

LEARN & BUILD

Basic Linux Commands

Submitted By:- Amisha Singh

Submitted To:- Ashish Mittal

ASSIGNMENT-2

Introduction

- Linux is Free and Open Source.
- It is a UNIX like multi-user, multitasking operating system with the X Windows GUI, which can work on multiple hardware platform.
- Since Linux is free and Open Source, there are many flavours to linux.





- 1.ls** – Displays information about files in the current directory.
- 2.pwd** – Displays the current working directory.
- 3.mkdir** – Creates a directory.
- 4.cd** – To navigate between different folders.
- 5.rmdir** – Removes empty directories from the directory lists.
- 6.cp** – Moves files from one directory to another.
- 7.mv** – Rename and Replace the files
- 8.rm** – Delete files
- 9.uname** – Command to get basic information about the OS
- 10.locate**– Find a file in the database
- 11.touch** – Create empty files
- 12.ln** – Create shortcuts to other files

- 13.cat** – Display file contents on terminal
- 14.clear** – Clear terminal
- 15.ps**- Display the processes in terminal
- 16.man** – Access manual for all Linux commands
- 17.grep**- Search for a specific string in an output
- 18.echo**- Display active processes on the terminal
- 19.wget** – download files from the internet
- 20.whoami**- Create or update passwords for existing users
- 21.sort**- sort the file content
- 22.cal**- View Calendar in terminal
- 23.whereis** – View the exact location of any command types after this command
- 24.df** – Check the details of the file system
- 25.wc** – Check the lines, word count, and characters in a file using different options

ls command

- ❖ The simple ls command lists the contents of your current working directory.

Example-:

```
Editor  Tab 1  + 59 min
Initialising Kubernetes... done

controlplane $ ls
filesystem
controlplane $
```

pwd command

- ❖ The [pwd command](#) is mostly used to print the current working directory on your terminal.

Command-:

```
1 pwd
```

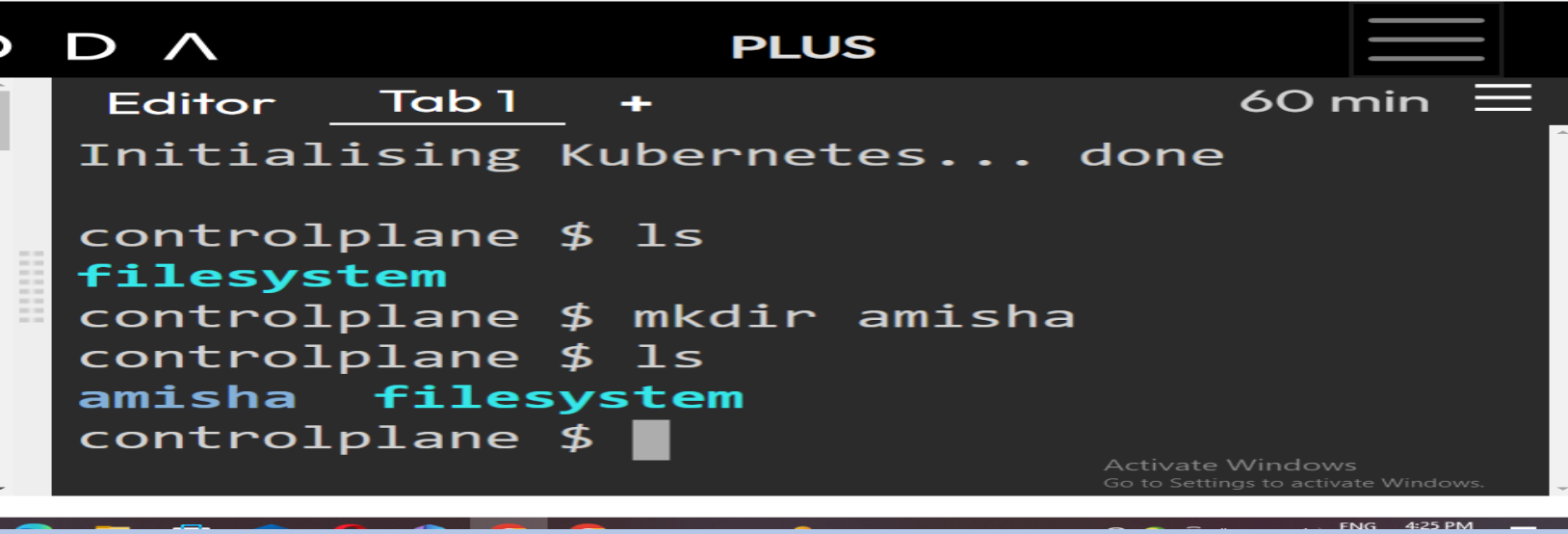
Output-:

```
/home/cg/root/63b31828bde9b
```

