A Guide To Linux for University of Toronto Undergraduate Physics Students

Yigit Ozcelik 2T3

November 20, 2023

Contents

Introduction

What Is This?

This document is a guide to what Linux is and how to use it for the purposes of academic writing and scientific computing. It is aimed primarily at undergraduate physics students at the University of Toronto, where the author obtained his degree in physics in 2023.

This document will introduce you to Linux, make a case for why you might want to use it, describe how to install and use a Linux operating system on your computer (or in a virtual machine), introduce you to some fundamental Linux concepts (the terminal, package managers, etc.), and then focus on setting up important tools for a physicist-in-training at UofT (such as a Python coding environment and a LaTeX setup).

What Is Linux?

Linux is an **operating system-** an operating system ('OS') is responsible for managing the interactions between hardware and the other software running on the computer. Linux serves as an alternative to Mac OS or Windows.

However, there are several fundamental differences between Linux and Windows/Mac OS. Here are a few:

Linux has 'distributions'. Essentially, Linux is very modular; you can modify even the most fundamental parts of Linux. This has led to different versions of Linux being created by different projects to serve different needs and priorities. These different versions are called **Linux distributions,** or **Linux distros** for short. Different distros come with different preinstalled programs, different looks ('desktop environments'), different update schedules and so on. Some distros are Ubuntu, Fedora, Arch Linux and Gentoo.

Linux is 'free' open source software. Open source software refers to software whose source code is publicly available - you can use the source code to examine or modify the program, including compiling it yourself. 'Free' software in this context (also called 'libre software') refers to software that you can use for any purpose, including commercial purposes. Software that is both free and open source is called FOSS (Free and Open Source Software) for short.

Crucially, the **Linux kernel,** the core of a Linux OS, is FOSS. This means that anyone (including you!) can build an OS around the Linux kernel, for any purpose, for free - and you can access and edit the kernel's source code to do so. This renders Linux far more versatile than both Windows and Mac OS (which are very much not FOSS).

Linux focuses on using the terminal, rather than graphical applications. A terminal is a program that lets you execute commands on your computer using text prompts. These commands can both execute text-based programs - meant to be used within a terminal (such as `cd`) - and graphical programs (e.g. the `libreoffice` command runs the graphical LibreOffice program).

A **graphical program** here refers to a program that creates a window on the screen - this is likely the case for almost all programs you interact with regularly. In contrast, a **text program** is accessed through a terminal.

A great part of the power of a Linux system comes from how terminal commands are deeply integrated into the operating system. You can use text commands in the terminal to manipulate files, run text-based programs, update programs (see below) and to do a million other things that might be cumbersome or downright unfeasible to do in a graphical program.

Linux uses 'package managers'. The vast majority of programs installed on a Linux machine are managed (tracked, (re)installed, updated and removed) by a program called a 'package manager'. A 'package' in this context means 'a bunch of files, possibly including programs'. You can think of a package manager as an app store of sorts, where all files and programs you need are bundled into various packages.

A Linux distribution almost always comes with a package manager, such as `apt` or `pacman`. The package manager used by a distribution is very important - it influences how and where you install packages, for example. Different package managers also take different types of commands.

Why Use Linux?

There are a large variety of reasons to use Linux. Many servers and various devices around the world use the Linux kernel as a base for their operating systems. Linux operating systems are reliable, quite secure and give you full control of your system. Linux is very customizable, so you can use it in a lot of different scenarios (servers, lightweight installations, and so on).

Furthermore, Linux's focus on FOSS means that most Linux software is free to use, in sharp contrast to most Windows and Mac OS apps. The vast majority of Linux distributions are completely free of charge to download and use.

For our purposes (scientific computing), learning Linux and command line tools will let you effectively manipulate data, easily install Python and similar software, interface with computer clusters during research, and do many other things without worrying about, say, a Windows update interrupting your workflow.

As you read on through the rest of this guide, I hope you will find your own reasons for switching to Linux as well.

Why NOT use Linux?

There are some factors to consider before committing to installing Linux on your computer (or a virtual machine) - unfortunately, as with any large-scale software, Linux operating systems do have their drawbacks.

