

Introduction to Robot Operating System (ROS 1)

Dr. Essa Alghannam

توزيع الدرجات ونمط الاختبارات مقرر نظم تشغيل

#	نمط الاختبار	نسبة المئوية للاختبار	الاسبوع
Test 1	تحريري	15%	السادس
Test 2	حاسوبي	15%	الثاني عشر
Practical exam	حاسوبي	20%	الثالث عشر
Final exam	حاسوبي	50%	الخامس عشر

Prerequisites:

- Basic knowledge (ubuntu operating system and terminal commands)
- Basic programming experience (C++, Python)
- knowledge of linear algebra and calculus.
- knowledge of basic robotics concepts.

Course Learning Outcomes:



- Students will be able to install Ubuntu and use terminal commands.
- Students will be able to install and configure ROS 1 on their systems.
- Students will understand the core concepts of ROS: nodes, topics, services, messages, actions and ROS graph.
- Students will be able to write ROS nodes using C++ and Python.
- Students will be able to use ROS tools simulation.
- Students will be able to apply ROS to robotic applications, such as navigation.

Course Syllabus:



introduction to Ubuntu

- what is ubuntu? why use it?
- overview of Linux vs Windows.
- Ubuntu files and folders.
- installing Ubuntu
- setting up a system.
- command-line basics

Course Syllabus:



Introduction to ROS

- What is ROS? Why use it?

Installing ROS (noetic) on your Laptops directly or using the virtual image.

Navigating the ROS Filesystem

Creating and Building a ROS Package

Course Syllabus:



ROS Basics Part 1

- Setting up a ROS workspace. Creating and Building a ROS Package (package.xml and cmake file)
- understanding ROS nodes, topics and messages

Course Syllabus:



ROS Basics Part 2

- Creating Packages and writing nodes with roscpp and rospy / multiple examples

Course Syllabus:



ROS Basics Part 3

- Understanding ROS Services and Parameters
- Using `rqt_console` and `roslaunch`
- Turtlesim Examples

Course Syllabus:



ROS Basics Part 4

- Creating a ROS msg and srv
- Writing and examining a Simple Publisher and Subscriber (C++ and Python)
- Writing and examining a Simple Service and Client (C++ and Python)

Course Syllabus:



Introduction to wheeled mobile Robotics

- Overview of Robotics Concepts: kinematics, dynamics, sensors, actuators.
- Cosine Switch Control examples
- Path planning algorithms.
- Applying control algorithms with turtlesim + examples (cosine switch control)

Course Syllabus:



Introduction to RVIZ + Understanding URDF Model Designing a 3D robot with URDF language

Course Syllabus:



ROS Simulation

- Introduction to Gazebo: a physics-based simulator for ROS.
- Creating simple Gazebo worlds, navigation maps and robots.
- Running basic navigation simulations and Path Planning in `Gazebo`.
- path planning and obstacle avoidance

Project



Students work on individual or group projects to apply ROS knowledge.

Project Examples:

- Autonomous navigation of WMR in a simulated environment.
- Autonomous navigation of Robot arm in a simulated environment, and execute manipulation tasks.
- Implementation of Object detection and tracking.
- Implementation of a simple robot control system.
- Project presentations and demos.

Resources:



- * ROS Wiki: <http://wiki.ros.org>
- * ROS Tutorials: <http://wiki.ros.org/ROS/Tutorials>
- * ROS Discourse: <https://discourse.ros.org>



UBUNTU

Linux *Unix-like*

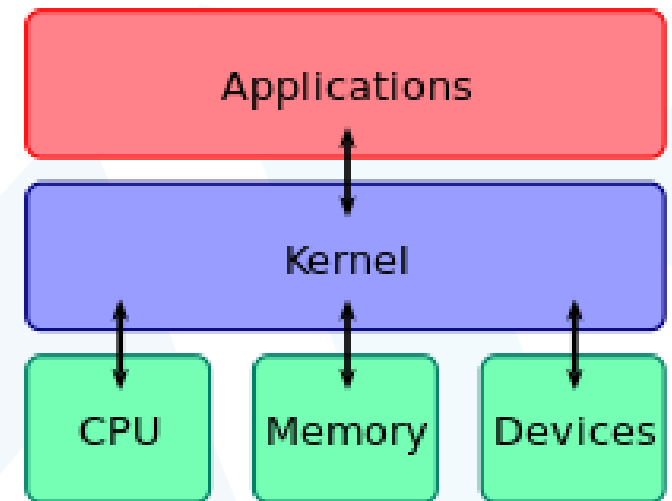


- Not an Operating System (OS), but a Kernel.
- it shares core principles with the original Unix system.
- Linux is open-source software, meaning its source code is freely available for anyone to modify, and distribute.
 - **WINDOWS Kernel type: Hybrid (Windows NT kernel)**
 - **Ubuntu Kernel type : Debian kernel, distribution of the Linux kernel**

- The kernel is a computer program at the core of a computer's operating system and is responsible for translating the command into something that can be understood by the computer.
- It is the interface between hardware (devices which are available in Computer) and Application software like Visual Studio

The main functions of the kernel are:

- memory management
- network management
- device driver
- file management
- process management: preventing and mitigating conflicts between different processes.



Linux VS windows



	Ubuntu (Linux):	Windows
Core Philosophy:	Open-source, community-driven, highly customizable, known for stability and security . It emphasizes user freedom and control .	closed-source, developed by Microsoft , focused on ease of use and compatibility
Cost	Free and open-source , meaning it's completely free to download and use.	Typically requires a paid license for most users.
User Interface:	Uses the GNOME desktop environment by default, known for its clean and modern design with a focus on simplicity.	Windows 11 and 10 introduce a more modern, a familiar and intuitive interface often considered user-friendly for beginners. with features like a Start Menu and taskbar .

Difference between Linux and windows in folder structures



	Ubuntu (Linux):	Windows
The Root (The Starting Point):	The root folder is represented by a single forward slash (`/`) and is the foundation of the entire file system.	The root folder is typically represented by a drive letter (like `C:\`), followed by a hierarchical structure. Each drive is essentially a separate "island" with its own file system.
Case Sensitivity	File and folder names are case sensitive. `MyFile` is different from `myfile`.	File and folder names are generally case-insensitive, `MyFile` and `myfile` are considered the same.
Mount Points	Devices like hard drives, USB drives, and network shares are often mounted into the file system at specific locations (e.g., `/mnt/usb_drive`)	assigned drive letters (like `D:`, `E:`, etc.) and appear as separate drive
Hierarchical Nature:	Both Linux and Windows use a hierarchical file system. This means that folders are organized within folders, and so on, creating a tree-like structure.	

These releases of Ubuntu are available

<https://releases.ubuntu.com/20.04/>

Standard support

LTS Releases

Ubuntu 24.04.1 LTS (Noble Numbat) ›

Ubuntu 22.04.4 LTS (Jammy Jellyfish) ›

Ubuntu 20.04.6 LTS (Focal Fossa) ›

Releases - Ubuntu Wiki



List of releases

[Ubuntu Website release cycle page](#)

Current

Version	Code name	Docs	Release	End of Standard Support	End of Life
Ubuntu 22.10	Kinetic Kudu	Release Notes	October 20, 2022	July 2023	July 2023
Ubuntu 22.04.1 LTS	Jammy Jellyfish	Release Notes	August 11, 2022	April 2027	April 2032
Ubuntu 22.04 LTS	Jammy Jellyfish	Release Notes	April 21, 2022	April 2027	April 2032
Ubuntu 20.04.5 LTS	Focal Fossa	Changes	September 1, 2022	April 2025	April 2030
Ubuntu 20.04.4 LTS	Focal Fossa	Changes	February 24, 2022	April 2025	April 2030
Ubuntu 20.04.3 LTS	Focal Fossa	Changes	August 26, 2021	April 2025	April 2030
Ubuntu 20.04.2 LTS	Focal Fossa	Changes	February 4, 2021	April 2025	April 2030
Ubuntu 20.04.1 LTS	Focal Fossa	Changes	August 6, 2020	April 2025	April 2030
Ubuntu 20.04 LTS	Focal Fossa	Release Notes	April 23, 2020	April 2025	April 2030
Ubuntu 18.04.6 LTS	Bionic Beaver	Changes	September 17, 2021	April 2023	April 2028
Ubuntu 18.04.5 LTS	Bionic Beaver	Changes	August 13, 2020	April 2023	April 2028
Ubuntu 18.04.4 LTS	Bionic Beaver	Changes	February 12, 2020	April 2023	April 2028
Ubuntu 18.04.3 LTS	Bionic Beaver	Changes	August 8, 2019	April 2023	April 2028
Ubuntu 18.04.2 LTS	Bionic Beaver	Changes	February 15, 2019	April 2023	April 2028
Ubuntu 18.04.1 LTS	Bionic Beaver	Changes	July 26, 2018	April 2023	April 2028
Ubuntu 18.04 LTS	Bionic Beaver	Release Notes	April 26, 2018	April 2023	April 2028

- Ubuntu 20.04 was released in 2020 (20), April (04).
- Ubuntu 22.04 was released in 2022 (22), April (04).

•The last ".4" part of 20.04.4 is called the "point release." It means this is the 4 revision of the 20.04.

LONG TIME STABLE

- Both 20.04 and 22.04 are [LTS](#). This means the version 20.04 gets long term support (5 years). Part of the long term support means support for new hardware.



Select an image

Ubuntu is distributed on three types of images described below.

Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 1024MiB of RAM to install from this image.

64-bit PC (AMD64) desktop image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

Server install image

The server install image allows you to install Ubuntu permanently on a computer for use as a server. It will not install a graphical user interface.

64-bit PC (AMD64) server install image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

- The designation "amd64" in Ubuntu (and other Linux distributions) refers to the 64-bit architecture compatible with the x86-64 instruction set, which was initially developed by AMD. This architecture is fully compatible with Intel's 64-bit processors.
- both AMD and Intel implement the same x86-64 standard.
- The terms "armhf" and "amd64" refer to different CPU architectures and instruction sets used in computing devices.
- armhf: ARM hard float (armhf) is an architecture that is based on the ARM (Advanced RISC Machine) architecture. It is specifically designed for 32-bit ARM processors.
- Commonly found in mobile devices, embedded systems, and increasingly in servers and IoT devices due to its power efficiency.

Option to install

Single boot

Dual boot

Virtual Machine



v/s

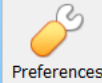


Virtualbox

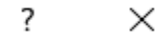


Oracle VM VirtualBox Manager

File Machine Help



Create Virtual Machine



Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: 

Folder:


ISO Image:

Edition:

Type: 

Version:

Skip Unattended Installation

 Detected OS type: Ubuntu (64-bit). This OS type can be installed unattendedly. The install will start after this wizard is closed.

