

Assignment 02 — Basics of Linux and Open-Source Tools

Course: ETCCCP105

— Computer Science Fundamentals & Career Pathways

Name: *Agrima Mishra*

Student ID: 2501010207

Section : B

Semester: Odd

Date of submission: 18 – Nov – 2025

Link: [Git Hub link](#)

Introduction

Linux is one of the most important operating systems used in the fields of programming, networking, cybersecurity, and servers. Unlike Windows, Linux works mainly through commands, which makes it faster and more flexible. In this assignment, I learned how to install Linux using Windows Subsystem for Linux (WSL), which allowed me to run Ubuntu directly on my Windows laptop without needing a separate computer or virtual machine. After installation, I explored different Linux commands such as creating folders, copying and moving files, checking system information, managing processes, and viewing network details.

I also learned how to write simple shell scripts. These scripts helped me automate tasks like taking backups, monitoring the system, and downloading files. Working with the Linux terminal gave me practical experience, and I got a clear idea of how real programmers and cybersecurity professionals use Linux every day. This introduction to Linux has helped me understand the basics in a simple and hands-on way.

COMMAND TABLE

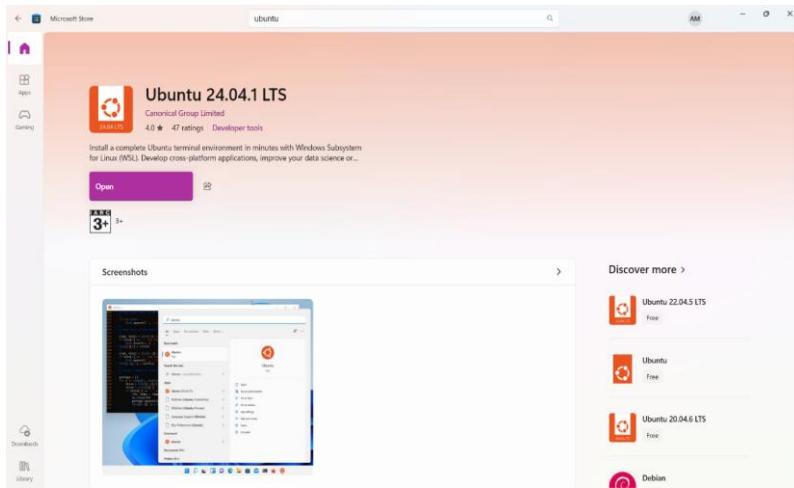
Command (syntax)	Short description	When / why used
pwd	Print Working Directory — shows full path of current directory.	Confirm where you are in the filesystem before running file operations or giving paths.
ls / ls -l / ls -a	List files and directories (long format, show hidden files).	Inspect directory contents, check file names, permissions, sizes and hidden files.
cd <dir> / cd // cd	Change directory (cd with no args → home).	Move around the filesystem to the directory where you want to work.
tree or ls -R	Show directory tree recursively.	Get a quick overview of folder structure (useful before backups or packaging).
mkdir <dir>	Create a new directory.	Create project folders, output directories, or structure for scripts.
touch <file>	Create an empty file (or update timestamp).	Quickly create placeholder files or update file timestamps.
cp <src> <dest>	Copy files or directories (cp -r for directories).	Make backups or duplicate files before editing.
mv <old> <new>	Move or rename files/directories.	Rename files or move them into other directories for organization.
rm <file> / rm -r <dir>	Remove file or directory (recursive).	Delete unwanted files/folders — use carefully (no recycle bin).
chmod <mode> <file>	Change file permissions (e.g., chmod 755 file).	Make scripts executable, restrict or allow read/write/execute access.

Command (syntax)	Short description	When / why used
ps / ps aux	List processes running in current shell or system.	Find PIDs to manage or investigate running processes.
top	Real-time interactive process monitor (CPU/memory).	Troubleshoot high CPU/memory usage and see active processes live.
sleep <seconds> &	Run sleep in background (use & to background).	Create a simple background job for testing background/kill behavior.
kill <PID> / kill -9 <PID>	Send signal to process (terminate).	Stop misbehaving or test processes by PID; -9 forces kill if needed.
ping <host> -c <n>	Send ICMP echo requests (e.g., ping google.com -c 4).	Check network reachability and round-trip latency to a host.
ip a (or ip addr show)	Show network interfaces and IP addresses.	Verify IP addresses, interfaces status, and troubleshoot networking.
netstat -tulpn (or ss -tulpn)	Show listening ports and active network sockets.	Find which services are bound to which ports for troubleshooting.
date	Display current date and time (formatable).	Timestamp logs, name backup files, or verify system clock.
df -h	Show disk free/used space (human-readable).	Check available storage before copying, backing up, or installing packages.
uname -a	Show kernel and system information (OS, kernel version, arch).	Document system details for reports, debugging or compatibility checks.
history	Show shell command history.	Provide an audit trail of commands you ran for documentation or grading.

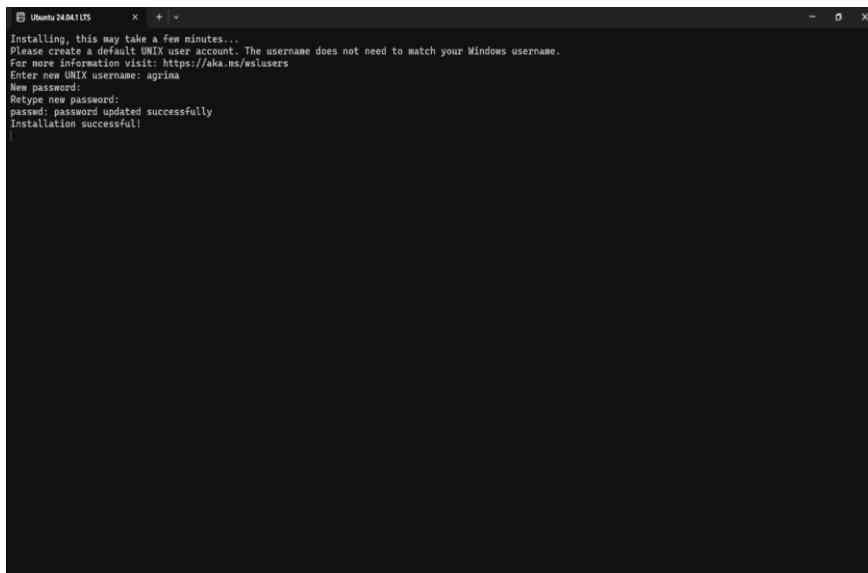
LINUX INSTALLATION DOCUMENTATION

Platform used: Windows Subsystem for Linux (WSL) — Ubuntu 24.04 LTS

Step 1- Installed Ubuntu 24.04 from Microsoft Store.



Step 2- Opening Ubuntu installing and creating the username for the first terminal login.



Step 3- Updating the Ubuntu by using the following commands

```
sudo apt update
```

```
sudo apt upgrade -y
```

```
aptimp@AGMA - x + * -
```

Installing this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: <https://aka.ms/wslusers>

Enter new UNIX username: agrima
New password:
Retype password:
password: password updated successfully
Installation successful

To log in as the administrator (user "root"), use "sudo <command>".

See "["Run sudo_root"](#)" for details.

