remote_host.com
```

To specify the alternate pid file

```
$ sudo rsyslogd -i /var/
```

To do a config check

```
$ sudo rsyslogd -N 1
```

● **299 runlevel**

runlevel - Print previous and current SysV runlevel

0 - Halt

1 - Single-user mode

2 - Not used (user-definable)

3 - Full multi-user mode

4 - Not used (user-definable)

5 - Full multi-user mode (with an X-based login screen)

6 - Reboot

To see the current runlevel of the system

$ runlevel

N 3

Full multi-user mode in CLI mode

To temporarily change the runlevel to 5 (Full multi-user mode with an X-based login screen) from 3

$ init 5

To permanently change the runlevel to 5 from 3

$ sudo vim /etc/default/grub

GRUB_CMDLINE_LINUX="5"

:x

$ sudo update-grub

$ sudo reboot

● **300 runuser**

runuser - run a command with substitute user and group ID
runuser cannot be used by non-root users

Syntax:
runuser - username -c [commands...]

runuser - user1 -c 'mkdir -p ~/sample.txt'
runuser - user1 -c 'ls -l'
runuser - user1 -c 'df -Th'

To run multiple commands
runuser -user1 -c 'mkdir -p ~/sample.txt; ls -l; df -Th '

● **301 scp**

SCP (Secure Copy Protocol) - is a network protocol used to securely copy files/folders between Linux systems on a network

syntax:

$ scp [option] [user_name@source_host:path/to/source/file] [user_name@target_host:target/path]

To copy a file from local to remote server

$ scp example.txt root@remote_server:/home/user1/remote_dir

To copy a file from a remote server to the local host

$ scp remote_server:/home/remote_dir/example.txt home/user1/Documents

To copy a file from one remote server to another

$ scp root@remote_server1:/home/user1/remote_dir/example.txt root@remote_server2:/home/user2/Documents

To copy multiple files with scp

$ scp sample1.txt sample2.txt root@remote_server:/home/user1/remote_dir

To copy a folder from local host to remote server recursively

$ scp -r example_folder root@remote_server:/home/user1/remote_dir

To copy a file with scp using a specific port

$ scp -P 2222 sample_example.txt root@remote_server:/home/user1/remote_dir

To copy a file with scp in quiet mode

$ scp -q example.txt root@remote_server:/home/user1/remote_dir

To copy a file with scp in verbose mode

$ scp -v example.txt root@remote_server:/home/user1/remote_dir

To copy a file with scp and limit bandwidth for scp to 50 KB/s

$ scp -l 400 example.txt root@remote_server:/home/user1/remote_dir

To copy a file with scp faster

$ scp -C example.txt root@remote_server:/home/user1/remote_dir

To copy a file with scp preserving file attributes

$ scp -p example.txt root@remote_server:/home/user1/remote_dir

● **302 script**

script - is used to typescript or record all terminal processes.

syntax

```
$ script [option] [filename]
```

To start, type "script" without specifying any parameters

```
$ script
$ echo "This is ILUGC"
$ exit
```

it will create typescript file to save the recorded information

To start the typescript, run any random command and save it in a text file

```
$ script "ilugc.txt"
$ echo "This is klug"
$ date
$ time
$ cal
$ exit
```

now check with

```
$ vim ilugc.txt
```

To append the output, retaining the prior content of the file ilugc.txt

```
$ script -a ilugc.txt
$ free -h
$ df -Th
$ echo "This is test message"
$ exit
```

now check with

```
$ vim ilugc.txt
```

To Run the command rather than an interactive shell

```
$ script -c cal calender.txt
```

To run in quiet mode

```
$ script -q msg.txt
$ echo "Test message"
$ exit
$ vim msg.txt
```

To capture the terminal activity step by step and appears like a video

```
$ script --timing=time_log ilugc
$ echo "test message1"
$ echo "test message2"
$ pwd
$ ls
$ free
$ exit
```

To replay the terminal activity like video

```
$ scriptreplay --timing=time_log ilugc
```

To print help options

```
$ script --help
```

To run flush after each write

```
$ script -f sample.txt
$ pwd
$ exit
```

```
$ vim sample.txt
```

To save the script in some specific directory

```
$ script --force /home/ilugc/test.txt
```

● **303 sed**

sed - stream editor for filtering and transforming text

Basic text substitution using 'sed'

```
$ echo "indian linux user group" | sed 's/indian/kanchi/'
```

Replace all instances of a text in a particular line of a file using 'g' option

```
$ cat linux.txt
```

linux is a very popular os.

linux is easy to use. linux is easy to learn.

linux is a versatile os

To make all occurrences to change from linux to unix

```
$ sed 's/linux/unix/g' linux.txt
```

To replace words or characters with ignore character case

```
$ sed 's/linux/unix/gi' myfile.txt
```

To make the occurrences to change from linux to unix in line 2

```
$ sed '2 s/linux/unix/g' linux.txt
```

To Replace the second occurrence only of a match on each line

```
$ sed 's/linux/unix/g2' linux.txt
```

To Parenthesize first character of each word

```
$ echo "Welcome To Indian Linux User Group" | sed 's/\(\b[A-Z]\)/\(\1\)/g'
```

To Replace string on a range of lines

```
$ sed '1,2 s/linux/unix/' linux.txt
```

To Delete a particular line ex. 5th line

```
$ sed '5d' filename.txt
```

To Delete last line

```
$ sed '$d' filename.txt
```

To Delete line from range x to y

```
$ sed 'x,yd' filename.txt
$ sed '5,10d' filename.txt
```

To Delete from nth to last line

```
$ sed '12,$d' filename.txt
```

To Delete pattern matching line

```
$ sed '/pattern/d' filename.txt
$ sed '/abc/d' filename.txt
```

To View a range of lines of a document

```
$ sed -n '5,10p' myfile.txt
```

To view the entire file except a given range

```
$ sed '5,10d' myfile.txt
```

To Insert spaces in files

```
$ sed G linux.txt
```

To insert two blank lines

```
$ sed 'G;G' linux.txt
```

● **304 select**

select - is a very useful bash command for bash menu creation

syntax:

select v in data_list

do

statement1

Statement2

Statement3

done

Creating a simple menu

$ vim select.sh

```bash
#!/bin/bash
# Define the menu list here
select os in debian ubuntu fedora arch kali
do
echo "You have chosen $os"
done
```

$ bash select.sh

● <u>305 seq</u>

seq - print a sequence of numbers

syntax
```
$ seq [option] FIRST INCREMENT LAST
```

To display Seq LAST , default increment is 1
```
$ seq 20
```

To print with Seq FIRST and LAST
```
$ seq 5 20
```

To print Seq FIRST INCREMENT LAST
```
$ seq 4 2 50
```

To add the strings before the sequence number
```
$ seq -f NUM%02g 10
```

To print the formatted strings starting from 5 and incrementing by 5 up to 25
```
$ seq -f NUM%02g 5 5 25
```

To display the output on the same line
```
$ seq -s " " 5 5 50
```

To append the leading zeros to equalize the width
```
$ seq -ws " " 3 2 120
```

To display with separators

```
$ seq -s - 100
```

To generate a sequence containing the floating values

```
$ seq -f "%f" 2 0.5 10
```

● **306 service**

service - is used to control SysVinit services through SysVinit scripts

syntax

$ service [options] [service] [subcommand]

To check the status of service

$ sudo service rsyslog status

$ sudo service sshd status

To stop a service

$ sudo service rsyslog stop

$ sudo service sshd stop

To restart a service

$ sudo service rsyslog restart

$ sudo service sshd restart

$ sudo service vsftpd restart

To list the status for all the programs

$ sudo service --status-all

 To do a full restart

$ sudo service service_name --full-restart

● <u>307 set</u>

set - is a built-in Linux shell command that displays and sets the names and values of shell and Linux environment variables

syntax:

```
$ set [options] [arguments]
```

To list all settings

```
$ set
```

```
$ set debian redhat arch
```

To list all the parameters in the order of $1 $2 $3

```
$ echo "$*"
```

```
$ echo $1
```

```
$ echo $2
```

```
$ echo $3
```

To unset All Positional Parameters

```
$ set --
```

To print out an error on the shell if the script runs into an undefined variable

```
$ vim example1.sh
```

```
#!/bin/bash
```

```
set -u
```

```
echo $foo
```

```
echo "welcome to ilugc"
```

```
:x
```

```
$ bash example.sh
```

To display an Error If a Command Is Non-existent

$ vim example2.sh

```
#!/bin/bash
set -e
foo bar
echo "welcome to klug"
:x
```

$ bash example2.sh

To Display an Error in Piped Commands

$ vim example3.sh

```
#!/bin/bash
set -eo pipefail
foobar | echo "Hi welcome to chennaipy"
echo "welcome to ilugc"
:x
```

$ bash example3.sh

To set allexport and notify options

$ set -o allexport -o notify

● <u>308 setfacl</u>

setfacl - set file access control lists

syntax

```
$ setfacl option file
```

To modify ACLs of file to give read and write permission to user1

```
$ setfacl -m u:user1 rw file
```

To remove all extended ACL entries

```
$ setfacl -b file.txt
```

To remove entries from the ACL of file, To remove group 'ilugc' from a file's ACL

```
$ setfacl -x g:ilugc file.txt
```

To remove the default ACL

```
$ setfacl -k file
```

To apply operations to all files and directories recursively

```
$ setfacl -m g:ilugc:rw -R directory
```

To restore a permission backup

```
$ setfacl --restore=file
```

To copy the ACL of one file to another

```
$ getfacl example.txt | setfacl --set-file=- f sample.txt
```

● **309 setsid**

setsid - run a program in a new session

syntax

```
$ setsid [options] program [arguments]
```

```
$ cat add.sh
#!/bin/bash
# Take input from user and calculate sum.
read -p "Enter first number: " num1
read -p "Enter second number: " num2
sum=$(( $num1 + $num2 ))
echo "Sum is: $sum"
:x
```

To execute shell script in a new session.
```
$ sudo setsid ./add.sh
```

To set the controlling terminal to the current one.
```
$ sudo setsid -c ./add.sh
```

To wait for the execution of the program to end, and return
the exit value of this program as the return value of setsid