mkdir command

❖ The `mkdir` command is used to create a new directory.

Example:-



```
Editor  Tab 1  +  60 min  ≡
Initialising Kubernetes... done

controlplane $ ls
filesystem
controlplane $ mkdir amisha
controlplane $ ls
amisha filesystem
controlplane $
```

Activate Windows
Go to Settings to activate Windows.

ENG 4:25 PM

cd command

❖ The `cd command` uses to change the current working directory to another directory.

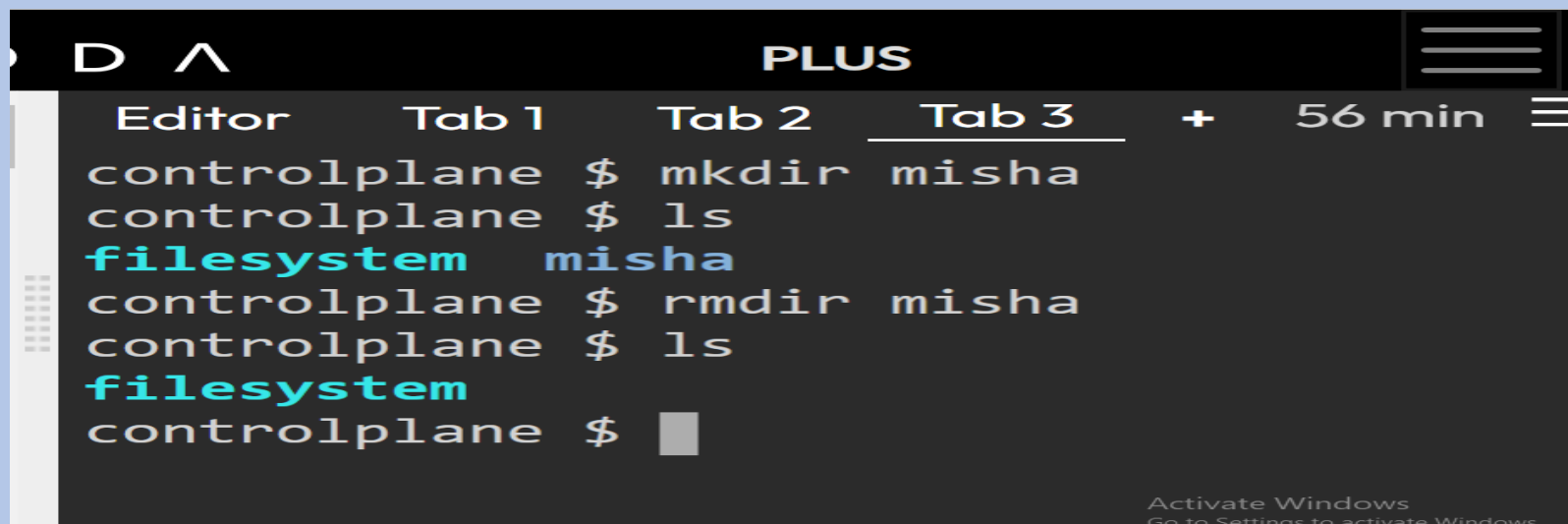
Example-:

```
[root@ip-172-31-41-3 ~]# pwd
/root
[root@ip-172-31-41-3 ~]# cd amisha
[root@ip-172-31-41-3 amisha]# pwd
/root/amisha
```

rmmdir command

- ❖ The [rmmdir command](#) is used to delete permanently an empty directory.