Most notably, a lot of proprietary software (Adobe suite, Microsoft Office, most video games) is not available officially for Linux. There are open source alternatives to these, of course (GIMP replaces Photoshop, Libreoffice replaces Word, Steam Proton lets you play most video games), but this can be an obstacle to using Linux as your daily operating system.

Another thing to consider is that Linux is a paradigm shift from more common operating systems. What I mean by the phrase ‘paradigm shift’ is that getting started with Linux is a learning and adaptation process. Especially early on, you will need to do a lot of research and troubleshooting to get things working the way you want them to. (The flipside to this is that there usually *is* a way to get things working exactly how you want!)

1 Getting Started: Installing Ubuntu

1.1 Ubuntu?

Ubuntu is generally regarded as the most popular Linux distribution, especially for beginners. There is a colossal community of Ubuntu users out there, full of people who have had the same questions as you. This is the most important asset a beginner-friendly distribution could have. As such, I will be using the most recent version of Ubuntu (at the time of writing) in this guide. That version is **Ubuntu 23.10** - the version number stands for October 2023.

1.2 Where should I install Ubuntu?

You can install Ubuntu in a few ways. Most importantly, you can either directly install Ubuntu onto your hard drive (a 'bare metal' install) or you can install Ubuntu into a **virtual machine-** a program that acts like a computer within your computer.

The latter option is much safer and tidier in terms of learning the ropes. I would only recommend installing Ubuntu to a hard drive if you are using a spare computer (e.g. an old laptop) - if you choose to do so, **back up all of the data on your computer onto a separate hard drive!** It is possible to mess up to such a degree that completely reinstalling Ubuntu is easier than fixing the issue manually - reinstalling Ubuntu might even be the only reliable solution! Reinstalling Ubuntu on a virtual machine is much safer and easier than reinstalling Ubuntu to a hard drive.

We will be installing Ubuntu into a virtual machine in this guide.

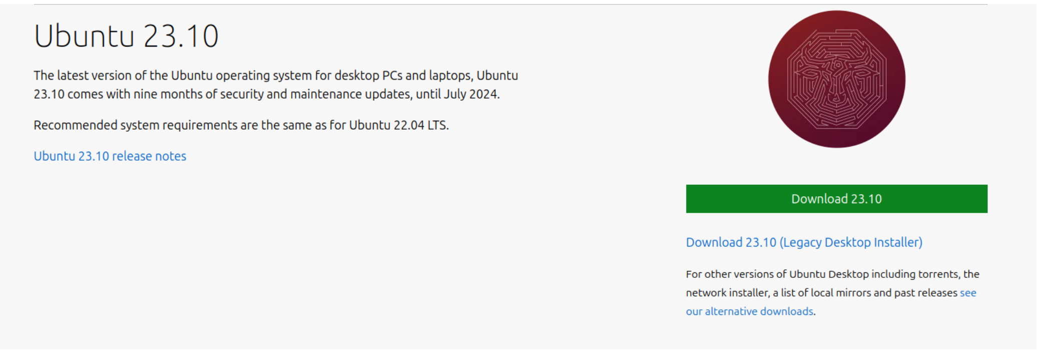
1.3 VirtualBox

The virtual machine software I have decided to use is **VirtualBox,** which you can download from `https://www.virtualbox.org/wiki/Downloads` . Just go to this URL and download the appropriate file for your 'host' system (i.e. the operating system of your actual computer), and then follow the instructions in the setup.

Once you have installed VirtualBox, you can proceed to installing Ubuntu into VirtualBox.

1.4 Downloading Ubuntu 23.10 and Installing into VirtualBox

Navigate to `https://ubuntu.com/download/desktop` and scroll down. Under the "Ubuntu 23.10" section, click the button labeled "Download 23.10". This will download the file `ubuntu-23.10.1-desktop-amd64.iso` [1] .



Ubuntu 23.10

The latest version of the Ubuntu operating system for desktop PCs and laptops, Ubuntu 23.10 comes with nine months of security and maintenance updates, until July 2024.