```
$ setsid -w ./add.sh
```

● <u>310 sfdisk</u>

sfdisk - display or manipulate a disk partition table

To display partitions of all the disks in system
```
$ sudo sfdisk -l
```

To view disk partitions of a specific device
```
$ sudo sfdisk -l /dev/vda
```

To display Total Partition Size
```
$ sudo sfdisk -s
```

To display Partition size of specific partition
```
$ sudo sfdisk -s /dev/vda
```

To dump disk partition details to text file
```
$ sudo sfdisk -d /dev/vda > vda.out
```

To display Disk Geometry for Debugging
```
$ sudo sfdisk -G
```

To display the kernel's idea of the geometry of the device
```
$ sudo sfdisk -g
```

To display All Disk Partition types
```
$ sudo sfdisk -T
```

To display the partition type of a specific partition
```
$ sudo sfdisk --print-id  /dev/vda 1
```

● **311 sftp**

sftp - is a secure remote file transfer utility based on File Transfer Protocol (FTP)

To Connect to SFTP

$ sftp user@192.168.122.50

sftp>

To get help commands

sftp> help

To Check Present Working Directory

local working directory

sftp> lpwd

remote working directory

sftp> pwd

To List files with sftp

On Remote

sftp> ls

On local machine

sftp> lls

To upload file using sftp

sftp > put sample.py

To upload multiple files using sftp

sftp > mput *.py

To download files using sftp

 sftp > get example.py

To download multiple files using sftp

sftp > mget *.pl

To switching directories in sftp

On Remote

sftp > cd Documents

on local machine

sftp > lcd Documents

To create new directory on local

sftp > mkdir local_dir

To create new directory on remote

sftp > lmkdir remote_dir

To remove directories using sftp

sftp > rm file.txt

sftp > rmdir dir1

To exit sftp shell

sftp > exit

● 312 sg

sg - execute command as different group ID

To execute command as different group ID

```
$ sg group-name -c 'command'
$ sg dev-group -c 'sleep 100'
$ sg admin-group -c 'ping 8.8.8.8'
```

● **313 sh**

sh - is a command language interpreter that executes commands read from a command line string, the standard input, or a specified file.

To Invoke the Bourne shell

$ sh

To run the bash script

$ sh example.sh

● <u>314 sha1sum , sha256sum</u>

sha1sum - compute and check SHA1 message digest

syntax

```
$ sha1sum [OPTION] [FILE_NAME]
```

To create the SHA-1 of a file

```
$ sha1sum example.txt
```

To create the SHA-256 of a file

```
$ sha256sum example.txt
```

To write the SHA-1 of a file to a file

```
$ sha1sum example.txt > example.sha1
```

add some contents to example.txt
and check with

```
$ sha1sum -c example.sha1
```

example.txt: FAILED

sha1sum: WARNING: 1 computed checksum did NOT match

updated the SHA-1 file against the example.txt

```
$ sha1sum new.txt > example.sha1
$ sha1sum -c example.sha1
```

example.txt: OK

To forcefully change the command's capabilities to read in binary mode

```
$ sha1sum -b example.sha1
```

To print help commands

```
$ sha1sum - -help
```

To get a list of other sha-related commands

```
$ sha + TAB
```

● <u>315 shar</u>

shar - create a shell archive

$ sudo apt install sharutils
provides shar package

To create a shar archive file.
$ shar *.txt > mytext.shar
$ shar *.pdf > mypdf.shar
$ shar *.mp3 > mymp3.shar
$ shar *.c > myc.shar

remove all .txt , .pdf , .mp3 , .c files

and then,
To extract shar archive file
$ unshar mytext.shar
$ unshar mypdf.shar
$ unshar mymp3.shar
$ unshar myc.shar

● **316 shasum**

shasum - Print or Check SHA Checksums

syntax

$ shasum [OPTION]... [FILE]...

$ cat sha_example.txt

This is test message for shasum

$ shasum sha_example.txt

$ sha1sum sha_example.txt

$ sha256sum sha_example.txt

or

$ shasum -a 256 sha_example.txt

$ sha512sum sha_example.txt

To generate the SHA value for text on the console

$ sha256sum -t

This is test message (Enter)

Ctrl+d

● <u>317 shred</u>

shred - overwrite a file to hide its contents, and optionally delete it

syntax:
```
$ shred [filename]
```

```
$ shred sample.txt
```

To designate number of times to overwrite a file
```
$ shred -n 2 example.txt
```

To overwrite and delete a file
```
$ shred -uv sample1.txt
```

To selectively overwrite bytes of text
```
$ shred -s [number_of_bytes] [filename]
```
```
$ shred -s 10 password.txt
```

To run shred with verbose mode
```
$ shred -v file.txt
```

To change permissions to allow writing if necessary
```
$ shred -f [filename]
```
```
$ shred -f file.txt
```

To hide shredding from the file system.
```
$ shred -z filename
```
```
$ shred -z file.txt
```

To Display shred Basic Details and Version and help options

```
$ shred --version
```

```
$ shred --help
```

● 318 shuf

shuf - writes a random permutation of the input lines to standard output

shuf command without any option

```
$ shuf
1
2
3
ctrl+d
3
1
2
```

```
$ cat file.txt
msg1
msg2
msg3
msg4
msg5
msg6
```

To shuffle all the lines in the file
```
$ shuf file.txt
```

To add the number of lines to shuffle
```
$ shuf -n 3 file.txt
```

To make shuf as a range

```
$ shuf -i 10-30
```

To get only one output from the randomized range

```
$ shuf -i 10-100 -n 1
```

To shuffle a range and allow displaying three outputs which can be repetitive.

```
$ shuf -i 10-100 -n 3 -r
```

To use shuf as a List

```
$ shuf -e a b c d
$ shuf -e 1 2 3 4
```

To randomize a list and decide to only output two output lines

```
$ shuf -e -n 2 1 2 3 4
$ shuf -e -n 2 a b c d
```

To write output to a file

```
$ shuf -i 10-20 -o file.txt
$ cat file.txt
```

● <u>319 showkey</u>

showkey - examine the codes sent by the keyboard

To examine the codes sent by the keyboard
$ sudo showkey

To display print options
$ sudo showkey -h

To start showkey in scan code dump mode
$ sudo showkey -s

To start showkey in keycode dump mode
$ sudo showkey -k

To start showkey in `ascii' dump mode.
$ sudo showkey -a

● **320 shutdown**

shutdown - is used to shutdown the system in a safe way

syntax

$ shutdown [OPTIONS] [TIME] [MESSAGE]

To shutdown the system at a specified time 6 P.M

$ sudo shutdown 18:00

To schedule a system shutdown in 30 minutes from now

$ sudo shutdown +30

To shutdown the system immediately

$ sudo shutdown now

To shutdown the system in 30 minutes from now and notify the users
with message "system upgrade"

$ sudo shutdown +30 "system upgrade"

To halt your system

$ sudo shutdown -H

To make shutdown power-off machine

$ sudo shutdown -P

To reboot using shutdown

$ sudo shutdown -r

To specify a time argument and a custom message

```
$ sudo shutdown -r +10 "system upgrade"
```

To cancel a scheduled shutdown

```
$ sudo shutdown -c
```

To cancel a scheduled shutdown, and to broadcast a message to all users

```
$ sudo shutdown -c "reboot is cancelled"
```

● **321 size**

size - list section sizes and total size of binary files

```
$ cat hello.c
```
```
#include <stdio.h>
int main() {
    // printf() displays the string inside quotation
    printf("Hello, World!");
    return 0;
}
```

```
$ gcc hello.c -o hello.o
```

```
$ size hello.o
```

To display in octal
```
$ size -o hello.o
```

To display in hexadecimal
```
$ size -x hello.o
```

To display in system V format and hexadecimal values
```
$ size -Ax hello.o
```

● **322 skill**

skill - is used to send signals to users and process

To halt/stop user1

```
$ skill -STOP -u user1
```

To resume already halted user1

```
$ skill -CONT -u user1
```

To kill and logout user1

```
$ skill -KILL -u user1
```

To kill and logout all users

```
$ skill -KILL -v /dev/pts/*
```

To stop 3 users user1, user2, user3

```
$ skill -STOP user1 user2 user3
```

● <u>323 sleep</u>

sleep - is used to delay for a fixed amount of time during the execution of any script

syntax

```
$ sleep number[suffix]
```

To make sleep command without any suffix

```
$ sleep 20s
```

To display help options

```
$ sleep --help
```

```
$ cat sleep.sh
```
```
#!/bin/bash
echo "Waiting for 10 seconds..."
sleep 10
echo "Task Completed"
:x
```

```
$ sh sleep.sh
```
```
Waiting for 10 seconds...
Task Completed
```

sleep command in the terminal with other commands

```
$ ll && sleep 20 && pwd && free -h
```

● **324 slogin**

slogin - is an alias for the ssh client, which is used to connect securely to a remote shell.

To login to a remote server
$ slogin 192.168.122.50
$ slogin user1@192.168.122.50

To login to remote server with port
$ slogin -p 22 user1@192.168.122.50

To to login as user1 on the remote machine
$ slogin -l user1 192.168.122.50

● **325 snap**

snap - is used to install, configure, refresh and remove snaps

$ sudo snap install snap-store

To Install Snap Apps

$ sudo snap install <package_name>

To list Installed snaps

$ snap list

To Search for Snaps

$ snap find <search_term>

To update snaps

$ snap refresh <package_name>

To print which snap packages have available updates

$ sudo snap refresh --list

To downgrade Snaps

$ sudo snap revert <package_name>

To remove Snaps

$ sudo snap remove <package_name>

To remove a snap without generating a snapshot

$ sudo snap remove <package_name> --purge

To Disable Snaps

```
$ sudo snap disable <package_name>
```

To Enable a disabled snap

```
$ sudo snap enable <package_name>
```

To List All Running Services

```
$ snap services
```

To list the services of a single snap

```
$ snap services <package_name>
```

To Start, Restart, and Stop Snap Services

