

Name: Adwait S Purao

UID: 2021300101

Batch: B2

Experiment no. : 1

**Aim : Installation of Pop OS and running File ,
Directory ,Networking , Utility Commands**

Theory :

Linux is a free and open-source operating system, based on Unix, that is widely used for server, desktop, and embedded systems. Linux is known for its stability, security, and customization capabilities, and it is used by individuals, corporations, and governments worldwide. The source code for Linux is freely available, allowing users and developers to modify and distribute the operating system. The popularity of Linux has led to the development of many distributions, or "distros," each with its own unique features and target users. Some of the most popular distros include Ubuntu, Fedora, and CentOS.

Development: Linux was created in 1991 by Linus Torvalds, a student at the University of Helsinki, Finland. It has since grown into a massive project with contributions from thousands of developers all over the world.

Community: Linux is developed and maintained by a large and dedicated community of developers, users, and organizations. There is a strong sense of collaboration and cooperation within the community, with everyone working together to improve the operating system.

Open-Source: As mentioned earlier, Linux is open-source software, which means that the source code is available to everyone, allowing anyone to use, modify, and distribute the operating system. This has led to the creation of many different distributions and applications, making Linux a highly versatile platform.

Security: Linux is known for its security, thanks in part to its open-source development model. With the source code available for everyone to inspect, security vulnerabilities can be quickly identified and fixed. Additionally, the architecture of Linux is designed with security in mind, making it a secure platform for both personal and enterprise use.

Customization: Linux is highly customizable, with users able to modify everything from the look and feel of the operating system to the software that runs on it. There are countless tools and resources available for customizing Linux, making it a great choice for users who want a unique and personalized computing experience.

Wide Range of Applications: Linux can be used for a wide range of applications, from servers and supercomputers to smartphones and consumer electronics. It is a popular choice for web servers,

database servers, and scientific computing, and is used by many businesses and organizations for mission-critical applications.

Pop OS:

Pop!_OS is a free and open-source Linux distribution, based upon Ubuntu, and featuring a customized GNOME desktop environment known as COSMIC. The distribution is developed by American Linux computer manufacturer System76. Pop!_OS is primarily built to be bundled with the computers built by System76, but can also be downloaded and installed on most computers.[3]

Pop!_OS provides full out-of-the-box support for both AMD and Nvidia GPUs. It is regarded as an easy distribution to set up for gaming, mainly due to its built-in GPU support. Pop!_OS provides default disk encryption, streamlined window and workspace management, keyboard shortcuts for navigation as well as built-in power management profiles. The latest releases also have packages that allow for easy setup for TensorFlow and CUDA.[4][5]

Pop!_OS is maintained primarily by System76, with the release version source code hosted in a GitHub repository. Unlike many other Linux distributions, it is not community-driven, although outside programmers can contribute, view and modify the source code. They can also build custom ISO images and redistribute them under another name

Reasons to use POP OS:

1. Separate NVIDIA/AMD ISO

In case you have a system with Nvidia GPU, it is not feasible to make it work using Nouveau (an open-source driver for Nvidia cards).

Well, you get a working display, but you cannot utilize the benefits of your graphics card.

For instance, you will encounter stuttering and performance issues with almost every task that needs to utilize the GPU.

Hence, the proprietary Nvidia drivers are the only solution to make things right.

And, to conveniently add the proprietary driver out of the box, the distro needs to include it in the ISO itself or offer a separate ISO with Nvidia's latest available proprietary driver.

While you can always manually install the proprietary drivers, it is not always a flawless experience.

The procedure should be similar to that of installing Nvidia drivers in Linux Mint for most of the popular Linux distros if you are curious.

Driver Manager | Linux Mint (Cinnamon)

Note that, at times, you will need to troubleshoot or re-install the correct (or an older driver version of the driver) to use Nvidia graphics properly.

But, if you get a separate ISO for Nvidia-powered systems, or an Nvidia graphics installation mode (like Zorin OS), it eliminates the hassle of installing Nvidia drivers manually.

This is not just about the ease of use, but Pop!_OS always works out of the box with the help Nvidia ISOs offered.

I have had issues with Linux Mint, Ubuntu, and a few other distributions when I tried to manually install the proprietary drivers. So, Pop!_OS is a breeze to set up for my system with Nvidia graphics.

2. Automatic Tiling Manager

Without a tiling manager, you need to constantly drag and drop (and move) your active windows to organize them for quick access.

So, a window tiling manager makes it convenient to automatically organize the active windows of any application that you launch.

Of course, the experience with a tiling manager will differ as per your screen size, but even if you have a modest 27-inch size display, it should be a noticeable experience.

I find myself using it all the time, even without a big monitor (or a dual-monitor setup).

I have never used it on a laptop, but at the end of the day, having an automatic window tiling manager (that you can enable/disable) sounds like a useful feature.

It helps with enhancing your multi-tasking capabilities without necessarily moving to different workspaces.

3. Ease of Use

Why do you think Ubuntu is one of the best Linux distributions?

It manages to offer an easy-to-use modern desktop experience for years now.

Impressively, System76 has managed to polish the user experience in Pop!_OS up a notch on top of Ubuntu.

Hence, Pop!_OS is not just another Ubuntu distro, but the presence of an application launcher, application library, dock, workspace customizations, and various other optimizations lead to a unique desktop experience.

For my use-case, it feels better than Ubuntu. Not to forget, it also includes some of its tools like Popsicle that comes in handy later down the road.

4. Newer Linux Kernel

Pop!_OS 21.10 featuring Linux Kernel 5.15.8

Ubuntu may not feature the latest and greatest Linux Kernel, especially when it comes to LTS releases.

However, Pop!_OS tries its best to offer the latest Linux Kernel releases even for the LTS releases. For instance, Ubuntu 20.04.3 LTS packs in Linux Kernel 5.11, but Pop!_OS 20.04 LTS features Linux Kernel 5.13.

For most users, this may not be a big deal, but Pop!_OS pushes newer Linux Kernel faster than most other Linux distributions.

If you want your latest and greatest hardware to work well with Linux distros, Pop!_OS is a fantastic bet.

5. Application Launcher

Application launcher comes in incredibly handy to quickly launch an app or navigate between active windows.

The ability to invoke the launcher using a shortcut (or maybe a macro linked to it, as I use it) makes it one of the quickest ways to launch applications and navigate between them.

Of course, you can install tools like Ulauncher to achieve the same. However, having it pre-configured and working out of the box without any troubleshooting is convenient.

6. Fast Upgrade Process

Even though upgrading Linux distributions is usually hassle-free, Pop!_OS tries its best to refine and improve the upgrade process with every update.

For instance, with Pop!_OS 21.10, they introduced improvements that would prevent conflicts in the upgrade process and make it a seamless process.

In fact, I upgraded from Pop!_OS 21.04 to 21.10 in a few clicks without any issues.

7. Thrives to Improve the Modern Desktop Experience

Pop!_OS may not be a lightweight Linux distro, but it focuses on providing a top-notch experience for modern desktop users.

While there are plenty of good Linux distributions for older computers, it is equally important to aim for newer-gen hardware.

Pop!_OS does that incredibly well considering System76 also tests the distribution on latest hardware configurations with their laptops.

In addition, Pop!_OS also offers out-of-the-box tweaks that makes things easy for a user in terms of OS experience. While Linux is all about tweaking and taking control of your system, Pop!_OS seems to be succeeding in making it a mainstream option and a viable alternative to Ubuntu that could prove to be better for some.

Uninstalling Virtual Box:

Command: sudo apt-get remove --purge virtual box

```
bash: /home/spit/catkin_ws/devel/setup.bash: No such file or directory
bash: /home/spit/ws_moveit/devel/setup.bash: No such file or directory
spit@spit-ThinkCentre-M70s:~$ sudo apt-get remove --purge virtualbox
[sudo] password for spit:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  dctrl-tools dkms libfwupdplugin1 libgsoap-2.8.91 liblzf1 libsdl1.2debian
  libvncserver1 virtualbox-dkms
Use 'sudo apt autoremove' to remove them.
The following packages will be REMOVED:
  virtualbox* virtualbox-qt*
0 upgraded, 0 newly installed, 2 to remove and 267 not upgraded.
After this operation, 173 MB disk space will be freed.
Do you want to continue? [Y/n] Y
(Reading database ... 309347 files and directories currently installed.)
Removing virtualbox-qt (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Removing virtualbox (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for shared-mime-info (1.15-1) ...
Processing triggers for desktop-file-utils (0.24-1ubuntu3) ...
(Reading database ... 308980 files and directories currently installed.)
Purging configuration files for virtualbox (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Processing triggers for systemd (245.4-4ubuntu3.18) ...
```

Installing Virtual Box:

```
spit@spit-ThinkCentre-M70s:~$ sudo apt-get install virtualbox
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libfwupdplugin1
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  virtualbox-qt
Suggested packages:
  vde2 virtualbox-guest-additions-iso
The following NEW packages will be installed:
  virtualbox virtualbox-qt
0 upgraded, 2 newly installed, 0 to remove and 267 not upgraded.
Need to get 0 B/43.4 MB of archives.
After this operation, 173 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Selecting previously unselected package virtualbox.
(Reading database ... 308978 files and directories currently installed.)
Preparing to unpack .../virtualbox_6.1.38-dfsg-3~ubuntu1.20.04.1_amd64.deb ...
Unpacking virtualbox (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Selecting previously unselected package virtualbox-qt.
Preparing to unpack .../virtualbox-qt_6.1.38-dfsg-3~ubuntu1.20.04.1_amd64.deb ...
Unpacking virtualbox-qt (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Setting up virtualbox (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Setting up virtualbox-qt (6.1.38-dfsg-3~ubuntu1.20.04.1) ...
Processing triggers for desktop-file-utils (0.24-1ubuntu3) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for systemd (245.4-4ubuntu3.18) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for shared-mime-info (1.15-1) ...
spit@spit-ThinkCentre-M70s:~$
```

Linux Commands:

Linux directory commands

1. pwd Command

The **pwd** command is used to display the location of the current working directory.

```
spit@spit-ThinkCentre-M70s:~$ pwd
/home/spit
```

2. mkdir Command

The **mkdir** command is used to create a new directory under any directory.

```
spit@spit-ThinkCentre-M70s:~$ mkdir new_dir
```

3. rmdir Command

The **rmdir** command is used to delete a directory.

```
spit@spit-ThinkCentre-M70s:~$ rmdir new_dir
```

4. ls Command

The `ls` command is used to display a list of content of a directory.

```
spit@spit-ThinkCentre-M70s:~$ ls
  a.out      div.c      Downloads    Pictures    snap       Videos
 Desktop   Documents   Music        Public     Templates  'VirtualBox VMs'
spit@spit-ThinkCentre-M70s:~$
```

5. cd Command

The `cd` command is used to change the current directory.

```
spit@spit-ThinkCentre-M70s:~$ cd Downloads/
spit@spit-ThinkCentre-M70s:~/Downloads$
```

6. cp Command

The `cp` command is used to copy a file or directory.

Syntax:

To copy in the same directory:

```
cp <existing file name> <new file name>
```

```
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$ sudo cp DSA3.pdf DSA3copy.pdf
[sudo] password for coderadwait:
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$
```

My pdfs				
	Name	Date modified	Type	Size
	CH-2	24-12-2022 11:33	Adobe Acrobat D...	2,047 KB
	CH-2.pdf.Zone.Identifier	24-12-2022 11:33	IDENTIFIER File	1 KB
	DAA1Writing	02-02-2023 19:55	Adobe Acrobat D...	620 KB
	DAA1Writing.pdf.Zone.Identifier	02-02-2023 19:55	IDENTIFIER File	1 KB
	DSA3	26-09-2022 20:18	Adobe Acrobat D...	1,658 KB
	DSA3copy	05-02-2023 10:07	Adobe Acrobat D...	1,658 KB

7. mv Command

The `mv` command is used to move a file or a directory from one location to another location.

Syntax:

```
mv <file name> <directory path>
```

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ sudo mv DSA3.pdf Ubuntu\New_Directory  
[sudo] password for coderadwait:  
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

8. rm Command

rm stands for **remove** here. rm command is used to remove objects such as files, directories, symbolic links and so on from the file system like UNIX. To be more precise, rm removes references to objects from the filesystem, where those objects might have had multiple references (for example, a file with two different names). **By default, it does not remove directories.**

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ ls  
DSA1.pdf  DSA2.pdf  DSA4.pdf  Mytxt.txt
```

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ sudo rm Mytxt.txt  
[sudo] password for coderadwait:  
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ ls  
DSA1.pdf  DSA2.pdf  DSA4.pdf
```

9. head Command

The **head** command is used to display the content of a file. It displays the first 10 lines of a file.

```
spit@spit-ThinkCentre-M70s:~/Downloads$ head file1.txt  
I'm studying in spit
```

10. id Command

The **id** command is used to display the user ID (UID) and group ID (GID).

```
spit@spit-ThinkCentre-M70s:~/Downloads$ id  
uid=1000(spit) gid=1000(spit) groups=1000(spit),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),120(lpadmin),132(lxd),133(sambashare)
```

Linux Networking commands

1. ip Command

Linux **ip** command is an updated version of the ipconfig command. It is used to assign an IP address, initialize an interface, disable an interface.

Syntax:

ip a or ip addr

```
spit@spit-ThinkCentre-M70s:~/Downloads$ 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 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eno1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether d8:bb:c1:c5:c5:35 brd ff:ff:ff:ff:ff:ff
    altname enp0s31f6
        inet 172.16.40.111/24 brd 172.16.40.255 scope global dynamic noprefixroute eno1
            valid_lft 1816sec preferred_lft 1816sec
        inet6 fe80::ffff1:f7d:4cb7:c17d/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
spit@spit-ThinkCentre-M70s:~/Downloads$
```

2. ping Command

The **ping** command is used to check the connectivity between two nodes, that is whether the server is connected. It is a short form of "Packet Internet Groper."

```
spit@spit-ThinkCentre-M70s:~/Downloads$ ping google.com
PING google.com (216.239.38.120) 56(84) bytes of data.
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=1 ttl=55 time=4.80 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=2 ttl=55 time=3.55 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=3 ttl=55 time=3.59 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=4 ttl=55 time=3.21 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=5 ttl=55 time=3.39 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=6 ttl=55 time=3.35 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=7 ttl=55 time=3.33 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=8 ttl=55 time=3.24 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=9 ttl=55 time=4.02 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=10 ttl=55 time=3.47 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=11 ttl=55 time=3.35 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=12 ttl=55 time=3.35 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=13 ttl=55 time=3.59 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=14 ttl=55 time=3.44 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=15 ttl=55 time=3.30 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=16 ttl=55 time=3.31 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=17 ttl=55 time=3.38 ms
^C
--- google.com ping statistics ---
17 packets transmitted, 17 received, 0% packet loss, time 16024ms
rtt min/avg/max/mdev = 3.212/3.510/4.801/0.370 ms
```

3. host Command

The **host** command is used to display the IP address for a given domain name and vice versa. It performs the DNS lookups for the DNS Query.

```
spit@spit-ThinkCentre-M70s:~/Downloads$ host javatpoint.com
javatpoint.com has address 172.67.211.76
javatpoint.com has address 104.21.23.133
javatpoint.com has IPv6 address 2606:4700:3036::6815:1785
javatpoint.com has IPv6 address 2606:4700:3031::ac43:d34c
javatpoint.com mail is handled by 5 alt2.aspmx.l.google.com.
javatpoint.com mail is handled by 1 aspmx.l.google.com.
javatpoint.com mail is handled by 10 alt4.aspmx.l.google.com.
javatpoint.com mail is handled by 5 alt1.aspmx.l.google.com.
javatpoint.com mail is handled by 10 alt3.aspmx.l.google.com.
spit@spit-ThinkCentre-M70s:~/Downloads$
```

4. ifconfig command

You can use the **ifconfig** command to assign an address to a network interface and to configure or display the current network interface configuration information. The **ifconfig** command must be used at system startup to define the network address of each interface present on a system. After system startup, it can also be used to redefine an interfaces address and its other operating parameters. The network interface configuration is held on the running system and must be reset at each system restart. The **ifconfig** command interprets the **IFF_MULTICAST** flag and prints its value if it is set.

```
students@students-HP-280-G3-MT:~$ ifconfig
enp4s0    Link encap:Ethernet HWaddr 3c:52:82:63:e7:90
          inet addr:172.16.31.188 Bcast:172.16.31.255 Mask:255.255.255.0
          inet6 addr: fe80::4c7:4c55:2f2e:5cd6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:455 errors:0 dropped:0 overruns:0 frame:0
          TX packets:120 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:129118 (129.1 KB) TX bytes:13386 (13.3 KB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:392 errors:0 dropped:0 overruns:0 frame:0
          TX packets:392 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:32523 (32.5 KB) TX bytes:32523 (32.5 KB)

students@students-HP-280-G3-MT:~$
```

5. Nslookup Command

Name server lookup (nslookup) is a command-line tool that lets you find the internet protocol (IP) address or domain name system (DNS) record of a specific hostname. This command also allows reverse DNS lookup by inputting the IP addresses of the corresponding domains.

The nslookup tool is useful for DNS-related tasks, such as server testing or troubleshooting issues. To use this tool, type “nslookup” into a command-line interface (CLI) such as the Command Prompt on Windows or Terminal on Linux and macOS.

```
students@students-HP-280-G3-MT:~$ nslookup spit.ac.in
Server:      127.0.1.1
Address:     127.0.1.1#53

Name:   spit.ac.in
Address: 172.16.10.2
Name:   spit.ac.in
Address: 172.16.10.3
Name:   spit.ac.in
Address: 172.16.10.6

students@students-HP-280-G3-MT:~$
```

6. Traceroute

Linux traceroute command is a network troubleshooting utility that helps us determine the number of hops and packets traveling path required to reach a destination. It is used to display how the data transmitted from a local machine to a remote machine. Loading a web page is one of the common examples of the traceroute. A web page loading transfers data through a network and routers. The traceroute can display the routes, IP addresses, and hostnames of routers over a network. It can be useful for diagnosing network issues.

```
students@students-HP-280-G3-MT:~$ traceroute google.com
The program 'traceroute' can be found in the following packages:
 * inetutils-traceroute (You will have to enable component called 'universe')
 * traceroute (You will have to enable component called 'universe')
Try: sudo apt install <selected package>
```

7. Sudo apt-get install

sudo apt-get install command is used to download the latest version of your desired application from an online software repository pointed to by your sources.list configuration file and install that application on your Linux machine.

A good analogy is to think of your computer as a factory and the apt-get command as the manager in that factory who is responsible for the installation of new equipment, removal of equipment that is no longer needed, and update desired equipment to their latest versions, while maintaining records of the equipment names and versions which are currently present in the factory.

```
students@students-HP-280-G3-MT:~$ sudo apt-get install traceroute
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  traceroute
0 upgraded, 1 newly installed, 0 to remove and 155 not upgraded.
Need to get 53.6 kB of archives.
After this operation, 130 kB of additional disk space will be used.
WARNING: The following packages cannot be authenticated!
  traceroute
Install these packages without verification? [y/N] y
Get:1 http://httpredir.debian.org/debian jessie/main traceroute amd64 1:2.0.20-2+b1 [53.6 kB]
Fetched 53.6 kB in 2s (22.1 kB/s)
Selecting previously unselected package traceroute.
(Reading database ... 244949 files and directories currently installed.)
Preparing to unpack .../traceroute_1%3a2.0.20-2+b1_amd64.deb ...
Unpacking traceroute (1:2.0.20-2+b1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up traceroute (1:2.0.20-2+b1) ...
update-alternatives: using /usr/bin/traceroute.db to provide /usr/bin/traceroute
(traceroute) in auto mode
update-alternatives: using /usr/bin/lft.db to provide /usr/bin/lft (lft) in auto
mode
update-alternatives: using /usr/bin/traceproto.db to provide /usr/bin/traceproto
(traceproto) in auto mode
update-alternatives: using /usr/sbin/tcptraceroute.db to provide /usr/sbin/tcptr
aceroute (tcptraceroute) in auto mode
students@students-HP-280-G3-MT:~$
```

8. ARP command

The arp stands for the “Address Resolution Protocol” and it makes changes in the kernel’s table which contains the arp addresses.