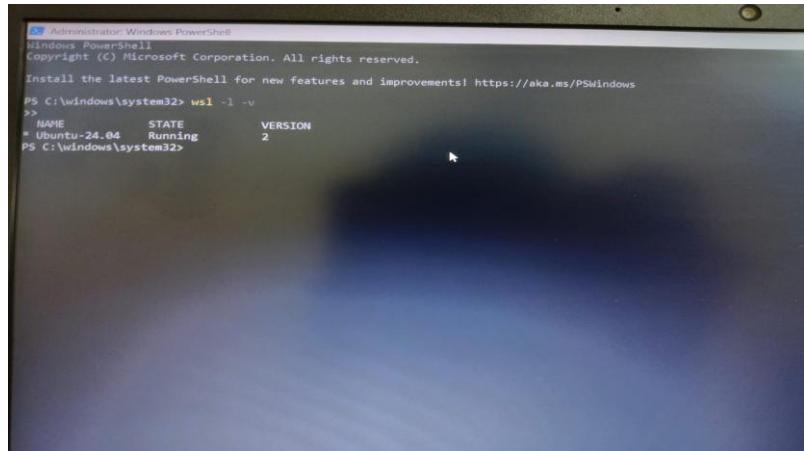
```
agrima@AGMA:~$ sudo apt update  
[sudo] password for agrima:  
sudo apt upgrade -y  
[sudo] password for agrima:  
Get:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]  
Get:2 http://archive.ubuntu.com/ubuntu noble InRelease  
Get:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Get:4 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1309 kB]  
Get:5 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Get:6 http://archive.ubuntu.com/ubuntu noble-security/main amd64 Packages [15.8 MB]  
Get:7 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [215 kB]  
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21.5 kB]  
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [100 kB]  
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [980 kB]  
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [208 kB]  
Get:12 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.1 kB]  
Get:13 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [13.4 kB]  
Get:14 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [486 kB]  
Get:15 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [210 kB]  
Get:16 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [486 kB]  
Get:17 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [33 kB]  
Get:18 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Metadata [500 B]  
Get:19 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [27.4 kB]  
Get:20 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [506 B]  
Get:21 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n= Metadata [312 B]  
Get:22 http://archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]  
Get:23 http://archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]  
GPN [ 6 Packages store 8 8 ] {23 Components- amd64 23.8 kB/3871 kB 14 }
```

```
agrima@AGRIMA: ~          x  +  x

Setting up landscape-common (24.02-0ubuntu5.6) ...
Setting up netplan.io (1.1.2+git20200610-0ubuntu1) ...
Setting up libsystemd-dbus (250.0-0ubuntu0.24.04.2) ...
Setting up python3-software-properties (0.99.49.3) ...
Setting up python3-requests (2.31.0+dfsg-1ubuntu1.1) ...
Setting up liblandscape-client (2.02-0ubuntu0.6) ...
Setting up libsystemd0 (250.0-0ubuntu0.24.04.2) ...
Installing new version of config file /etc/apparmor.d/ubuntu_pro_apt_news ...
Installing new version of config file /etc/apparmor.d/ubuntu_pro_esp_cache ...
Installing new version of config file /etc/apt/apt.conf.d/20apt-esp-hook.conf ...
Setting up libsystemd0 (250.0-0ubuntu0.24.04.2) ...
Setting up ubuntu-pro-client-110n (20ubuntu0.24.04) ...
Setting up ufw (0.1.18-20.04.3) ...
Setting up ufw-privo-service (0.1.18-20.04.3) ...
Setting up liblqr-1-mesalibamdgpu (25.0.7-0ubuntu0.24.04.2) ...
Setting up cloud-init (0.20.0-20.04.2) ...
Installing new version of config file /etc/cloud/templates/sources.list.debian.deb822.tmpl ...
Installing new version of config file /etc/cloud/templates/sources.list.ubuntu.deb822.tmpl ...
Setting up ubuntu-minimal (1.559.2) ...
Setting up libsystemd0 (250.0-0ubuntu0.24.04.2) ...
Setting up libapt-inst-3.0-0ubuntu1.3 (3.24.01-0ubuntu1.3) ...
Setting up ufw-setup (0.5.10-24.04) ...
Installing new version of config file /etc/update-motd.d/99-wsl ...
Setting up libapt-inst-3.0-0ubuntu1.3 (3.24.01-0ubuntu1.3) ...
Setting up apt-report-control-center (2.28.1-0ubuntu3.8) ...
Setting up apt-report (2.28.1-0ubuntu1.8) ...
apt-report-autoport.service is a disabled or a static unit not running, not starting it.
Setting up python3-disupgrade (1:24.04.12) ...
Setting up libapt-inst-3.0-0ubuntu1.3 (3.24.01-0ubuntu1.3) ...
Setting up python3-update-manager (1:24.04.12) ...
Setting up update-manager-core (1:24.04.12) ...
Processing triggers for sgml-base (1:2.3-1ubuntu1) ...
Processing triggers for libfontconfig1 (7.1-3build2) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.6) ...
Processing triggers for systemd (255.4-0ubuntu8.11) ...
Processing triggers for libfontconfig1 (7.1-3build2) ...
Processing triggers for dbus (1.10.18-0ubuntu0.1) ...
Setting up packagekit (1.2.8-2ubuntu1.2) ...
Setting up packagekit-tools (1.2.8-2ubuntu1.2) ...
Setting up software-properties-common (0.99.49.3) ...
Setting up ubuntu-wsl (1.559.2) ...
agrima@AGRIMA: ~
```

Step 4- Checking WSL version using the command

wsl -l -v



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

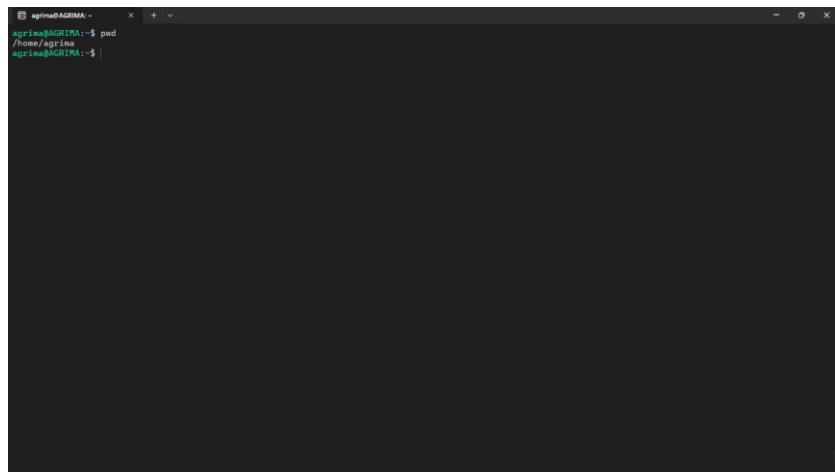
PS C:\windows\system32> wsl -l -v
>
NAME          STATE      VERSION
* Ubuntu-24.04    Running       2
PS C:\windows\system32>
```

LINUX COMMANDS

1. File Navigation:

1). **pwd**- Print Working Directory — shows current directory path.

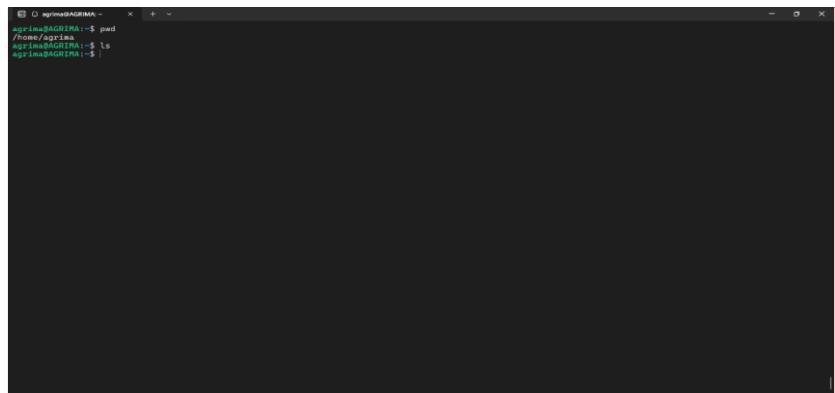
Used - Use to confirm your current location in filesystem.