Help

Expert Mode

Back

Next

Cancel


Create Virtual Machine


Unattended Guest OS Install Setup

You can configure the unattended guest OS install by modifying username, password, and hostname. Additionally you can enable guest additions install. For Microsoft Windows guests it is possible to provide a product key.

Username and Password

Username: ✓

Password: 

Repeat Password: 

Additional Options

Product Key:

Hostname: ✓

Domain Name:

Install in Background

Guest Additions

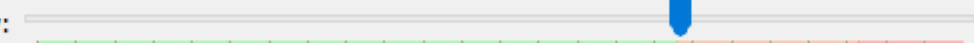
Guest Additions ISO:

Help Back Next Cancel

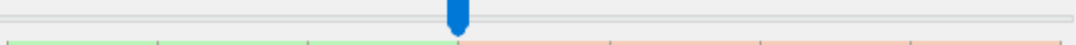
Create Virtual Machine

Hardware

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory:  8539 MB

4 MB 12288 MB

Processors:  4

1 CPU 8 CPUs

Enable EFI (special OSes only)

Help Back Next Cancel

Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

Create a Virtual Hard Disk Now

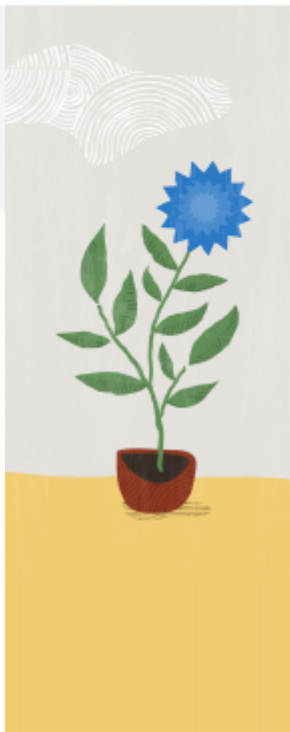
Disk Size: 73.86 GB
4.00 MB 2.00 TB

Pre-allocate Full Size

Use an Existing Virtual Hard Disk File
ubuntu-20.04.5-focal fossa.vdi (Normal, 100.00 GB)

Do Not Add a Virtual Hard Disk

Help Back Next Cancel



Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

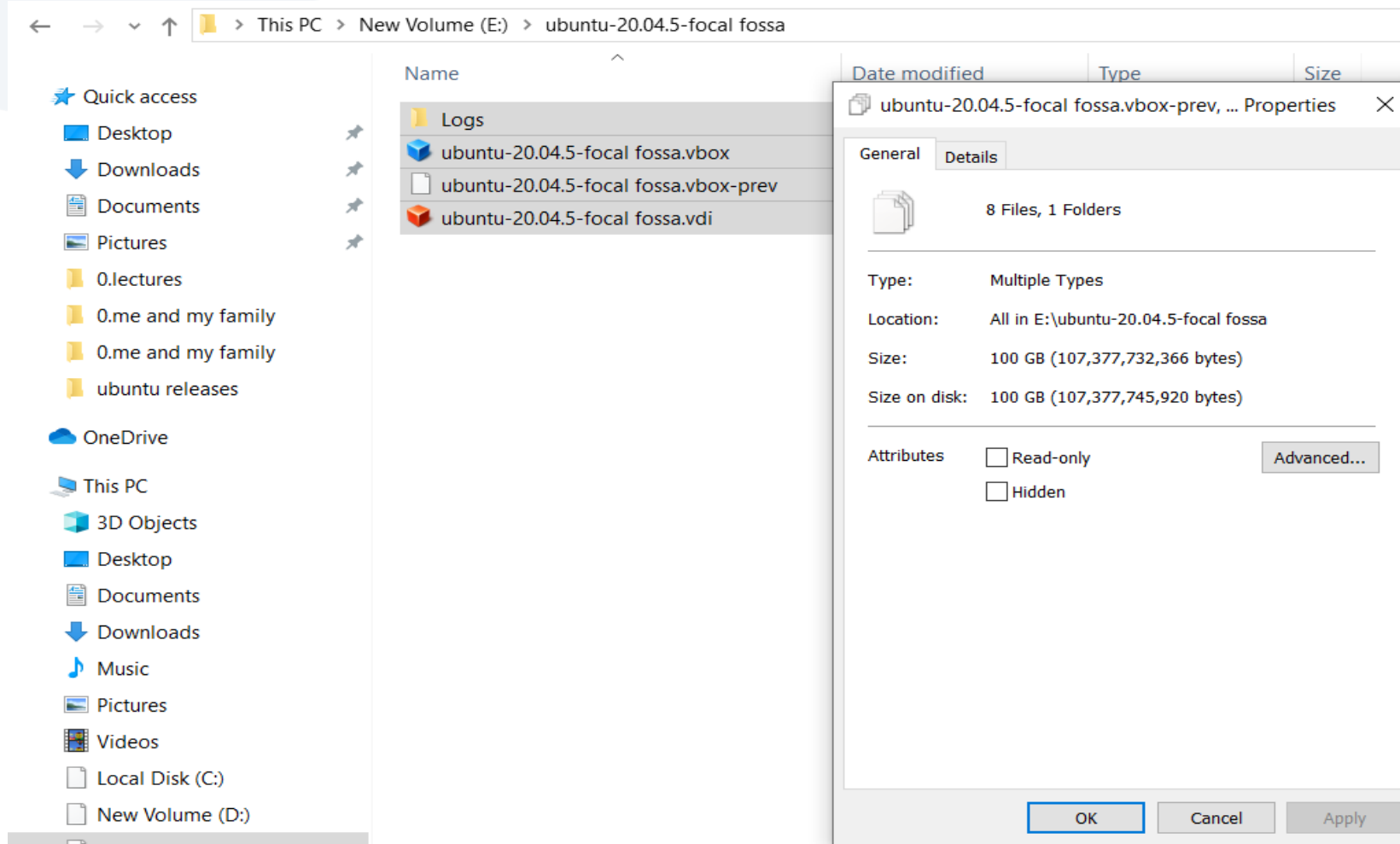
	Machine Name and OS Type
Machine Name	ubuntu 20.04.5
Machine Folder	D:/ubuntu 20.04.5
ISO Image	F:/E/3.softwares/Operating Systems/operating system/ubuntu/ubuntu rel...
Guest OS Type	Ubuntu (64-bit)
Skip Unattended Install	false
	Unattended Install
Username	essa
Product Key	false
Hostname/Domain Name	ubuntu.myguest.virtualbox.org
Install in Background	false
Install Guest Additions	false
	Hardware
Base Memory	8539
Processor(s)	4
EFI Enable	false
	Disk
Disk Size	73.86 GB
Pre-allocate Full Size	true

Help

Back

Finish

Cancel



The screenshot shows a Windows File Explorer window with the address bar set to "This PC > New Volume (E:) > ubuntu-20.04.5-focal fossa". The left sidebar shows the navigation pane with "New Volume (D:)" selected. The main pane displays a list of files and folders:

Name
Logs
ubuntu-20.04.5-focal fossa.vbox
ubuntu-20.04.5-focal fossa.vbox-prev
ubuntu-20.04.5-focal fossa.vdi

Overlaid on the right is the "Properties" dialog box for the selected file, titled "ubuntu-20.04.5-focal fossa.vbox-prev, ... Properties". The "General" tab is active, showing:

- 8 Files, 1 Folders
- Type: Multiple Types
- Location: All in E:\ubuntu-20.04.5-focal fossa
- Size: 100 GB (107,377,732,366 bytes)
- Size on disk: 100 GB (107,377,745,920 bytes)
- Attributes: Read-only, Hidden

Buttons at the bottom of the dialog include "OK", "Cancel", and "Apply".



Tools

64 **ubuntu-20.04.5-focal fossa**
Powered Off



General

Name: ubuntu-20.04.5-focal fossa
Operating System: Ubuntu (64-bit)

System

Base Memory: 10025 MB
Processors: 6
Execution Cap: 54%
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

Display

Video Memory: 60 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Secondary Device 0: [Optical Drive] Empty
Controller: SATA
SATA Port 0: ubuntu-20.04.5-focal fossa.vdi (Normal, 100.00 GB)

Audio

Host Driver: Default
Controller: ICH AC97

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

USB

USB Controller: OHCI, EHCI
Device Filters: 0 (0 active)

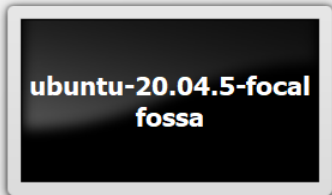
Shared folders

None

Description

None

Preview



How do I reduce the icon size of this sidebar in gnome?



- click on show application then Search appearance or setting
- show application -> Setting -> appearance → icon size
- Gnome Software, the default graphical software manager in Ubuntu.

Make a screenshot



PrtSc – Takes a screenshot of the entire Ubuntu desktop.

ALT + PrtSc – This keyboard shortcut will take a screenshot of the currently selected window.

SHIFT + PrtSc – Takes a screenshot of a selected area.

Make a screencast



You can make a video recording of what is happening on your screen:

1. Press **Ctrl+Alt+Shift+R** to start recording what is on your screen.

A red circle is displayed in the top right corner of the screen when the recording is in progress.

2. Once you have finished, press **Ctrl+Alt+Shift+R** again to stop the recording.

3. The video is automatically saved in your `Videos` folder in your home folder, with a file name that starts with `Screencast` and includes the date and time it was taken.



If you do not have a `Videos` folder, the videos will be saved in your home folder instead.