The arp command allows users to manipulate the neighbor cache or ARP table. It is contained in the Net-tools package along with many other notable networking commands (such as ifconfig). The arp command has since been replaced by the ip neighbour command.

```
students@students-HP-280-G3-MT:~$ arp -a
? (172.16.31.189) at f4:4d:30:4f:7e:e6 [ether] on enp4s0
? (172.16.31.1) at e0:07:1b:c2:64:60 [ether] on enp4s0
? (172.16.31.208) at 3c:52:82:67:82:4d [ether] on enp4s0
students@students-HP-280-G3-MT:~$
```

9. netstat

Linux netstat command refers to the network statistics.

It provides statistical figures about different interfaces which include open sockets, routing tables, and connection information.

Syntax:

netstat

Output:

Observe the output displaying all the open sockets.

Variations in netstat command

Below are few variations of the netstat command used.

1) To display the programs

```
coderadwait@LAPTOP-LG4IQEFB:~$ netstat -p
(No info could be read for "-p": geteuid()=1000 but you should be root.)
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State      PID/Program name
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type            State         I-Node    PID/Program name      Path
coderadwait@LAPTOP-LG4IQEFB:~$
```

10. Dig Command

Linux dig command stands for Domain Information Groper. This command is used in DNS lookup to query the DNS name server. It is also used to troubleshoot DNS related issues.

It is mainly used to verify DNS mappings, MX Records, host addresses, and all other DNS records for a better understanding of the DNS topography.

This command is an improvised version of nslookup command.

Syntax:

```
dig <domainName>
```

```
coderadwait@LAPTOP-LG4IQEFB:~$ dig google.com

; <>> DiG 9.18.1-1ubuntu1.2-Ubuntu <>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 950
;; flags: qr rd ad; QUERY: 1, ANSWER: 9, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;google.com.           IN      A

;; ANSWER SECTION:
google.com.          0       IN      A      172.217.166.174
ns3.google.com.       0       IN      A      216.239.36.10
ns1.google.com.       0       IN      A      216.239.32.10
ns4.google.com.       0       IN      A      216.239.38.10
ns2.google.com.       0       IN      A      216.239.34.10
ns3.google.com.       0       IN      AAAA   2001:4860:4802:36::a
ns1.google.com.       0       IN      AAAA   2001:4860:4802:32::a
ns4.google.com.       0       IN      AAAA   2001:4860:4802:38::a
ns2.google.com.       0       IN      AAAA   2001:4860:4802:34::a

;; Query time: 20 msec
;; SERVER: 172.18.240.1#53(172.18.240.1) (UDP)
;; WHEN: Sat Feb 04 14:41:32 IST 2023
;; MSG SIZE  rcvd: 342
```

Linux Utility Commands

1. find Command

The [find](#) command is used to find a particular file within a directory. It also supports various options to find a file such as byname, by type, by date, and more.

The following symbols are used after the find command:

(.) : For current directory name

(/) : For root

Syntax:

1. find . -name "*.pdf"

```
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$ find . -name "*.pdf"
./DSA3.pdf
./CH-2.pdf
./DAA1Writing.pdf
```

2. locate Command

The [locate](#) command is used to search a file by file name. It is quite similar to find command; the difference is that it is a background process. It searches the file in the database, whereas the find command searches in the file system. It is faster than the find command. To find the file with the locates command, keep your database updated.

Syntax:

locate <file name>

```
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$ locate DSA3
/My pdfs/DSA3.pdf
/mnt/c/Users/aspur/OneDrive/DSA/ALL_PROGRAMS/C/DSA3_S11_PALINDROME.c
/mnt/c/Users/aspur/OneDrive/DSA/DSA_PDFS/DSA3.pdf
/mnt/c/Users/aspur/OneDrive/DSA/DSA_WORD_DOCS/DSA3.docx
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$
```

3. date Command

The [date](#) command is used to display date, time, time zone, and more.

Syntax:

1. date

```
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$ date  
Sat Feb 4 15:20:45 IST 2023
```

4. cal Command

The [cal](#) command is used to display the current month's calendar with the current date highlighted.

Syntax:

1. cal <

```
coderadwait@LAPTOP-LG4IQEFB:~$ cal  
February 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
```

5. sleep Command

The [sleep](#) command is used to hold the terminal by the specified amount of time. By default, it takes time in seconds.

Syntax:

```
sleep <time>
```

```
coderadwait@LAPTOP-LG4IQEFB:/My pdfs$ sleep 4
```

6. time Command

The [time](#) command is used to display the time to execute a command.

Syntax:

1. time

```
coderadwait@LAPTOP-LG4IQEFB:~$ time  
real    0m0.000s  
user    0m0.000s  
sys     0m0.000s  
coderadwait@LAPTOP-LG4IQEFB:~$
```

7. df Command

The [df](#) command is used to display the disk space used in the file system. It displays the output as in the number of used blocks, available blocks, and the mounted directory.

Syntax:

```
df
```

```
coderadwait@LAPTOP-LG4IQEFB:~$ df  
Filesystem      1K-blocks        Used   Available  Use% Mounted on  
/dev/sdb       263174212    1326356  248409700   1% /  
none           1956092         4    1956088   1% /mnt/wsl  
tools          412187644  146466696  265720948  36% /init  
none           1956092         4    1956088   1% /run  
none           1956092         0    1956092   0% /run/lock  
none           1956092         0    1956092   0% /run/shm  
none           1956092         0    1956092   0% /run/user  
tmpfs          1956092         0    1956092   0% /sys/fs/cgroup  
drivers        412187644  146466696  265720948  36% /usr/lib/wsl/drivers  
lib            412187644  146466696  265720948  36% /usr/lib/wsl/lib  
drvfs          24779772      84844   24694928   1% /mnt/a  
drvfs          32541692      690060  31851632   3% /mnt/b  
drvfs          412187644  146466696  265720948  36% /mnt/c  
drvfs          29287420      127224  29160196   1% /mnt/e  
coderadwait@LAPTOP-LG4IQEFB:~$
```

8. mount Command

The [mount](#) command is used to connect an external device file system to the system's file system.

Syntax:

```
mount -t type <device> <directory>
```

9. clear Command

Linux **clear** command is used to clear the terminal screen.

Syn-tax:

```
coderadwait@LAPTOP-LG4IQEFB:~$ mount
/dev/sdb on / type ext4 (rw,relatime,discard,errors=remount-ro,data=ordered)
none on /mnt/wsl type tmpfs (rw,relatime)
tools on /init type 9p (ro,relatime,dirsync,aname=tools;fmask=022,loose,access=client,msize=65536,trans=fd,rfd=7,wfd=7)
none on /dev type devtmpfs (rw,nosuid,relatime,size=1954012k,nr_inodes=488503,mode=755)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,noatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,noatime)
devpts on /dev/pts type devpts (rw,nosuid,noexec,noatime,gid=5,mode=620,ptmxmode=000)
none on /run type tmpfs (rw,nosuid,noexec,noatime,mode=755)
none on /run/lock type tmpfs (rw,nosuid,nodev,noexec,noatime)
none on /run/shm type tmpfs (rw,nosuid,nodev,noatime)
none on /run/user type tmpfs (rw,nosuid,nodev,noexec,noatime,mode=755)
binfmt_misc on /proc/sys/fs/binfmt_misc type binfmt_misc (rw,relatime)
tmpfs on /sys/fs/cgroup type tmpfs (rw,nosuid,nodev,noexec,relatime,mode=755)
cgroup2 on /sys/fs/cgroup/unified type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,cpuset)
cgroup on /sys/fs/cgroup/cpu type cgroup (rw,nosuid,nodev,noexec,relatime,cpu)
cgroup on /sys/fs/cgroup/cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,cpuacct)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,blkio)
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,memory)
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,devices)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,freezer)
cgroup on /sys/fs/cgroup/net_cls type cgroup (rw,nosuid,nodev,noexec,relatime,net_cls)
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,perf_event)
cgroup on /sys/fs/cgroup/net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,net_prio)
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,hugetlb)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,pids)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,rdma)
drivers on /usr/lib/wsl/drivers type 9p (ro,nosuid,nodev,noatime,dirsync,aname=drivers;fmask=222;dmask=222,mmap,access=client,msize=65536,trans=fd,rfd=4,wfd=4)
lib on /usr/lib/wsl/lib type 9p (ro,nosuid,nodev,noatime,dirsync,aname=lib;fmask=222;dmask=222,mmap,access=client,msize=65536,trans=fd,rfd=4,wfd=4)
drvfs on /mnt/a type 9p (rw,noatime,dirsync,aname=drvfs;path=A:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
drvfs on /mnt/b type 9p (rw,noatime,dirsync,aname=drvfs;path=B:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
drvfs on /mnt/c type 9p (rw,noatime,dirsync,aname=drvfs;path=C:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
drvfs on /mnt/e type 9p (rw,noatime,dirsync,aname=drvfs;path=E:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
coderadwait@LAPTOP-LG4IQEFB:~$
```

clear

Before:

```
drivers on /usr/lib/wsl/drivers type 9p (ro,nosuid,nodev,noatime,dirsync,aname=drivers;fmask=222;dmask=222,mmap,access=client,msize=65536,trans=fd,rfd=4,wfd=4)
lib on /usr/lib/wsl/lib type 9p (ro,nosuid,nodev,noatime,dirsync,aname=lib;fmask=222;dmask=222,mmap,access=client,msize=65536,trans=fd,rfd=4,wfd=4)
drvfs on /mnt/a type 9p (rw,noatime,dirsync,aname=drvfs;path=A:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
drvfs on /mnt/b type 9p (rw,noatime,dirsync,aname=drvfs;path=B:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
drvfs on /mnt/c type 9p (rw,noatime,dirsync,aname=drvfs;path=C:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
drvfs on /mnt/e type 9p (rw,noatime,dirsync,aname=drvfs;path=E:\;uid=0;gid=0;symlinkroot=/mnt/,mmap,access=client,msize=262144,trans=virtio)
coderadwait@LAPTOP-LG4IQEFB:~$ clear
```

After:



10. exit Command

Linux **exit** command is used to exit from the current shell. It takes a parameter as a number and exits the shell with a return of status number.

Syntax:

```
exit
```

```
coderadwait@LAPTOP-LG4IQEFB: ~
coderadwait@LAPTOP-LG4IQEFB:~$ exit
```

Linux File Commands

1. touch Command

The **touch** command is used to create empty files. We can create multiple empty files by executing it once.

```
spit@spit-ThinkCentre-M70s:~/Downloads$ touch file1.txt
spit@spit-ThinkCentre-M70s:~/Downloads$ touch f3.txt f2.txt
spit@spit-ThinkCentre-M70s:~/Downloads$ ls
f2.txt  Fedora-Workstation-Live-x86_64-37-1.7.iso          file1.txt
f3.txt  Fedora-Workstation-Live-x86_64-37-1.LfMzbw5r.7.iso.part
```

2. cat Command

The **cat** command is a multi-purpose utility in the Linux system. It can be used to create a file, display content of the file, copy the content of one file to another file, and more.

Syntax:

```
cat [OPTION]... [FILE]...
```

Create a file, execute it as follows:

```
cat > <file name>
```

```
// Enter file content
```

Press "CTRL+ D" keys to save the file. To display the content of the file, execute it as follows:

```
cat <file name>
```

```
spit@spit-ThinkCentre-M70s:~/Downloads$ cat > file1.txt
I'm studying in spit
spit@spit-ThinkCentre-M70s:~/Downloads$ cat file1.txt
I'm studying in spit
```

3. tail command

It is the complementary of head command. The tail command, as the name implies, print the last N number of data of the given input. By default it prints the last 10 lines of the specified files. If more than one file name is provided then data from each file is preceded by its file name.

-n num: Prints the last ‘num’ lines instead of last 10 lines. **num** is mandatory to be specified in command otherwise it displays an error. This command can also be written as without symbolizing ‘n’ character but ‘-’ sign is mandatory.

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ tail -n 1 NewText.txt
Thank youcoderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

```
Thank youcoderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ cat NewText.txt
Hi
I'm Adwait
How are you
I'm working on wsl
Hope your are doing well
Thank youcoderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ tail -n 2 NewText.txt
Hope your are doing well
Thank youcoderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

4. more Command

more command is used to view the text files in the command prompt, displaying one screen at a time in case the file is large (For example log files). The more command also allows the user do scroll up and down through the page

```

coderadwait@LAPTOP-LG4IQEFB:/My pdfs$ more -f DSA3.pdf
%PDF-1.5
%>>>
1 0 obj
<</Type/Catalog/Pages 2 0 R/Lang(en-IN) /StructTreeRoot 62 0 R/MarkInfo<</Marked true>>>
endobj
2 0 obj
<</Type/Pages/Count 16/Kids[ 3 0 R 8 0 R 15 0 R 19 0 R 23 0 R 27 0 R 29 0 R 31 0 R 33 0 R 36 0 R 39 0 R 44 0 R 48 0 R 50
0 R 52 0 R 55 0 R] >>
endobj
3 0 obj
<</Type/Page/Parent 2 0 R/Resources<</Font<</F1 5 0 R>>/XObject<</Image7 7 0 R>>/ProcSet[/PDF/Text/ImageB/ImageC/ImageI
>>/MediaBox[ 0 0 595.32 841.92] /Contents 4 0 R/Group<</Type/Group/S/Transparency/CS/DeviceRGB>>/Tabs/S/StructParents 0
>>
endobj
4 0 obj
<</Filter/FlateDecode/Length 2302>>
stream
xmoxFn nna?Elk@@[@@s@EH@@!@4E@D,@@)@@_@@R@@@,Q@@y@gfV@@?@@p@@X@@@RH@@-@@Z I@@RI@@@S@@Z -@@
=@@Q@@DH@@1[@@i@@~]E@@@ @@xs@@[@@@(`@@c@@L@@@H@@)-@@r@@桂@@qb@@@t@@s@@9@@@@,@@T?H7v@@X@@tC|L@@N@@^@@i@@-
R@@v@@^@@s@@@A@@^@@v@@'@@
@@i@@Gi@@Ic@@@'s@@@=@@6@@ e%[vx8@@@@ @@[ @@@!g\@@s @@GI@@%b@@7 r@` @@p@@D@@~d@@@Q@@@'@@w@@80^x@@A@@
@@
@@F@@k@@ p!Ao@@/0@@@ : x@@@@#@@+@@ZPj@@EL@@J1@@!@@-@@9 @@Q@@
@@
@@ !@@N7&@@ @@EN@@A@@ - |@@S@@@ '@@@9[F-]
@@ 3@@c@@Y@@@b~#@@@'1@@@T @@b@@@.@@K@@B@@@Q@@@Q@@@k@@L@@w@@J@@Km@@}@@}@@o@@?5@@B@@LX@@@7@@@1@@y@@@}
@@
@@--More--(0%)

```

5. less Command

The less command is a Linux terminal pager that shows a file's contents one screen at a time. It is useful when dealing with a large text file because it doesn't load the entire file but accesses it page by page, resulting in fast loading speeds.

```

coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ less NewText.txt

```

```

coderadwait@LAPTOP-LG4IQEFB:/MyPdfs
Hi
I'm Adwait
How are you
I'm working on wsl
Hope your are doing well
Thank you
NewText.txt (END)

```

6. wc Command

It is used to find out number of lines, word count, byte and characters count in the files specified in the file arguments.

By default it displays four-columnar output.

First column shows number of lines present in a file specified, second column shows number of words present in the file, third column shows number of characters present in file and fourth column itself is the file name which are given as argument.

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ wc Mytxt.txt
2 5 21 Mytxt.txt
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

7. uniq Command

The uniq command in Linux is a command-line utility that reports or filters out the repeated lines in a file.

In simple words, uniq is the tool that helps to detect the adjacent duplicate lines and also deletes the duplicate lines. uniq filters out the adjacent matching lines from the input file(that is required as an argument) and writes the filtered data to the output file.

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ cat Mytxt.txt
HI
How are u
How are u
How are u
How are u
Hello
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ uniq Mytxt.txt
HI
How are u
Hello
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

8. Tac Command

tac command in Linux is used to concatenate and print files in reverse. This command will write each FILE to standard output, the last line first. When no file is specified then this command will read the standard input.

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ tac NewText.txt
Thank youHope your are doing well
I'm working on wsl
How are you
I'm Adwait
Hi
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

9. Sort Command

SORT command is used to sort a file, arranging the records in a particular order. By default, the sort command sorts file assuming the contents are ASCII. Using options in the sort command can also be used to sort numerically.

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ sort NewText.txt
Hi
Hope your are doing well
How are you
I'm Adwait
I'm working on wsl
Thank you
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

10. Grep Command

The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern. The pattern that is searched in the file is referred to as the regular expression (grep stands for global search for regular expression and print out).

Syntax:

grep [options] pattern [files]

- Case insensitive search :** The **-i** option enables to search for a string case insensitively in the given file. It matches the words like “UNIX”, “Unix”, “unix”.

```
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ grep -i "UNIx" OStext.txt
unix is great os. unix was developed in Bell labs.
Unix linux which one you choose.
uNix is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$ cat OStext.txt
unix is great os. unix was developed in Bell labs.
learn operating system.
Unix linux which one you choose.
uNix is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.coderadwait@LAPTOP-LG4IQEFB:/MyPdfs$
```

11. File Command

file command is used to determine the type of a file. *file* type may be of human-readable(e.g. ‘ASCII text’) or MIME type(e.g. ‘text/plain; charset=us-ascii’). This command tests each argument in an attempt to categorize it.

It has three sets of tests as follows:

- filesystem test:** This test is based on the result which returns from a *stat* system call. The program verifies that if the file is empty, or if it’s some sort of special file. This test causes the file type to be printed.
- magic test:** These tests are used to check for files with data in particular fixed formats.
- language test:** This test search for particular strings which can appear anywhere in the first few blocks of a file.

Options:

- b, –brief :** This is used to display just file type in brief mode.

Syntax:

file -b filename

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ file -b Hello.py
ASCII text
```

* option : Command displays the all files's file type.

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ file *
Hello.py: ASCII text
Myfile.txt: ASCII text
OS1.docx: Microsoft Word 2007+
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

12. Append text to the end of file

In Linux, the echo command can be used for displaying a line of string/text that is passed as the **arguments**. This command is a built-in that is mostly and widely used in various batch files and shell scripts to outcome status test to a file and screen.

Syntax of the echo command

1. echo [option] [string]

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ echo "Hey there I'm Adwait"
Hey there I'm Adwait
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ echo "Hey there I'm Adwait" >>
 myfile.txt
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cat myfile.txt
Hello, there I'm Adwait
Hey there I'm Adwait
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ echo "I study in spit" >> myfile.txt
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cat myfile.txt
Hello, there I'm Adwait
Hey there I'm Adwait
I study in spit
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

13. Appending one file to another

The >> operator redirects output to a file. If the mentioned file doesn't exist the file is created and then the text is appended to the file. When we use it with cat command it appends the first file to the second and the changes are made in the second file and the first file remains unchanged.