```
agrimal@AGRIMA: ~
agrimal@AGRIMA:~$ pwd
/home/agrima
agrimal@AGRIMA:~$ |
```

2). **ls** - List files in the directory. examples: ls -l, ls -a

Used - Inspect files and directories.

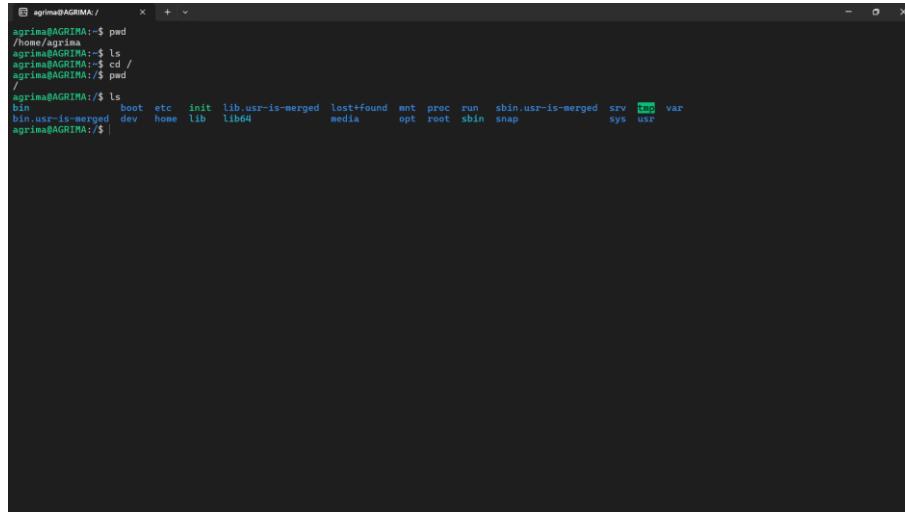


```
agrimal@AGRIMA: ~
agrimal@AGRIMA:~$ pwd
/home/agrima
agrimal@AGRIMA:~$ ls
agrimal@AGRIMA:~$ |
```

Commented [AM1]: ls shows files except hidden ones while ls -a command shows all the files including the hidden ones

3). **cd/** - Change directory. cd / goes to root.

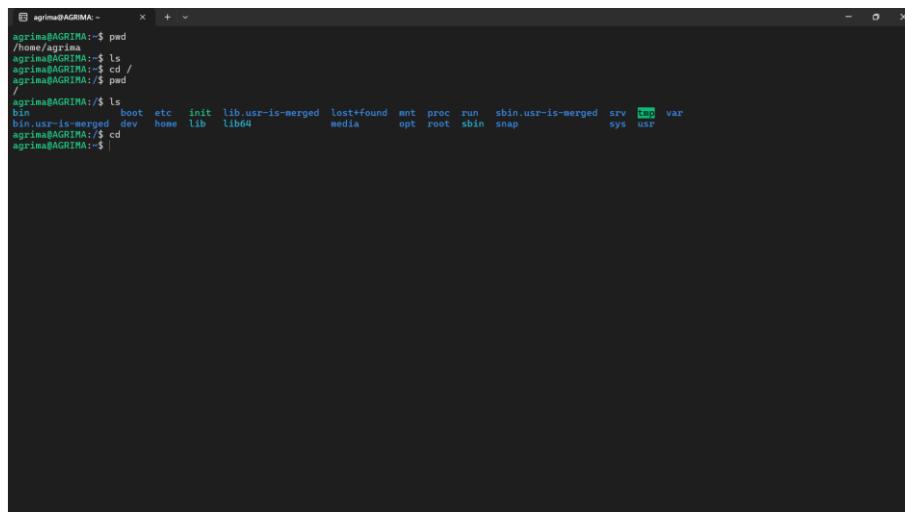
Used - Navigate the filesystem.



```
agrimald@AGRIMA:~$ pwd
/home/agrimald
agrimald@AGRIMA:~$ ls
agrimald@AGRIMA:~$ cd /
agrimald@AGRIMA:/$ pwd
/
agrimald@AGRIMA:/$ ls
bin  boot  etc  init  lib  lib usr-is-merged  lost+found  mnt  proc  run  sbin  sbin usr-is-merged  srv  tmp  var
bin usr-is-merged  dev  home  lib  lib64  media  opt  root  sbin  snap  sys  usr
agrimald@AGRIMA:/$ |
```

4). **cd** is used to **change directory**, return to home directory.

Used – Navigate filesystem with ease accessing directory and exiting back to home directory through this command.



```
agrimald@AGRIMA:~$ pwd
/home/agrimald
agrimald@AGRIMA:~$ ls
agrimald@AGRIMA:~$ cd /
agrimald@AGRIMA:/$ pwd
/
agrimald@AGRIMA:/$ ls
bin  boot  etc  init  lib  lib usr-is-merged  lost+found  mnt  proc  run  sbin  sbin usr-is-merged  srv  tmp  var
bin usr-is-merged  dev  home  lib  lib64  media  opt  root  sbin  snap  sys  usr
agrimald@AGRIMA:/$ |
```

Commented [AM2]: Cd / is root directory while cd is home directory

5). tree –

Installing the demo tree consisting sample directories and files in tree form.

-- By **sudo apt install tree** command

Commented [AM3]: Used a tree installing command to show tree which are files and folders with the root directory or parent they are in