Example-:

A screenshot of a Windows terminal window with a dark background. The window title bar shows 'D ^' and 'PLUS'. The terminal has three tabs: 'Editor', 'Tab 1', and 'Tab 2', with 'Tab 2' being the active tab. The terminal shows a series of commands and their outputs. The commands are: 'mkdir misha', 'ls', 'rmmdir misha', and 'ls'. The outputs are: 'filesystem misha' (after the first ls), and 'filesystem' (after the second ls). The prompt is '\$' for each command. At the bottom right, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.'

```
controlplane $ mkdir misha
controlplane $ ls
filesystem misha
controlplane $ rmmdir misha
controlplane $ ls
filesystem
controlplane $
```

cp command

❖ The **cp command** is used to copy files and directories.

Command-:

```
1 ls
2 cp first.txt second.txt
3 ls
```

Output-:

```
first.txt  main.sh
first.txt  main.sh second.txt
```



mv command

❖ The [mv command](#) is generally used for renaming the files in Linux.

Command-:

```
1 ls
2 mv first.txt renamed.txt
3 ls
```

Output-:

```
first.txt  main.sh
main.sh    renamed.txt
```



rm command

❖ [rm command](#) in Linux is generally used to delete the files created in the directory.

Command-:

```
1 ls
2 rm renamed.txt
3 ls
```

Output-:

```
main.sh  renamed.txt
main.sh
```

uname command

❖ The [uname command](#) is used to check the complete OS information of the system. Check out the command and the output below.

Command-:

```
1  uname
```

Output-:

```
SMP Sun Dec 04 08:06:28 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
```

locate command

❖ The [locate command](#) is generally used to locate the files in the database.

Command-:

```
1 rm first.txt
2 locate first.txt
```

Output-:

```
locate -e first.txt
```

touch command

- ❖ The **touch command** creates an empty file , change and modify timestamps of a file.

Example-:

```
Editor  Tab 1  + 59 min
Initialising Kubernetes... done

controlplane $ ls
filesystem
controlplane $ touch file1
controlplane $ ls
file1 filesystem
controlplane $ touch file1.text
controlplane $ touch file2
controlplane $ touch file3
controlplane $ ls
file1      file2  filesystem
file1.text file3
controlplane $
```

Activate Windows
Go to Settings to activate Windows.

In command

❖ The [In command](#) is used to create a shortcut link to another file. This is among the most important Linux commands to know if you want to operate as a Linux administrator.

Command-:

```
1 mkdir Demo
2 mkdir Linked
3 ln -s Demo Linked
```

Output-:

```
Linked/Demo
```

cat command

- ❖ The [cat command](#) is the simplest command to use when you want to see the contents of a particular file. The only issue is that it simply unloads the entire file to your terminal. If you want to navigate around a huge file, should use **less** command alternatively.

Command-:

```
1 cat files.txt
```

Output-:

```
this is a File
```

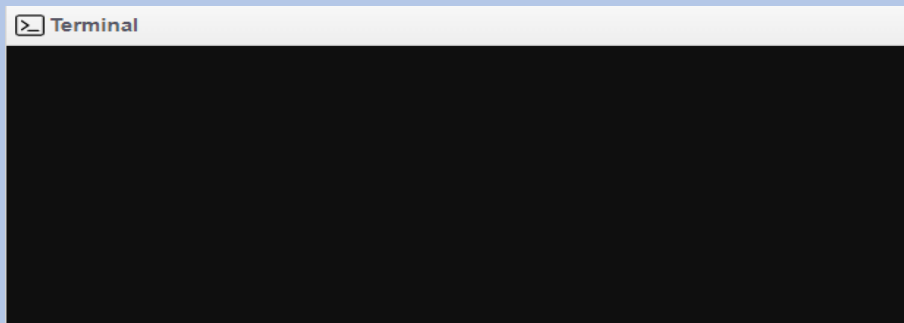
clear command

❖ The [clear command](#) is a standard command to clear the terminal screen.