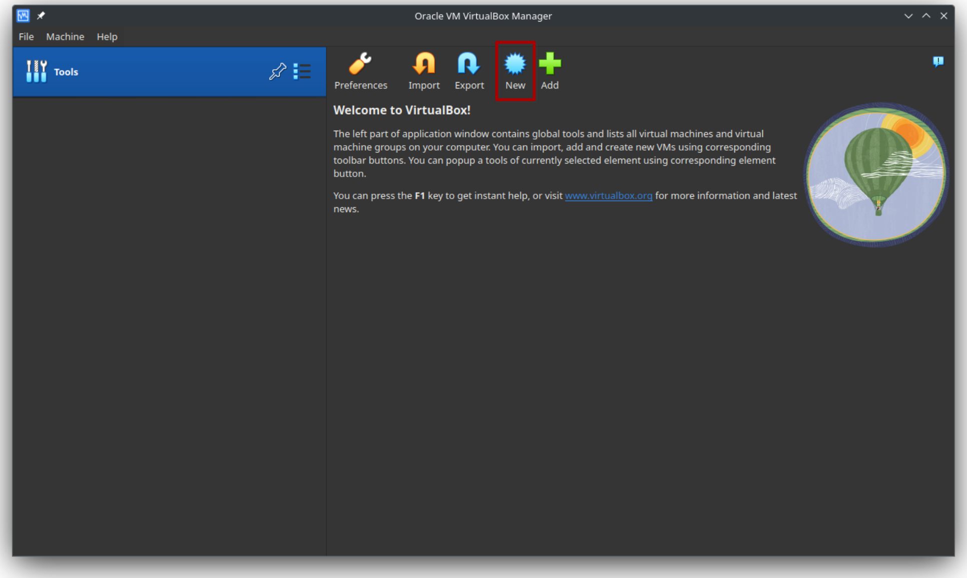
Recommended system requirements are the same as for Ubuntu 22.04 LTS.

Ubuntu 23.10 release notes

Download 23.10 (Legacy Desktop Installer)

For other versions of Ubuntu Desktop including torrents, the network installer, a list of local mirrors and past releases see our alternative downloads.

Once the `.iso` file is downloaded, open VirtualBox.



Welcome to VirtualBox!

The left part of application window contains global tools and lists all virtual machines and virtual machine groups on your computer. You can import, add and create new VMs using corresponding toolbar buttons. You can popup a tools of currently selected element using corresponding element button.

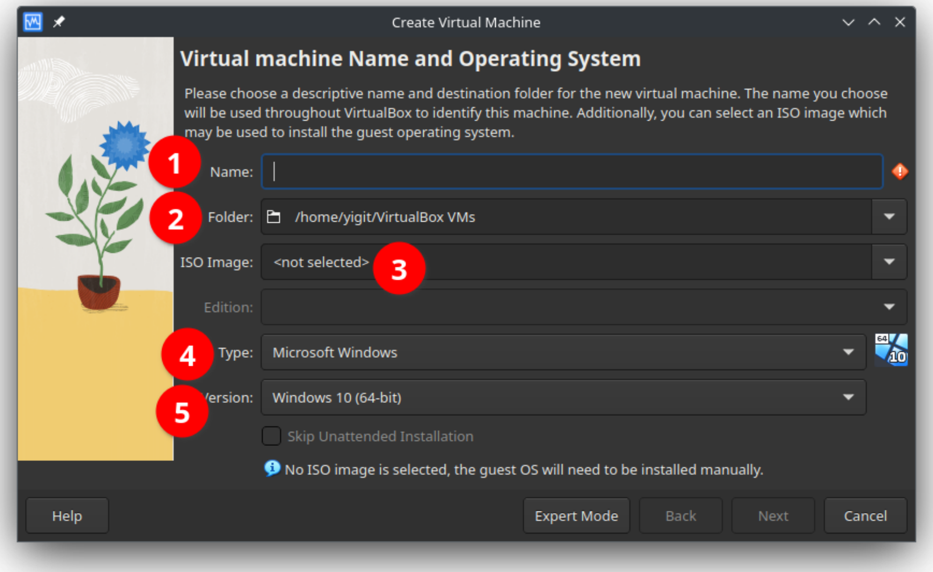
You can press the **F1** key to get instant help, or visit www.virtualbox.org for more information and latest news.