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cat Myfile.txt >> MyDesc.txt
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cat Myfile.txt
Hello, there I'm Adwait
Hey there I'm Adwait
I study in spit
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cat MyDesc.txt
I like playing table tennis and Drawing
Hello, there I'm Adwait
Hey there I'm Adwait
I study in spit
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ exit
```

14. chmod Command

In Unix-like operating systems, the **chmod** command is used to change the access mode of a file.

The name is an abbreviation of **change mode**.

Syntax :

```
chmod [reference] [operator] [mode] file...
```

Options:

The chmod command supports the following command-line options:

-c, --changes: It is similar to the verbose option, but the difference is that it is reported if a change has been made.

-f, --silent, --quiet: It is used to suppress the error messages.

-v, --verbose: It is used to display a diagnostic for every processed file.

--no-preserve-root: It is used for not treating the backslash symbol ('/'), especially (the default).

--preserve-root: If this option is used, it will fail to operate recursively on backslash ('/').

--reference=RFILE: It is used to specify the RFILE's mode alternatively MODE values.

-R, --recursive: It is used to change files and directories recursively.

--help: It is used to display the help manual having a brief description of usage and support options.

--version: It is used to display the version information.

the user can read, write, and execute it;

1. the **user can read, write, and execute it;**
2. members of your **group can read and execute it;** and
3. **others may only read it.**

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ chmod u=rwx,g=rx,o=r Myfile.txt
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ chmod -rwx Myfile.txt
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cat Myfile.txt
cat: Myfile.txt: Permission denied
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

15. ps Command

Linux provides us a utility called **ps** for viewing information related with the processes on a system which stands as abbreviation for “**Process Status**”. ps command is used to list the currently running processes and their PIDs along with some other information depends on different options. It reads the process information from the virtual files in **/proc** file-system. /proc contains virtual files, this is the reason it’s referred as a virtual file system.

ps provides numerous options for manipulating the output according to our need.

Syntax –

```
ps [options]
```

1. Simple process selection : Shows the processes for the current shell –

PID – the unique process ID

TTY – terminal type that the user is logged into

TIME – amount of CPU in minutes and seconds that the process has been running

CMD – name of the command that launched the process.

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ ps
  PID TTY          TIME CMD
 6032 pts/0    00:00:00 bash
 7616 pts/0    00:00:00 ps
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

2. View Processes : View all the running processes use either of the following option with ps –

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ ps -A
  PID TTY          TIME CMD
    1 ?        00:00:01 systemd
    2 ?        00:00:00 kthreadd
    3 ?        00:00:00 rcu_gp
    4 ?        00:00:00 rcu_par_gp
    5 ?        00:00:00 slub_flushwq
    6 ?        00:00:00 netns
    8 ?        00:00:00 kworker/0:0H-events_highpri
   10 ?       00:00:00 mm_percpu_wq
   11 ?       00:00:00 rcu_tasks_rude_
```

```
7517 ? 00:00:00 Web Content
7561 ? 00:00:00 kworker/2:0-events
7569 ? 00:00:01 kworker/9:2-events
7571 ? 00:00:00 Web Content
7595 ? 00:00:00 kworker/u24:3
7634 ? 00:00:00 kworker/11:1-events
7635 ? 00:00:00 kworker/7:1-events
7649 pts/0 00:00:00 ps
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

View all the processes except session leaders :

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ ps -d
  PID TTY      TIME CMD
    2 ? 00:00:00 kthreadd
    3 ? 00:00:00 rcu_gp
    4 ? 00:00:00 rcu_par_gp
    5 ? 00:00:00 slab_flushwq
    6 ? 00:00:00 netns
    8 ? 00:00:00 kworker/0:0H-events_highpri
  7634 ? 00:00:00 kworker/11:1-events
  7635 ? 00:00:00 kworker/7:1-events
  7686 ? 00:00:00 kworker/10:0-events
  7687 ? 00:00:00 kworker/2:2-events
  7707 ? 00:00:00 kworker/6:2
  7712 ? 00:00:00 kworker/u24:0-events_unbound
  7713 ? 00:00:00 kworker/1:0-events
  7718 pts/0 00:00:00 ps
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

16. alias

The alias command lets you give your own name to a command or sequence of commands. You can then type your short name, and the shell will execute the command or sequence of commands for you.

```
alias cls=clear
```

This sets up an alias called `cls`. It will be another name for `clear`. When you type `cls`, it will clear the screen just as though you had typed `clear`.

```
7739 ? 00:00:00 kworker/7:0
7770 ? 00:00:00 kworker/9:0-events
7796 ? 00:00:00 Web Content
7834 ? 00:00:00 Web Content
7883 pts/0 00:00:00 ps
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ alias cls=clear
```

```
7739 ? 00:00:00 kworker/7:0
7770 ? 00:00:00 kworker/9:0-events
7796 ? 00:00:00 Web Content
7834 ? 00:00:00 Web Content
7883 pts/0 00:00:00 ps
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ cls
```

17. curl Command

The curl command is a tool to retrieve information and files from Uniform Resource Locators (URLs) or internet addresses.

The curl command may not be provided as a standard part of your Linux distribution. Use apt-get to install this package onto your system if you're using Ubuntu or another Debian-based distribution. On other Linux distributions, use your Linux distribution's package management tool instead.

```
sudo apt-get install curl
```

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ sudo apt-get install curl
[sudo] password for spit:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libfwupdplugin1 libxmlb1
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 31 not upgraded.
Need to get 161 kB of archives.
After this operation, 413 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.15 [161 kB]
Fetched 161 kB in 0s (527 kB/s)
Selecting previously unselected package curl.
(Reading database ... 229616 files and directories currently installed.)
Preparing to unpack .../curl_7.68.0-1ubuntu2.15_amd64.deb ...
Unpacking curl (7.68.0-1ubuntu2.15) ...
Setting up curl (7.68.0-1ubuntu2.15) ...
Processing triggers for man-db (2.9.1-1) ...
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

18. finger

The finger command gives you a short dump of information about a user, including the time of the user's last login, the user's home directory, and the user account's full name.

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ finger spit
Login: spit                                     Name: spit
Directory: /home/spit                         Shell: /bin/bash
On since Mon Feb  6 09:11 (IST) on :0 from :0 (messages off)
No mail.
No Plan.
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

19. free

The `free` command gives you a summary of the memory usage with your computer. It does this for both the main Random Access Memory (RAM) and swap memory. The `-h` (human) option is used to provide human-friendly numbers and units. Without this option, the figures are presented in bytes.

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ free -h
              total        used        free      shared  buff/cache   available
Mem:       7.5Gi       2.3Gi       2.1Gi      451Mi       3.0Gi       4.4Gi
Swap:      2.0Gi          0B       2.0Gi
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$
```

20. groups

The `groups` command tells you which groups a user is a member of.

```
spit@spit-ThinkCentre-M70s:~/Desktop/2021300101$ groups spit
spit : spit adm cdrom sudo dip plugdev lpadmin lxd sambashare
```

21. history

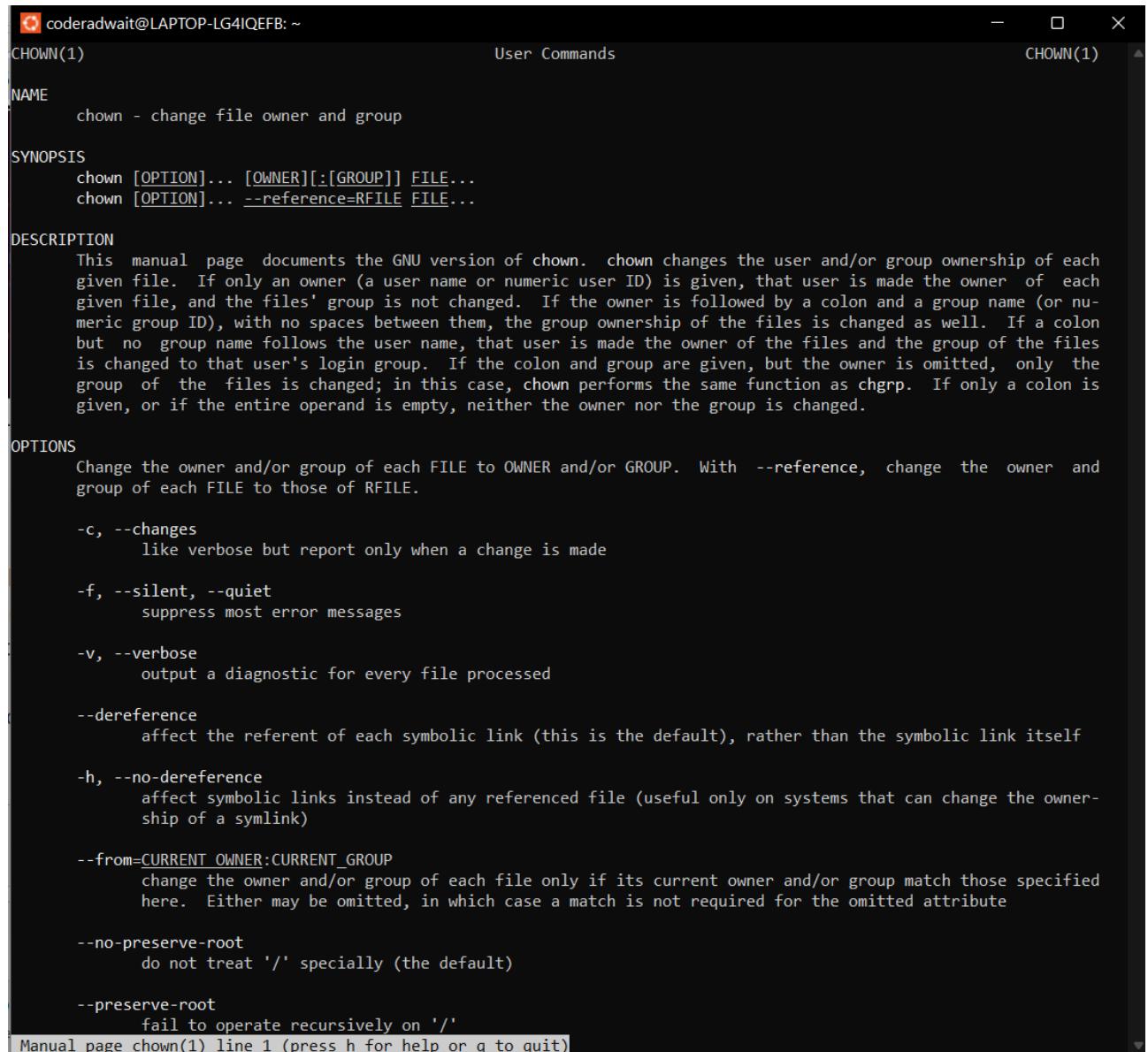
The `history` command lists the commands you have previously issued on the command line. You can repeat any of the commands from your history by typing an exclamation point ! and the number of the command from the history list.

```
coderadwait@LAPTOP-LG4IQEFB: ~
coderadwait@LAPTOP-LG4IQEFB:~$ history
 1  clear
 2  pwd
 3  cd ..
 4  cd MyPdfs/
 5  mv DSA3.pdf Ubuntu\New_Directory
 6  sudo mv DSA3.pdf Ubuntu\New_Directory
 7  wc Mytxt.txt
 8  cat Mytxt.txt
 9  uniq Mytxt.txt
10  clear
11  cat Mytxt.txt
12  uniq Mytxt.txt
13  ls
14  rem DSA2.pdf
15  rm DSA2.pdf
16  rm Mytxt.txt
17  sudo rm Mytxt.txt
18  ls
19  tail -n 1 NewText.txt
20  cat NewText.txt
```

22. man

The man command displays the “man pages” for a command in less . The man pages are the user manual for that command. Because man uses less to display the man pages, you can use the search capabilities of less.

```
coderadwait@LAPTOP-LG4IQEFB:~$ man chown
```



```
coderadwait@LAPTOP-LG4IQEFB: ~
CHOWN(1)                               User Commands                               CHOWN(1)

NAME
    chown - change file owner and group

SYNOPSIS
    chown [OPTION]... [OWNER][:GROUP]] FILE...
    chown [OPTION]... --reference=RFILE FILE...

DESCRIPTION
    This manual page documents the GNU version of chown. chown changes the user and/or group ownership of each given file. If only an owner (a user name or numeric user ID) is given, that user is made the owner of each given file, and the files' group is not changed. If the owner is followed by a colon and a group name (or numeric group ID), with no spaces between them, the group ownership of the files is changed as well. If a colon but no group name follows the user name, that user is made the owner of the files and the group of the files is changed to that user's login group. If the colon and group are given, but the owner is omitted, only the group of the files is changed; in this case, chown performs the same function as chgrp. If only a colon is given, or if the entire operand is empty, neither the owner nor the group is changed.

OPTIONS
    Change the owner and/or group of each FILE to OWNER and/or GROUP. With --reference, change the owner and group of each FILE to those of RFILE.

    -c, --changes
        like verbose but report only when a change is made

    -f, --silent, --quiet
        suppress most error messages

    -v, --verbose
        output a diagnostic for every file processed

    --dereference
        affect the referent of each symbolic link (this is the default), rather than the symbolic link itself

    -h, --no-dereference
        affect symbolic links instead of any referenced file (useful only on systems that can change the ownership of a symlink)

    --from=CURRENT_OWNER:CURRENT_GROUP
        change the owner and/or group of each file only if its current owner and/or group match those specified here. Either may be omitted, in which case a match is not required for the omitted attribute

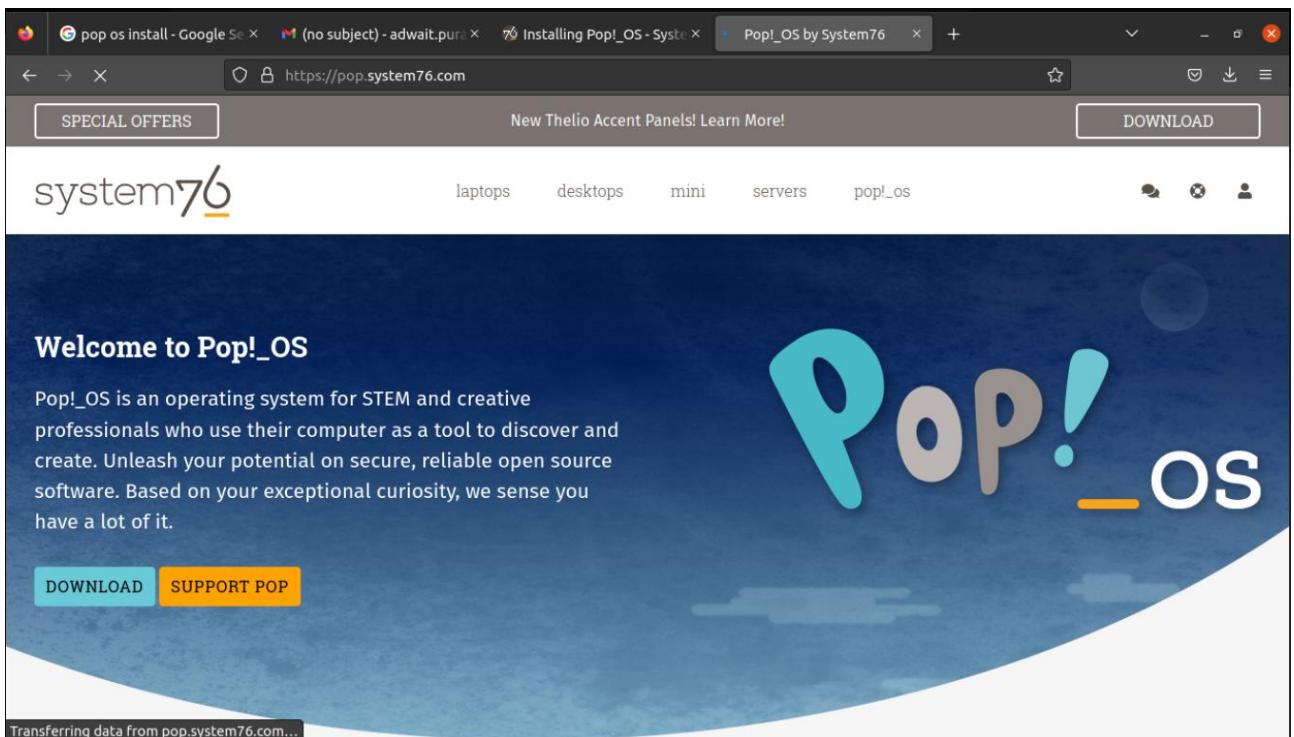
    --no-preserve-root
        do not treat '/' specially (the default)

    --preserve-root
        fail to operate recursively on '/'
```

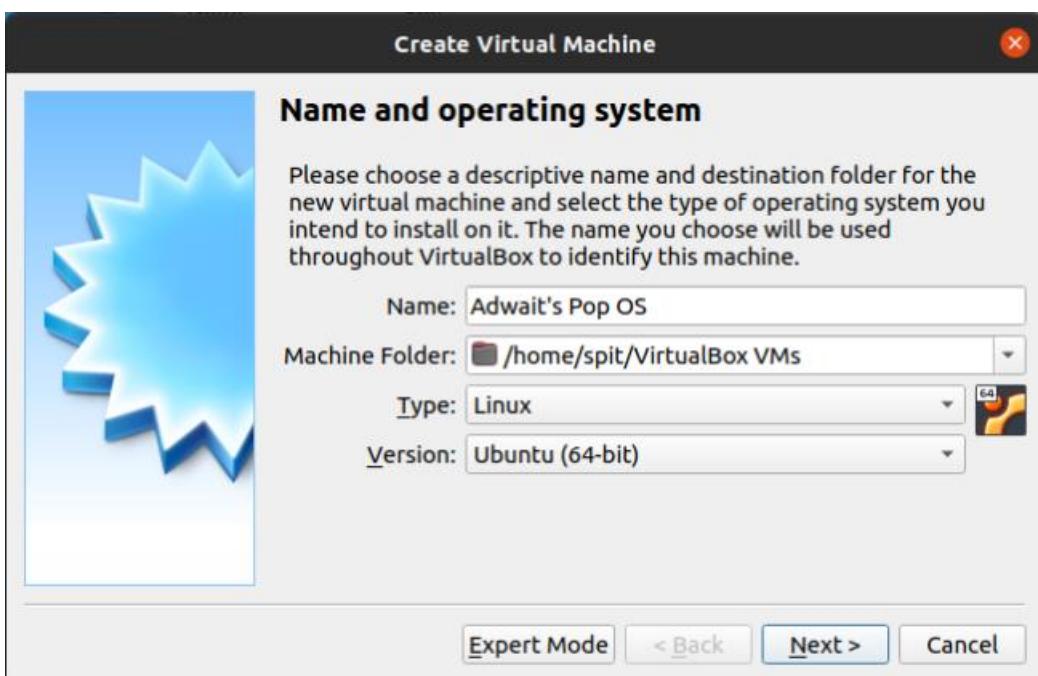
Manual page chown(1) line 1 (press h for help or q to quit)

Installation of Pop OS

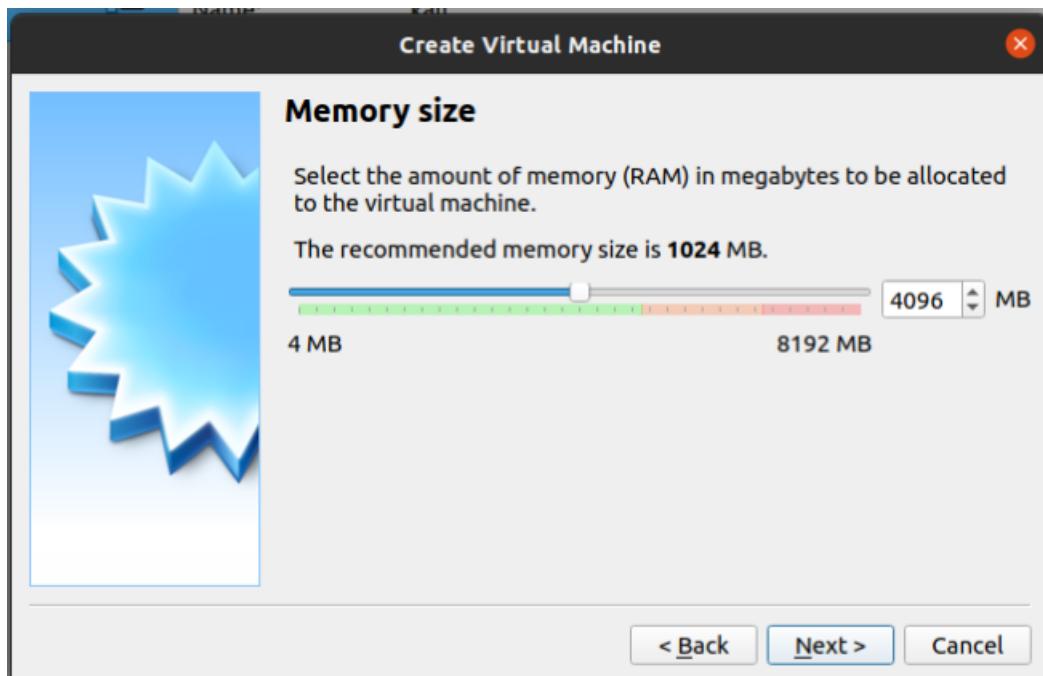
Site of Pop OS



Initializing Virtual Box



Initializing Memory Size



Choosing the file type of Hard-disk

Create Virtual Hard Disk

Hard disk file type



Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

VDI (VirtualBox Disk Image)
 VHD (Virtual Hard Disk)
 VMDK (Virtual Machine Disk)