```
$ sudo snap restart <package_name>
$ sudo snap start <service_name>
$ sudo snap stop <package_name>
```

To prevent a snap service from starting on boot

```
$ sudo snap stop --disable <service_name>
```

To make a service to start on next boot

```
$ sudo snap start --enable <service_name>
```

To Download and Install Snap Apps Offline

```
$ snap download <package_name>
```

To display history of changes made to system

```
$ snap changes
```

To change to a different channel

```
$ sudo snap refresh <package_name> --channel=<channel_name>
$ sudo snap refresh youtube-dl --channel=stable
```

● **326 sort**

sort - is used to sort a file, arranging the records in a particular order.

$ cat file.txt

assam

tamilnadu

chattisgarh

delhi

gujarat

himachal pradesh

kerala

bihar

To sort arrange the file.txt

$ sort file.txt

To Save Output to File

$ sort file.txt > sorted.txt

To Check for Sorting in File

$ sort -c file.txt

To Sort Multiple Files

$ sort file1.txt file2.txt

To Sort in Reverse Order

$ sort -r file.txt

To Remove Duplicate Entries

```
$ sort -u file.txt
```

To Sort by Months

```
$ ls -l > month.txt
$ sort -Mk6 month.txt
```

To Randomly Sort Data

```
$ sort -R sorted.txt
```

● <u>327 source</u>

source - is a built-in shell command that reads and executes the file content in the current shell.

syntax

$ source [filename] [arguments]

To pass commands and arguments

$ cat example.txt

free -h

pwd

date

time

uptime

$ source example.txt

To read variables from a file

$ cat example.sh

#!/bin/bash

VAR1="a"

VAR2="b"

VAR3="c"

create bash script

$ cat sample.sh

#!/bin/bash

source example.sh

echo "VAR1 is $VAR1"

echo "VAR2 is $VAR2"

```
echo "VAR3 is $VAR3"
```

```
$ source sample.sh
```

To refresh the current shell environment
```
$ alias ll = 'ls -l'
```
```
$ alias c = 'clear'
```
```
$ c
```
```
$ ll
```
This command only works in the current shell session

To make it permanent
```
$ sudo nano ~/.bashrc
```
```
alias ll = 'ls -l'
```
```
alias c = 'clear'
```
```
:x
```
Refresh the current shell environment and make it permanent for the user
```
$ source ~/.bashrc
```

To make it system wide change
```
$ sudo vim /etc/profile
```
```
alias ll = 'ls -l'
```
```
alias c = 'clear'
```
```
:x
```

```
# source /etc/profile
```

● <u>328 split</u>

split - is used to split large files into smaller files

syntax

```
$ split {options} {file_name} {prefix}
```

```
$ cat example.txt
```
This is line 1

This is line 2

This is line 3

This is line 4

This is line 5

This is line 6

This is line 7

This is line 8

This is line 9

This is line 10

To split example.txt with verbose option

```
$ split example.txt --verbose
```

To split files with customize line numbers

```
$ split -l5 example.txt --verbose
```

To split files with file size

```
$ split -b 4 example.txt --verbose   (in bytes)
$ split -b 2K <file_name> --verbose (in Kb)
$ split -b 2M <file_name> --verbose (in Mb)
$ split -b 1G <file_name> --verbose (in Gb)
```

To create Split files with numeric suffix instead of alphabetic

```
$ split -d example.txt --verbose
x00
```

To split file with customize suffix

```
$ split -l5 example.txt ilugc_file
```

To generate n chunks output files

```
$ split -n5 <file_name>
```

To Prevent Zero Size Split output files

```
$ split -n60 -e example.txt
```

To create split output files of customize suffix length

```
$ split -b 4 example.txt -a 3
$ split -b 4 example.txt -a 4
```

To split ISO file and merge it into a single file

```
$ split -n5 ubuntu-22.04.iso Split_IS0_
```
it will split ISO file into 5 pieces

To merge these files into a single

```
$  cat Split_IS0_a* > ubuntu22.04_new.iso
```

To verify the integrity of merge file using md5sum
before split of iso

```
$ md5sum ubuntu22.04.iso
```
after split and merge
```
$ md5sum ubuntu22.04_new.iso
```

● <u>329 ss</u>

ss - is used to dump socket statistics, and to investigate sockets

To list all Connections
$ ss

To list Listening and Non-listening Ports
$ ss -a

To list Listening Sockets
$ ss -l

To list all TCP connections
$ ss -t

To list all listening TCP connections
$ ss -lt

To list all UDP connections
$ ss -ua

To list all listening UDP connections
$ ss -lu

To display PID of sockets
$ ss -p

To display summary statistics

```
$ ss -s
```

To display IPv4 and IPv6 socket connections

```
$ ss -4
$ ss -6
```

To filter connections by port number

```
$ ss -at '( dport = :22 or sport = :22 )'
$ ss -at '( dport = :80 or sport = :80 )'
$ ss -at '( dport = :ssh or sport = :ssh )'
$ ss -at '( dport = :http or sport = :http )'
```

● <u>330 ssh</u>

ssh - is a program for logging into a remote machine and for executing commands on a remote machine

To access remote server
```
$ ssh 192.168.122.50
$ ssh my.server.in
```

To specify a username for SSH connection
```
$ ssh username@hostname_or_ip
$ ssh -l username@hostname_or_ip
$ ssh -l user1@192.168.122.50
$ ssh -l user1@my.server.in
```

To use a different port number for ssh connection
```
$ ssh my.server.in -p 2222
$ ssh 192.168.122.50 -p 2222
```

To run a command on a remote server from a local computer
```
$ ssh 192.168.122.50 rm ~/Desktop/test_file.txt
```

To execute multiple commands using SSH on remote nodes
```
$ ssh 192.168.122.50 command1; command2; command3
```

To make ssh to use protocol version 2
```
$ ssh -2 user1@192.168.122.50
```

To Print debug information
```
$ ssh -v user1@192.168.122.50
```

To increase the level of verbosity

```
$ ssh -vv user1@192.168.122.50
```

To get more level of verbosity

```
$ ssh -vvv user1@192.168.122.50
```

To enable X11 forwarding with ssh command

```
$ ssh -X ip_address
$ ssh -X 192.168.122.50
```

To hide the error message

```
$ ssh -q 192.168.122.50
```

To change any default value to other possible values during ssh

```
$ ssh -o option=value ip_address
$ ssh -o Port=2222 192.168.122.50
```

● <u>331 ssh-add</u>

ssh-add - adds private key identities to the OpenSSH
authentication agent

To create the public key and the private key
$ ssh-keygen -t rsa

To keep the identity of the agents for 600 seconds.
$ ssh-agent -t 600
To make ssh-agent command for non-interactive authentication
$ eval $(ssh-agent)

To add the private key passphrase to ssh-agent
$ ssh-add

To list my private keys cached by ssh-agent
$ ssh-add -l

To list all public key parameters of all identities
$ ssh-add -L

To remove all cached ssh-agent private keys
$ ssh-add -D

To lock the SSH Agent
$ ssh-add -x
To unlock SSH Agent
$ ssh-add -X

● **332 ssh-agent**

ssh-agent - is a program to hold private keys used for public key authentication

To start the ssh-agent

```
$ ssh-agent
```

To stop / kill the ssh-agent

```
$ ssh-agent -k
```

To run ssh-agent in debug mode

```
$ ssh-agent -d
```

To set bind socket name

```
$ ssh-agent -a ~/demo-ssh-socket $SHELL
```

To set expiry time for keys

```
$ ssh-agent -t 1800 $SHELL
$ ssh-agent -t 10D $SHELL
```

● 333 ssh-copy-id

ssh-copy-id - use locally available keys to authorise logins on a remote machine

create ssh key

```
$ ssh-keygen -t rsa
```

To copy ssh key to remote machine

```
$ ssh-copy-id -i /home/ilugc/.ssh/id_rsa.pub user@192.168.122.101
```

To enable forced mode to copy ssh key to remote machine

```
$ ssh-copy-id -f -i /home/dhana/.ssh/id_rsa.pub
dhana@192.168.122.101
```

To perform dry-run that prints the keys intended for installation without installing them on the remote host

```
$ ssh-copy-id -n -i /home/dhana/.ssh/id_rsa.pub
dhana@192.168.122.101
```

To connect to a remote host when the default SSH port is not being used.

```
$ ssh-copy-id -i -p <custom_port> /home/dhana/.ssh/id_rsa.pub
dhana@192.168.122.101
```

● 334 sshd

sshd - is the daemon program for ssh

To use either IPv4 or IPv6 only

/usr/sbin/sshd -4

/usr/sbin/sshd -6

To display debug modes

/usr/sbin/sshd -D

/usr/sbin/sshd -d -d -d

To send the error messages to the standard error

/usr/sbin/sshd -d -e

To use own config file for sshd apart default /etc/ssh/sshd_config

/usr/sbin/sshd -f /root/conf/custom_sshd.conf

To customize SSHD grace time

/usr/sbin/sshd -g 180

To specify options in command line

/usr/sbin/sshd -o "AllowUsers user3 user4"

● **335 ssh-keygen**

ssh-keygen - is used to generate a public/private authentication key pair

To generate ssh key without any arguments

$ ssh-keygen

To define key type

$ ssh-keygen -t rsa

$ ssh-keygen -t dsa

To define bit size default is 2048

$ ssh-keygen -b 4096

To assign passphrase

$ ssh-keygen -P "myp@ssword"

To change passphrase of the private key

$ ssh-keygen -p

To create keys with custom filename

$ ssh-keygen -f my-rsa-key

To add custom comment to the key

$ ssh-keygen -C "This key is for my.server.in"

To change comment of the key

$ ssh-keygen -c

To hash the content of known_hosts file

```
$ ssh-keygen -H
```

To delete all the keys related to 192.168.122.101 host from known_hosts file