```
aptitude upgrade -y <--  
[sudo] password for optimus:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  libgl1-mesa-dri libglapi-mesa libgbm-1.0-1 libgles2-mesa libgles2-dri=2.0.0-0ubuntu1  
Use 'sudo apt autoremove' to remove them.  
The following NEW packages will be installed:  
  tree  
tree is already the newest version.  
0 to remove and 2 not upgraded.  
Nothing to get or upgrade.  
After this operation, 111 kB of additional disk space will be used.  
Do you want to continue? [Y/n] y  
Reading package lists... done  
Building dependency tree... done  
Reading state information... done  
Fetching 47.4 kB in 2s (24.9 kB/s)  
Preparing to unpack .../tree_2.1.1~ubuntu3.24.04.2_amd64.deb ...  
(Reading database ... 40882 files and directories currently installed.)  
Preparing to unpack .../tree_2.1.1~ubuntu3.24.04.2_amd64.deb ...  
Unpacking tree (2.1.1~ubuntu3.24.04.2) ...  
Setting up tree (2.1.1~ubuntu3.24.04.2) ...  
Processing triggers for man-db (2.12.0-1ubuntu2) ...  
processingtree-3  
optimus@Opti
```

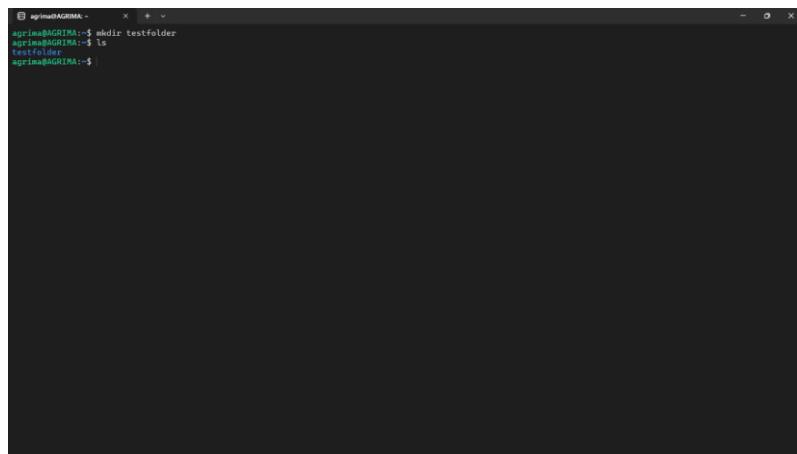
File & Directory Management Commands:

1). **mkdir**- make directory.

mkdir testfolder created a folder.

Syntax- mkdir <folder name>

Used- Create project or storage directories.



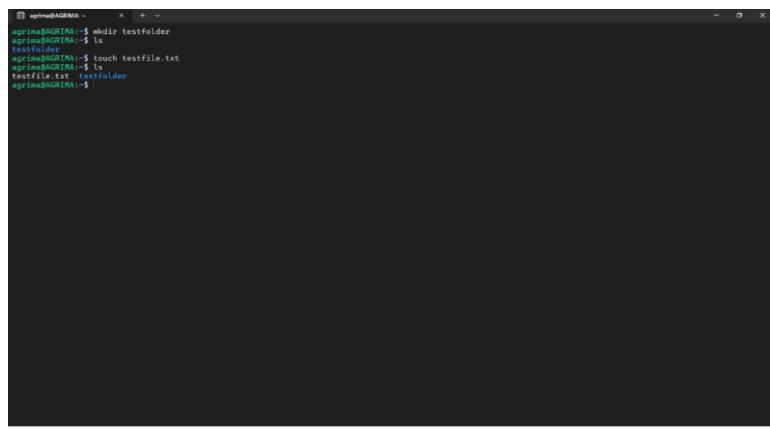
```
agrimal@AGRIMA:~$ mkdir testfolder
agrimal@AGRIMA:~$ ls
testFolder
agrimal@AGRIMA:~$
```

2). **touch** - Create an empty file.

Syntax- touch <file>

--touch testfile.txt.

Used- Quickly create files to test or use as placeholders.

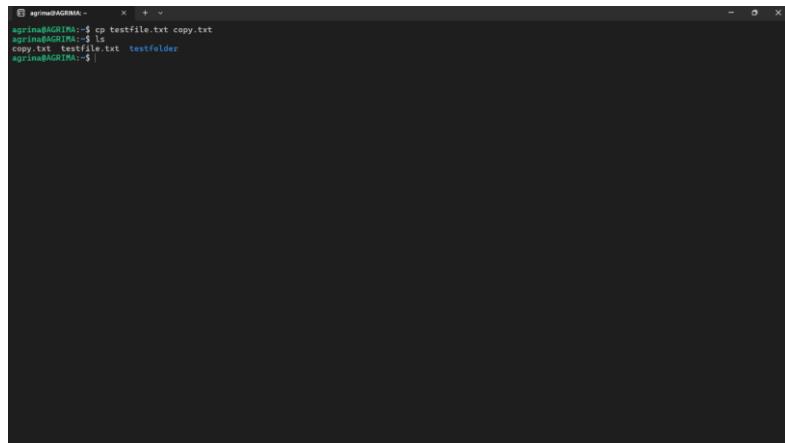


```
agrimal@AGRIMA:~$ mkdir testFolder
agrimal@AGRIMA:~$ ls
testFolder
agrimal@AGRIMA:~$ touch testfile.txt
agrimal@AGRIMA:~$ ls
testfile.txt
testFolder
agrimal@AGRIMA:~$
```

3). cp – Copy files

Syntax- cp source dest

Used- Duplicate files for backup or editing safely.



```
aprina@AGRIMA:~$ cp testfile.txt copy.txt
aprina@AGRIMA:~$ cp copy.txt testfile.txt testfolder
aprina@AGRIMA:~$
```

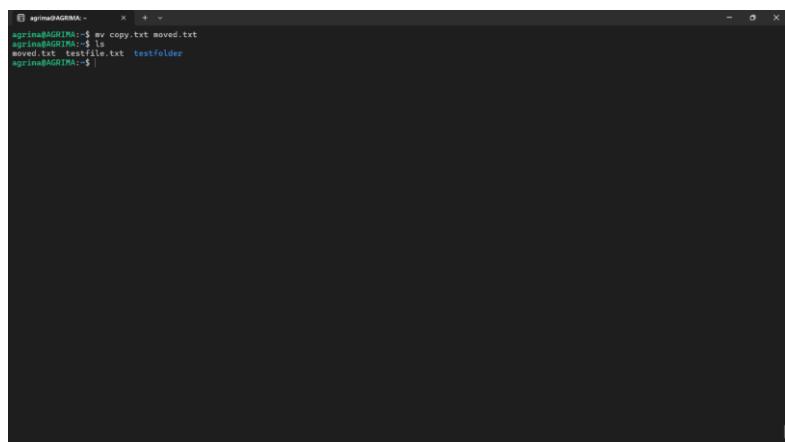
A terminal window titled "Terminal" showing the execution of the cp command. The user copies "testfile.txt" to "copy.txt" and then moves "copy.txt" to "testfile.txt" in the "testfolder" directory. The terminal window has a dark background with white text.

4). mv - Move or rename files.

Rename files or relocate them.

Syntax- mv old.txt new.txt

--- mv copy.txt moved.txt



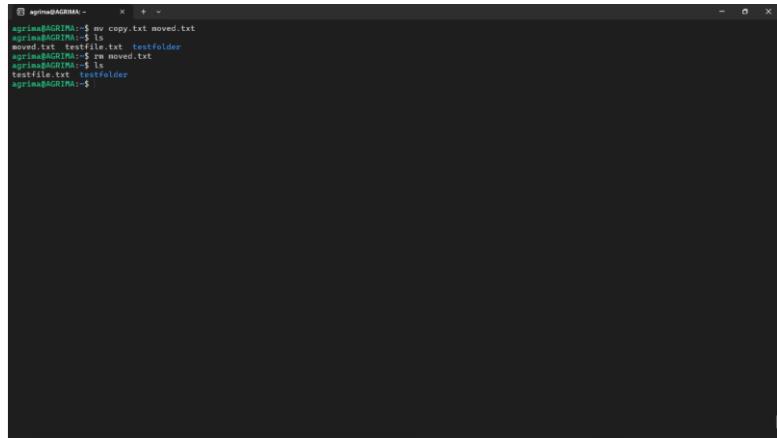
```
aprina@AGRIMA:~$ mv copy.txt moved.txt
aprina@AGRIMA:~$ ls
moved.txt testfile.txt testfolder
aprina@AGRIMA:~$
```

A terminal window titled "Terminal" showing the execution of the mv command. The user renames "copy.txt" to "moved.txt". The terminal window has a dark background with white text.

5). rm – remove files

Syntax- rm <file>

Delete unwanted files