[Expert Mode](#) [**Next >**](#) [Cancel](#)

Create Virtual Hard Disk

Storage on physical hard disk



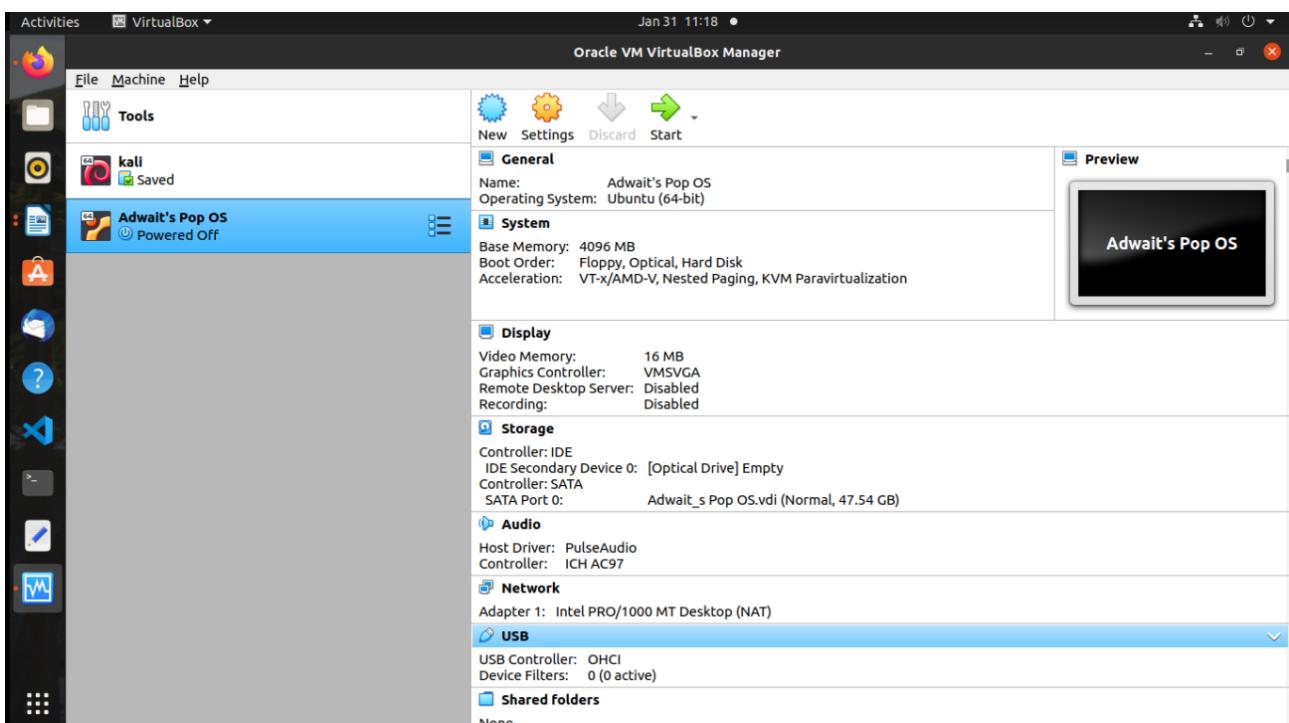
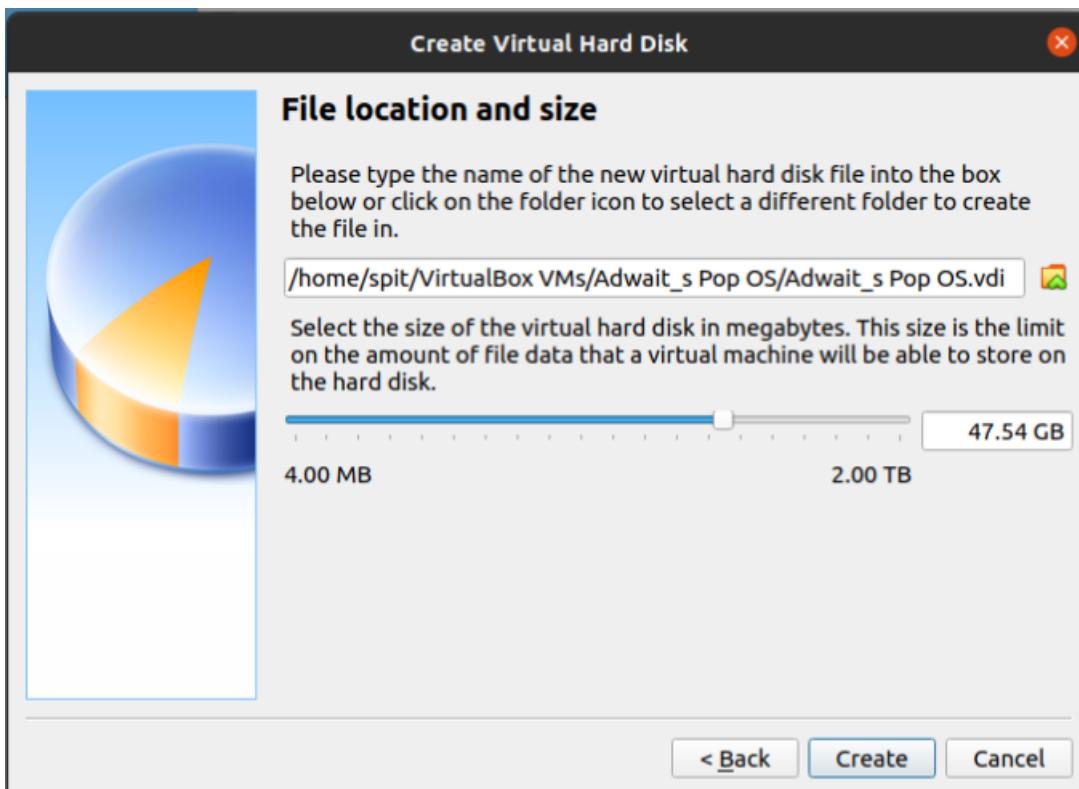
Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

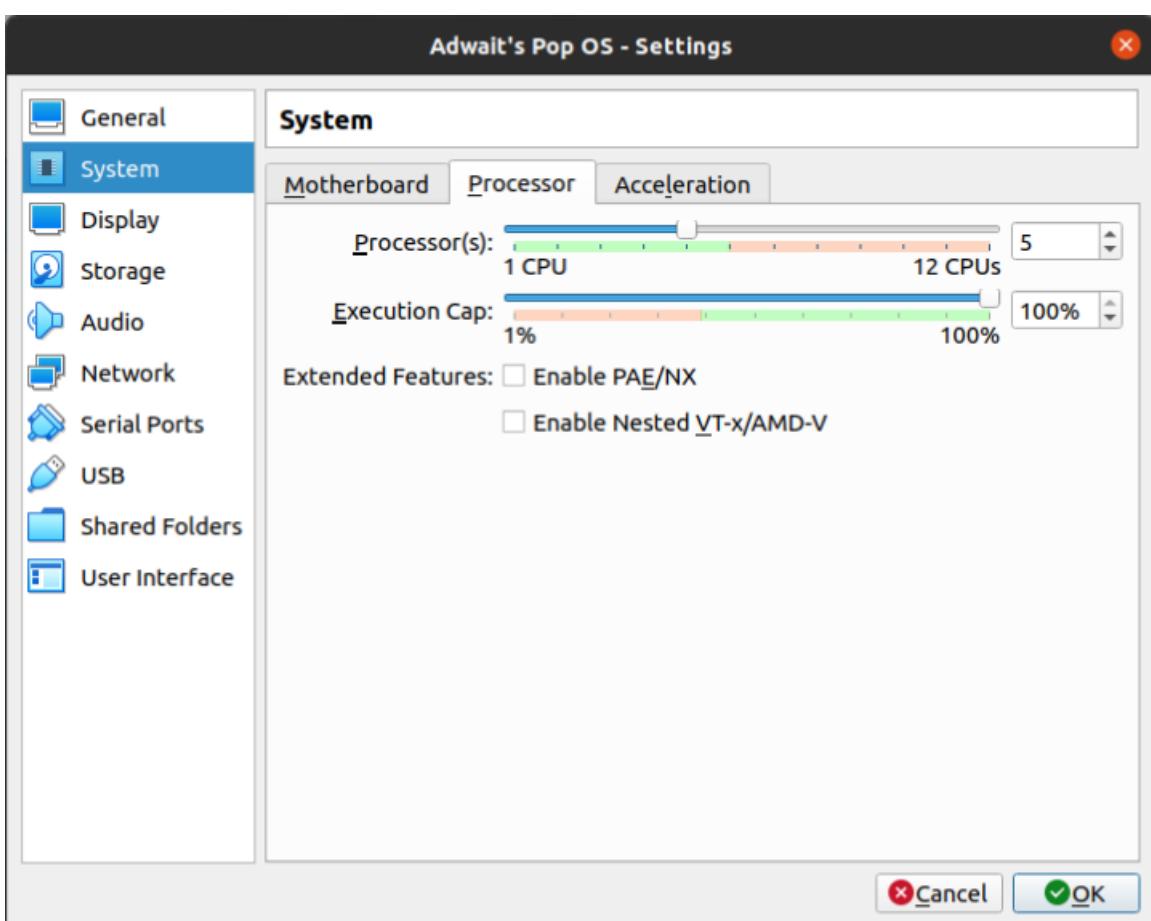
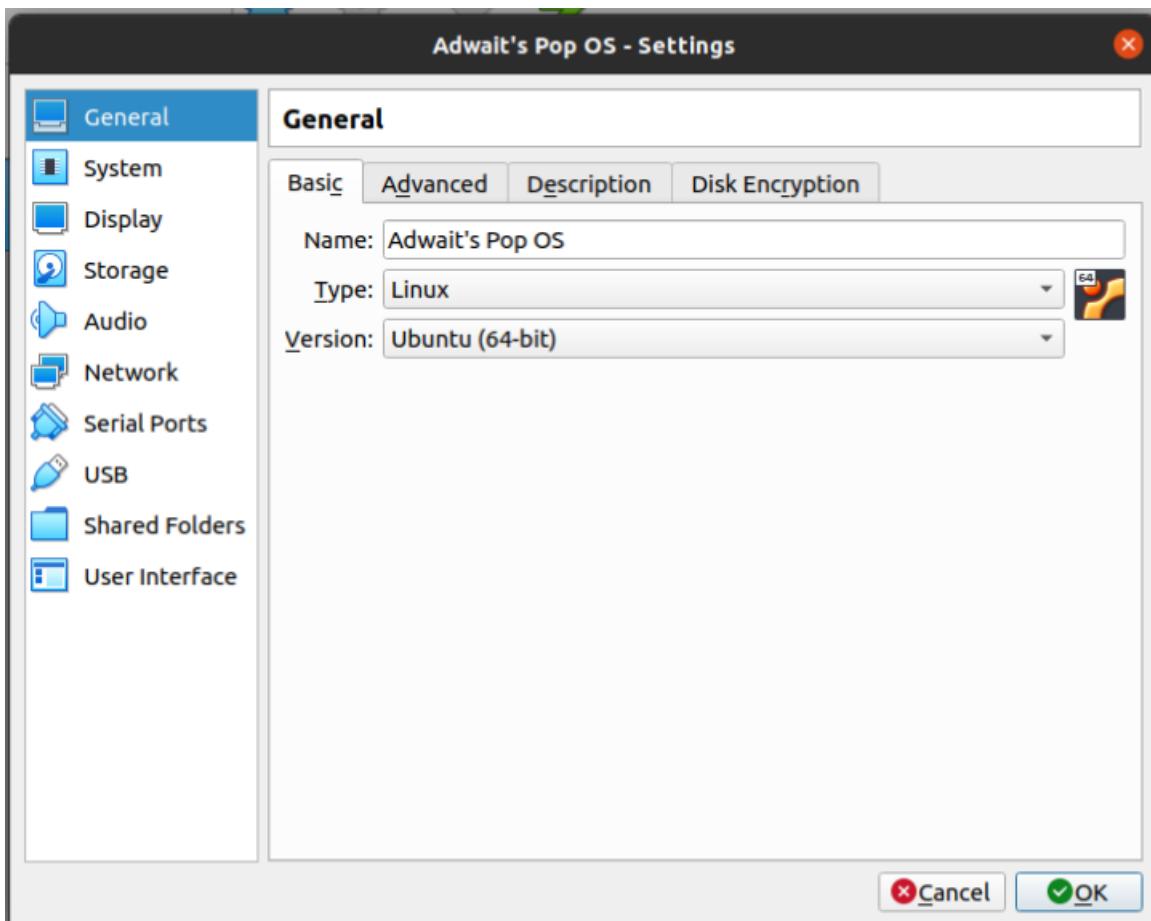
A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

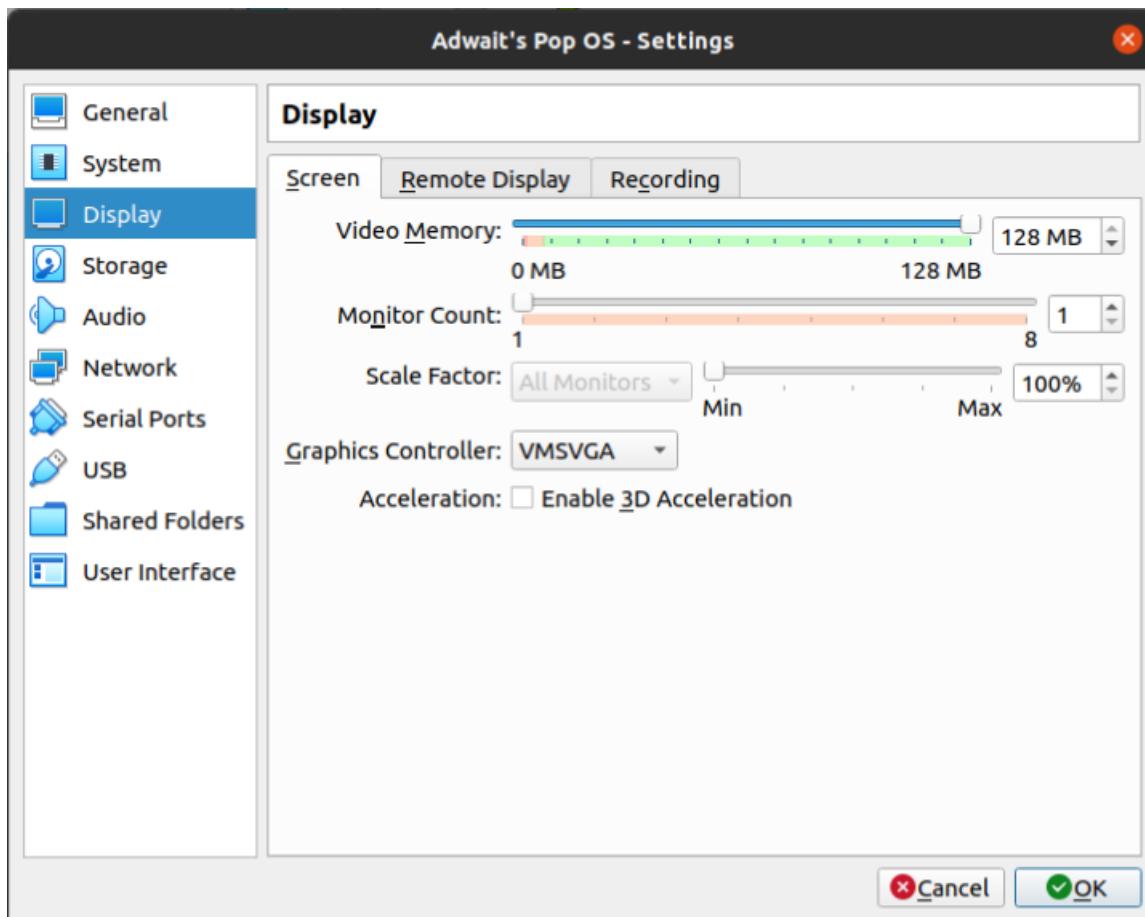
A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

Dynamically allocated
 Fixed size

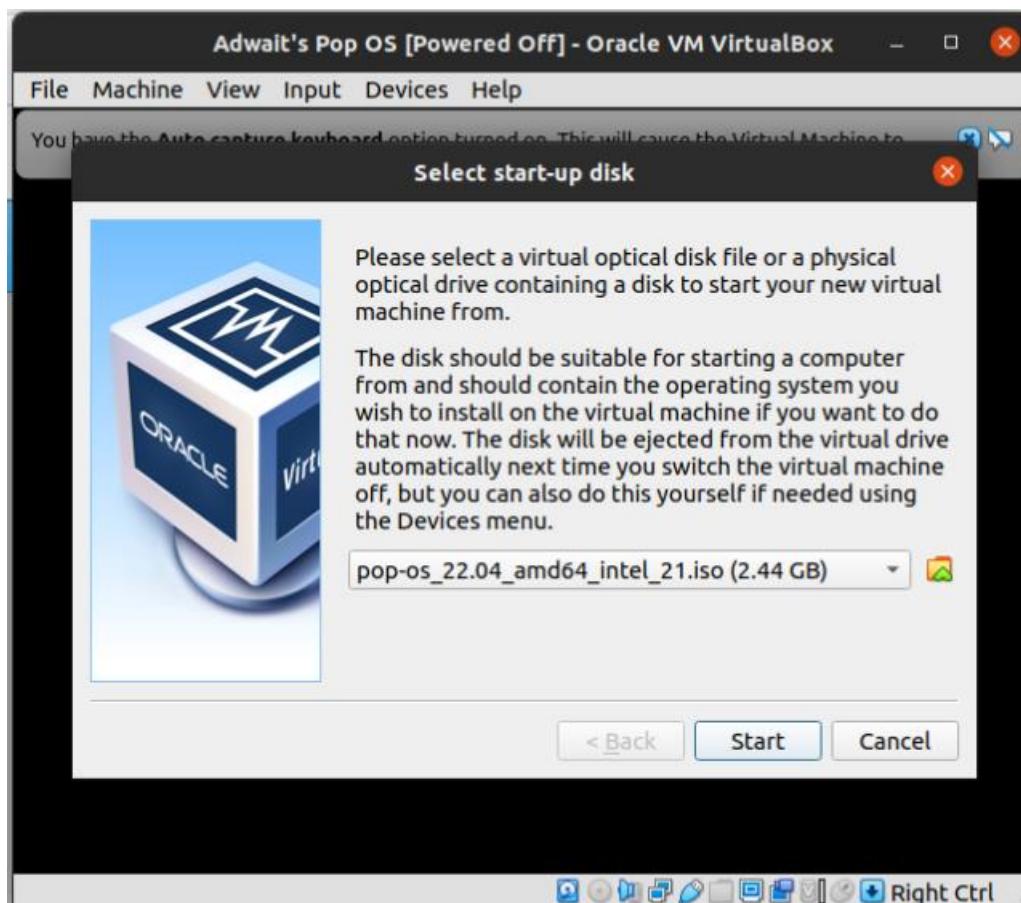
[**< Back**](#) [**Next >**](#) [Cancel](#)