Main Keyboard shortcuts

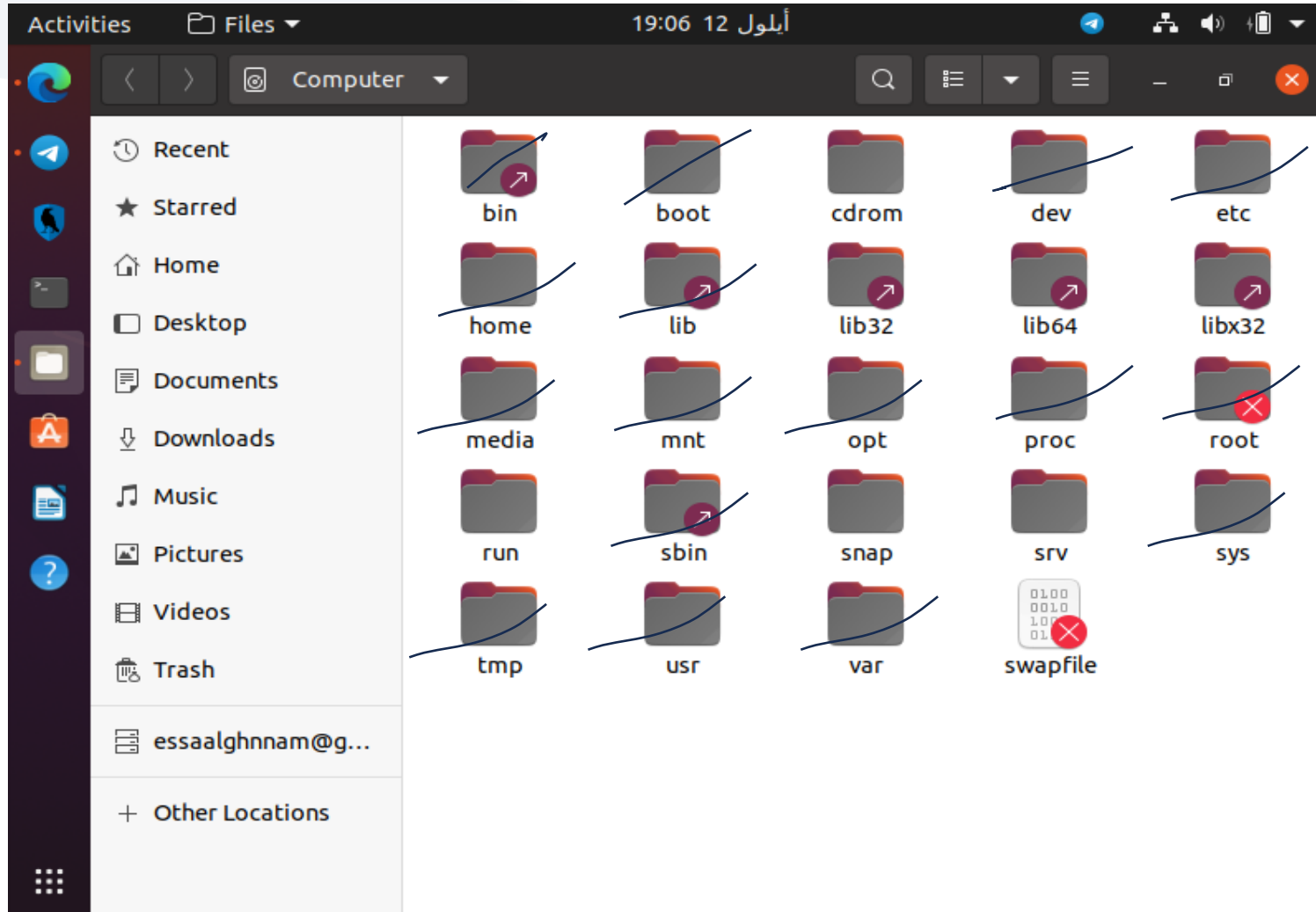
Libreoffice - text editor- folder - files

- ctrl+a
- ctrl+c
- ctrl + x
- ctrl + v

In terminal use:

- ctrl+shift +c
- ctrl+shift + v

Linux File system & Linux folder structure

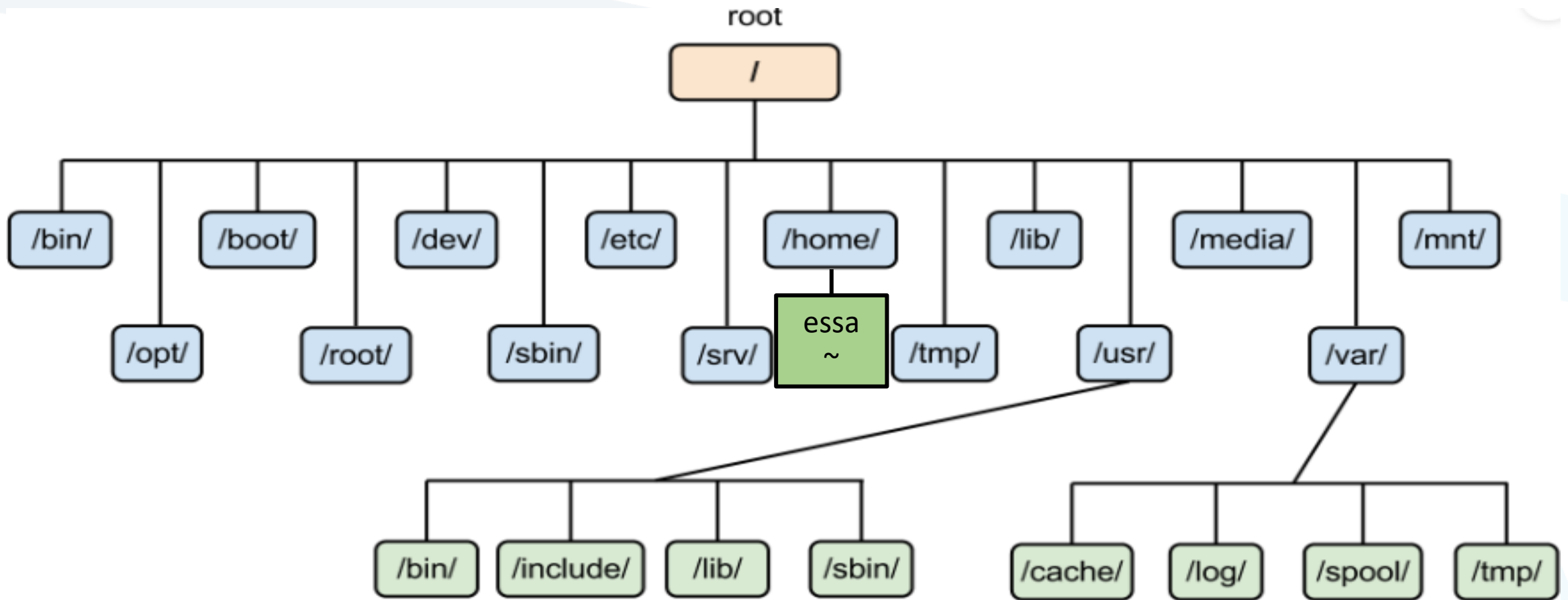


- the `/` (main*absolute* root) is the root of the entire file system. the top-level directory
- `/root` is the home directory of the superuser.

So, why is it named "root" and located under the main root?

- Historical Reasons: Linux inherited this structure from Unix systems.
- **Security: The `/root` folder is isolated from regular user directories. It's designed to be a secure location for the root user's files and configuration.**
- **Administrative Tasks: The root user has full access to the entire system and is typically used for system administration tasks.**

Linux File system & Linux folder structure



Linux File system & Linux folder structure



Linux File system & Linux folder structure



Directory	Details
/bin	is a place for most commonly used terminal commands, like ls, mount, rm, etc.
/sbin	"contains important administrative commands that should generally only be employed by the superuser.
/boot	contains files needed to start up the system, including the Linux kernel, a RAM disk image and bootloader configuration files.
/dev	contains all device files, which are not regular files but instead refer to various hardware devices on the system, including hard drives.
/etc	contains system-global configuration files, which affect the system's behavior for all users.
/home	User's directories for personal folders. home sweet home, this is the place for users' home directories.
/lib	library files required by system and other software or applications
/media	is intended as a mount point for external devices, such as hard drives or removable media (USB, CDs, DVDs).
/mnt	is also a place for mount points, but dedicated specifically to "temporarily mounted" devices, such as network filesystems.

Linux File system & Linux folder structure



Directory	Details
/opt	The "/opt" directory is typically structured with subdirectories for each software package, providing a clear and hierarchical organization of installed software. For example, if a software package called "example" is installed, it would have its own directory under "/opt" as "/opt/example".
/proc	is a virtual filesystem that provides a mechanism for kernel to send information to processes.
/root	All operating systems have a root directory. C:\ path is root directory in Windows system. It is the superuser's home directory (root user), not in /home/ to allow for booting the system even if /home/ is not available.
/tmp	is a place for temporary files used by applications.
/usr	contains the majority of user utilities and applications, and partly replicates the root directory structure, containing for instance, among others, /usr/bin/ and /usr/lib..
/var	is dedicated to variable data, such as logs, databases, websites, and temporary spool (e-mail etc.) files that persist from one boot to the next. A notable directory it contains is /var/log where system log files are kept.
/sys	is a virtual filesystem that can be accessed to set or obtain information about the kernel's view of the system.
/srv	can contain data directories of services such as HTTP (/srv/www/) or FTP.

Linux commands



Command	Description
cat	Utility to concatenate files to standard output
chgrp	Utility to change file group ownership
chmod	Utility to change file access permissions and mode flags
chown	Utility to change file owner
Cp	Utility to copy files and directories
date	Utility to print or set the system data and time
dd	Utility to convert and copy a file
df	Utility to report filesystem disk space usage
dmesg	Utility to print or control the kernel message buffer
echo	Utility to display a line of text
clear	Clear terminal
sudo/su	Execute commands as superuser

Linux commands

false	Utility to do nothing, unsuccessfully
hostname	Utility to show or set the system's host name
kill	Utility to send a kill signal to a process
ln	Utility to make links between files
login	Utility to begin a session on the system
ls	Utility to list directory contents
mkdir	Utility to make directories
mknod	Utility to make block or character special files
more	Utility to page through text
mount	Utility to mount a filesystem
mv	Utility to move/rename files
cd	Change directory. cd is a <u>shell builtin</u> . So it's part of the shell itself, not a separate executable.

cd is a shell builtin. So it's part of the shell itself, not a separate executable.