```
$ ssh-keygen -R 192.168.122.101
```

● <u>336 ssh-keyscan</u>

ssh-keyscan - is a utility for gathering the public SSH host keys of a number of hosts

To read all public keys from the host
$ ssh-keyscan 192.168.122.50

To read all public keys from the host at port number 22
$ ssh-keyscan -p 22 192.168.122.50

To read all public keys of the rsa type
$ ssh-keyscan -t rsa 192.168.122.50

To update the known_hosts file located in the path, ~/.ssh/known_hosts
$ ssh-keyscan -H 192.168.122.50 >> ~/.ssh/known_hosts

● **337 startx**

startx command launches an X11 session.

To start an X session at 16 bits color depth.
$ startx -- -depth 16

To start an X session at 100 dpi (dots per inch) resolution.
$ startx -- -dpi 100

To start an X session with support for multiple displays.
$ startx -- -layout Multihead

● **338 stat**

stat - display file or file system status

To view the file details

`$ stat example.txt`

To view information about multiple files

`$ stat sample.pdf example.txt`

To display file system status

`$ stat -f /home`

To display information in terse form

`$ stat -t sample.pdf`

`$ stat -t example.txt`

`$ stat -t sample.pdf example.txt`

To enable following of symbolic links

`$ stat /usr/share/zoneinfo/Asia/Calcutta`

To get information on the file that the links points

`$ stat -L /usr/share/zoneinfo/Asia/Calcutta`

To format sequencing

To display inode of a file only

`$ stat --printf='%i\n' example.txt`

To display access rights and uid (User ID)

`$ stat --printf='%a:%u\n' example.txt`

To print out a new line without requiring an additional operand.

```
$ stat --format='%a:%F' example.txt
```

To display help options

```
$ stat --help
```

● <u>339 strace</u>

strace - is a CLI tool for debugging and troubleshooting programs

To trace of all system calls made command free -h

$ strace free -h

$ strace df -Th

To Trace Linux Process PID

$ sudo strace -p <PID>

To get summary of linux process

$ sudo strace -c -p <PID>

To Print Instruction Pointer During System Call

$ sudo strace -i df -Th

$ sudo strace -i free -h

To show Time of Day For Each Trace Output Line

$ sudo strace -t df -Th

$ sudo strace -t free -h

To Print Command Time Spent in System Calls

$ sudo strace -T df -Th

$ sudo strace -T free -h

To trace Only Specific System Calls

$ sudo strace -e trace=write df -Th

$ sudo strace -e trace=write free -h

$ sudo strace -e trace=all df -Th

```
$ sudo strace -e trace=all free -h
```

To trace System Calls Based on a Certain Condition

To trace all system calls involving process management

```
$ sudo strace -q -e trace=process df -Th
$ sudo strace -q -e trace=process free -h
```

To trace all system calls that take a filename as an argument,

```
$ sudo strace -q  -e trace=file df -Th
$ sudo strace -q  -e trace=file free -h
```

To trace all system calls involving memory mapping

```
$ sudo strace -q -e trace=memory df -Th
$ sudo strace -q -e trace=memory free -h
```

To redirect Trace Output to File

```
$ sudo strace -o df_debug.txt df -Th
$ sudo strace -o free_debug.txt free -h
```

To show debugging information for strace

```
$ strace -d df -Th
$ strace -d free -h
```

● <u>340 strip</u>

strip - discard symbols and other data from object files

syntax
```
$ strip [options] objfile...
```

```
$ cat hello.c
#include <stdio.h>
int main() {
    // printf() displays the string inside quotation
    printf("Hello, World!");
    return 0;
}
```

```
$ gcc hello.c -o hello.out
```

To strip the symbol table
```
$ readelf -s hello.out
$ strip -s hello.out
```
check the symbol table
```
$ readelf -s hello.out
```

To remove debug symbols only
```
$ strip --strip-debug hello.out
```
check the symbol table
```
$ readelf -a hello.out
```

To Remove a particular section
```
$ readelf -S hello.out
```

strip the .gnu.version section from the executable

`$ strip -R .gnu.version hello.out`

check the list of sections

`$ readelf -S hello.out`

To Remove unneeded symbols

`$ strip --strip-unneeded hello.out`

`$ readelf -a hello.out`

To Shield a particular symbol from stripping

`$ strip -s -K hello.c hello.out`

`$ readelf -s hello.out`

To strip off a particular symbol

`$ strip -N hello.c hello.out`

`$ readelf -s hello.out`

To create a new stripped off file

`$ strip -s -o stripped_hello hello.out`

`$ ls -lart stripped_hello`

To preserve the access and modification date/time

check the access and modification time of the original file

`$ stat hello.out`

`$ strip -s -p hello.out`

● <u>341 su</u>

su - run a command with substitute user and group ID

su command without any option

```
$ su
```

su command to make the shell a login shell

```
$ su - user3
```

To switch to root user

```
$ su -
```

To print help options

```
$ su -h
```

To switch to a Different User

```
$ su -l user3
```

To Use su with sudo command

```
$ sudo su - user3
```

To run specific command as a different user

```
$ su -c pwd user3
```

To use a different shell

```
$ su -s /usr/bin/csh
```

To Execute a command as different user with su command

```
$ su -c <command> user3
```

To use a different user in the same environment

```
$ su -p user3
```

```
$ su -m user3
```

● **342 sudo**

sudo - allows a permitted user to execute a command as the superuser or another user

To run command as a root user

`$ sudo <command>`

`$ sudo pwd`

To run command as a different user

`$ sudo -u user3 <command>`

`$ sudo -u user3 pwd`

To List user privileges with sudo

`$ sudo -l`

To Display privileges for another user with sudo

`$ sudo -l -U user3`

To add a user to the sudo group

`$ sudo usermod -aG sudo user3`

To add users to the sudoers file

`$ sudo visudo`

To run command in the background

`$ sudo -b <command>`

To update sudoers files

`$ sudo -e <file>`

To update the user's cached credentials

```
$ sudo -v
```

To invalidate user's cached credentials

```
$ sudo -k
```

● <u>343 swaplabel</u>

swaplabel - print or change the label or UUID of a swap area

To print the swaplabel

`$ sudo swaplabel /dev/vda5`

To change the swaplabel, first make swapoff

`$ sudo swapoff -a`

then,

change the swaplabel name

`$ sudo swaplabel -L mynewswap /dev/vda5`

then,

`$ sudo swapon -a`

check with

`$ sudo swapon --show`

now new swaplabel is fixed

`$ sudo swaplabel /dev/vda5`

● <u>344 swapoff</u>

swapoff - disables swapping on the specified devices and files

To deactivate the swap

$ sudo swapoff /dev/vda5

$ sudo swapoff -v /dev/vda5

To disable all swaps from /proc/swaps

$ sudo swapoff -a

$ sudo swapoff -av

To check if the swap area has been disabled

$ free -h

or

$ sudo swapon --show

● **345 swapon**

swapon - enable devices and files for paging and swapping

syntax

$ swapon <option>

To enable all swap devices and files listed in the /etc/fstab

$ sudo swapon -a

$ sudo swapon -av

To enable swap device with device name

$ sudo swapon -a /dev/vda5

$ sudo swapon -av /dev/vda5

To display the swap usage summary of all enabled swap devices and files

$ sudo swapon -s

To enable the swap discards if the device supports the discard operation

$ sudo swapon -d

To reinitialize the swap space

$ sudo swapon -f

To display a summary table with selected columns

$ swapon --show=NAME,SIZE

To print the raw output format

```
$ sudo swapon --raw
```

● <u>346 sync</u>

sync - is used to synchronize the data of the temporary memory to the permanent storage memory

syntax

```
$ sync [option] [file]
```

To sync all the cached files to the permanent memory which belongs to the current user

```
$ sudo sync
```

To sync only file data, no unneeded metadata

```
$ sudo sync -d /home/ilugc/file1.txt
```

To sync all the files in dir

```
$ sudo sync -f /home/ilugc/Downloads
```

To sync the cache data of the mounted partition

```
$ sudo sync /dev/vda1
$ sudo sync /var/log/syslog
```

● **347 sysctl**

sysctl - configure kernel parameters at runtime

To view all current kernel parameters invoke the sysctl

`$ sudo sysctl -a`

To view the swappiness value

`$ sudo sysctl vm.swappiness`

`$ sysctl kernel.hostname`

To Modify the Kernel Parameters

`$ sudo sysctl -w parameter=value`

To enable IPv4 packet forwarding

`$ sudo sysctl -w net.ipv4.ip_forward=1`

To set a parameter permanently

`$ sudo sysctl -w net.ipv4.ip_forward=1 >> /etc/sysctl.conf`

To read values from file

`$ sudo sysctl -p /etc/sysctl.d/file_name.conf`

To print variable names without values

`$ sudo sysctl -N vm.swappiness`

To print values only of the given variable

`$ sudo sysctl -n vm.swappiness`

To read values from all system directories

```
$ sudo sysctl --system
```

To print version and help options

```
$ sudo sysctl --version
$ sudo sysctl --help
```

● **348 systemctl**

systemctl - Control the systemd system and service manager

syntax

$ systemctl [OPTION] [SERVICE]

To start service

$ sudo systemctl start mariadb.service

$ sudo systemctl start sshd

To stop service

$ sudo systemctl stop mariadb.service

$ sudo systemctl stop sshd

To restart or reload service

$ sudo systemctl reload mariadb.service

$ sudo systemctl restart mariadb.service

To reload a service default, but if reload is not available for that service. It will restart it

$ sudo systemctl reload-or-restart mariadb.service

To check status of service

$ sudo systemctl status mariadb.service

$ sudo systemctl status sshd

To enable service

$ sudo systemctl enable mariadb.service

$ sudo systemctl enable sshd

To disable service

```
$ sudo systemctl disable mariadb.service
$ sudo systemctl disable sshd
```

To check Service is enabled or disabled

```
$ sudo systemctl is-active mariadb.service
$ sudo systemctl is-active sshd
$ sudo systemctl is-enabled mariadb.service
$ sudo systemctl is-enabled sshd
```

To see the status of all services