Command-: *This was the terminal before the command.

```
1 $ ls
2 Demo
3 files.txt Linked main.sh NewFile Second
4 $ pwd
5 /home/cg/root/638c34db4d98e
6 $ cp Linked Non-Linked
7 cp: -r not specified; omitting directory 'Linked'
8 $ clear
```

Output-:



ps command

❖ [ps command](#) in Linux is used to check the active processes in the terminal.

Command-:

```
1 ps
```

Output-:

```
PID TTY          TIME CMD
8454 pts/521    00:00:00 bash
11982 pts/521    00:00:00 bash
11983 pts/521    00:00:00 ps
```

man command

- ❖ The `man command` displays a user manual for any commands or utilities available in the Terminal, including their name, description, and options.

Command-:

```
1 man -f ls
```

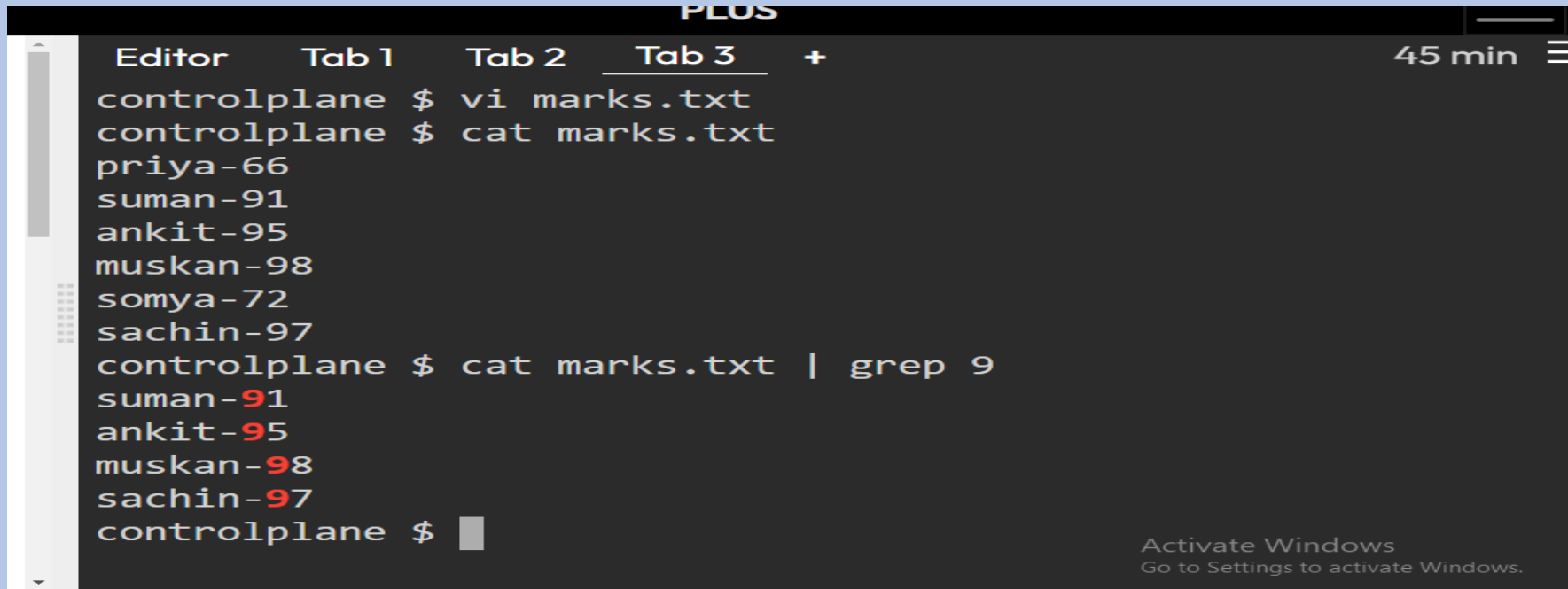
Output-:

```
ls (1) - list directory contents
```

grep command

*The [grep command](#) is used to find a specific string in a series of outputs.