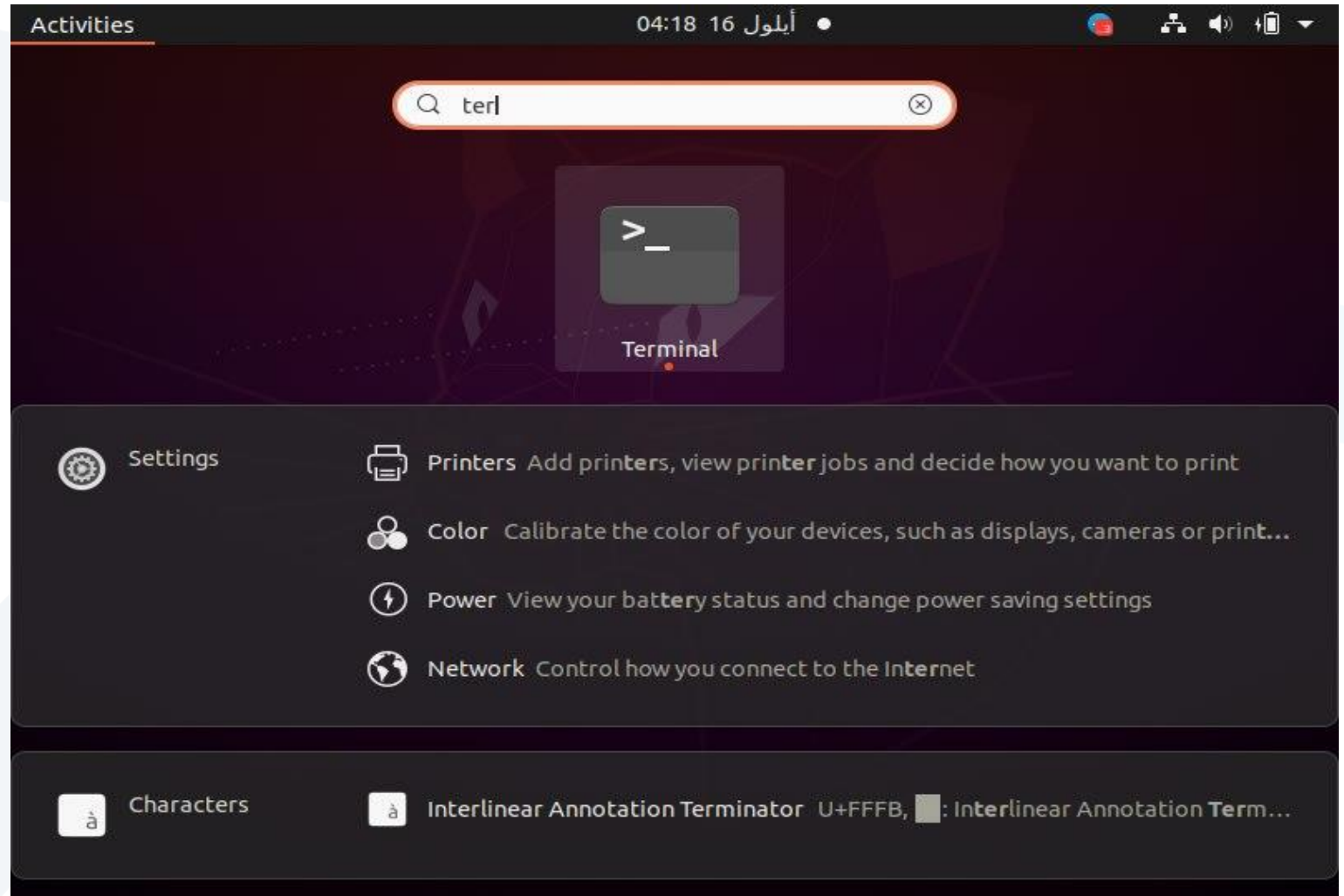
Linux commands



ps	Utility to report or list the running processes status
pwd	Utility to print name of current working directory
rm	Utility to remove files or directories
rmdir	Utility to remove empty directories
sed	The `sed` stream editor
sh	POSIX compatible command shell
stty	Utility to change and print terminal line settings
su	Utility to change user ID
sync	Utility to flush filesystem buffers
true	Utility to do nothing, successfully
umount	Utility to unmount file systems
uname	Utility to print system information (operating system name or unix name)
id	Returns user id

Terminal

Keyboard Shortcut	Use of the keyboard shortcut
Ctrl-Alt-t	Open a new Terminal
Ctrl-d	Close a terminal
Ctrl-c	Cancels the currently running command.
Ctrl-l	Clears the screen just like clear command



Terminal in (home/essa) ~



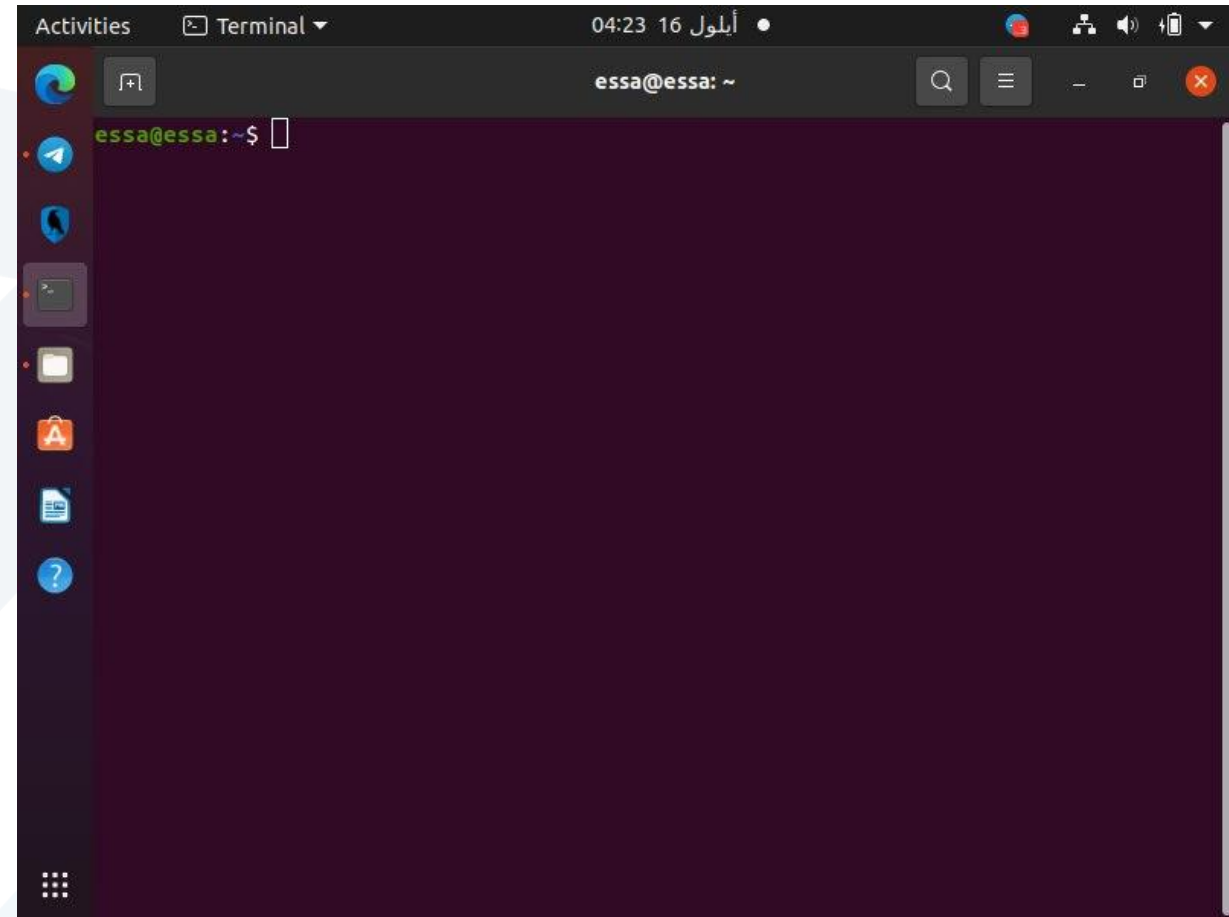
`essa@essa:~\$` represents a Linux or Unix command prompt.

1 `essa`: This is the `username` of the `user` logged in to the system.

2 `essa`: This is the `hostname` of the `machine` the user is logged into. It's likely that the user is logged into a machine named "essa".

3 `~`: This is the `home directory` of the user.

`$`: This indicates that the user is a `regular user, not a superuser (root)`. If the prompt ended with a `#``, it would mean the user has root privileges.



how to open terminal in specific directory



Nautilus is the default file manager in the GNOME desktop environment,

what Nautilus does:

- **File Browsing:** It lets you **navigate your file system**, view **directories** and files, and organize them.
- **File Management:** You can create, rename, delete, move, copy, and paste files and folders.
- **Open Files:** Double-clicking on a file will launch the associated application to open it (like a text editor for a `.txt`` file).
- **Search:** You can easily find files by name or content.
- **Permissions:** You can manage file permissions, controlling who has access to certain files.
- **Tabbed Interface:** You can have multiple file windows open in separate tabs, making it easy to work with multiple directories.

1. Quit nautilus fully

```
nautilus -q
```

2. Make sure python3-nautilus is installed:

```
sudo apt install python3-nautilus
```

Go to any path then right click in any empty space and select "open in terminal"

open terminal in specific directory



```
essa@essa: ~  
essa@essa:~$ ~  
bash: /home/essa: Is a directory  
essa@essa:~$ ~/Desktop  
bash: /home/essa/Desktop: Is a directory  
essa@essa:~$ cd ~  
  
essa@essa:~$ ls ~  
catkin_ws  Documents  Music      Public  Templates  
Desktop    Downloads  Pictures    snap    Videos  
essa@essa:~$
```