```
$ sudo systemctl list-units --type=service
```

To List services by status

```
$ sudo systemctl list-units --type=service --state=active
$ systemctl list-units --type=service --state=running
$ systemctl list-units --type=service --state=stopped
$ systemctl list-units --type=service --state=enabled
$ systemctl list-units --type=service --state=disabled
$ systemctl list-units --type=service --state=failed
```

To prevent service from starting

```
$ sudo systemctl mask {servicename}
```

To Kill a service with signal 15

```
$ sudo systemctl kill {servicename}
```

To Kill a service with signal 9

```
$ sudo systemctl kill -s 9 {servicename}
```

To Reboot

```
$ sudo systemctl reboot
```

To shutdown

```
$ sudo systemctl poweroff
```

To change default to GUI

```
$ sudo systemctl set-default graphical.target
```

To switch to multi-user.target

```
$ sudo systemctl isolate multi-user.target
```

To reload daemon

```
$ systemctl daemon-reload
```

To find version

```
$ systemctl --version
```

● **349 systemd**

systemd - is a system and service manager for Linux operating systems

To find boot process duration

$ systemd-analyze

$ systemd-analyze time

To analyze time taken by each process at boot

$ systemd-analyze blame

To analyze critical chain at boot

$ systemd-analyze critical-chain

To Determine the initial start-up transaction

$ systemd --test

To make not to pipe output into a pager

$ systemd --test --no-pager

To Dump understood unit configuration items

$ systemd --dump-configuration-items

To dump exposed bus properties

$ systemd --dump-bus-properties

● <u>350 systemd-analyse</u>

systemd-analyse - Analyze and debug system manager

To determine system boot-up performance statistics

$ systemd-analyze

To Print time required to boot the machine

$ systemd-analyze time

To Print list of running units ordered by time to init

$ systemd-analyze blame

To Print a tree of the time critical chain of units

$ systemd-analyze critical-chain

Output state serialization of service manager

$ systemd-analyze dump

Output SVG graphic showing service initialization

$ systemd-analyze plot

To List load directories for units

$ systemd-analyze unit-paths

To List exit status definitions

$ systemd-analyze exit-status

To Analyze security of unit

$ systemd-analyze security

● <u>351 systemd-machine-id-setup</u>

systemd-machine-id-setup - Initialize the machine ID in
/etc/machine-id

To clear the original "machine-id"

$ sudo rm /etc/machine-id

To create the new "machine-id" value

$ sudo systemd-machine-id-setup

To verify the new value of machine-id.

$ cat /etc/machine-id

● <u>351 systemd-machine-id-setup</u>

● **352 tac**

tac - is used to concatenate and print files in reverse

syntax

$ tac [OPTION]... [FILE]...

$ cat example.txt

This is line 1

This is line 2

This is line 3

This is line 4

This is line 5

$ cat example1.txt

This is msg 1

This is msg 2

This is msg 3

This is msg 4

This is msg 5

To print files in reverse

$ tac example.txt

To attach the separator before instead of after

$ tac -b example.txt example1.txt

To make the STRING as the separator

$ tac --separator="STRING" example1.txt example.txt

To interpret the separator as a regular expression

```
$ tac -r example.txt example1.txt
```

● 353 tail

tail - output the last part of files

To display the last 10 lines of a file

```
$ tail /var/log/messages
```

To Display the last N lines in a file

```
$ tail -n 4 /var/log/messages
```

To Print filename header

```
$ tail -v example.txt
$ tail -v /var/log/messages
```

To Display the last n lines from multiple files

```
$ tail -n 4 file1.txt file2.txt
$ tail -n 4 /var/log/messages /var/log/apache2/access.log
```

To Save the output of tail command to a text file

```
$ tail -n 10 /var/log/apache2/access.log  > output.txt
```

To append to output.txt

```
$  tail -n 10 /var/log/messages >> output.txt
```

To monitor real-time log files

```
$ tail -f /var/log/messages
```

To use pipes

```
$ tail -file.txt | sort
$ tail /var/log/messages | sort
$ tail /var/log/messages | tail -n 6 | sort
```

To Print N number of bytes data from a file

```
$ tail -c 500 /var/log/messages
$ tail -c 10M /var/log/messages
```

To print help options

```
$ tail --help
```

● <u>354 taskset</u>

taskset - is used to set or retrieve the CPU affinity of a running process

To fetch the CPU Affinity of a Process

$ pidof sshd

$ taskset -p <PID_of_sshd>

To get the CPU range of a process

$ taskset -cp <PID_of_sshd>

To change the CPU affinity of a process by assigning the new value to the PID of the existing process

$ taskset -p 0x7 <PID_of_sshd>

To Change CPU range

$ taskset -cp 0,2 <PID_of_sshd>

To set the CPU core

$ taskset -c 3,5 sshd

$ taskset -c 0-4 sshd

● **355 tar**

tar - an archiving utility

syntax
```
$ tar <options> <files>
```

To create an archive file
```
$ tar -cvf myfile.txt /home/ilugc/
```

To create tar.gz Archive File
```
$ tar cvzf mydoc.tar.gz /home/ilugc/Documents
```

To create tar.bz2 archive file
```
$ tar cvfj mydownloads.tar.bz2 /home/ilugc/Downloads
```

To untar tar Archive file
```
$ tar -xvf mydoc.tar
```
To untar tar archive file in different location
```
$ tar -xvf mydoc.tar -C /home/ilugc/Music/
```

To uncompress tar.gz Archive File
```
$ tar -xvf mydoc.tar.gz
```

To Uncompress tar.bz2 Archive File
```
$ tar -xvf mydownloads.tar.bz2
```

To list Content of tar Archive File
```
$ tar -tvf mydoc.tar
```

To List Content tar.gz Archive File

`$ tar -tvf mydoc.tar.gz`

To List Content tar.bz2 Archive File

`$ tar -tvf mydownloads.tar.bz2`

To Untar Single file from tar File

`$ tar -xvf files.sh.tar file.sh`

To Untar Single file from tar.gz File

`$ tar -zxvf backup.tar.gz backup.html`

To Untar Single file from tar.bz2 File

`$ tar -jxvf Python.tar.bz2 demo.py`

To Untar Multiple files from tar, tar.gz, and tar.bz2 File

`$ tar -xvf file.tar "file1" "file2" "file3"`

`$ tar -zxvf myfile.tar.gz "file1" "file2" "file3"`

`$ tar -jxvf myfiles.tar.bz2 "file1" "file2" "file3"`

To Extract Group of Files using Wildcard

`$ tar -xvf Python.tar --wildcards '*.py'`

`$ tar -zxvf Perl.tar.gz --wildcards '*.pl'`

`$ tar -jxvf Bash.tar.bz2 --wildcards '*.sh'`

To add files or directories to the existing tar archive files

`$ tar -rvf file.tar file.txt`

`$ tar -rvf file.tar demo.py`

To Add Files or Directories to tar.gz and tar.bz2 Files

```
$ tar -rvf file.tar.gz sample.txt
$ tar -rvf Php.tar.bz2 sample.php
```

To verify any tar or compressed archive file

```
$ tar tvfW sample.tar
```

To Check the Size of the tar, tar.gz, and tar.bz2 Archive File

```
$ tar -czf - sample.tar | wc -c
$ tar -czf - mydocs.tar.gz | wc -c
$ tar -czf - mydownloads.tar.bz2 | wc -c
```

To Exclude Files and Directories When Creating Tar File

```
$ tar --exclude='example.txt' -zcvf file.tar.gz /home/ilugc
$ tar --exclude='/home/ilugc/songs' -zcvf backup.tar.gz
/home/ilugc
```

To exclude files with specific file extensions when creating a tar archive file,

```
$ tar --exclude='*.txt' -zcvf backup.tar.gz /home/ilugc
$ tar --exclude='*.py' -zcvf backup.tar.gz /home/ilugc
```

To remove File and Directory from Tar Archive

```
$ tar --delete -f backup.tar.gz sample.txt
$ tar --delete -f backup.tar.gz '/home/ilugc/songs'
```

To extract Specific File Extension in Tar Archive

```
$ tar -xvf backup.tar.gz --wildcards '*.mp4'
```

● **356 tcpdump**

tcpdump - prints out a description of the contents of packets on a network interface

To capture packets from Specific Interface

`$ sudo tcpdump -i eth1`

To capture only N number of packets

`$ sudo tcpdump -c 4 -i eth1`

To display the package in ASCII format

`$ sudo tcpdump -A -i eth1`

To Display Available Interfaces

`$ sudo tcpdump -D`

To Display Captured Packets in HEX and ASCII

`$ sudo tcpdump -XX -i eth1`

To capture and save packets in a file

`$ sudo tcpdump -w 0002.pcap -i eth1`

To read captured packets file

`$ sudo tcpdump -r 0002.pcap`

To capture IP address packets

`$ sudo tcpdump -n -i eth1`

To capture only TCP packets.