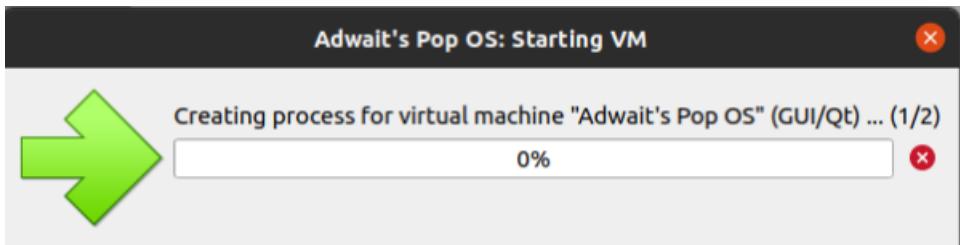






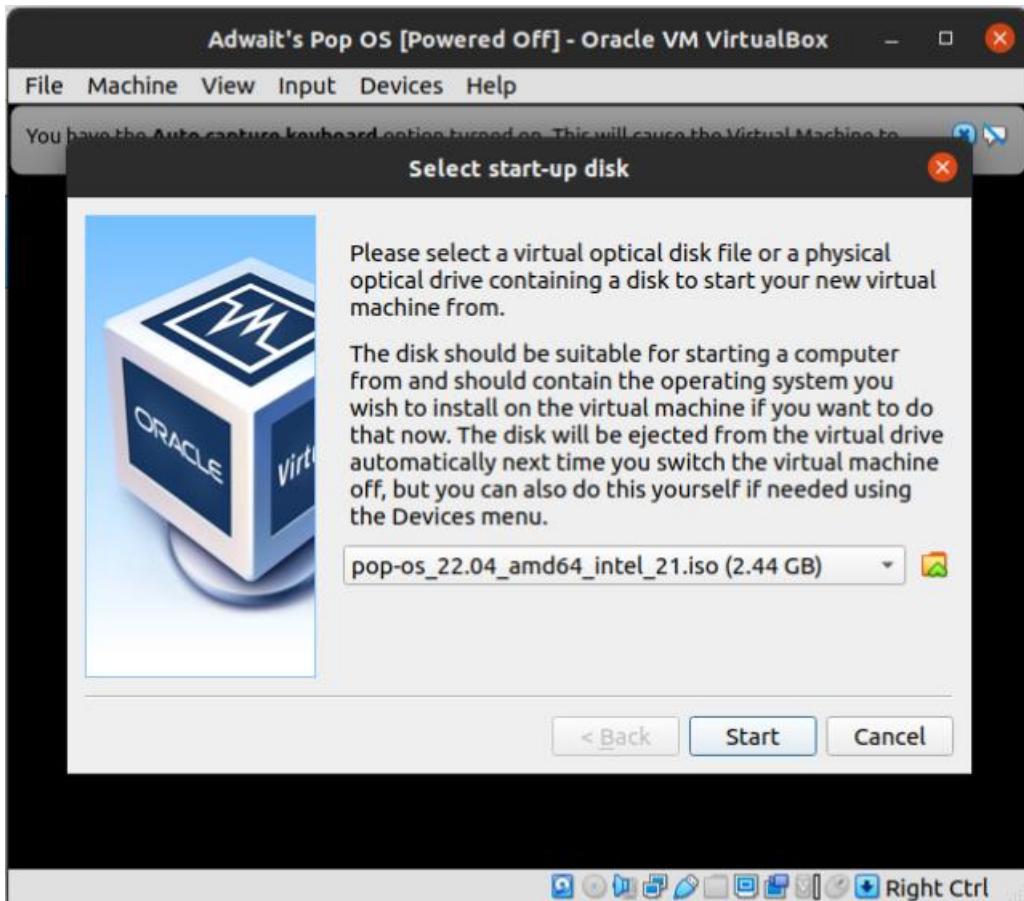
Select and start Pop OS



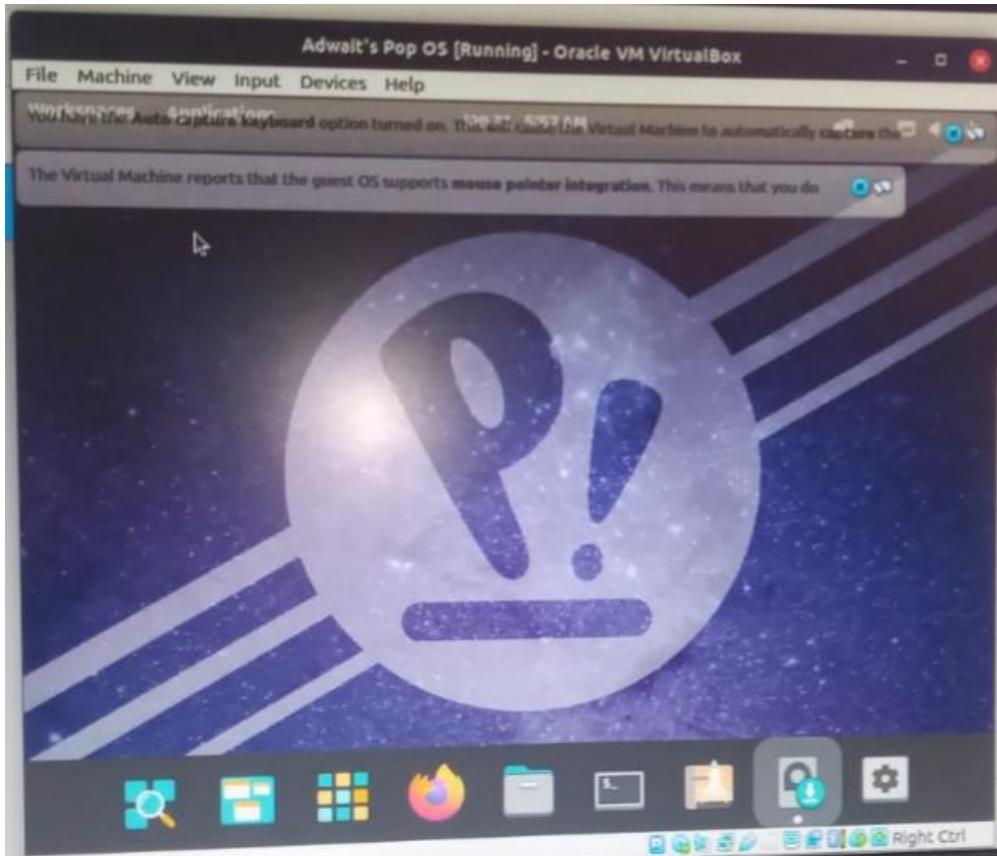


Click on cross.