ls
show folder contents

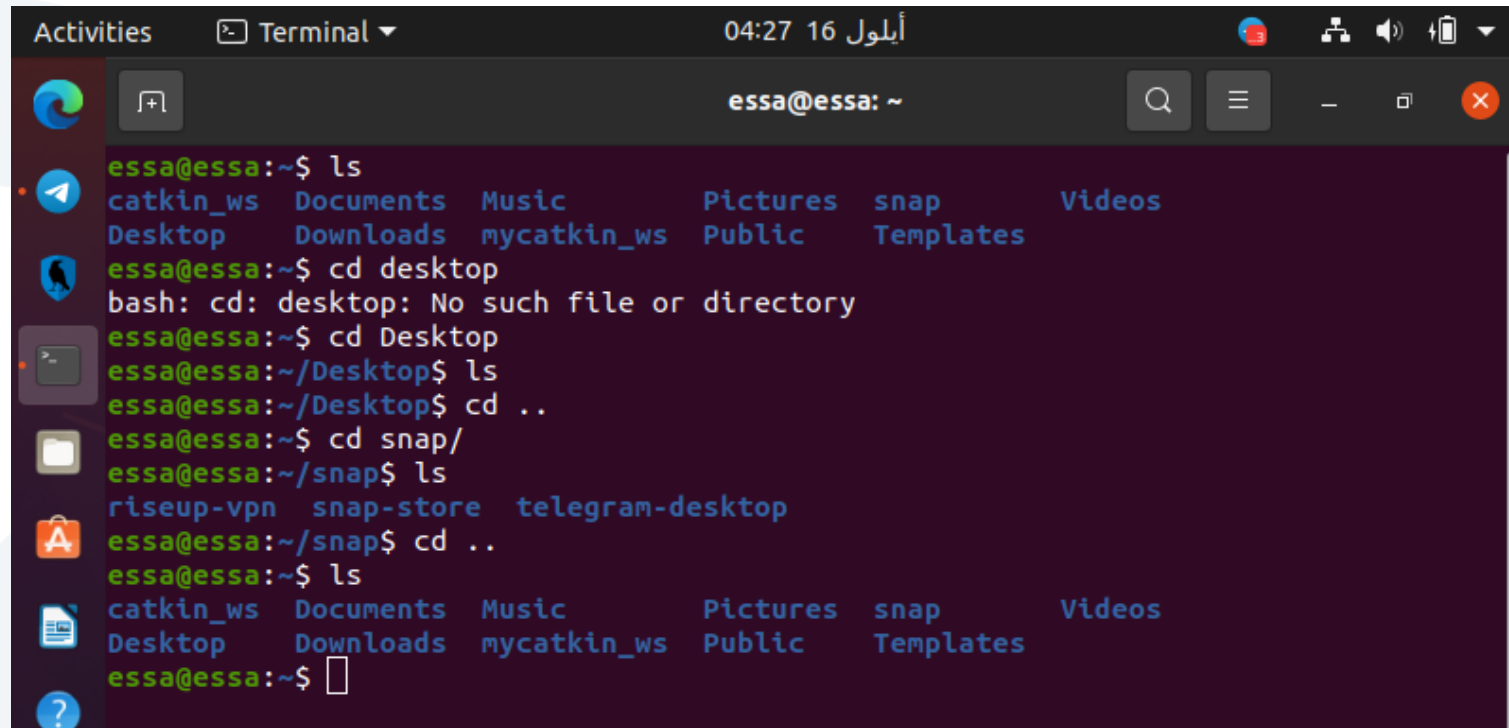
```
Activities Terminal 04:13 16 أيلول  
essa@essa: /  
essa@essa:/$ ls ~  
catkin_ws  Documents  Music      Pictures  snap      Videos  
Desktop    Downloads  mycatkin_ws  Public    Templates  
essa@essa:/$ ls  
bin  dev  lib  lib32  mnt  root  snap  sys  var  
boot  etc  lib32  lost+found  opt  run  srv  tmp  
cdrom  home  lib64  media  proc  sbin  swapfile  usr
```


cd

- ``cd`` is crucial: The ``cd`` (change directory) command is essential for **moving to different locations within the file system.**
- Paths: Use **forward slashes (/)** in Linux paths, not backslashes (\).
- **Tab** completion: Press ``Tab`` to auto-complete file and directory names when typing commands, saving time.
- **Current directory:** If you're already in the **desired directory in the terminal**, you don't need to use ``cd``.

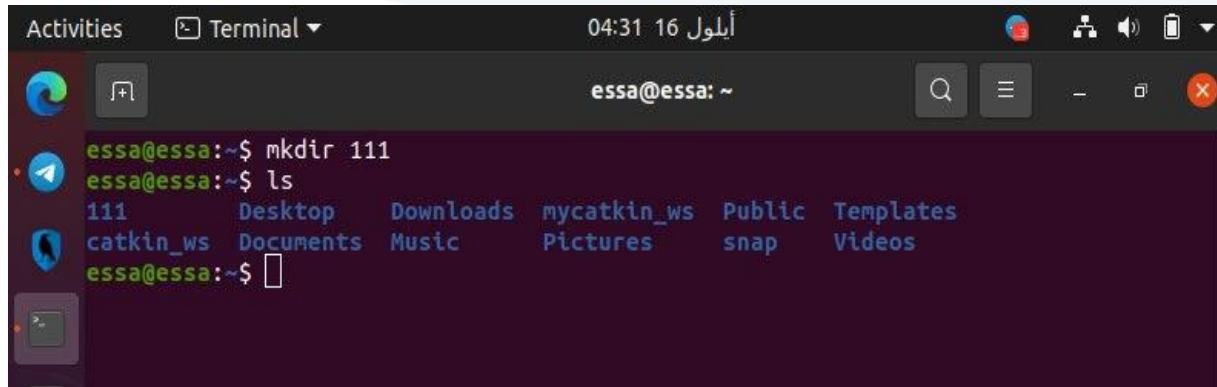
File & Directory Commands

1. To navigate into the root directory, use "cd /"
2. To navigate to your home directory, use "cd" or "cd ~"
3. To navigate up one directory level, use "cd .."
4. To navigate to the previous directory (or back), use "cd -"

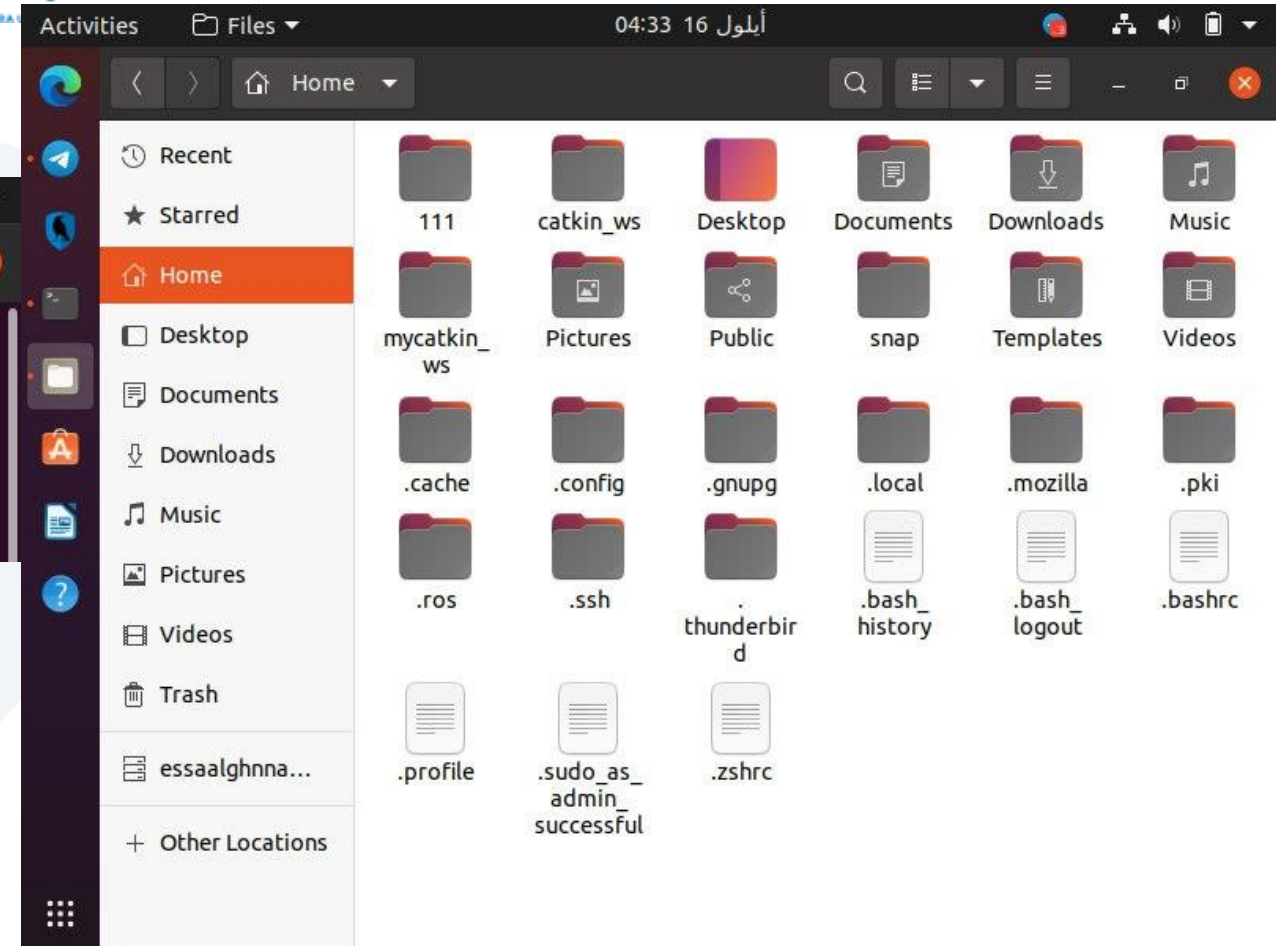


```
Activities Terminal 04:27 16 أيلول  
essa@essa: ~  
essa@essa:~$ ls  
catkin_ws Documents Music Pictures snap Videos  
Desktop Downloads mycatkin_ws Public Templates  
essa@essa:~$ cd desktop  
bash: cd: desktop: No such file or directory  
essa@essa:~$ cd Desktop  
essa@essa:~/Desktop$ ls  
essa@essa:~/Desktop$ cd ..  
essa@essa:~$ cd snap/  
essa@essa:~/snap$ ls  
riseup-vpn snap-store telegram-desktop  
essa@essa:~/snap$ cd ..  
essa@essa:~$ ls  
catkin_ws Documents Music Pictures snap Videos  
Desktop Downloads mycatkin_ws Public Templates  
essa@essa:~$
```

mkdir



```
Activities Terminal 04:31 16 أيلول  
essa@essa: ~  
essa@essa:~$ mkdir 111  
essa@essa:~$ ls  
111 Desktop Downloads mycatkin_ws Public Templates  
catkin_ws Documents Music Pictures snap Videos  
essa@essa:~$
```



1. ` `\$`: This indicates that you're entering the command in your terminal.
2. `mkdir`: This is the command to create a directory (folder).