```
aparna@Agneta: ~
aparna@Agneta:~$ mv copy.txt moved.txt
aparna@Agneta:~$ ls
moved.txt testfile.txt testfolder
aparna@Agneta:~$ mv moved.txt
aparna@Agneta:~$ ls
testfile.txt testfolder
aparna@Agneta:~$
```

Commented [AM4]: rm is used to remove a file but the individual has to be careful cause the deleted file won't be retrieved back again through any means

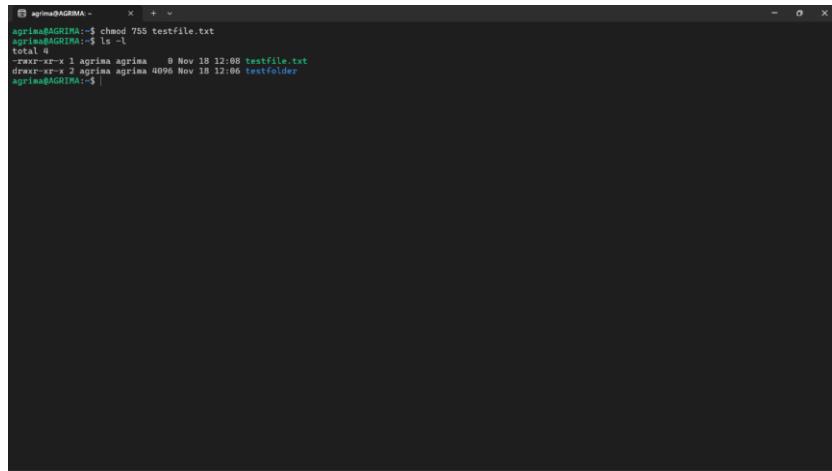
PERMISSIONS COMMANDS:

1). chmod- change permissions.

chmod 755 testfile.txt sets rwx r-x r-x.

Syntax- chmod 755 file

Controls read (r), write (w), execute (e) permissions



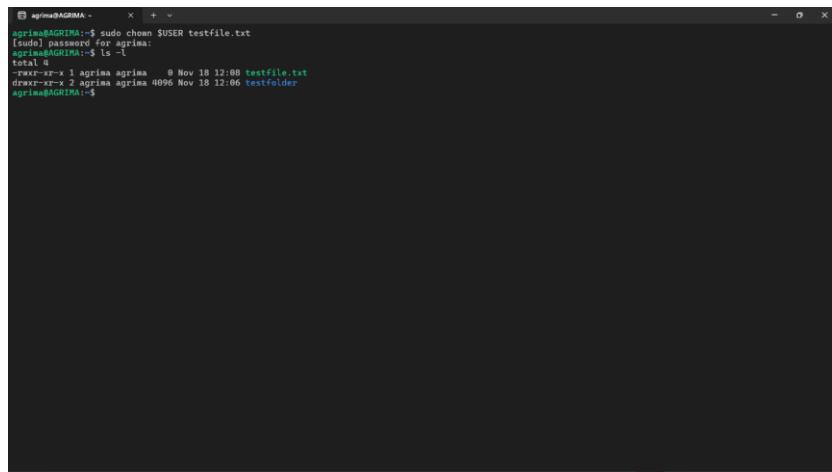
```
agrimal@AGRIMA:~$ chmod 755 testfile.txt
agrimal@AGRIMA:~$ ls -l
total 4
-rwxr-x-- 1 agrimal agrimal 0 Nov 18 12:08 testfile.txt
drwxr-xr-x 2 agrimal agrimal 4096 Nov 18 12:06 testfolder
agrimal@AGRIMA:~$
```

Commented [AM5]: These commands are very sensitive for security and privacy related task as it gives access to ones system and the charge which might create troubles and serious consequences if not handled sensitively and carefully hence one mustn't share or allow any of these without proper thinking or knowledge

2). chown- change the ownership of the file

Syntax- sudo chown \$USER file

Fix ownership issues or set for deployment.



```
agrimal@AGRIMA:~$ sudo chown $USER testfile.txt
[sudo] password for agrimal:
agrimal@AGRIMA:~$ ls -l
total 4
-rwxr-x-- 1 agrimal agrimal 0 Nov 18 12:08 testfile.txt
drwxr-xr-x 2 agrimal agrimal 4096 Nov 18 12:06 testfolder
agrimal@AGRIMA:~$
```

PROCESS MONITORING COMMANDS

1). **ps** - Show current shell processes.

ps (or ps aux)

Check running processes and their PIDs.

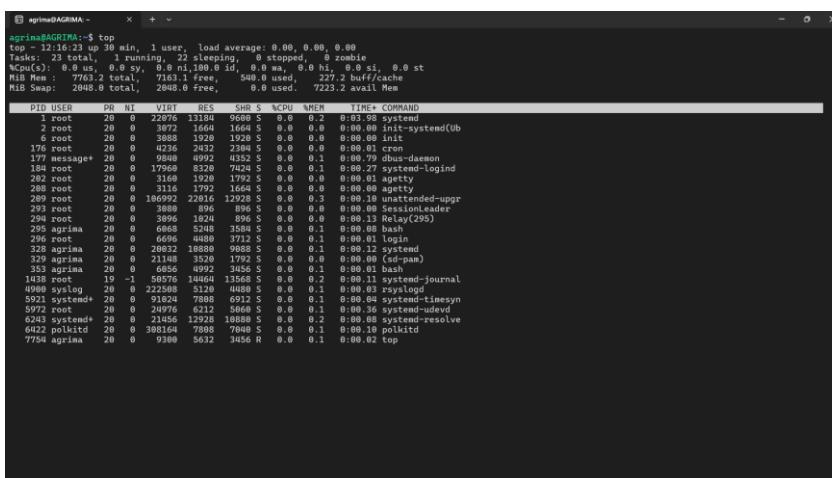


```
aprima@AGRIMA:~$ ps
PID TTY      TIME CMD
299 pts/0    00:00:00 bash
7751 pts/0    00:00:00 ps
aprima@AGRIMA:~$
```

2). **top**- Real-time process monitor (or htop if installed).

top

Monitor CPU, memory and running processes.



```
aprima@AGRIMA:~$ top
top: 12:16:23 up 30 min, 1 user, load average: 0.00 0.00 0.00
Tasks: 23 total, 0 running, 23 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
Mem: 7763.2 total, 7163.1 free, 540.0 used, 227.2 buff/cache
Swap: 2048.0 total, 2048.0 free, 0.0 used, 7223.2 avail Mem

PID USER      PR  NI  VIRT   RES   SHR S %CPU %MEM TIME+ COMMAND
 1 root      20   0 22876 13180  9688 S  0.0  0.2  0:03:98 systemd
 2 root      20   0 10240  10240  1024 R  0.0  0.0  0:00:00 systemd[0]
 3 root      20   0 3868  1929  1928 S  0.0  0.0  0:00:00 init
176 root      20   0 4236  2432  2384 S  0.0  0.0  0:00:01 cron
177 message+ 20   0 9840  4995  4352 S  0.0  0.0  0:00:79 dbus-daemon
180 root      20   0 17376  9126  8232 S  0.0  0.0  0:00:01 systemd-logind
262 root      20   0 3169  1920  1792 S  0.0  0.0  0:00:01 getty
268 root      20   0 3116  3792  1664 S  0.0  0.0  0:00:00 getty
289 root      20   0 18940  22716  12584 S  0.0  0.0  0:00:00 gdm@display0:upgr
291 root      20   0 3868  894  596 S  0.0  0.0  0:00:00 gdm@display0:leader
294 root      20   0 3896  1824  596 S  0.0  0.0  0:00:13 Relay(259)
295 agrima    20   0 6668  5240  3584 S  0.0  0.0  0:00:08 bash
296 agrima    20   0 6668  5240  3584 S  0.0  0.0  0:00:08 bash
328 agrima    20   0 20832  10880  9988 S  0.0  0.1  0:00:12 systemd
329 agrima    20   0 21148  3520  1792 S  0.0  0.0  0:00:00 (sd-pam)
333 agrima    20   0 6856  4992  4992 S  0.0  0.0  0:00:01 bash
1038 agrima   19  -1 58544 10456 13568 S  0.0  0.2  0:00:01 systemd-journal
4980 syslog    20   0 222588 5120 41888 S  0.0  0.1  0:00:03 rsyslogd
5921 systemd+ 20   0 91824 7886 6912 S  0.0  0.0  0:00:04 systemd-timesync
5979 systemd+ 20   0 10240  10240  1024 S  0.0  0.0  0:00:00 systemd-timesync
6241 systemd+ 20   0 21456 12928 10888 S  0.0  0.2  0:00:08 systemd-resolve
6422 polkitd+ 20   0 388164 7888 7048 S  0.0  0.1  0:00:10 polkitd
7754 agrima    20   0 9308  5632  3456 R  0.0  0.1  0:00:02 top
```

3). Kill – starts a background job and kills it

Syntax- sleep 1000 & + kill <PID>

Demonstrates starting background tasks and stopping them safely.