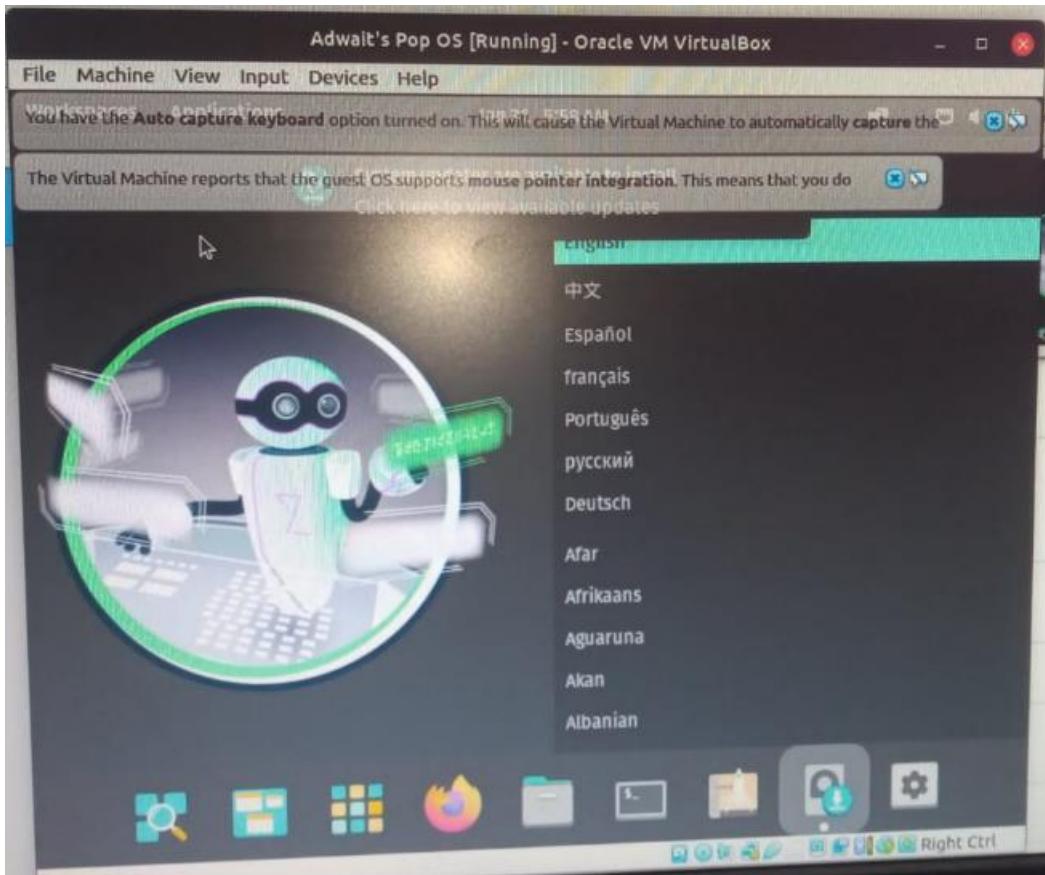
Selecting the start-up disk



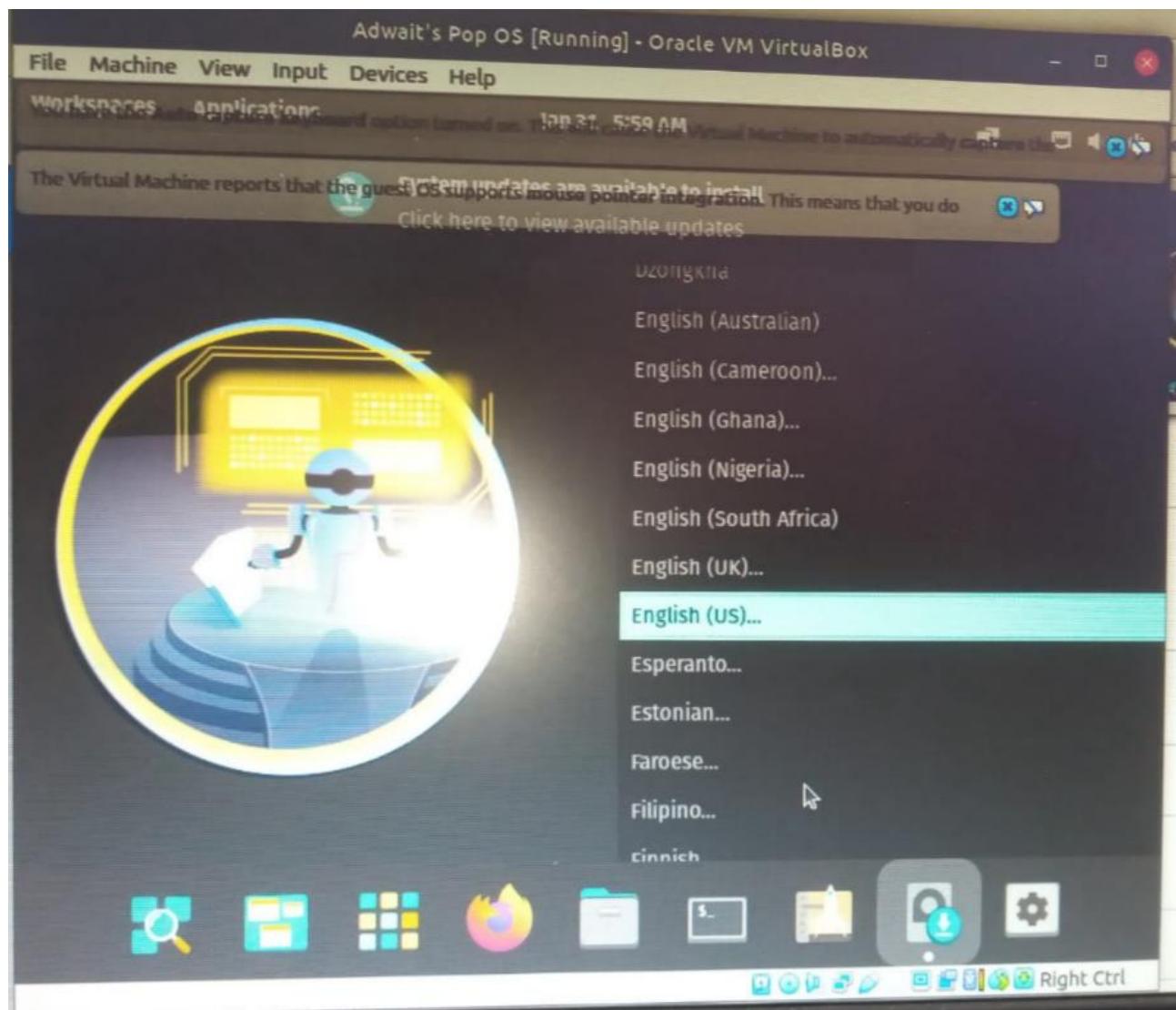
Starting display of Pop OS

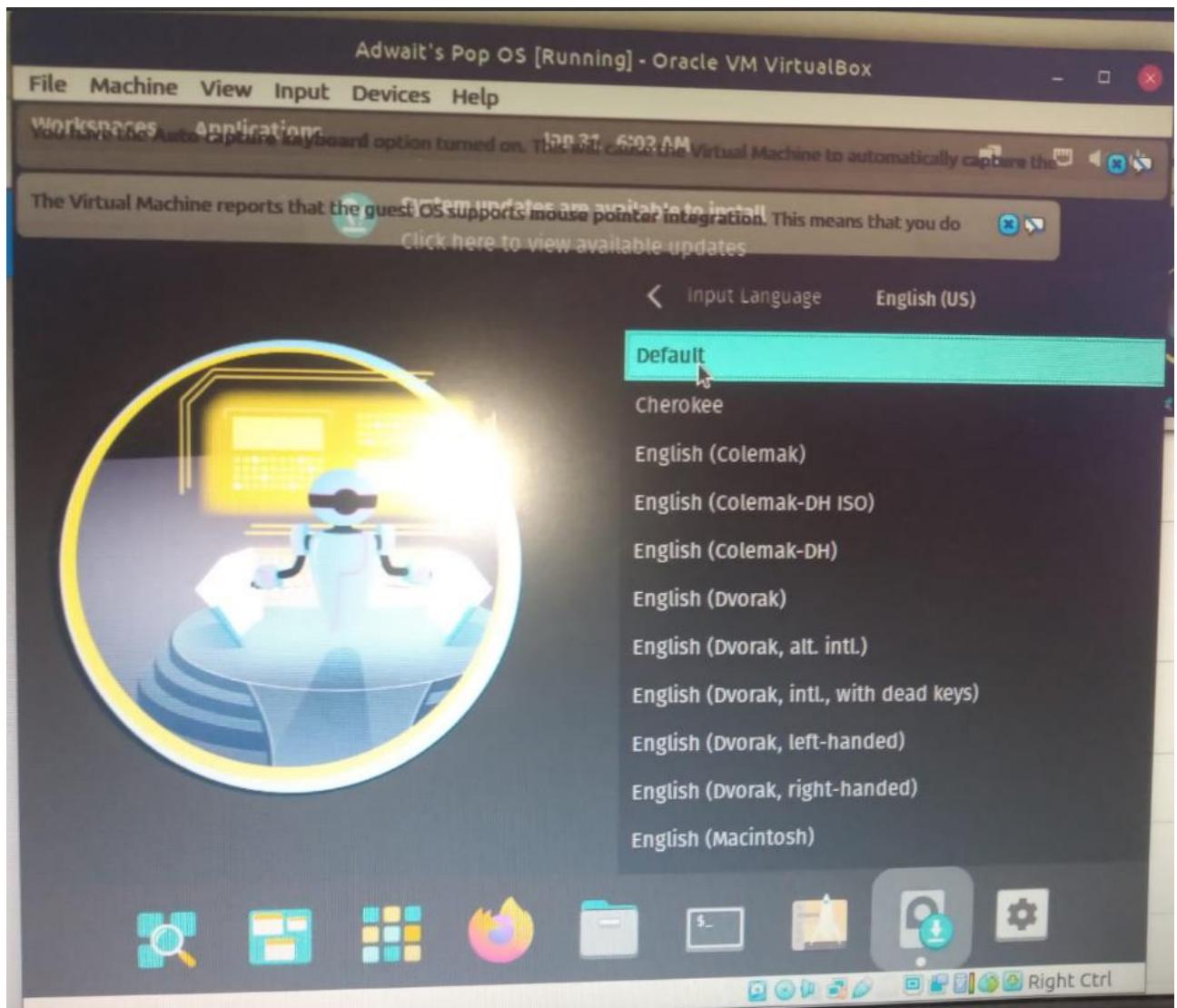


Select English Language

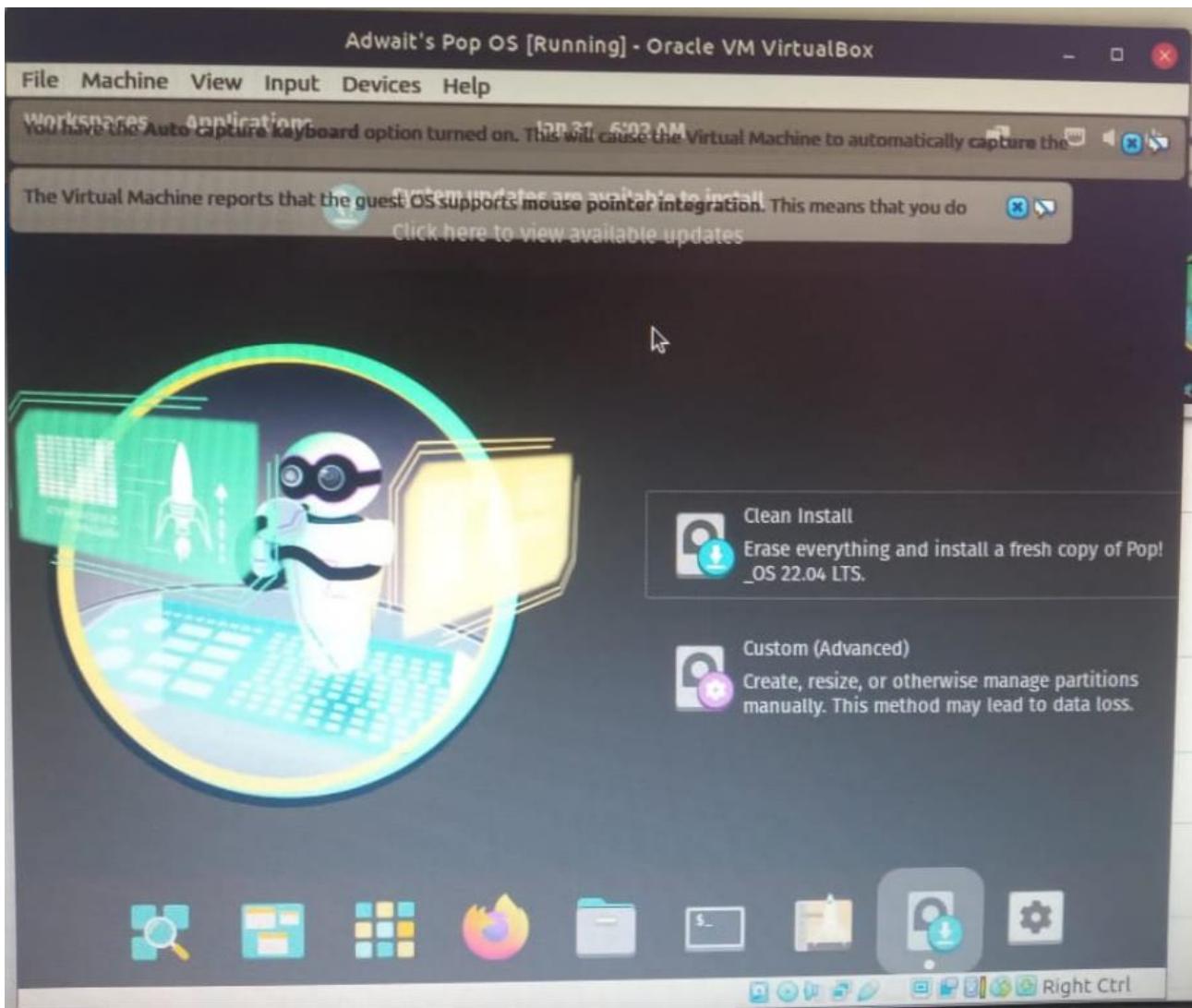


Select Country as United States

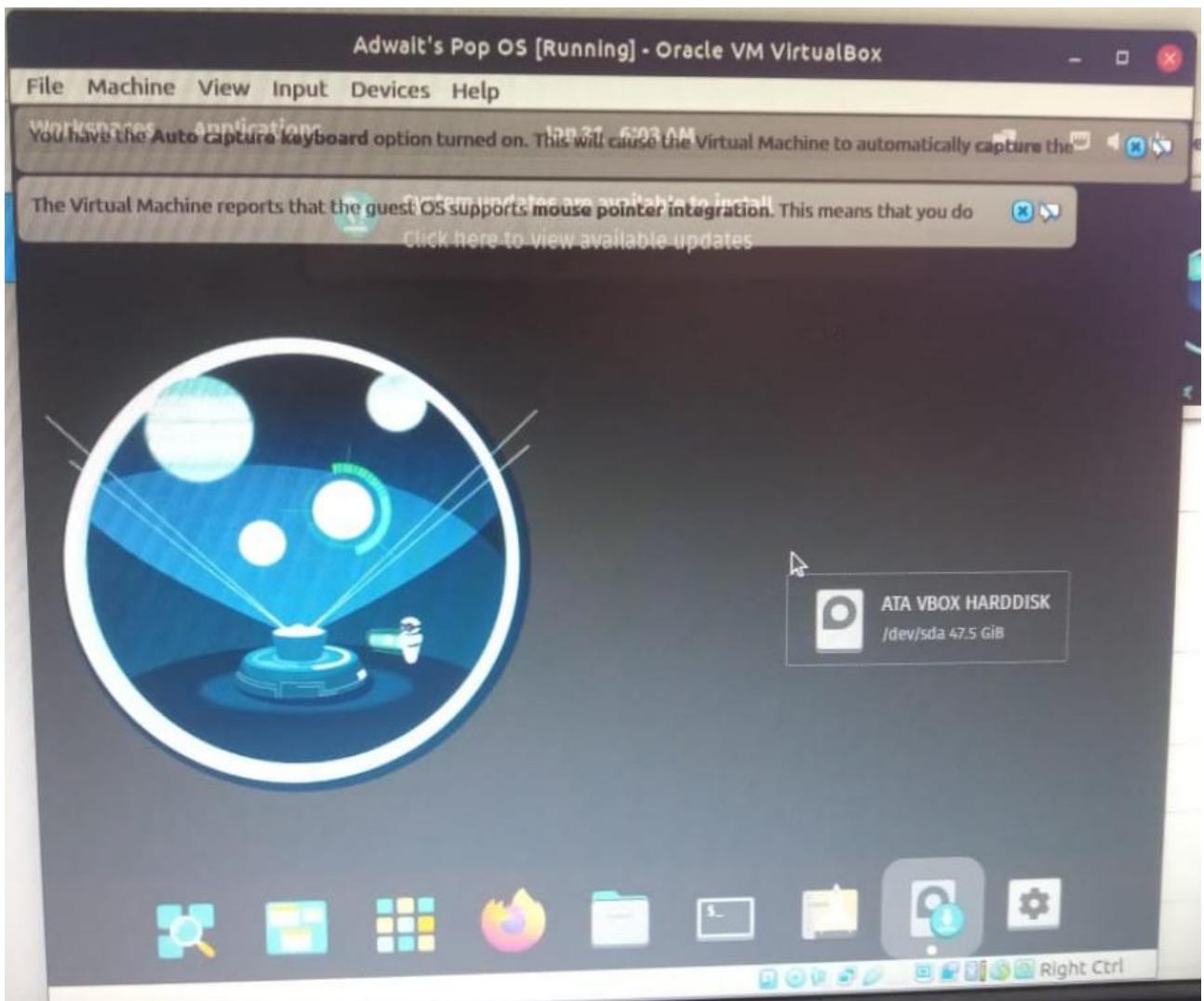




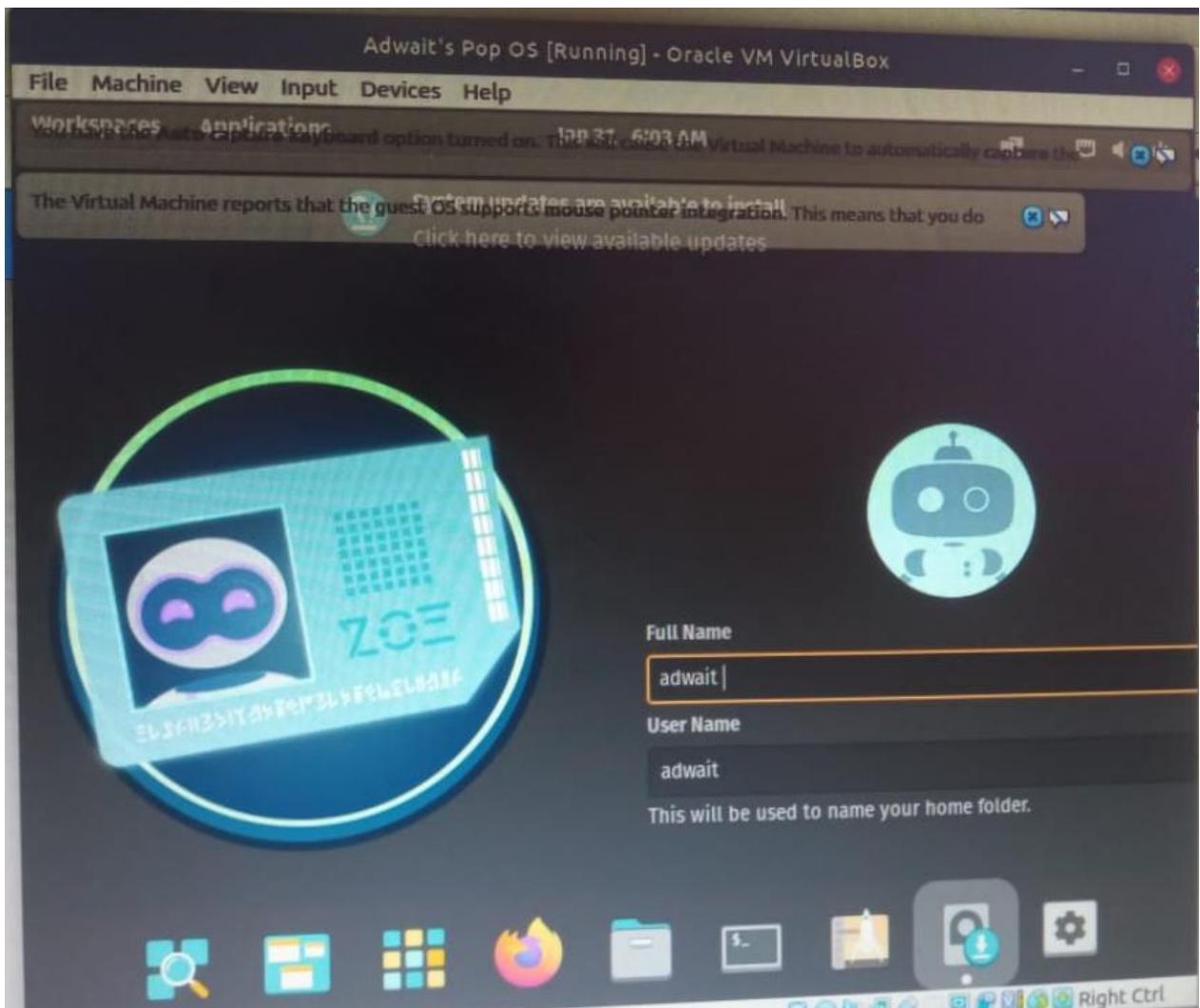
Click Clean Install Pop OS



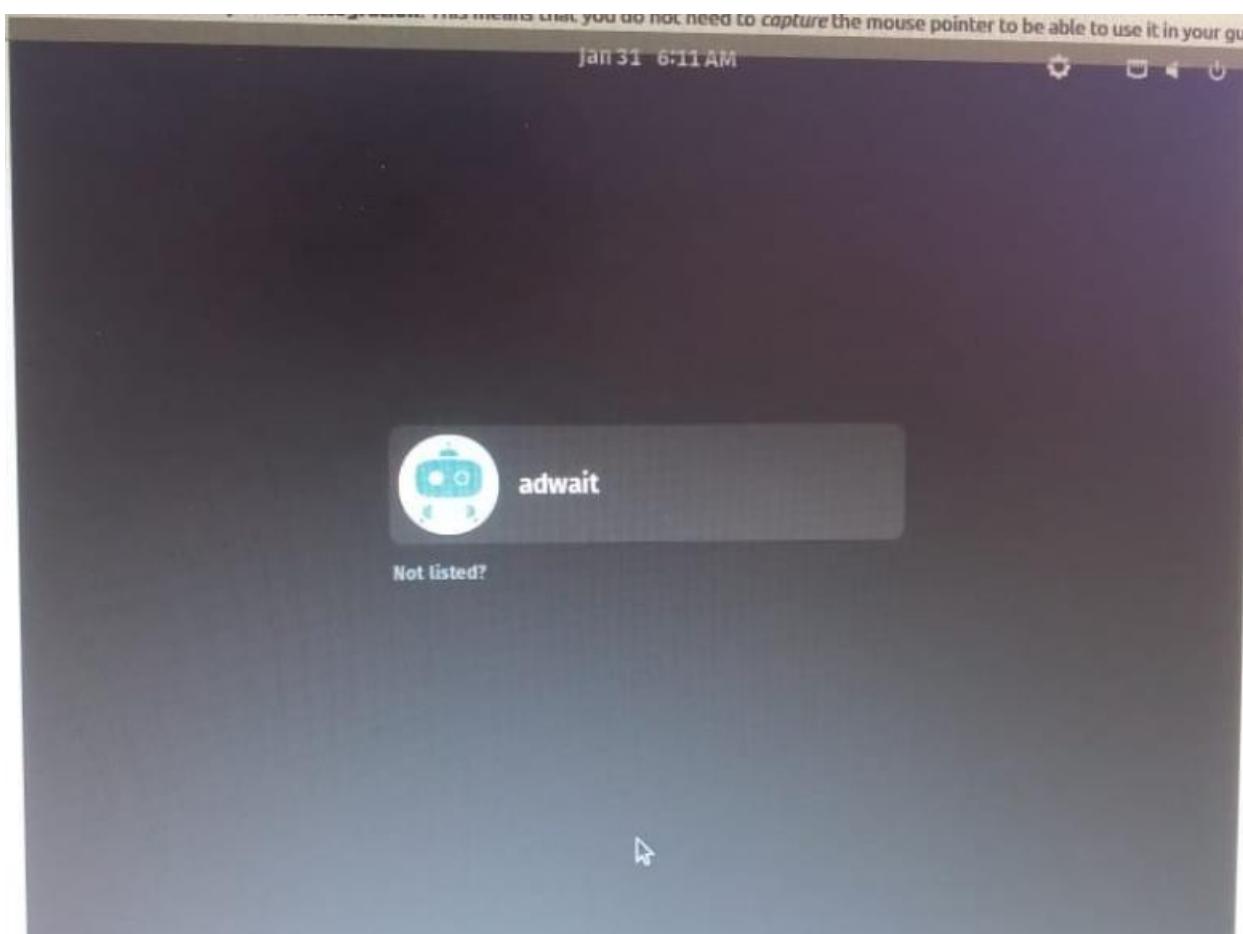
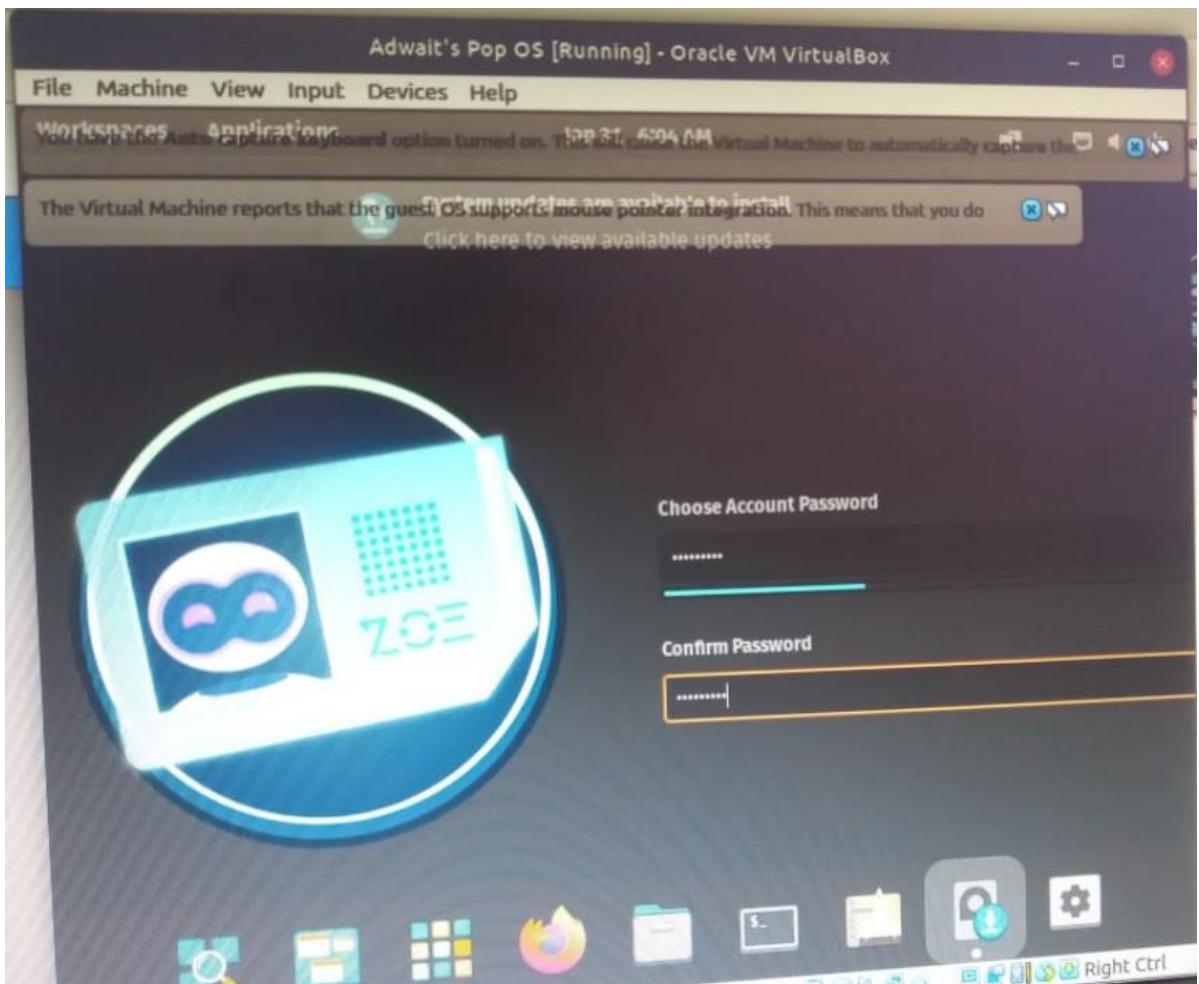
Click on ATA VBOX HARDISK



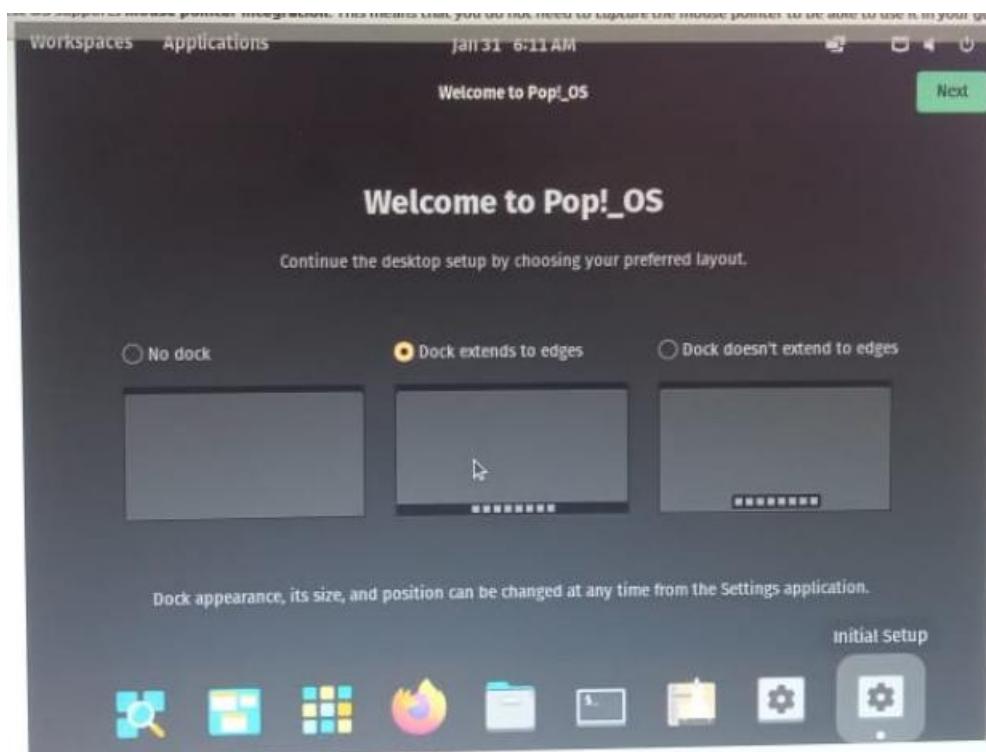
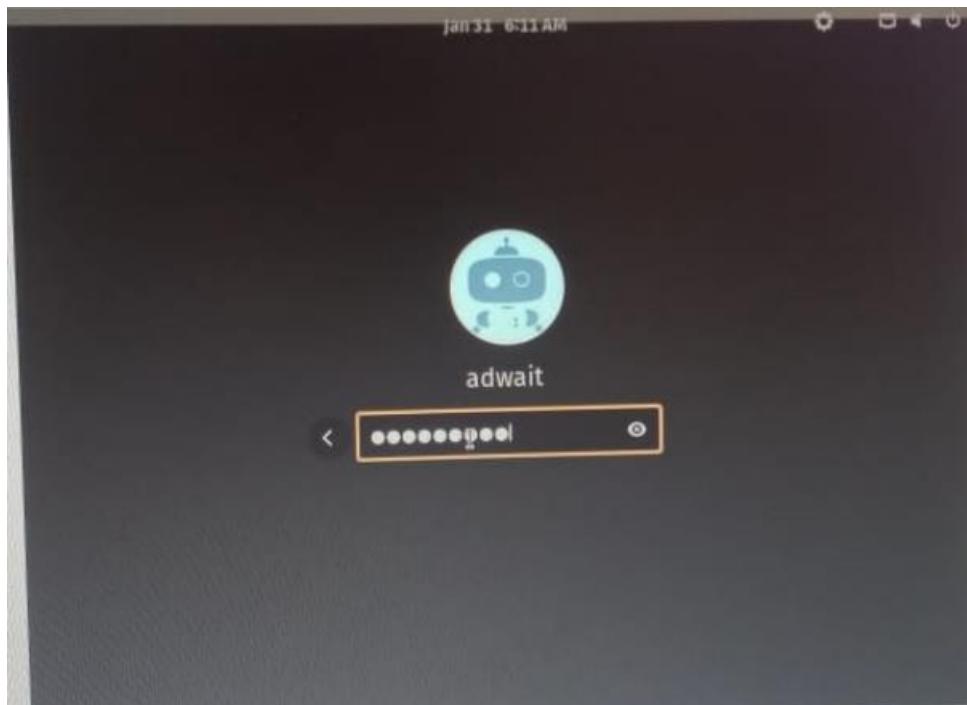
Enter your username.