mkdir -p



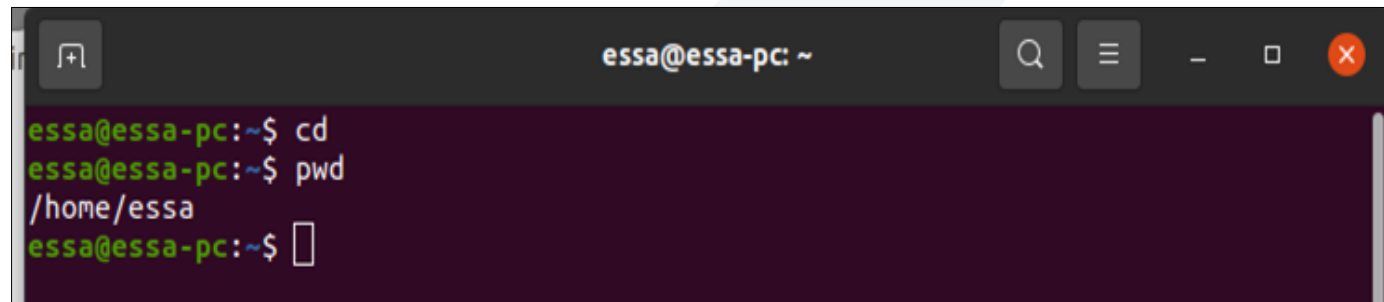
```
Activities Terminal 04:37 16 أيلول  
essa@essa: ~/Desktop  
essa@essa:~$ mkdir 111  
essa@essa:~$ ls  
111 Desktop Downloads mycatkin_ws Public Templates  
catkin_ws Documents Music Pictures snap Videos  
essa@essa:~$ mkdir /222  
mkdir: cannot create directory '/222': Permission denied  
essa@essa:~$ mkdir Desktop/222  
essa@essa:~$ cd ..  
essa@essa:/home$ cd ..  
essa@essa:/$ mkdir 333  
mkdir: cannot create directory '333': Permission denied  
essa@essa:/$ cd ~  
essa@essa:~$ cd Desktop  
essa@essa:~/Desktop$ ls  
222  
essa@essa:~/Desktop$
```

```
$ mkdir -p  
~/mycatkin_ws/src
```

1. ``$``: This indicates that you're entering the command in your terminal.
2. ``mkdir``: This is the command to create a directory (folder).
3. ``-p``: This is an **option** that tells ``mkdir`` **to create any parent directories that are needed**. Let's say you want to create a directory called ``~/mycatkin_ws/src``, but ``~/mycatkin_ws`` doesn't exist. Using ``-p`` will automatically create the parent directories

pwd

- To print the current working directory, we use the **pwd command** in the Linux system. pwd (print working directory)



```
essa@essa-pc: ~  
essa@essa-pc:~$ cd  
essa@essa-pc:~$ pwd  
/home/essa  
essa@essa-pc:~$
```

clear



- Type clear to remove everything in terminal

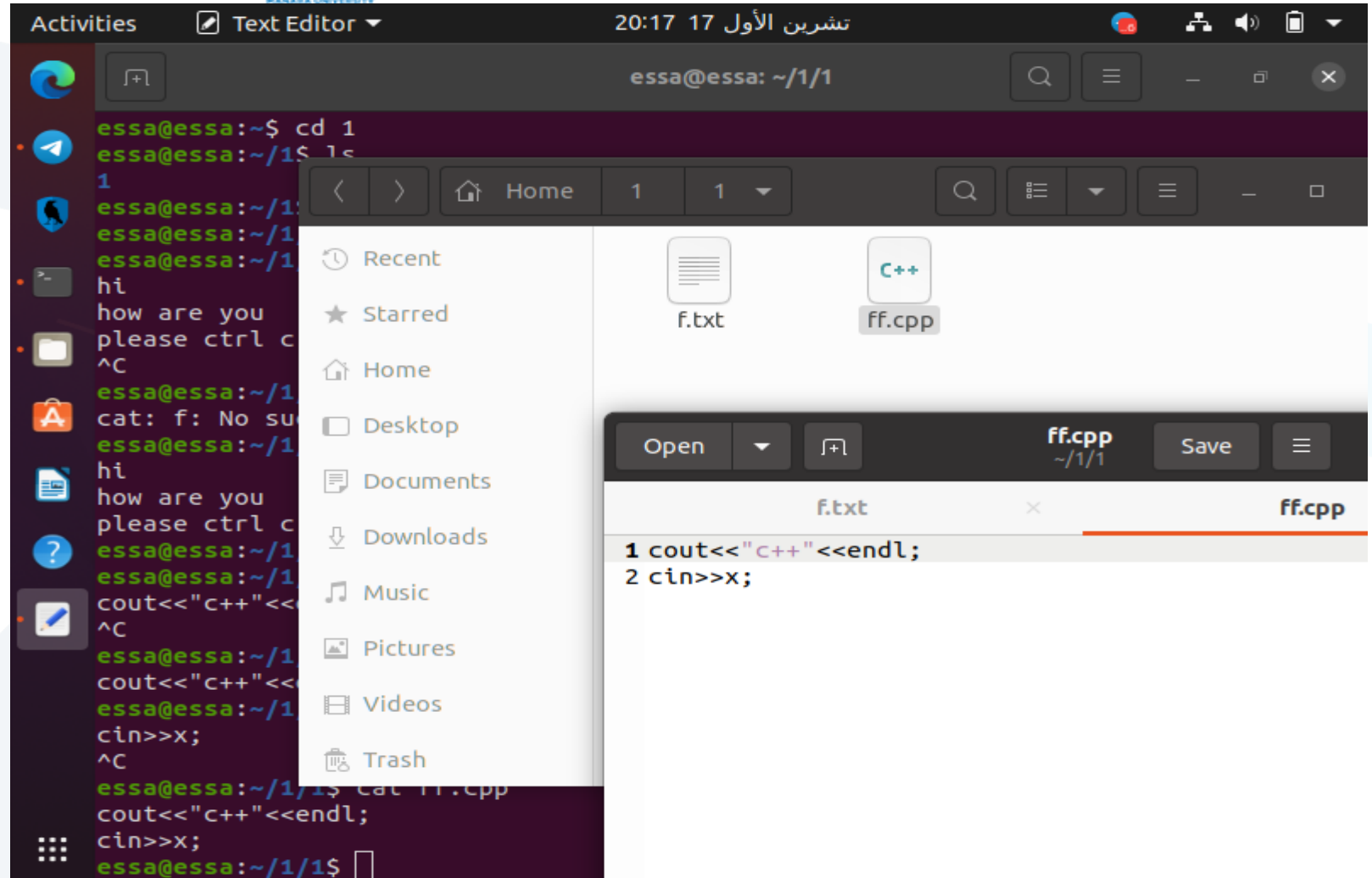
touch and cat

```
Activities Terminal 20:14 17 تشرين الأول  
essa@essa: ~/1/1  
essa@essa:~$ ls  
catkin_ws Documents Music Public Templates  
Desktop Downloads Pictures snap Videos  
essa@essa:~$ mkdir 1/1 2 3  
mkdir: cannot create directory '1/1': No such file or directory  
essa@essa:~$ mkdir 1/1  
mkdir: cannot create directory '1/1': No such file or directory  
essa@essa:~$ mkdir -p 1/1 2 3  
essa@essa:~$ ls  
1 3 Desktop Downloads Pictures snap Videos  
2 catkin_ws Documents Music Public Templates  
essa@essa:~$ cd 1  
essa@essa:~/1$ ls  
1  
essa@essa:~/1$ cd 1  
essa@essa:~/1/1$ touch f.txt  
essa@essa:~/1/1$ cat >> f.txt  
hi  
how are you  
please ctrl c to cancel  
^C  
essa@essa:~/1/1$ cat f  
cat: f: No such file or directory  
essa@essa:~/1/1$ cat f.txt  
hi  
how are you  
please ctrl c to cancel  
essa@essa:~/1/1$
```

```
Activities Text Editor 20:14 17 تشرين الأول  
Home 1 1  
Recent  
Starred  
Home  
Desktop  
Documents  
Downloads  
Music  
Pictures  
Videos  
Trash  
+ Other Locations  
f.txt  
Open Save  
1 hi  
2 how are you  
3 please ctrl c to cancel  
Plain Text Tab Width: 8 Ln 1, Col 1
```


touch and cat

```
essa@essa:~/1/1$ touch ff.cpp
essa@essa:~/1/1$ cat >> ff.cpp
cout<<"c++"<<endl;
^C
essa@essa:~/1/1$ cat ff.cpp
cout<<"c++"<<endl;
essa@essa:~/1/1$ cat >> ff.cpp
cin>>x;
^C
essa@essa:~/1/1$ cat ff.cpp
cout<<"c++"<<endl;
cin>>x;
essa@essa:~/1/1$
```



The screenshot shows a Linux desktop environment. At the top, the system tray displays the time as 20:17 on 17 October (تشرين الأول 17) and the user is logged in as 'essa@essa' in the directory '~/1/1'. The terminal window shows the following commands and output:

```
essa@essa:~$ cd 1
essa@essa:~/1$ ls
1
essa@essa:~/1$ touch ff.cpp
essa@essa:~/1$ cat >> ff.cpp
cout<<"c++"<<endl;
^C
essa@essa:~/1$ cat ff.cpp
cout<<"c++"<<endl;
essa@essa:~/1$ cat >> ff.cpp
cin>>x;
^C
essa@essa:~/1$ cat ff.cpp
cout<<"c++"<<endl;
cin>>x;
essa@essa:~/1$
```

The file manager window shows the contents of the directory ~/1/1, which includes two files: f.txt and ff.cpp. The ff.cpp file is highlighted. The file manager also shows a sidebar with navigation options: Recent, Starred, Home, Desktop, Documents, Downloads, Music, Pictures, Videos, and Trash. The ff.cpp file is open in a text editor window, showing the following code:

```
1 cout<<"c++"<<endl;
2 cin>>x;
```


rm and rmdir



Syntax : rm [OPTION]... FILE...

rm command is used to remove objects such as files, directories, symbolic links and so on from the file system like UNIX. once you delete the files then you are not able to recover the contents of files and directories.

```
$ ls
a.txt b.txt c.txt d.txt e.txt
Removing one file at a time
$ rm a.txt
$ ls
b.txt c.txt d.txt e.txt
Removing more than one file at
a time
$ rm b.txt c.txt
$ ls
d.txt e.txt
```

```
$ rm -i d.txt
rm: remove regular empty file 'd.txt'? y
$ ls
e.txt
$ rm e.txt
rm: remove write-protected regular empty
file 'e.txt'? n
$ ls
e.txt
$ rm -f e.txt
$ ls
```

rm and rmdir

```
Activities Terminal 20:18 17 تشرين الأول  
essa@essa: ~  
essa@essa:~/1/1$ rm f.txt  
essa@essa:~/1/1$ rm ff.cpp  
essa@essa:~/1/1$ ls  
essa@essa:~/1/1$ cd ..  
essa@essa:~/1$ cd ..  
essa@essa:~$ ls  
1 3 Desktop Downloads Pictures snap Videos  
2 catkin_ws Documents Music Public Templates  
essa@essa:~$ rmdir 1  
rmdir: failed to remove '1': Directory not empty  
essa@essa:~$ rmdir 2  
essa@essa:~$ rmdir -p 1  
rmdir: failed to remove '1': Directory not empty  
essa@essa:~$ ls  
1 catkin_ws Documents Music Public Templates  
3 Desktop Downloads Pictures snap Videos  
essa@essa:~$
```

rm and rmdir

Syntax: rmdir [option]... [directory]...

```
essa@essa:~$ ls
1  catkin_ws  Documents  Music      Public  Templates
3  Desktop    Downloads  Pictures    snap    Videos
essa@essa:~$ cd 1
?  essa@essa:~/1$ ls
1
essa@essa:~/1$ cd ..
```

```
essa@essa:~$ rm -r 1
essa@essa:~$ ls
3  Desktop    Downloads  Pictures    snap    Videos
catkin_ws  Documents  Music      Public      Templates
essa@essa:~$
```