```
agrima@AGRIMA:~$ sleep 1000 &
[1] 7788
agrima@AGRIMA:~$
```

Commented [AM6]: A fake background task is started through sleep command to perform the kill command

```
agrima@AGRIMA:~$ sleep 1000 &
[1] 7788
PID TTY TIME CMD
295 pts/0 00:00:00 bash
7788 pts/0 00:00:00 sleep
7788 pts/0 00:00:00 ps
agrima@AGRIMA:~$
```

```
agrima@AGRIMA:~$ sleep 1000 &
[1] 7788
PID TTY TIME CMD
295 pts/0 00:00:00 bash
7788 pts/0 00:00:00 sleep
7788 pts/0 00:00:00 ps
agrima@AGRIMA:~$ kill 7788
agrima@AGRIMA:~$ ps
PID TTY TIME CMD
295 pts/0 00:00:00 bash
7798 pts/0 00:00:00 ps
[1] 7788 terminated sleep 1000
agrima@AGRIMA:~$
```

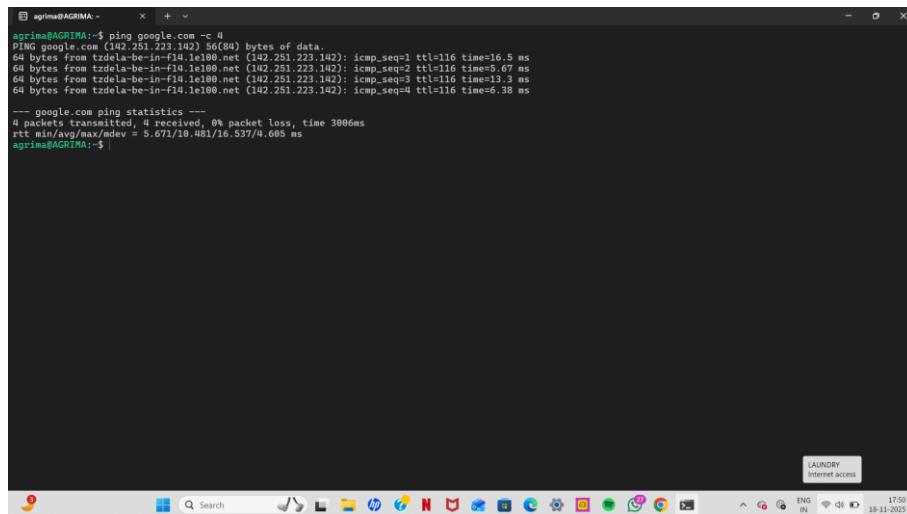
Commented [AM7]: Start a long-running background process

NETWORKING COMMANDS:

1). ping- Send ICMP echo requests to check connectivity.

ping google.com -c 4

Test network reachability and latency.

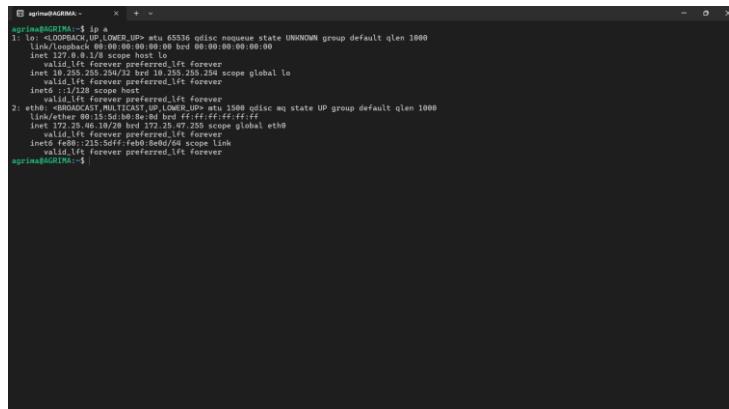


```
agrimal@AGRIMA:~$ ping google.com -c 4
PING google.com (142.251.223.142) 56(80) bytes of data.
40 bytes from tdeala-be-in-f14.1e100.net (142.251.223.142): icmp_seq=1 ttl=116 time=16.5 ms
64 bytes from tdeala-be-in-f14.1e100.net (142.251.223.142): icmp_seq=2 ttl=116 time=5.67 ms
64 bytes from tdeala-be-in-f14.1e100.net (142.251.223.142): icmp_seq=3 ttl=116 time=13.3 ms
64 bytes from tdeala-be-in-f14.1e100.net (142.251.223.142): icmp_seq=4 ttl=116 time=6.38 ms
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 5.671/10.481/16.537/4.685 ms
agrimal@AGRIMA:~$ |
```

2). ip - show network interfaces & IP addresses (modern replacement for ifconfig).

Syntax- ip a

Check IP, interface status.



```
agrimal@AGRIMA:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 127.255.255.255 scope global lo
        valid_lft forever preferred_lft forever
        link-layer brd ff:ff:ff:ff:ff:ff
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:00:00:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.32/24 brd 192.168.200.255 scope global eth0
        valid_lft forever preferred_lft forever
        link-layer brd ff:ff:ff:ff:ff:ff
    inet6 fe80::5e0:8eff:fe00:32/64 scope link
        valid_lft forever preferred_lft forever
agrimal@AGRIMA:~$
```

3). netstat -tulpn - Show active listening ports (install net-tools if needed).

Syntax- netstat -tulpn

find services bound to ports and troubleshoot network services.

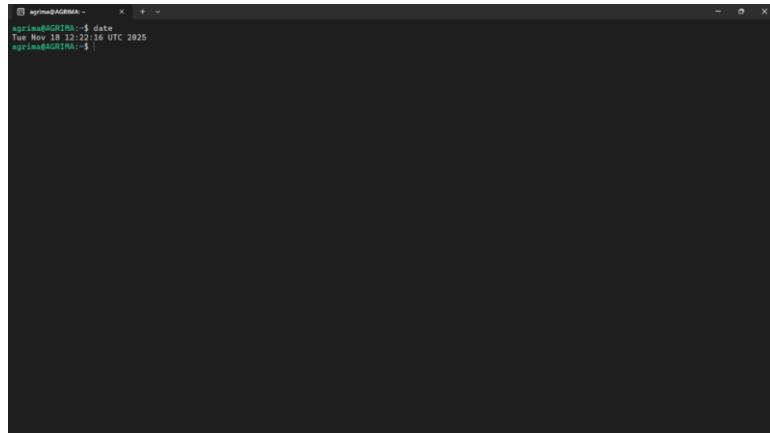
```
agrina@AGRIMA:~$ sudo apt install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
libdriv-nouveau2 libdriv-radeon1 libgl1-mesa libglapi-mesa liblvm1:i386 libxcb-dri2-0
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
net-tools
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 204 kB of archives.
After this operation, 813 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal amd64 net-tools amd64 2.10-0.lubuntu4.0 [204 kB]
Fetched 204 kB in 2s (129 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 48898 files and directories currently installed.)
Preparing to unpack .../net-tools_2.10-0.lubuntu4.0_amd64.deb ...
Unpacking net-tools (2.10-0.lubuntu4.0) ...
Setting up net-tools (2.10-0.lubuntu4.0) ...
Processing triggers for man-db (2.12.0-4build2) ...
agrina@AGRIMA:~$
```