Example-:

A terminal window titled 'PLEOS' with three tabs: 'Editor', 'Tab 1', and 'Tab 2'. The 'Editor' tab is active. The terminal shows a sequence of commands and their outputs. First, 'vi marks.txt' is run, followed by 'cat marks.txt', which displays a list of names and scores: priya-66, suman-91, ankit-95, muskan-98, somya-72, and sachin-97. Then, the command 'cat marks.txt | grep 9' is executed, resulting in the same list of names and scores, but with the scores 91, 95, 98, and 97 highlighted in red. The terminal ends with a prompt 'controlplane \$'.

```
PLEOS
Editor  Tab 1  Tab 2  Tab 3  +
45 min

controlplane $ vi marks.txt
controlplane $ cat marks.txt
priya-66
suman-91
ankit-95
muskan-98
somya-72
sachin-97
controlplane $ cat marks.txt | grep 9
suman-91
ankit-95
muskan-98
sachin-97
controlplane $
```

Activate Windows
Go to Settings to activate Windows.

echo command

❖ **echo command** in Linux is specially used to print something in the terminal.

Command-:

```
1 echo "Hello World"
```

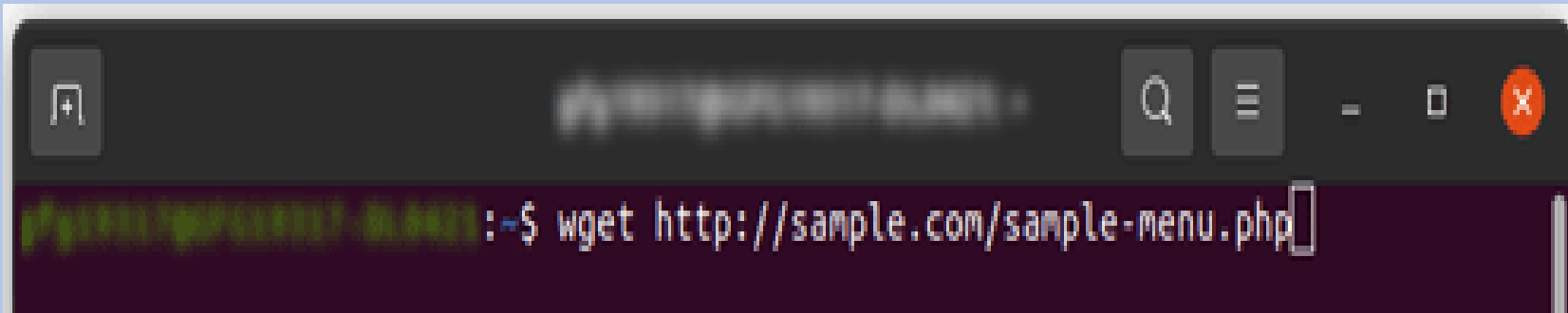
Output-:

```
Hello World
```

wget command

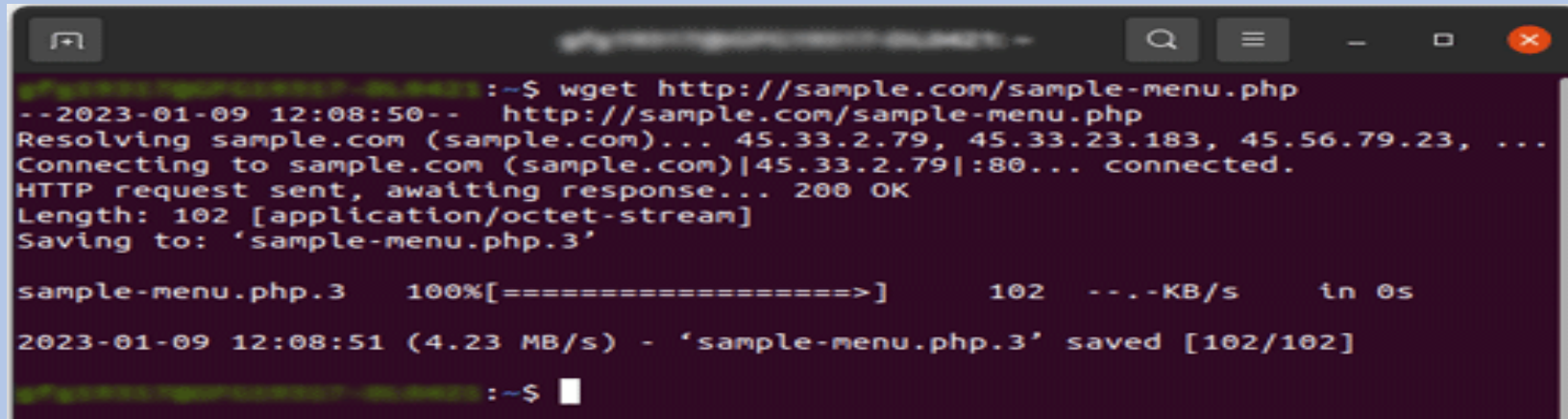
- ❖ The [wget command](#) in the Linux command line allows you to download files from the internet. It runs in the background and does not interfere with other processes.