- The rmdir or rm -d command is for removing empty directories.
- while the rm -r command deletes non-empty directories.

```
$ ls
```

```
A
```

```
$ cd A
```

```
$ ls
```

```
B C
```

```
$ ls B
```

```
a.txt b.txt
```

```
$ ls C
```

```
c.txt d.txt
```

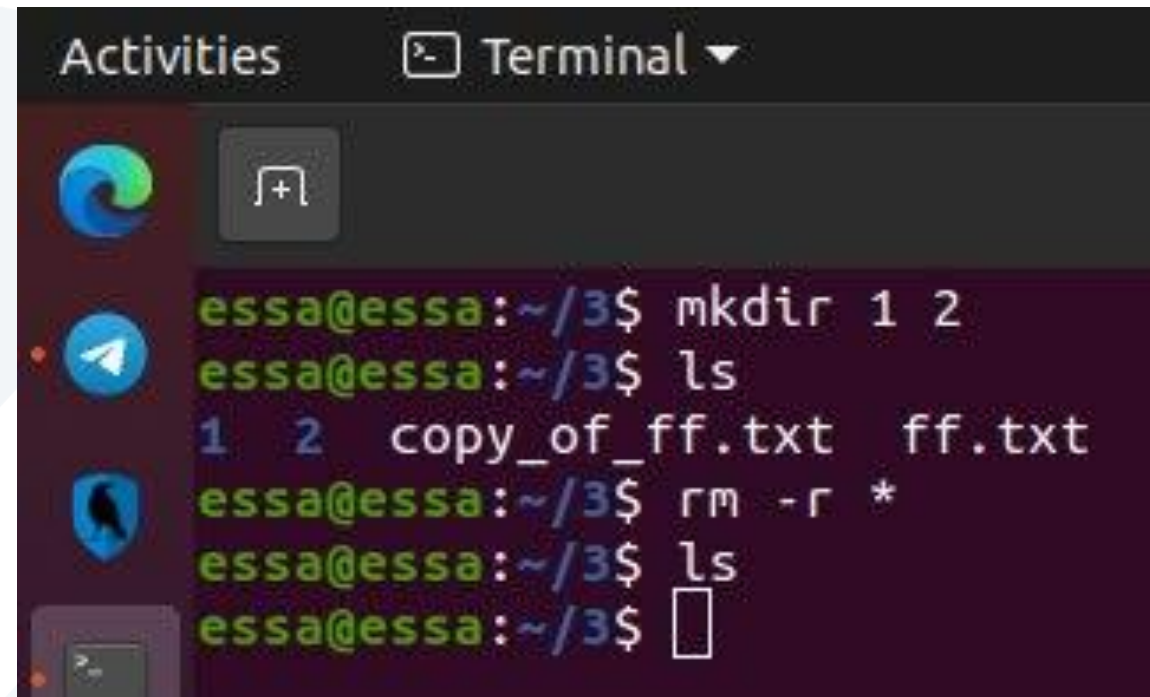
```
$ rm *
```

```
rm: cannot remove 'B': Is a directory
```

```
rm: cannot remove 'C': Is a directory
```

```
$ rm -r *
```

```
$ ls
```



```
Activities Terminal ▾  
essa@essa:~/3$ mkdir 1 2  
essa@essa:~/3$ ls  
1 2 copy_of_ff.txt ff.txt  
essa@essa:~/3$ rm -r *  
essa@essa:~/3$ ls  
essa@essa:~/3$
```

echo

```
essa@essa:~$ ls
3          Desktop  Downloads  Pictures  snap      Videos
catkin_ws Documents Music      Public   Templates
essa@essa:~$ touch 3/ff.txt
essa@essa:~$ cat >> 3/ff.txt
ffff
hi
how are you
^C
essa@essa:~$ echo 3/ff.txt
3/ff.txt
essa@essa:~$ cat 3/ff.txt
ffff
hi
how are you
essa@essa:~$ cat >> 3/ff.txt
new world
^C
essa@essa:~$ cat 3/ff.txt
ffff
hi
how are you
new world
essa@essa:~$
```

cp and mv

Rename a file `mv [source_file_name(s)] [Destination_file_name]`

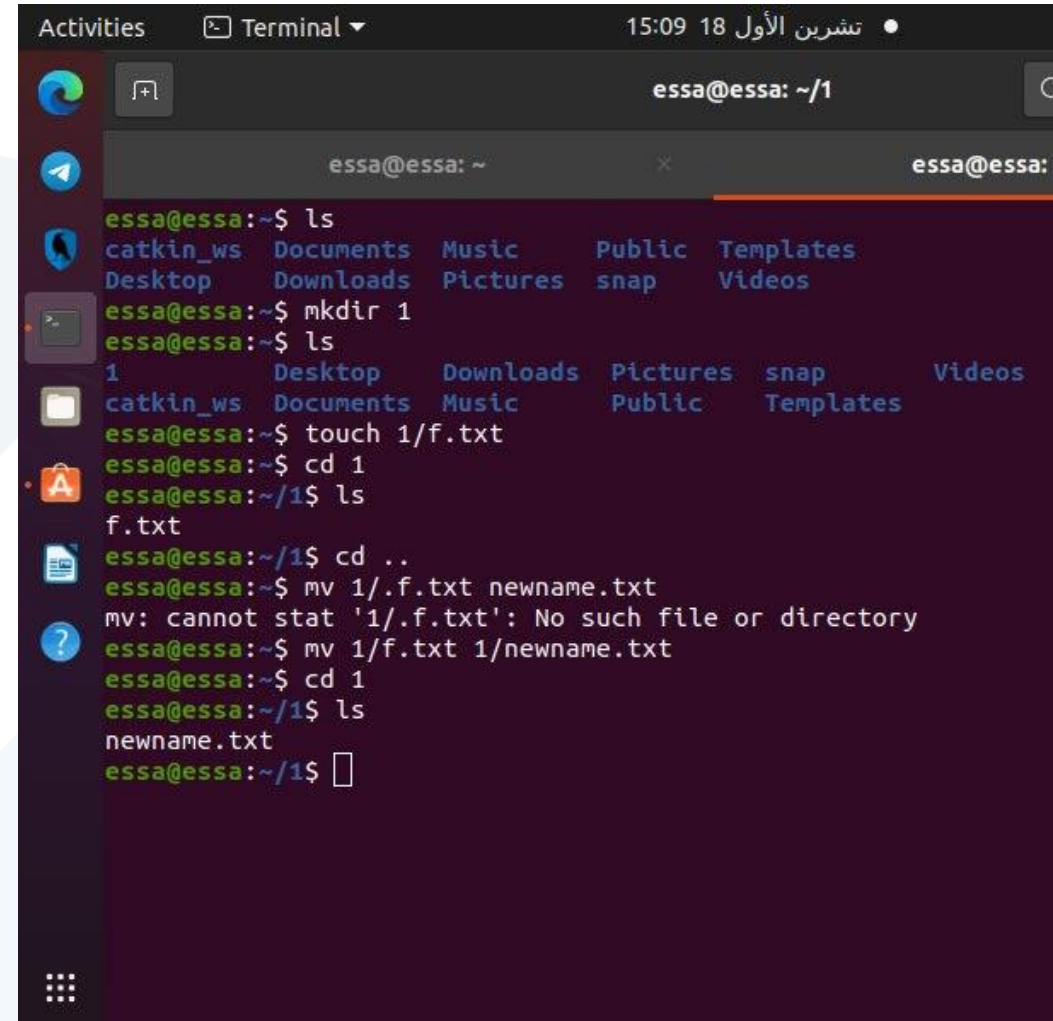
`mv [source_file_name(s)] [Destination_path]`

`mv [source_file_name_1] [source_file_name_2]`
`[source_file_name_.....] [Destination_path]`

`mv -i [source_file/directory_name(s)]`
`[Destination_file/directory_name/path]`

The “-i” option makes the “mv” command ask for confirmation

The -f option overrides this minor protection and overwrites the destination file forcefully and deletes the source file.



```
Activities Terminal 15:09 18 تشرين الأول
essa@essa: ~/1
essa@essa: ~$ ls
catkin_ws Documents Music Public Templates
Desktop Downloads Pictures snap Videos
essa@essa:~$ mkdir 1
essa@essa:~$ ls
1 Desktop Downloads Pictures snap Videos
catkin_ws Documents Music Public Templates
essa@essa:~$ touch 1/f.txt
essa@essa:~$ cd 1
essa@essa:~/1$ ls
f.txt
essa@essa:~/1$ cd ..
essa@essa:~$ mv 1/.f.txt newname.txt
mv: cannot stat '1/.f.txt': No such file or directory
essa@essa:~$ mv 1/f.txt 1/newname.txt
essa@essa:~$ cd 1
essa@essa:~/1$ ls
newname.txt
essa@essa:~/1$
```

cp and mv

```
essa@essa:~$ cat 3/ff.txt
ffff
? hi
how are you
new world
essa@essa:~$ cp 3/ff.txt 3/copy_of_ff.txt
essa@essa:~$ cd 3
essa@essa:~/3$ ls
copy_of_ff.txt  ff.txt
essa@essa:~/3$ cat copy_of_ff.txt
ffff
hi
how are you
new world
essa@essa:~/3$
```

cp --version

cp --help

rm --version

rm --help

How to know Ubuntu release



- Use the `lsb_release -a` command.
- Open file: the `/etc/lsb-release` or `/etc/os-release`
- `lsb_release -sc` command

```
essa@essa:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:   Ubuntu 20.04.6 LTS
Release:      20.04
Codename:     focal
essa@essa:~$ lsb_release -sc
focal
essa@essa:~$
```

```
essa@essa-VB:~$ lsb_release -a
No LSB modules are available.
Distributor ID:   Ubuntu
Description:      Ubuntu 20.04.6 LTS
Release: 20.04
Codename:         focal
```

sudo apt update & sudo apt upgrade



Sudo

```
sudo apt update  
sudo apt upgrade  
apt list --upgradable
```

make sure your Debian package index is up-to-date:

- 1. Updates the package lists on your system. It downloads information about the available software packages from the repositories.
- 2. Installs all updates for software packages you have already installed.
- 3. `sudo apt upgrade`: Run this command after `apt update` to install available software updates.