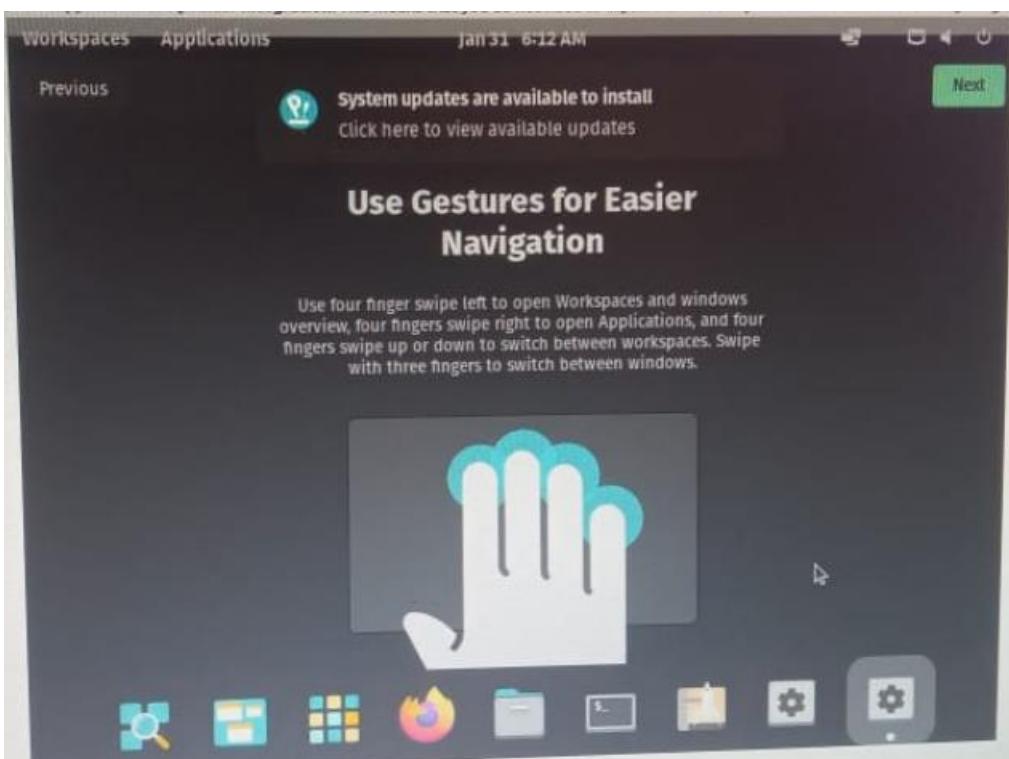
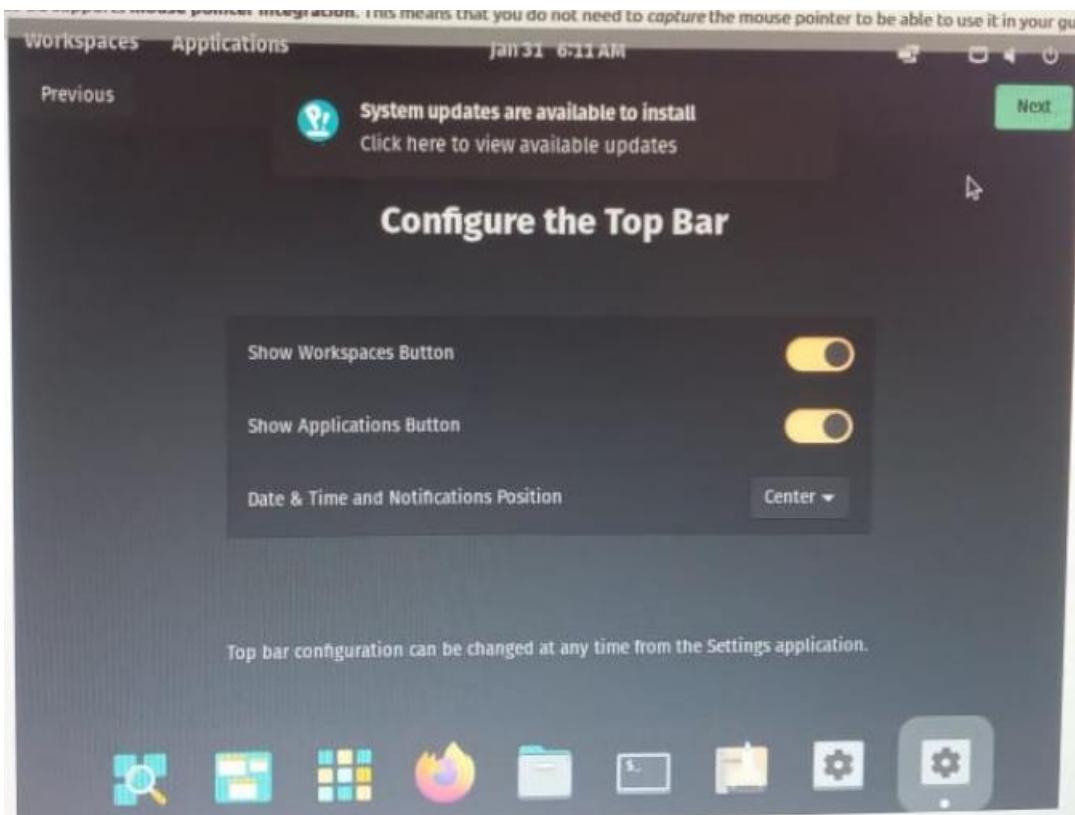


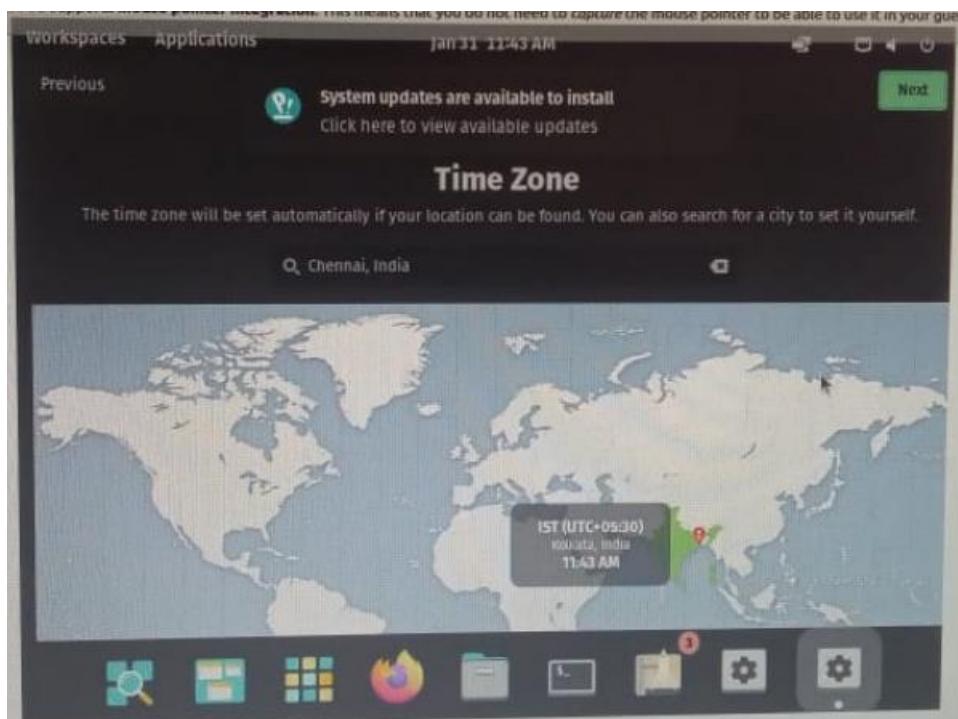
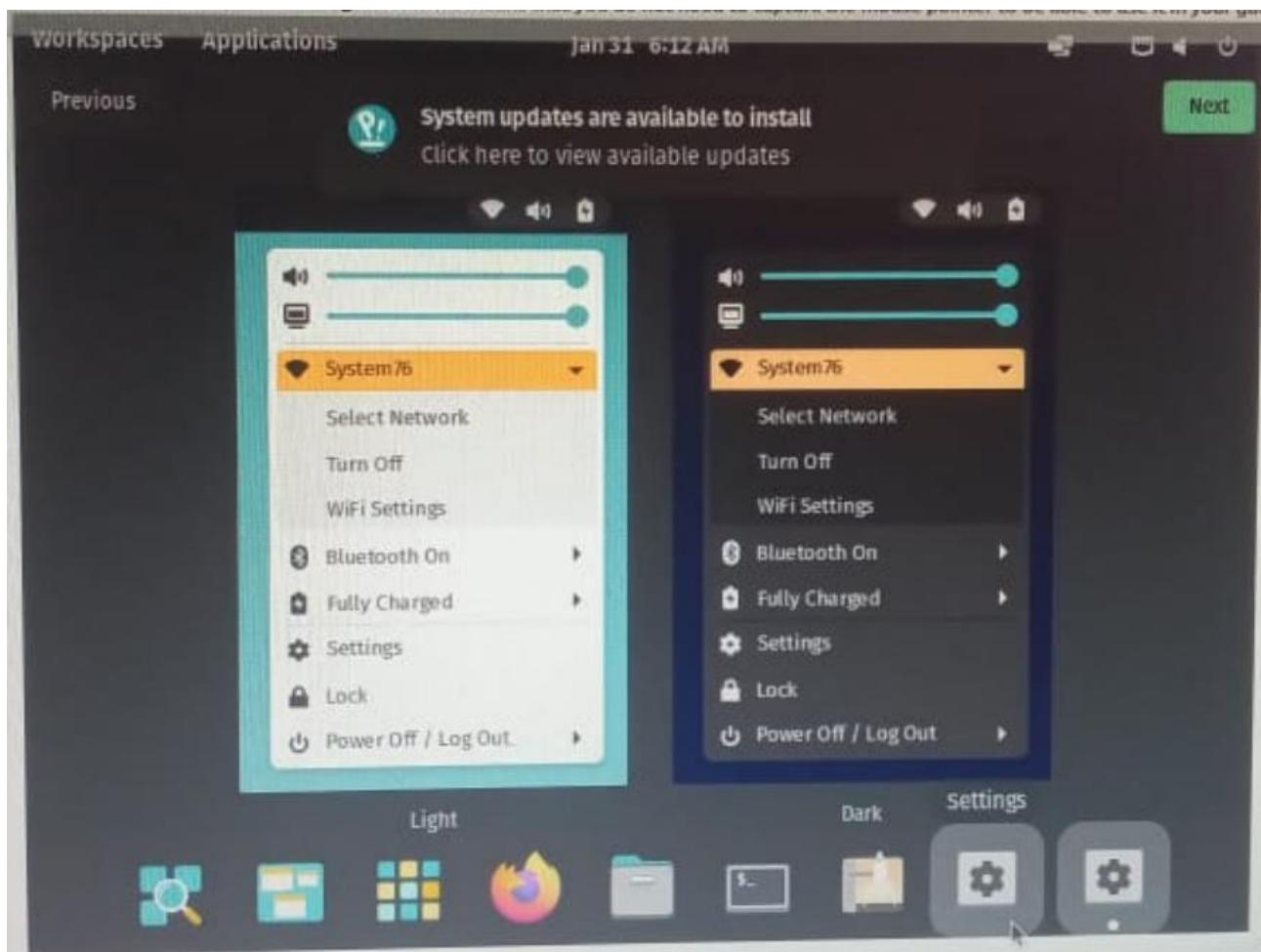
Set your password

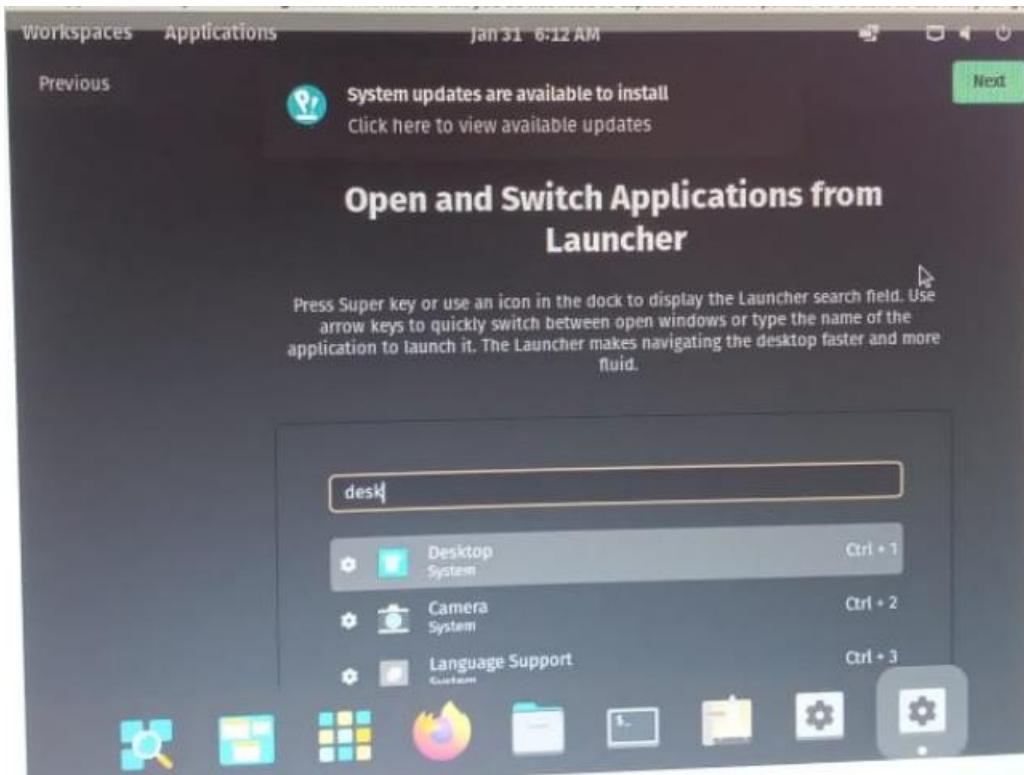
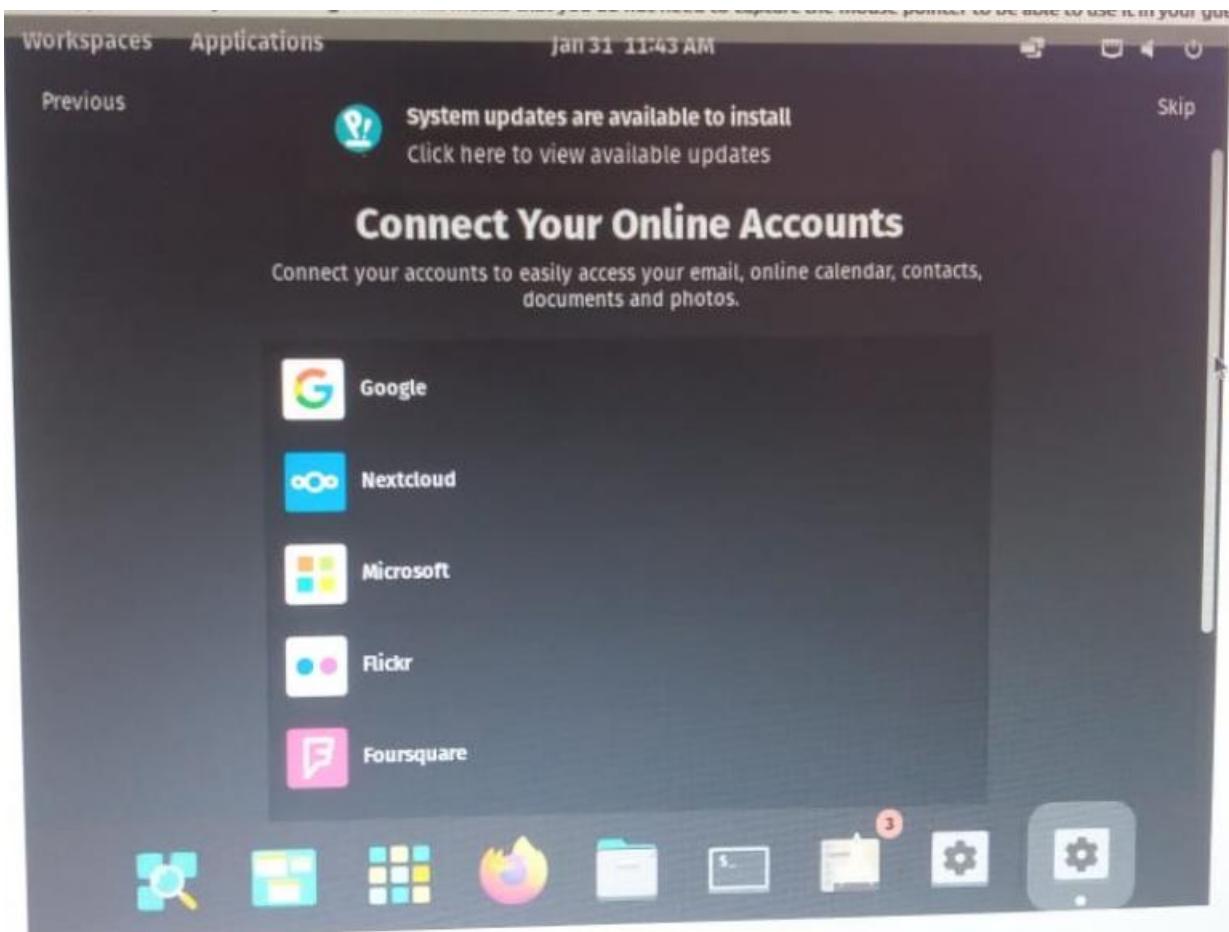