Command-:

A terminal window with a dark background. The command 'wget http://sample.com/sample-menu.php' is being entered at the prompt. The window has standard Linux window controls (minimize, maximize, close) and a search icon in the top right corner.

```
python@python:~$ wget http://sample.com/sample-menu.php
```

Output-:

A terminal window showing the output of the wget command. The output includes the command being run, the URL being resolved, the connection status, the HTTP response (200 OK), the file length (102 bytes), and the file being saved as 'sample-menu.php.3'. A progress bar shows 100% completion. The window has standard Linux window controls and a search icon in the top right corner.

```
python@python:~$ wget http://sample.com/sample-menu.php
--2023-01-09 12:08:50--  http://sample.com/sample-menu.php
Resolving sample.com (sample.com)... 45.33.2.79, 45.33.23.183, 45.56.79.23, ...
Connecting to sample.com (sample.com)|45.33.2.79|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 102 [application/octet-stream]
Saving to: 'sample-menu.php.3'

sample-menu.php.3  100%[=====]          102  --.-KB/s   in 0s

2023-01-09 12:08:51 (4.23 MB/s) - 'sample-menu.php.3' saved [102/102]

python@python:~$
```


whoami command

- ❖ The [whoami command](#) provides basic information that is extremely useful when working on multiple systems. In general, if you are working with a single computer, you will not require it as frequently as a network administrator.

Command-:

```
1 whoami
```

Output-:

```
acer
```



cal command

- ❖ The [cal command](#) is not the most famous command in the terminal but it functions to view the calendar for a particular month in the terminal. Let's see how this works.

Command-:

```
1 cal January 2023
```

Output-:

```
January 2023
Su Mo Tu We Th Fr Sa
 1  2  3  4  5  6  7  8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
```

whereis command

❖ [whereis command](#) in Linux is generally used to see the exact location of any command typed after this. Let's see how this performs.

Command-:

```
1 whereis printf
```

Output-:

```
printf: /usr/bin/printf /usr/include/printf.h
```

df command

❖ [df command](#) in Linux gets the details of the file system.

Command-:

```
1 df -h
```

Output-:

Filesystem	Size	Used	Avail	Use%	Mounted on
overlay	875G	120G	711G	15%	/
tmpfs	63G	0	63G	0%	/dev
tmpfs	63G	0	63G	0%	/sys/fs/cgroup
/dev/nvme0n1p3	875G	120G	711G	15%	/dev/init
shm	64M	0	64M	0%	/dev/shm
tmpfs	63G	0	63G	0%	/proc/acpi
tmpfs	63G	0	63G	0%	/proc/scsi
tmpfs	63G	0	63G	0%	/sys/firmware

wc command

[wc command](#) in Linux indicates the number of words, characters, lines, etc using a set of options.

- **wc -w** shows the number of words
- **wc -l** shows the number of lines
- **wc -m** shows the number of characters present in a file

Command-:

```
1 touch file.txt
2 echo -e "This file has only six words" > file.txt
3 wc -w file.txt
```

Output-:

```
6 file.txt
```



Thank
You ♥