- **sudo apt-get update:** This is the **older command** that originated from the **older `apt-get` package manager**.
- **sudo apt update:** This is the newer command that was introduced with the `apt` package manager, which is the recommended way to manage packages in modern Ubuntu versions.

Install `snapd` Package and The `gnome-software-plugin-snap` Package



Sudo

```
sudo apt install snapd gnome-software-plugin-snap
```

1. `sudo`: This is the "superuser do" command, granting you administrative privileges. You need this to install software that requires system-wide access.
 2. `apt`: This is the **Advanced Packaging Tool**, Ubuntu's primary package manager. It's responsible for installing, updating, and removing software on your system.
 3. `install`: This tells `apt` to install new software packages.
- * Install the `snapd` package, which enables the **Snap package manager**.
 - * Install the `gnome-software-plugin-snap` package, which adds **Snap support to the Gnome Software application**.



RiseupVPN

you can also install via the command line:

```
sudo snap install --classic riseup-vpn
```

Or

```
sudo apt install riseup-vpn
```

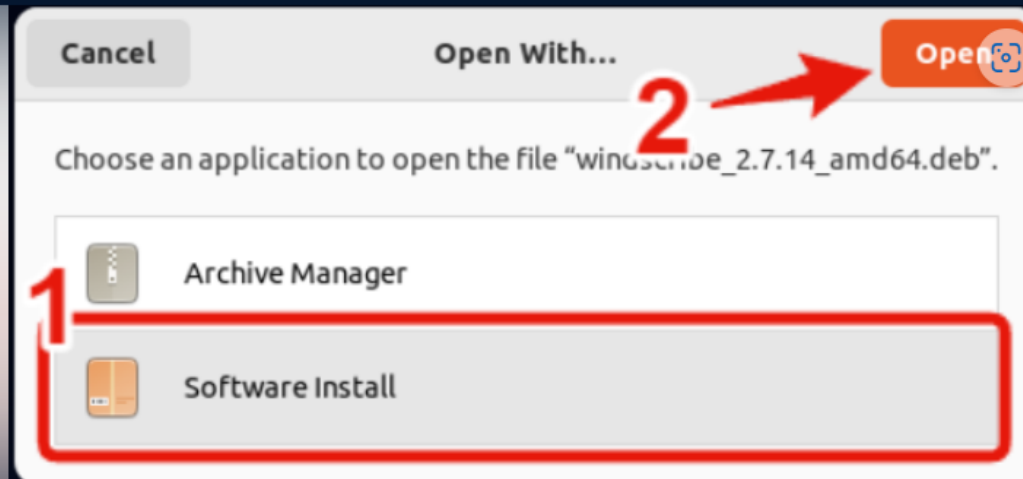


Windscribe for Linux

Debian/Ubuntu (AMD64)

Download 

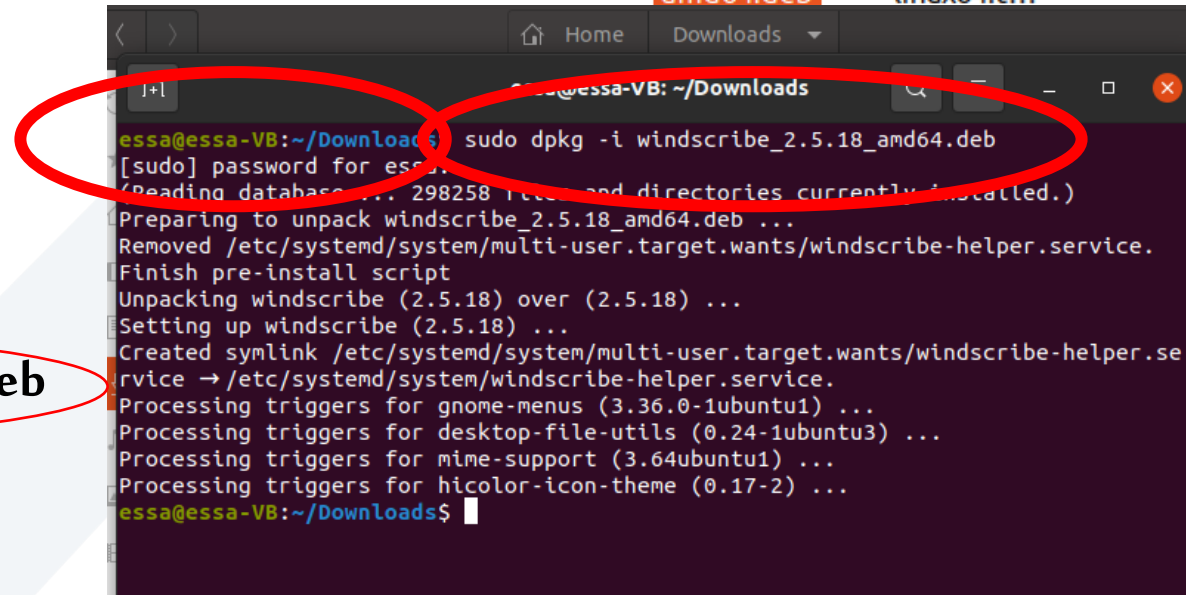
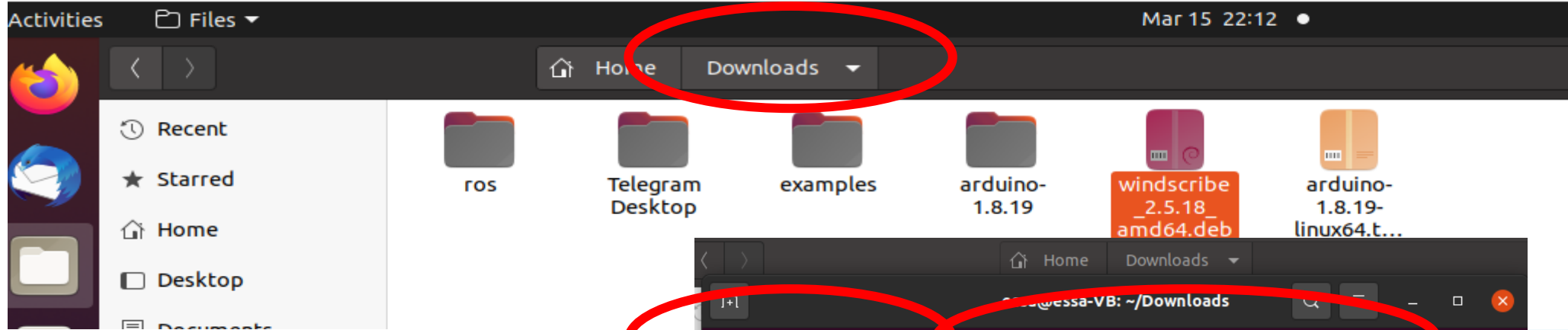
Select the **Software Install** option and then press **Open**



or

ubuntu 20.04 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help



```
essa@essa-VB: ~/Downloads$ sudo dpkg -i windscribe_2.5.18_amd64.deb
[sudo] password for essa:
(Reading database ... 298258 files and directories currently installed.)
Preparing to unpack windscribe_2.5.18_amd64.deb ...
Removed /etc/systemd/system/multi-user.target.wants/windscribe-helper.service.
Finish pre-install script
Unpacking windscribe (2.5.18) over (2.5.18) ...
Setting up windscribe (2.5.18) ...
Created symlink /etc/systemd/system/multi-user.target.wants/windscribe-helper.service → /etc/systemd/system/windscribe-helper.service.
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
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) ...
essa@essa-VB: ~/Downloads$
```

How to Install - Debian Package

`sudo dpkg -i windscribe_2.5.18_amd64.deb`



Install telegram

```
sudo snap install telegram-desktop
```

Install edge:

- `curl https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > microsoft.gpg`
- `sudo install -o root -g root -m 644 microsoft.gpg /etc/apt/trusted.gpg.d/`
- `sudo sh -c 'echo "deb [arch=amd64] https://packages.microsoft.com/repos/edge stable main" > /etc/apt/sources.list.d/microsoft-edge-dev.list'`
- `sudo rm microsoft.gpg`
- `sudo apt update && sudo apt install microsoft-edge-stable`

Install python IDLE



```
sudo apt update or sudo apt-get update  
sudo apt install idle3
```

```
python --version or python -V
```

Open a list of your currently installed programs



- Type `dpkg -l` into Terminal

Remove package

```
sudo apt-get remove nameOfProgram
```

- Or

```
sudo apt-get purge nameOfProgram
```

That will remove the program. It doesn't always remove the dependencies that were installed to make the program work. To clean up those package you can run this after:

```
sudo apt-get autoremove
```

- Example: `sudo apt-get remove windscribe`

Or From Open Ubuntu Software

Creating a New User

1. Creating a New User

- Open a Terminal: Press `Ctrl+Alt+T` to open a terminal window.
- Run the `useradd` Command: Use the `useradd` command to create a new user. For example:

```
sudo useradd new_user
```

- ❖ `sudo`: This command requires administrator privileges.
- ❖ `useradd`: The command to add a new user.
- ❖ `new_user`: Replace this with the desired username.

2. Setting a Password for the New User

- Use the `passwd` Command: After creating a new user, you need to set a password for them:

```
sudo passwd new_user
```

- You'll be prompted to enter and confirm the new password.

Creating a New User



3. Granting User Privileges (Optional)

- Use the `usermod` Command: By default, new users are created with standard permissions. If you need to grant administrative privileges (root access) to the new user, use the `usermod` command:

```
sudo usermod -aG sudo new_user
```

- This command adds the new user to the `sudo` group`, which allows them to use `sudo` to run commands with elevated privileges.

Creating a New User



Managing Users

1-Viewing Users: Use the `whoami` command to check which user you're logged in as, or use `users` to see all currently logged in users.

2- Removing Users: Use the `userdel` command to remove a user:

```
sudo userdel -r user_name
```

3- Changing Usernames: Use the `usermod` command to change a username:

```
sudo usermod -l new_username old_username
```

- **sudo killall apt apt-get**
- **Sudo poweroff**
- **Sudo reboot**

شكرا لحسن الاصغاء