```
agrina@AGRIMA:~$ netstat -tulpn
(Not all processes could be identified. Non-rooted process info
will not be shown. You would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53          0.0.0.0:*             LISTEN      -
tcp        0      0 127.0.0.54:53          0.0.0.0:*             LISTEN      -
tcp        0      0 127.0.0.54:2045        0.0.0.0:*             LISTEN      -
udp        0      0 127.0.0.54:53          0.0.0.0:*             LISTEN      -
udp        0      0 127.0.0.53:53          0.0.0.0:*             LISTEN      -
udp        0      0 127.0.0.253:2045       0.0.0.0:*             LISTEN      -
udp        0      0 127.0.0.1:723          0.0.0.0:*             LISTEN      -
udp6       0      0 ::1:323               ::*:*                  LISTEN      -
agrina@AGRIMA:~$
```

4). date – Show current date/time.

Syntax- date

Log timestamps; used in scripts for naming backups.

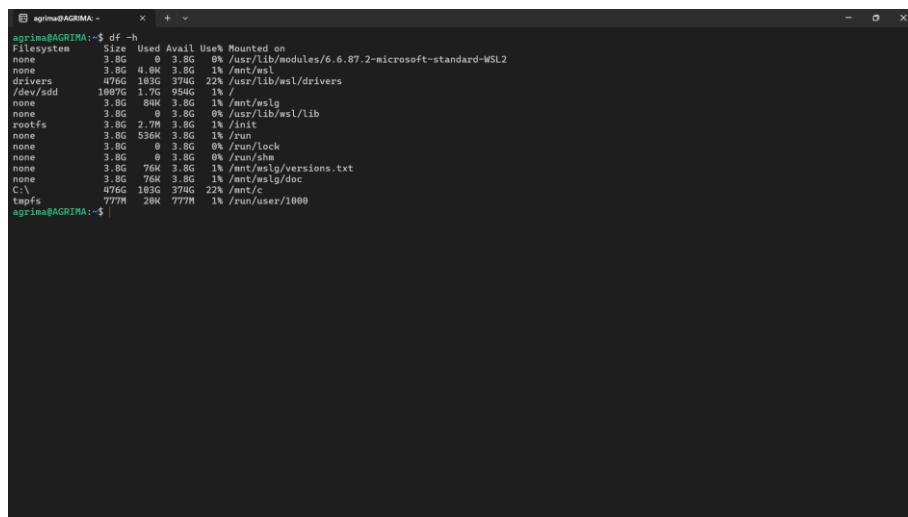


```
agrimald@AGRIMA:~$ date
Tue Nov 18 12:22:16 UTC 2025
agrimald@AGRIMA:~$ |
```

5). df -h - disk usage (human readable)

Syntax- df -h

Check free space before large operations like backups.



```
agrimald@AGRIMA:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
none            3.8G   0B  3.8G  0% /usr/lib/modules/6.6.87.2-microsoft-standard-WSL2
none            3.8G  4.0B  3.8G  1% /mnt/wsl
none            3.8G  3.8G  3.8G  0% /mnt/wsl/drivers
/dev/sda       18000M  1.7K  17999M  1% /dev/sda
none            3.8G  504K  3.8G  1% /mnt/wslg
none            3.8G  0  3.8G  0% /usr/lib/wsl/lib
none            3.8G  2.7M  3.8G  1% /init
rootfs          3.8G  536M  3.8G  1% /run
none            3.8G  0  3.8G  0% /run/lock
none            3.8G  0  3.8G  0% /run/shm
none            3.8G  76K  3.8G  1% /mnt/wslg/versions.txt
none            3.8G  76K  3.8G  1% /mnt/wslg/doc
C:\N          476G  103G  373G  22% /mnt/c
tmpfs          777M  20K  777M  1% /run/user/1000
agrimald@AGRIMA:~$ |
```

6). uname -a - Show kernel and system info.

Syntax- uname -a

Record system details for documentation or debugging.

```
agrinia@AGRIMA:~$ uname -a
Linux AGRIMA 6.6.87.2-microsoft-standard-wsl2 #1 SMP PREEMPT_DYNAMIC Thu Jun 5 18:30:46 UTC 2025 x86_64 x86_64 GNU/Linux
agriniag@AGRIMA:~$
```

7). history- Show command history (useful evidence).

Syntax- history, **use-** Provide an audit trail of commands you executed

Script Shells

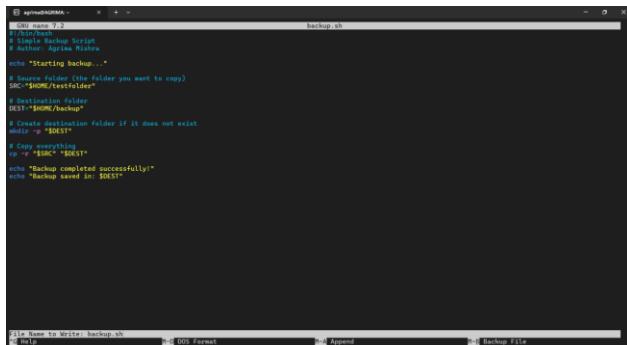
1). Backup a directory - backup.sh

Purpose: copy a specified folder into a backups folder named with timestamp.

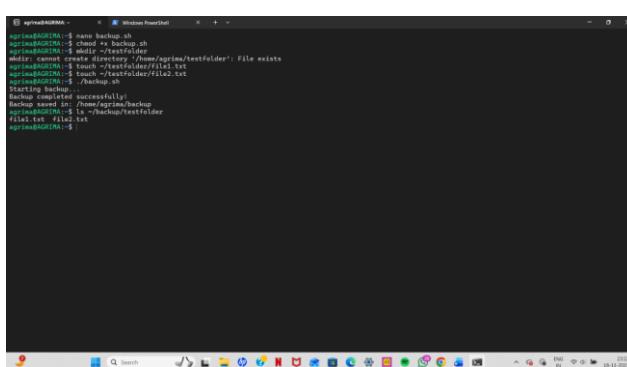
This script copies a folder called **testfolder** into a folder called **backup** inside your home directory.



```
gnome@GNOME:~$ nano backup.sh
```



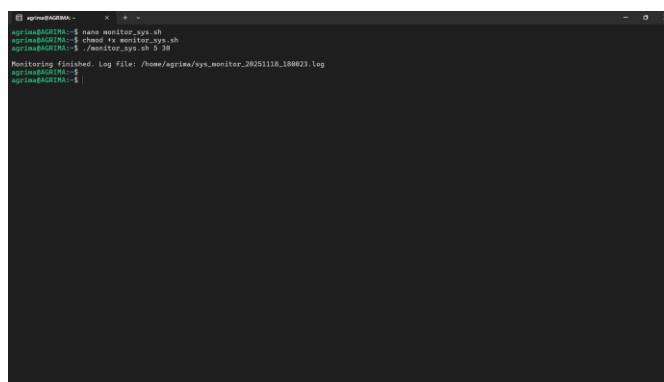
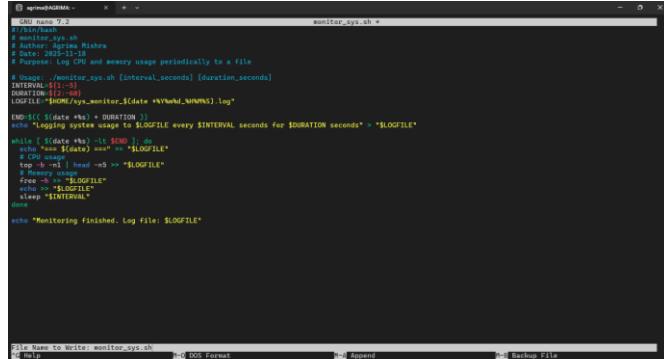
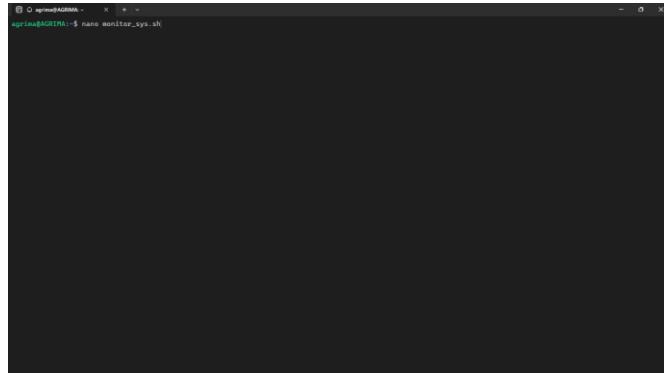
```
gnome@GNOME:~$ ./backup.sh
Starting backup...
Enter source folder you want to copy) SRC=/HOME/testfolder
Enter destination folder) DST=/HOME/backup
Create destination folder if it does not exist)
y/n? y
Copy everything)
y/n? y
Backup completed successfully!
Backup saved in: DST
```