```
$ sudo tcpdump -i eth1 tcp
```

To capture packet from specific port

```
$ sudo tcpdump -i eth1 port 22
```

To capture packets from source IP

```
$ sudo tcpdump -i eth1 src 192.168.122.50
```

To capture packets from destination IP

```
$ sudo tcpdump -i eth1 dst 192.168.122.100
```

● **357 tbl**

tbl - format tables for troff

To format a table in a file using the tbl command

```
$ tbl -C example.txt | troff
$ tbl -C example.txt | nroff
```

tbl command, piped to the troff

```
$ tbl example.txt | troff -T ascii
```

tbl with eqn

```
$ tbl example.txt | eqn | troff
```

● **358 tee**

tee - read from standard input and write to standard output and files

syntax
```
$ [command] | tee [options] [filename]
```

To create a file that stores information about a network interface
```
$ ifconfig enp1s0 | tee demo.txt
```

To append a line of text to a file
```
$ echo "This is demo msg " | tee -a demo.txt
```
check with
```
$ cat demo.txt
```

To write to multiple files
```
$ echo "This is demo msg " | tee demo1.txt demo2.txt
```

To hide the output
```
$ df -Th | tee sample.txt > /dev/null
```

To redirect output of one command to another command
```
$ cat example.txt | tee demo3.txt | grep "sample"
```

To Ignore Interrupts
```
$ ping ilugc.in | tee -i file.txt
```

● **359 telinit**

telinit - Change SysV runlevel

syntax

$ sudo telinit [options] {runlevel}

To change the runlevel for a system without rebooting or changing the /etc/inittab

$ sudo telinit 0   halt

$ sudo telinit 1   single-user mode

$ sudo telinit 2   Not used (user-definable)

$ sudo telinit 3   Full multi-user mode

$ sudo telinit 4   Not used (user-definable)

$ sudo telinit 5   Full multi-user mode (with an X-based login screen)

$ sudo telinit 6   Reboot

$ sudo telinit q   Reload daemon configuration equivalent systemctl daemon-reload

● **360 telnet**

telnet - is used for interactive communication with another host using the TELNET protocol

$ telnet -l [username] [remote machine IP] [port (optional)]

To connect to the device with specific IP address

$ telnet 192.168.122.105

To quit the connected machine

$ logout

To connect remote server with user and port number

$ telnet -l remote_user 192.168.122.105 3000

To connect to the telnet shell

$ telnet

telnet>

To print help options

$ telnet

telnet > h

To exit the telnet shell

telnet > quit

● **361 test**

test - check file types and compare values

To print "Yes, that's true." when 100 is greater than 99.

```
$ test 100 -gt 99 && echo "Yes, that's true." || echo "No, that's false."
```

To print "No, that's false." when 99 is greater than 100.

```
$ test 99 -gt 100 && echo "Yes, that's true." || echo "No, that's false."
```

To check if a file exists and is a regular file

```
$ test -f /etc/resolv.conf && echo "File /etc/resolv.conf found." || echo "File /etc/resolv.conf not found."
$ test -f /etc/ssh/sshd_config && echo "File /etc/ssh found." || echo "File /etc/ssh not found."
```

To print 0 when the expression is true and two strings are identical.

```
$ [ "awesome" = "awesome" ]; echo $?
```

To print 1 when the expression is false and two strings are not identical.

```
$  [ "awesome" = "1awesome" ]; echo $?
```

To print "1" because the expression is false

```
$ [ 7 -eq 10 ]; echo $?
```

To print "0" because the expression is true

```
$ [ 10 -eq 10 ]; echo $?
```

● **362 time**

time - run programs and summarize system resource usage

syntax

```
$ time [options] [command]
```

Real: This is the actual time taken by the processor of the computer to execute the command from pressing the button to complete the command.

User: CPU time that is taken by the user mode.

SYS: This is the time taken by the system or the Kernel to execute the command.

To calculate execution time of commands

```
$ time sleep 5
$ time free -h
$ time df -Th
$ time ping google.com
```

To display the time in portable POSIX format

```
$ time -p sleep 5
$ time -p free -h
$ time -p df -Th
$ time -p ping google.com
```

To calculate the update time

```
$ time sudo apt update
```

To save the time command output in the text file

```
$ /usr/bin/time -o sleep_output.txt sleep 5
$ /usr/bin/time -o df_output.txt df -Th
```

To get detailed summary of the executed time

```
$ /usr/bin/time -v sleep 5
$ /usr/bin/time -v df -Th
$ /usr/bin/time -v free -h
```

● **363 tgz**

tgz - makes a gzip'd tar archive

syntax

```
$ tgz [ destination [ source ...  ] ]

$ tar -cvzf <name of tarball>.tgz /path/to/source/folder
$ tar -cvzf <name of tarball>.tgz /path/to/source_file
$ tar -cvzf file.tgz /home/ilugc/Documents/file.pdf
```

● **364 timedatectl**

timedatectl - Control the system time and date

To display the current settings

```
$ timedatectl
```

To Show current time settings

```
$ timedatectl status
```

To Show properties of systemd-timedated

```
$ timedatectl show
```

To change settings

```
$ sudo timedatectl set-timezone Asia/Hong_Kong
$ sudo timedatectl set-timezone Asia/Tokyo
```

To list the time zones

```
$ timedatectl list-timezones | column
```

To display the number of time-zones available

```
$ timedatectl list-timezones | wc -l
```

To Show status of systemd-timesyncd

```
$ timedatectl timesync-status
```

To Show properties of systemd-timesyncd

```
$ timedatectl show-timesync
```

To Show all properties, including empty ones

```
$ timedatectl -a
```

To Set system time

```
$ timedatectl set-time <TIME>
```

To Set system time zone

```
$ timedatectl set-timezone <ZONE>
```

To Show this help options

```
$ timedatectl --help
```

● **365 timeout**

timeout - run a command with a time limit

syntax

```
$ timeout [OPTION] DURATION COMMAND [ARG]...
```

To specify timeout value (in seconds) with the command

```
$ timeout 5 ping ilugc.in
$ timeout 5h ping ilugc.in
$ timeout 5d ping ilugc.in
$ timeout 8 tail -f /var/log/syslog
```

we can specify time in seconds , minutes , hours and days

To make Preserve status returns an exit status even the process is terminated forcefully

```
$  timeout --preserve-status 2m ping facebook.com
```

Ctrl+c

```
$ Ctrl+c
$ echo $?
```

To set a signal to the command

```
$ timeout -s <command>
```

To check the available signals

```
$ kill -l
$ sudo timeout -s SIGKILL ping ilugc.in
```

To send signals by using a signal number

```
$ sudo timeout -s 9 ping ilugc.in
```

To kill the timeout command after 5 seconds if the connection is not established

```
$ sudo timeout -k 5 1m ping google.com
```

● **366 touch**

touch - change file timestamps

To Create an Empty File

$ touch file.txt

To Create Multiple Files

$ touch file1.txt file2.txt file3.txt file4.txt

To Change File Access and Modification Time

$ touch -a file.txt

To Set File Timestamp Using Date String

$ touch -d tomorrow demo

To Avoid Creating New File

$ touch -c file.txt

To Change File Modification Time

$ touch -m sample.txt

To Explicitly Set the Access and Modification times

$ touch -c -t YYDDHHMM sample.txt

To Use the time stamp of example.txt to sample.txt

$ touch -r sample.txt example.txt

To Create a File using a specified time

```
$ touch -t YYMMDDHHMM.SS demofile
$ touch -t 2211260600.26 demofile
```

● 367 tload

tload - graphic representation of system load average

syntax

$ tload [options] [terminal]

To display the representation of the average system load as a graph on the terminal

$ tload

To set delay between graph update in seconds

$ tload -d 2

To change the scale of the graph

$ tload -s 10

To print a graph of the current system load average to a specified tty

$ tload -d 2 -s 10 /dev/tty3

To run tload over the ssh based session

$ ssh -t user1@192.168.122.100 tload -d 2

● **368 tput**

tput - is used to query the terminfo terminal database and check
if that terminal supports a specific feature.

To Set the Cursor Position using tput cup

```
$ tput cup 5 6
$ tput cup 10 10
```

To Clear the Screen Using tput clear

```
$ tput clear
```

To Get the Number of Columns and Lines of a Terminal

```
$ tput cols
$ tput lines
```

To Execute Multiple tput Commands

```
$ tput -S <<END
> clear
> cup 2 4
> END
```

To Turn On and Turn Off Highlighting

```
$ echo `tput bold`ilugc`tput sgr0`
$ echo `tput sgr0`ilugc`tput sgr0`
```

To Underline Text using smul and rmul

```
$ echo `tput smul`ilugc`tput rmul`
$ echo `tput rmul`ilugc`tput rmul`
```

To Hide and Unhide the Cursor using civis and cnorm

```
$ tput civis
```

```
$ tput cnorm
```

● **369 tr**

tr - is used to translate and/or delete characters from stdin input and writes to stdout

```
$ cat example.txt
```

linux os is powerful

linux os is versatile

linux os is best

To change all lowercase letters in the text to uppercase and vice versa

```
$ cat example.txt | tr [:lower:] [:upper:]
```

or

```
$ cat example.txt | tr [a-z] [A-Z]
```

To save the results written to stdout in a file

```
$ cat example.txt | tr [a-z] [A-Z] >output.txt
```

To send input to tr using the input redirection and redirect the output to a file

```
$ tr [a-z] [A-Z] < linux.txt >output.txt
```

To delete characters and remove spaces

```
$ cat domain.txt
```

 www. ilugc. in

www. chennaipy. org

www. google. com

```
$ cat domains.txt | tr -d ''
```

To remove repeated characters in a sequence

```
$ cat domains.txt
 www. ilugc...........  innnn
www. chennaipy. org
www. google. com


$ cat domains.txt | tr -s ''
```

To delete all the letters and only leave the UID

```
$ echo "My UID is $UID" | tr -cd "[:digit:]\n"
```

To break a single line of words (sentence) into multiple lines

```
$ echo "My UID is $UID"
$ echo "My UID is $UID" | tr " "  "\n"
```

To translate multiple lines of words into a single sentence

```
$ cat uid.txt
My
UID
is
1000


$ tr "\n" " " < uid.txt
```

To translate single character, for instance, a space into a " : "

```
$ echo "ilugc.in =>linux,devops,cloud,programming" | tr " " ":"
$ echo "ilugc.in =>linux devops cloud programming" | tr " " ":"
```

- ## 370 traceroute

traceroute - traces path to a network host

To find the network path from my machine to google.com
$ traceroute google.com

To Disable IP address and host name mapping
$ traceroute google.com -n
$ traceroute ilugc.in -n

To Configure Response Wait Time
$ traceroute google.com -w 0.1

To configure number of queries per hop
$ traceroute google.com -q 5

To Configure the TTL value to start with
$ traceroute google.com -f 8
$ traceroute ilugc.in -f 10

● <u>371 trap</u>

trap - is used to execute a command when the shell receives any signal is called `trap`.

syntax
```
$ trap [options] "[arguments]" [signals]
```

To display the list of all commands associated with each condition
```
$ trap
```

To display the list of all signal names with number
```
$ trap -l
```

To Set `trap` command with signal number of SIGUP, SIGQUIT and SIGKILL
```
$ trap 'echo Trap command executed' 1 3 9
$ press Ctrl+C
$ ^CTrap command executed
$ trap
```

To Set trap command for ERR and EXIT
```
$ trap 'rm file.txt' err exit
$ trap
```

● **372 troff**

troff - it performs typesetting functions and formats documents

Troff the input file