Now click on the icon and enter the password

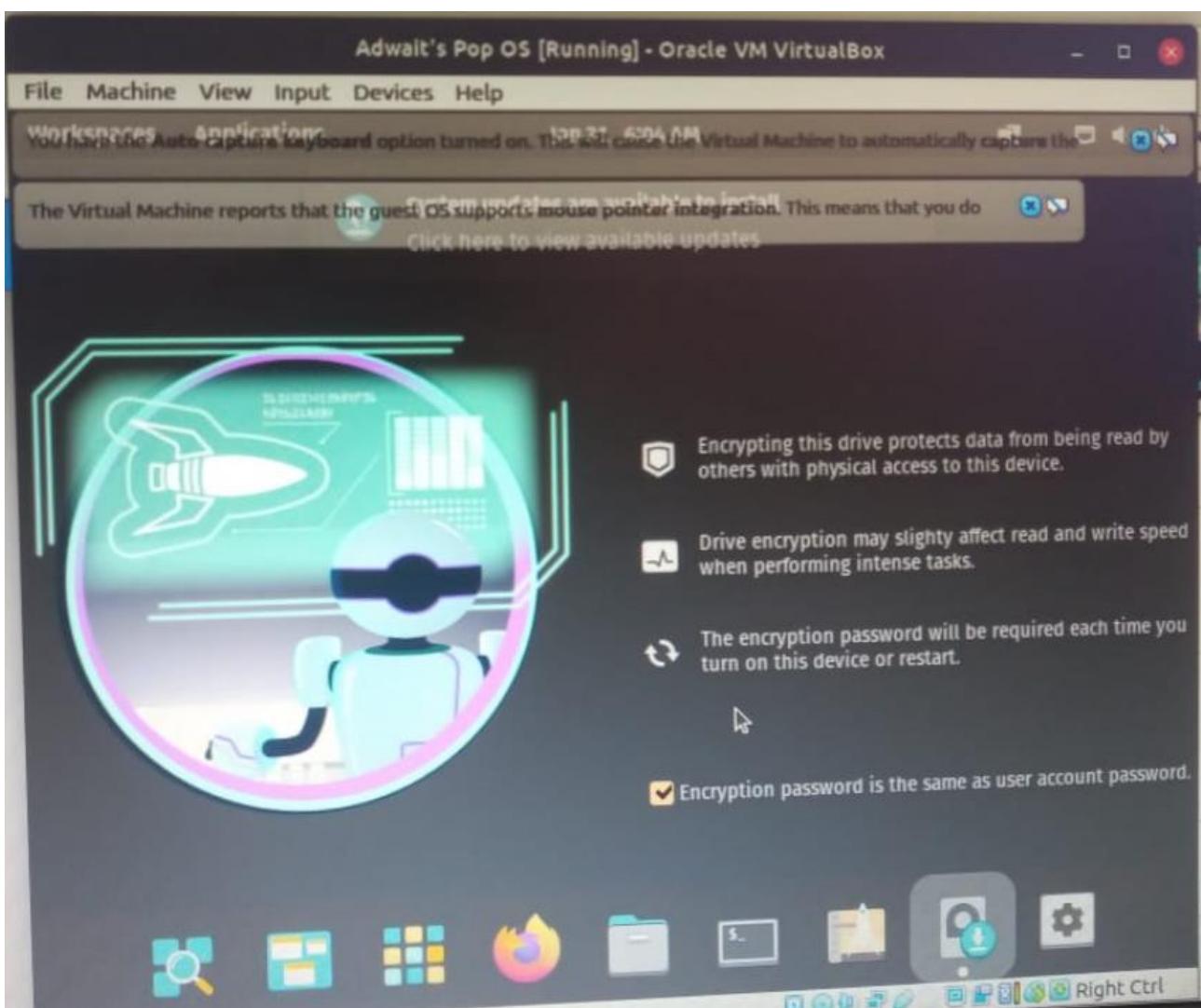


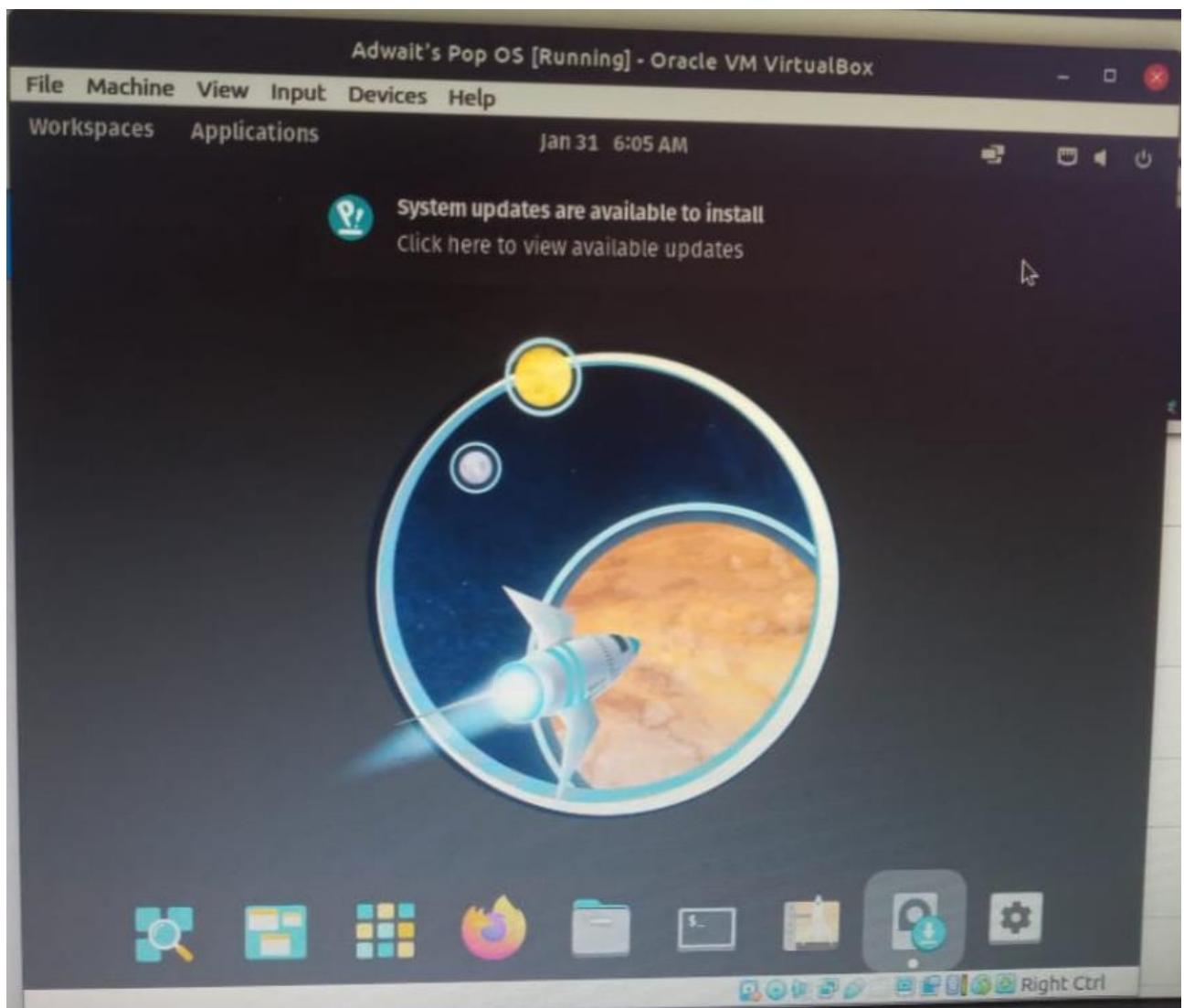




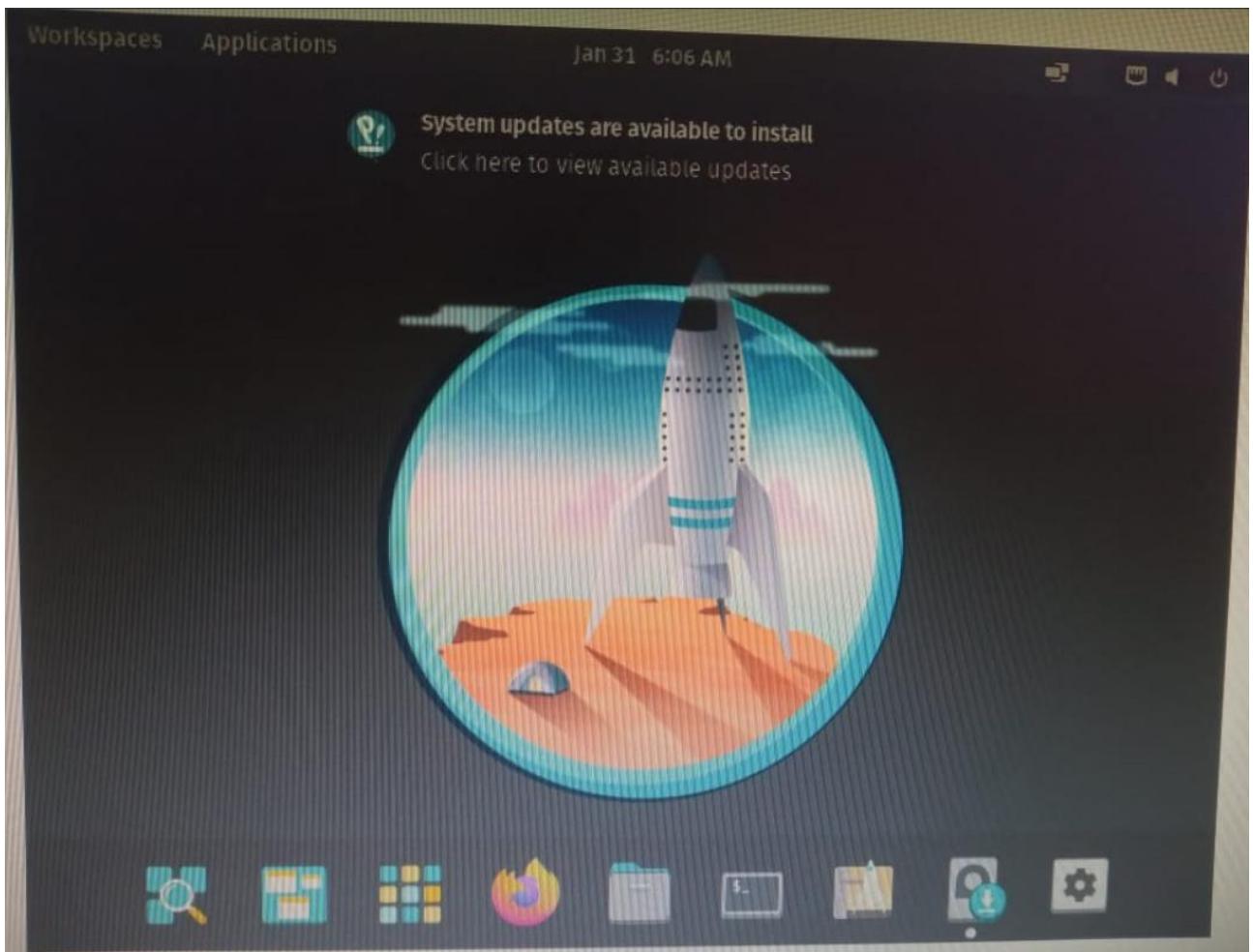


Click on the checkbox and Press Enter

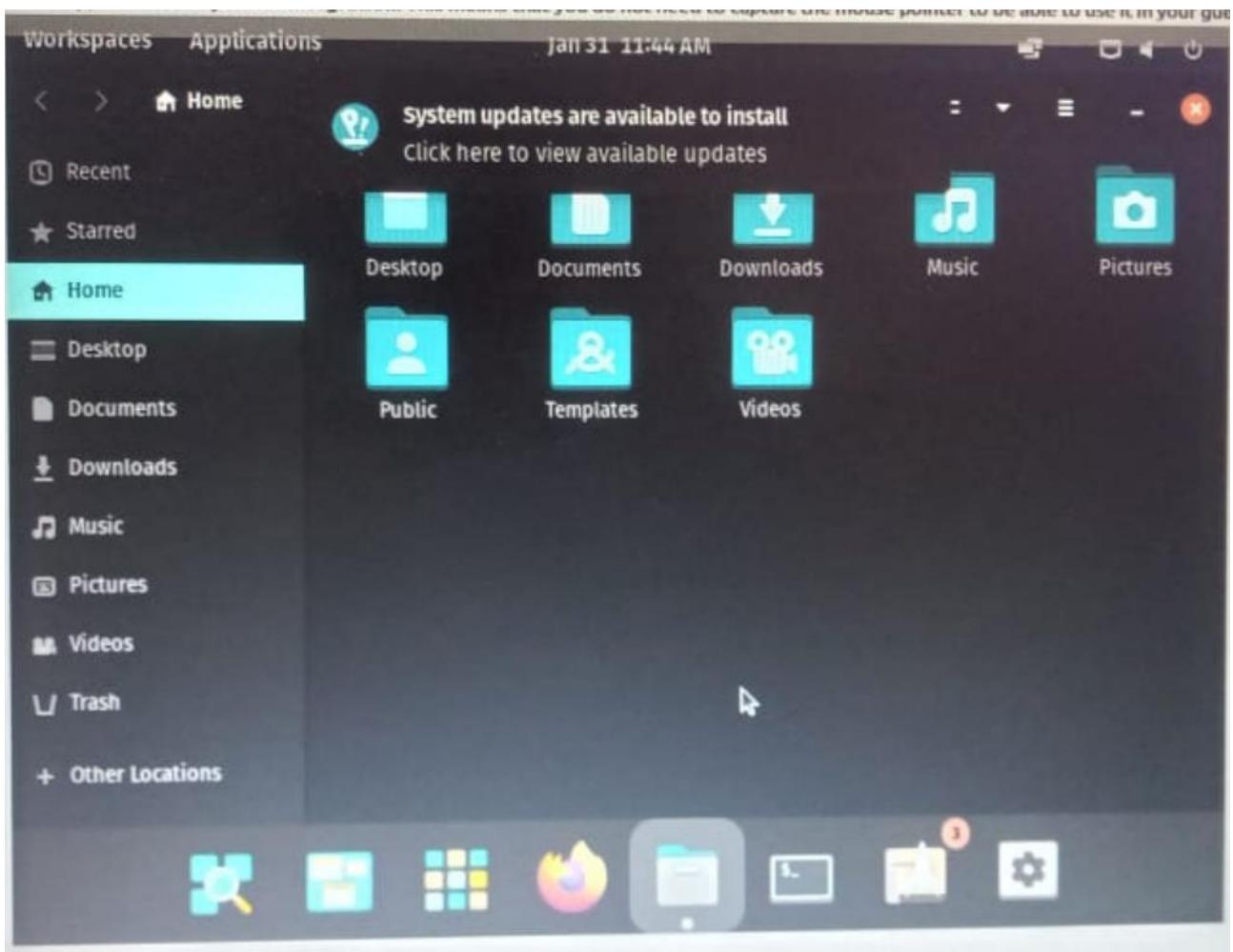




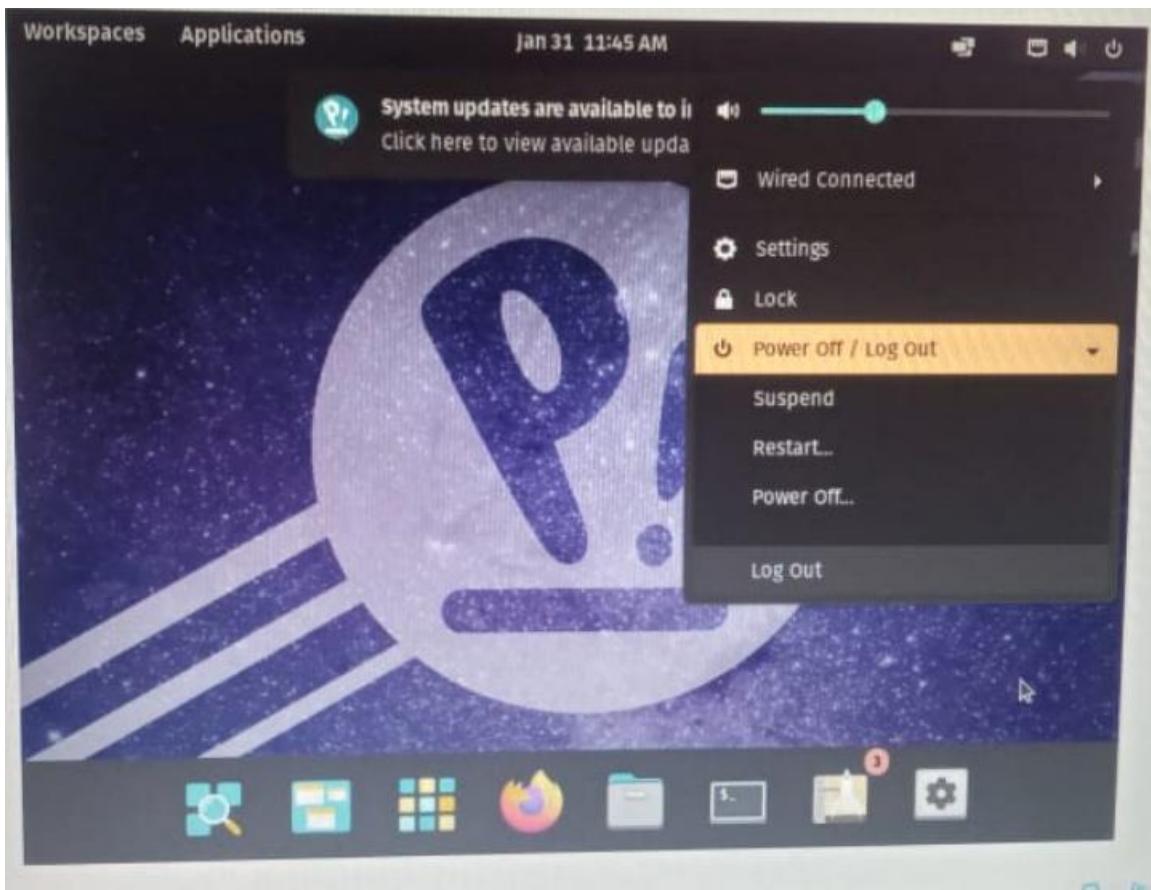
GUI of Pop OS



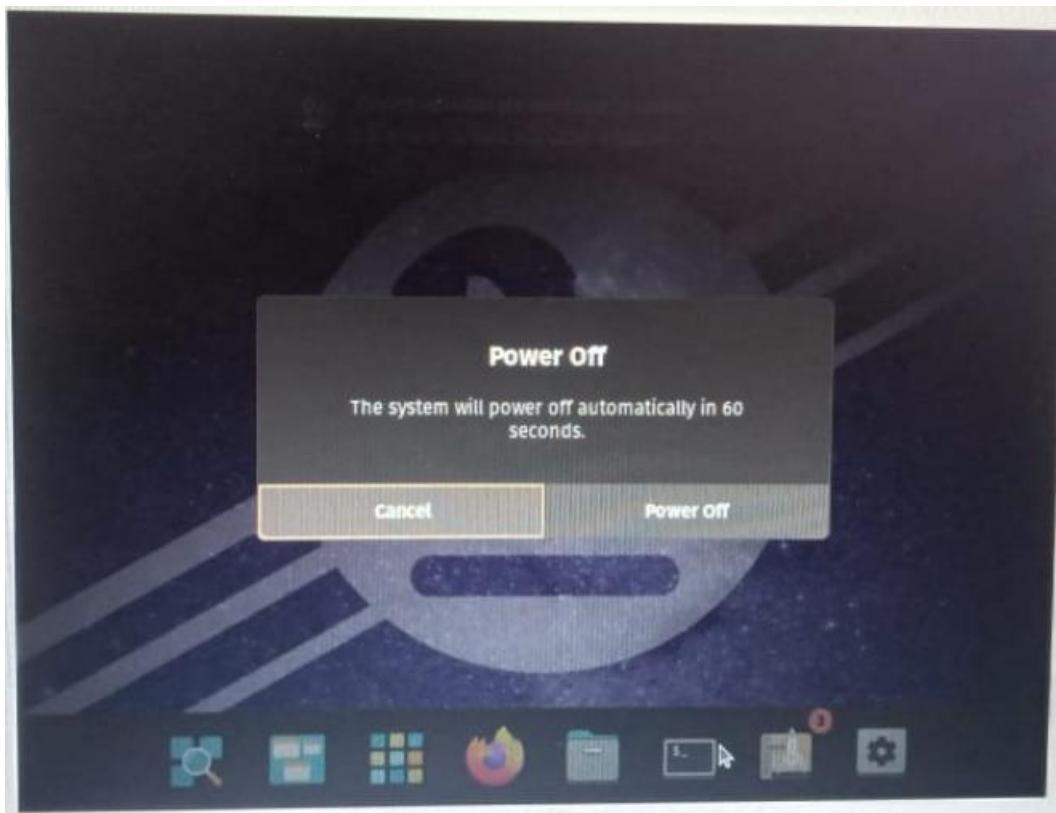
Checking whether the file manager is working or not



Toggling the volume



Turning power off



Conclusion:

I installed Pop OS of Linux from Chrome . I also ran Utility , Networking , File and Directory commands on Linux Terminal . I installed WSL in my own desktop and resolved the error which occurred while create new file or copying from one location to the other , by using “sudo” before each linux command . I also learnt about virtual machine in this experiment.