```
gnome@GNOME:~$ nano backup.sh
gnome@GNOME:~$ chmod +x backup.sh
gnome@GNOME:~$ ./backup.sh
Starting backup...
mkdir: cannot create directory '/home/gnome/testfolder': File exists
gnome@GNOME:~$ touch ./testfolder/file.txt
gnome@GNOME:~$ ./backup.sh
Starting backup...
Backup completed successfully!
Backup saved in: DST
gnome@GNOME:~$ ls -l /home/gnome/backup
total 0
gnome@GNOME:~$
```

2). CPU/Memory monitor — monitor_sys.sh

Purpose: log CPU and memory usage every N seconds to a log file.



```
agrinia@AGRIHA: ~ % nano monitor_sys.sh
#!/bin/bash
# GNU name "0.2"
#
# monitor_sys.sh
# Author: Agrinia Mishra
# Date: 2025-11-18
# Purpose: Log CPU and memory usage periodically to a file
# Usage: monitor_sys.sh [interval_seconds] [duration_seconds]
INTERVAL=$1
DURATION=$2
LOGFILE="/home/agrinia/monitor.log"
if [ -z $INTERVAL ] || [ -z $DURATION ]; then
    echo "Usage: $0 [interval] [duration]"
    exit 1
fi
echo "Logging system usage to $LOGFILE every $INTERVAL seconds for $DURATION seconds" > "$LOGFILE"
while [ 1 ]; do
    date "+%s" >> "$LOGFILE"
    echo " $(top -b -n1 | head -n3)" >> "$LOGFILE"
    free -m >> "$LOGFILE"
    echo " $(cat $LOGFILE)" >> "$LOGFILE"
    sleep "$INTERVAL"
done
echo "Monitoring finished. Log file: $LOGFILE"
```

```
agrinia@AGRIHA: ~ % nano monitor_sys.sh
agrinia@AGRIHA: ~ % nano monitor_sys.sh
agrinia@AGRIHA: ~ % chmod +x monitor_sys.sh
agrinia@AGRIHA: ~ % ./monitor_sys.sh 5 30
Monitoring finished. Log file: /home/agrinia/monitor.log
agrinia@AGRIHA: ~ %
```

3). Automated download — download_file.sh

Purpose: download a file with wget or curl into a predefined directory.

```
agrima@AGRIMA ~ % ~
agrima@AGRIMA: ~ $ cd ~/scripts
agrima@AGRIMA:~/scripts$ nano download_file.sh
agrima@AGRIMA:~/scripts$
```

```
GNU nano 7.2
download_file.sh
# Author: Agnieszka Michta
# Date: 2023-11-10
# Purpose: Download a file from the internet to a downloads folder
# Usage: ./download_file.sh URL

URL="$1"
DEST_DIR="/home/agrima/downloads"
FILENAME=$(basename "$URL")

if [ -z "$URL" ]; then
    echo "Usage: $0 <url>"
    exit 1
fi

echo "Trying $URL"
# Try wget first, then curl
if command -v wget >/dev/null 2>&1; then
    wget -O "$DEST_DIR/$FILENAME" "$URL"
elif command -v curl >/dev/null 2>&1; then
    curl -O -o "$DEST_DIR/$FILENAME" "$URL"
else
    echo "wget or curl not found. Install one and retry."
    exit 1
fi

if [ $? -eq 0 ]; then
    echo "Downloaded to ${DEST_DIR}/${FILENAME}"
    exit 0
else
    echo "Download Failed."
    exit 1
fi
```

File Name to Write: download_file.sh

```
agrima@AGRIMA: ~ % ~
agrima@AGRIMA: ~ $ cd ~/scripts
agrima@AGRIMA:~/scripts$ nano download_file.sh
agrima@AGRIMA:~/scripts$ agrima@AGRIMA:~/scripts$ chmod +x download_file.sh
agrima@AGRIMA:~/scripts$ ./download_file.sh https://www.w3.org/WAI/ER/tests/xhtml/testfiles/resources/pdf/dummy.pdf
--2023-11-10 18:10:18-- https://www.w3.org/WAI/ER/tests/xhtml/testfiles/resources/pdf/dummy.pdf
Resolving www.w3.org (www.w3.org) [192.31.22.19]:2060... connected.
HTTP request sent, awaiting response... 200 OK
Length: 195440 (192K)
Saving to: '/home/agrima/downloads/dummy.pdf'

/home/agrima/downloads/dummy.pdf 100%[=====] 12.95K --=80/s in 0s
2023-11-10 18:10:18 (95.4 MB/s) - '/home/agrima/downloads/dummy.pdf' saved [13264/13264]

Downloaded to /home/agrima/downloads/dummy.pdf
agrima@AGRIMA:~/scripts$ ls -~/downloads
dummy.pdf
agrima@AGRIMA:~/scripts$
```

Challenges faced

Setting up WSL and choosing Ubuntu 24.04 (initially tried 22.04), capturing screenshots while installing, and learning permission management (chmod/chown).

Learning outcomes: Gained experience installing Ubuntu via WSL, using 20 core Linux commands, writing shell scripts for backup/monitoring/download, and publishing to GitHub.

Real-world applications: System administration, DevOps automation, creating routine backups, monitoring servers, and automating deployments or data retrieval.

Reflection

I now know how Linux can function properly on a Windows system and how WSL operates.

I learned how effective Linux is at managing files, keeping an eye on system resources, verifying networks, and managing processes throughout the assignment. I also practiced a number of fundamental commands.

One of the most beneficial aspects was writing shell scripts, which demonstrated to me how tasks can be automated with a few lines of code. Future courses like programming, cloud computing, ethical hacking, and cybersecurity will benefit from these abilities. All things considered, this assignment has given me a solid basis for learning Linux and delving into more complex subjects. I feel more equipped now to use Linux in practical academic projects and technical professions.

I felt more comfortable using Linux after creating folders, copying files, and examining system details. The part I most enjoyed was writing shell scripts because it demonstrated to me how little scripts can carry out practical tasks automatically. It helped me understand why Linux is preferred by professionals because it offers complete control and completes tasks quickly.

I also encountered some problems, such as missing packages and permission errors, but resolving them taught me even more. My familiarity with Linux has improved as a result of this assignment, and my interest in ethical hacking and cybersecurity has grown. I am eager to learn more commands, tools, and scripting techniques because I believe that learning Linux will be very beneficial for my future career.