```
$ troff example.txt
```

troff command

```
$ troff
Hello world!
^D
```

To pipe the output of troff

```
$ troff > output
Welcome to Linux world
^D
```

To append the output of troff

```
$ troff >> output
Welcome to python
^D
```

● <u>373 truncate</u>

truncate - shrink or extend the size of a file to the specified
size

To Clear contents of a file with truncate

$ truncate -s 0 example.txt

To truncate a file to a specific size

$ truncate -s 200K example.txt

To extend file size

$ truncate -s +400K example.txt

To reduce file size

$ truncate -s -100k example.txt

● **374 tsort**

tsort - perform topological sort

To perform topological sort

```
$ cat demo.txt
a b c
d e
f
g z
b c d h
```

```
$ tsort demo.txt
```

To redirect output to a file

```
$ tsort > demo_out.txt
```

● <u>375 tune2fs</u>

tune2fs - is used to manipulate the filesystem parameters of a ext 2/3/4 type file system

syntax

$ sudo tune2fs option device

To list file system parameters

$ sudo tune2fs -l /dev/vda1

To set the volume label

$ sudo tune2fs -L my_vol /dev/vda1

To find out maximum mount count and mount count

$ sudo tune2fs -l /dev/vda1 | grep -i mount

To set maximum mount count

$ sudo tune2fs -c 80 /dev/vda1

To Display Check Interval of a filesystem

$ sudo tune2fs -l /dev/vda1 | grep interval

To set time intervals between two filesystem checks

$ sudo tune2fs -i 5m /dev/vda1

$ sudo tune2fs -i 5d /dev/vda1

$ sudo tune2fs -i 5w /dev/vda1

To disable filesystem check on reboot set the maximum mount count to -1 or 0.

```
$ sudo tune2fs -c 0 /dev/vda1
```

```
$ sudo tune2fs -i 0 /dev/vda1
```

To set mount counts

```
$ sudo tune2fs -C 30 /dev/vda1
```

To Set the last-mounted directory

```
$ sudo tune2fs -M mount_mydir /dev/vda1
```

To Set the last time checked of a filesystem

```
$ sudo tune2fs -T now /dev/vda1
```

To Change the error behavior of a filesystem

```
$ sudo tune2fs -e continue /dev/vda1
```

To generate a new random UUID of the filesystem

```
$ sudo tune2fs -U random /dev/vda1
```

● <u>376 type</u>

type - is used to display information about the command type

syntax
$ type [OPTIONS] FILE_NAME

To find the type of ls command
$ type ls

To find the type of wc command
$ type wc
$ type type

To display more than one argument
$ type df free sleep head

To display the command is an alias, keyword or a function and path
of an executable
$ type -a pwd
$ type -a ls

To print a single word describing the type of the command
$ type -t ls
$ type -t pwd
$ type -t df
$ type -t du
$ type -t while

To display the name of the disk file which would be executed by the shell

```
$ type -p bash
```

To search the PATH for an executable file on the disk even if the command is not file

```
$ type -P pwd
```

To suppress shell function lookup

```
$ type -f while
```

```
$ type -f ls
```

```
$ type -f pwd
```

● **377 udevadm**

udevadm - is used to manage udev. It takes various subcommands, each of which performs a certain task to modify the behavior of the systemd-udevd daemon and related components

To get help options
$ udevadm --help
$ udevadm info --help
$ udevadm trigger --help
$ udevadm settle --help
$ udevadm control --help
$ udevadm monitor --help
$ udevadm test --help
$ udevadm test-builtin --help

To query the Udev database for the device path of /dev/vda1
$ udevadm info --query=path --name=/dev/vda1

To query the device attributes from the udev database
$ udevadm info /dev/vda1

To query the Udev database for all device information for /dev/vda1
$ udevadm info --query=all --name=/dev/xvda
$ udevadm info --query=all --name=/dev/xvda1

To print all sysfs properties of /dev/vda1
$ udevadm info --attribute-walk --name=/dev/vda
$ udevadm info --attribute-walk --name=/dev/vda1

To Trigger all udev rules

```
$ sudo udevadm trigger
```

To Reload all udev rules

```
$ sudo udevadm control --reload-rules
```

To List attributes of a device

```
$ udevadm info --attribute-walk --path /dev/vda1
```

To Monitor all device events

```
$ udevadm monitor
```

To Print uevents sent out by the kernel

```
$ udevadm monitor --kernel
```

To Print device events after being processed by udev

```
$ udevadm monitor --udev
```

● **378 ulimit**

ulimit - is a built-in Linux shell command that allows viewing or limiting system resource amounts that individual users consume

Syntax

$ ulimit [flags][limit]

To set the physical memory size in kilobytes

$ ulimit -m 10000

To view the detailed soft limits for the current user

$ ulimit -Sa

To view the detailed hard limits for the current user

$ ulimit -Ha

To find the resource amount that the current user has access to use

$ ulimit

To get a detailed report with all resource limits for the current user

$ ulimit -a

To limit the process number to 10

$ ulimit -u 10

To set the maximum file size that a user can make

$ ulimit -f 100

To limit maximum virtual memory

`$ ulimit -v 1500`

To limits the number of simultaneously opened files

`$ ulimit -n 10`

To display maximum users process

`$ ulimit -u`

To show maximum memory size limit

`$ ulimit -v`

To display the Hard limit

`$ ulimit -Hn`

To display Soft Limit

`$ ulimit -Sn`

To check out the max scheduling priority of the current user

`$ ulimit -e`

To check the maximum stack size of the current user

`$ ulimit -s`

To report the time each process is allowed to run

`$ ulimit -t`

To check out how many file descriptors a process

`$ ulimit -n`

To check out the help options

```
$ ulimit --help
```

● <u>379 unalias</u>

unalias - is used to remove entries from the current user's list of aliases

To list all aliases

$ alias

or

$ alias -p

if user had an alias named h for the history command

alias could be removed

$ unalias h

To remove all aliases from the current session

$ unalias -a

● <u>380 ufw</u>

ufw - program for managing a netfilter firewall

To check ufw status

```
$ sudo ufw status
```

To enable ufw

```
$ sudo ufw enable
```

To disable ufw

```
$ sudo ufw disable
```

To show ufw rules, along with their numbers

```
$ sudo ufw status numbered
```

To reset ufw to its default state

```
$ sudo ufw reset
```

To view firewall reports

```
$ sudo ufw show added
$ sudo ufw show raw
```

To set ufw logging levels

```
$ sudo ufw logging on
$ sudo ufw logging high
$ sudo ufw logging medium
$ sudo ufw logging low
$ sudo ufw logging full
```

To allow service

```
$ sudo ufw allow openssh
```

To deny service

```
$ sudo ufw deny openssh
```

To view all application profiles

```
$ sudo ufw app list
```

To get more information about a particular profile and defined rules

```
$ sudo ufw app info 'Apache'
```

To reject outgoing FTP connections

```
$ sudo ufw reject out ftp
```

To add a new rule at a specific number

```
$ sudo ufw insert 1 allow 80/tcp
$ sudo ufw insert 2 allow 443/tcp
```

To Delete rules by specifying their numbers

```
$ sudo ufw status numbered
$ sudo ufw delete 1    (will delete above rule allow 80/tcp)
```

To Deleting rules by service

```
$ sudo ufw delete allow ftp
$ sudo ufw delete allow openssh
```

To allow or deny a specific port for a given IP address

```
$ sudo ufw allow from 192.168.122.100 to any port 8081
$ sudo ufw deny from 192.168.122.100 to any port 8080
```

To allow an entire subnet

```
$ sudo ufw allow from 10.0.1.0/16
```

To Allow Network Subnets to Specific Port

```
$ sudo ufw allow from 192.168.1.0/24 to any port 22
```

To Allow Specific Network Interface

```
$ sudo ufw allow in on eth1 to any port 22
```

To open all ports for a particular IP address

```
$ sudo ufw allow from 192.168.122.100
```

To allow all ports in a range by specifying a port range

```
$ sudo ufw allow 20:450/tcp
```

To get a numbered list of added rules

```
$ sudo ufw status numbered
```

To check all added rules before starting the firewall

```
$ sudo ufw show added
```

To Allow only TCP traffic over HTTP/80

```
$ sudo ufw allow http/tcp
$ sudo ufw allow 80/tcp
```

To deny incoming FTP traffic

```
$ sudo ufw deny ftp/tcp
$ sudo ufw deny 21/tcp
```

To add a new rule to allow SSH

```
$ sudo ufw allow ssh
$ sudo ufw allow 22/tcp
```

To rate limiting service

```
$ sudo ufw limit ssh/tcp
```

To print verbose

```
$ sudo ufw status verbose
```

To reload ufw

```
$ sudo ufw reload
```

To Dry Run UFW Rules

```
$ sudo ufw --dry-run enable
```

● <u>381 umask</u>

umask - is used to set default permissions for files or directories the user creates.

To calculate umask value

$ umask

To Displays the current mask

$ umask -p

To set and update the default umask value

$ umask 543

To view symbolic representation

$ umask -S

To set the default permissions for all new files or folders to 644 and 755 then umask value is

$ umask 022

for folders 777-022 = 755

for files 666-022 = 644

● <u>382 umount</u>

umount - it is used to unmounts a mounted filesystem, informing the system to complete any pending read or write operations, and safely detaching it

To unmount all the filesystems mentioned in /etc/mtab
$ sudo umount -a

To forcefully unmount a filesystem
$ sudo umount -f /backup

To lazy unmount of a filesystem
$ sudo umount -l /backup

To unmount a Partition or Filesystem
$ sudo umount /mnt

To recursively unmount a target with all its children
$ sudo umount -R /mnt

To umount without writing in /etc/mtab
$ sudo umount -n /mnt

To unmount more than mount point in a single execution
$ sudo umount /mnt /backup

● **383 uname**

uname - print system information

To print uname without options

$ uname

To print all information

$ uname -a

To print the kernel name

$ uname -s

To print the network node hostname

$ uname -n

To print the kernel release

$ uname -r

To print the kernel version

$ uname -v

To print the machine hardware name

$ uname -m

To print the processor type

$ uname -p

To print the hardware platform

$ uname -i

To print the operating system

```
$ uname -o
```

To display help options and version

```
$ uname -h
$ uname --version
```

● **384 unattended-upgrade/s**

unattended-upgrade - automatic installation of security (and other) upgrades

To simulate installing updates, do not actually do it

$ sudo unattended-upgrade --dry-run

To print debug messages

$ sudo unattended-upgrade -d

To make apt/libapt print verbose debug messages

$ sudo unattended-upgrade --apt-debug -v

To print info messages

$ sudo unattended-upgrade -v

To make Only download, do not even try to install

$ sudo unattended-upgrade --download-only

To make upgrade in minimal steps

$ sudo unattended-upgrade --minimal-upgrade-steps

To upgrade all packages together instead of in smaller sets

$ sudo unattended-upgrade --no-minimal-upgrade-steps

To show this help message

$ sudo unattended-upgrade -h

● <u>385 uncompress</u>

uncompress - it is used to uncompress files that were compressed using the compress command.

```
$ cat demo.txt
```
This is line one

This is line two

This is line three

This is line four

This is line five

This is line six

This is line seven

This is line eight

This is line nine

This is line ten

To compress the file
```
$ compress demo.txt
```
demo.txt.Z

To uncompress the file
```
$ uncompress demo.txt.Z
```


To write to the standard output
```
$ uncompress -c demo.txt.Z
```


To print verbose output
```
$ uncompress -v demo.txt.Z
```

To uncompress forcefully

```
$ uncompress -f demo.txt.Z
```

```
$ uncompress -cvf demo.txt.Z
```

combining with tar command

```
$ tar -cvf - dir1 dir2 | compress > file.tar.Z
```

To uncompress

```
$ uncompress -c file.tar.Z|tar -xvf  -
```

● <u>386 unexpand</u>

unexpand - it is used to convert blanks in each FILE to tabs, writing to standard output

syntax

$ unexpand [OPTION] [FILE]

To convert blanks in a file to tabs and write the output to stdout

$ unexpand sample.txt

To write the output to a file

$ unexpand sample.txt > unexpand_output.txt

To convert blanks to tabs, reading from standard output

$ unexpand

To convert all blanks, instead of just initial blanks

$ unexpand -a sample.txt

To convert only leading sequences of blanks

$ unexpand --first-only sample.txt

To have tabs a certain number of characters apart

$ unexpand -t 4 sample.txt

To mention multiple tab positions

$ unexpand -t 5,10,15 sample.txt

● **387 uniq**

uniq - it is used to report or filter out repeated lines in a file.

```
$ cat demo.txt
```
redhat

debian

ubuntu

ubuntu

centos

fedora

fedora

fedora

fedora

To report or filter out for lines that are adjacent and repeated

```
$ uniq demo.txt
```

To only print unique lines

```
$ uniq -u demo.txt
```

To prefix lines by the number of occurrences

```
$ uniq -c demo.txt
```

To only print duplicate lines, one for each group

```
$ uniq -d demo.txt
```

To print all duplicate lines

```
$ uniq -D demo.txt
```

To print all duplicate lines but allow separating groups with
an empty line

```
$ uniq --all-repeated=prepend demo.txt
```

To ignore differences in case when comparing

```
$ uniq -i demo.txt
```

To skip comparing the first X number of characters in each line

```
$ uniq -s 3 demo.txt
```

To compare only the first X number of characters

```
$ uniq -w 4 demo.txt
```

● **388 unlink**

unlink - is a command-line utility for removing a single file.

Call the unlink function to remove the specified FILE

syntax

$ unlink filename

$ unlink dir_name

To delete symbolic link

$ unlink linkname

$ ll

lrwxrwxrwx 1 dhana dhana   23 Dec 1 10:08 example.txt -> /home/dhana/example.txt

$ unlink example.txt

To delete symbolic link directory

$ unlink link_dir_name

$ ll

lrwxrwxrwx 1 dhana dhana   18 Dec 1 10:10 Music -> /home/dhana/Music/

$ unlink Music

● **389 unmkinitramfs**

unmkinitramfs - extract content from an initramfs image

syntax

$ unmkinitramfs [options] initramfs-file directory

To extract the /boot/initrd.img-5.14.0-8-amd64 file into the current working directory

$ unmkinitramfs /boot/initrd.img-5.14.0-8-amd64 .

To print verbose output

$ unmkinitramfs -v /boot/initrd.img-5.14.0-8-amd64 .

• **390 unset**

unset - is used to remove variables or functions.

syntax
$ unset <var-name>

$ VAR1='welcomt to linux'
$ echo $VAR1
$ unset VAR1
$ echo $VAR1

$ export ILUGC=foss
$ printenv ILUGC
$ unset ILUGC
$ printenv ILUGC

● **391 unzip**

unzip - list, test and extract compressed files in a ZIP archive

To extract all files from the zip archive
$ unzip file.zip

To unzip the file to another directory
$ unzip file.zip -d /home/ilugc/Documents

To display the content of the zip file without extracting
$ unzip -l file.zip

To extract zip files with suppressing output
$ unzip -q file.zip

To exclude files from extracting a zip file
$ unzip file.zip -x file1 file2

Not to overwrite the existing files
$ unzip -n file.zip

To unzip multiple files
$ unzip '*.zip'
$ unzip '*.txt'

To unzip password protected files
$ unzip -P Password file.zip

To extract in quiet mode

```
$ unzip -q file.zip
```

To overwrite existing files

```
$ unzip -o file.zip
```

To view detailed contents of a zipped file

```
$ unzip -Z file.zip
```

To unzip one or more files from an archive

```
$ unzip file.zip mydoc.txt
```

• **392 update-grub2**

update-grub2 - update the grub

To update grub

$ sudo update-grub2

To generate a grub2 config file

$ sudo update-grub2 -o /boot/grub/grub.cfg

● **393 update-passwd**

update-passwd - safely update /etc/passwd, /etc/shadow and /etc/group

syntax

$ update-passwd [options]

To make a dry run without any changes

$ sudo update-passwd -n /etc/passwd

$ sudo update-passwd -n /etc/shadow

$ sudo update-passwd -n /etc/group

To print options

$ sudo update-passwd -h

● **394 uptime**

uptime - is used to print how long the system has been running

syntax

$ uptime [-options]

uptime command without any options

$ uptime

To show uptime in pretty format

$ uptime -p

To display the date/time since when the system has been running

$ uptime -s

● **395 usb-creator-gtk**

usb-creator-gtk - Ubuntu startup disk creation tool for Gtk+

To start the startup disk creator
```
$ sudo usb-creator-gtk
```

To provide a source image to pre-populate the UI
```
$ sudo usb-creator-gtk -i /path/to/ubuntu22.04.iso
```

To allow writing to system-internal devices
```
$ sudo usb-creator-gtk --allow-system-internal
```

- ## **396 usb-devices**

usb-devices - print USB device details

To print usb device details

$ usb-devices

● <u>**397 useradd**</u>

useradd - create a new user or update default new user information

To add a new user

```
$ sudo useradd user1
```

To create a user with a different home directory

```
$ sudo useradd -d /home/klug user2
```

To create a user with a specific User ID

```
$ sudo useradd -u 1004 user3
```

To create a user with a specific Group ID

```
$ sudo -u 1004 -g admin user4
```

To add a user to multiple groups

```
$ sudo useradd -G admin,devops,dev user5
```

To add a user without home directory

```
$ sudo useradd -M user6
```

To create a user with an account expiry date

```
$ sudo useradd -e 2024-04-04 user7
```

To create a user with password expiry date

```
$ sudo -e 2024-04-04 -f 50 user1
```

To add a user with custom comments

```
$ sudo useradd -c "welcome to user2" user2
```

To add a user without a login shell

```
$ sudo useradd -s /sbin/nologin user3
```

To create user with home directory

```
$ sudo useradd -m user4
```

● <u>398 userdel</u>

userdel - delete a user account and related files

syntax

$ userdel [OPTIONS] USERNAME

To delete a user account

$ sudo userdel user1

To remove the user's home directory and mail spool

$ sudo userdel -r user2

To forcefully remove the user account

$ sudo userdel -f user3

To display help options

$ sudo userdel --help

To remove any SELinux(Security-Enhanced Linux) user mapping for the user's login.

$ sudo userdel -Z user2

● <u>399 usermod</u>

usermod - modify a user account

syntax

```
$ usermod [options] username
```

To add Information to user account

```
$ sudo usermod -c "This is test message" user1
```

To change user home directory

```
$ sudo usermod -d /home/user3/ user2
```

To set user account expiry date

```
$ sudo usermod -e 2025-04-04 user3
```

To change user primary group

```
$ sudo usermod -g devops user4
```

To add group to an existing user

```
$ sudo usermod -G web user5
```

To add supplementary and primary group to user

```
$ sudo usermod -aG wheel user6
```

To change user login name

```
$ sudo usermod -l new_username old_username
$ sudo usermod -l user7 user6
```

To lock user account

```
$ sudo usermod -L user7
```

To unlock user account

```
$ sudo usermod -U user7
```

To move user home directory to new location from /home/user7 to /var/user7

```
$ sudo usermod -d /var/user7/ -m user7
```

To create unencrypted password for user

```
$ sudo usermod -p passcode123 user7
```

To change user shell

```
$ sudo usermod -s /bin/sh user1
```

To change user ID

```
$ sudo usermod -u 666 user2
```

To modify the UID and GID.

```
$ sudo usermod -u 555 -g 665 user3
```

● **400 users**

users - print the user names of users currently logged in to the current host

To print the user name currently logged in the host

$ users