[1] Here, 'amd64' refers to the system architecture of your computer. Carry on with the instructions even if your file name is slightly different.

You can create a virtual machine in a few ways in VirtualBox - we will be creating one from the `.iso` file we downloaded earlier. To do so, click "New" on the icon toolbar (see figure above). A dialog box will appear (see below).

Sidebar: We chose the "New" option because `.iso` file we downloaded is not the Ubuntu operating system itself - it is an **installation medium,** which we will use to install Ubuntu proper into a virtual machine[2].

Back to installation. When you click "New", the dialog box that opens will have several fields we need to fill.



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: /home/yigit/VirtualBox VMs

ISO Image: <not selected>

Edition:

Type: Microsoft Windows

Version: Windows 10 (64-bit)

□ Skip Unattended Installation

No ISO image is selected, the guest OS will need to be installed manually.

1. **Name:** Enter a name of your choice for the virtual machine.

2. **Folder:** Choose the folder where the virtual machine will be created. I recommend creating a dedicated folder for this purpose.

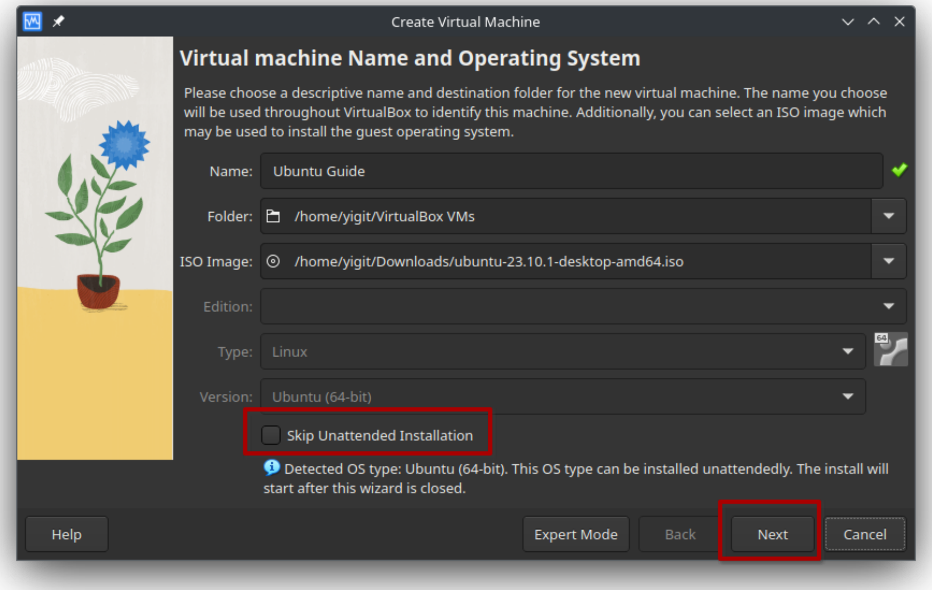
[2] If you are installing Ubuntu directly to a hard drive, you will need to flash this file to a USB drive and boot your computer from that USB drive. The instructions to do this vary slightly between different computer brands and models.

3. **ISO Image:** You need to point VirtualBox to the installation medium. Click the drop-down arrow on the right-hand side of this field, select "Other..." and then find the `.iso` file you downloaded earlier in the file picker.

4. **Type:** This will be automatically set to "Linux" when you select the ISO.

5. **Version:** This will be automatically set to "Ubuntu (64-bit)" when you select the ISO.

Check the "Skip Unattended Installation" box. We will install the "guest OS" (Ubuntu) manually[3].



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: Ubuntu Guide

Folder: /home/yigit/VirtualBox VMs

ISO Image: /home/yigit/Downloads/ubuntu-23.10.1-desktop-amd64.iso

Edition:

Type: Linux

Version: Ubuntu (64-bit)

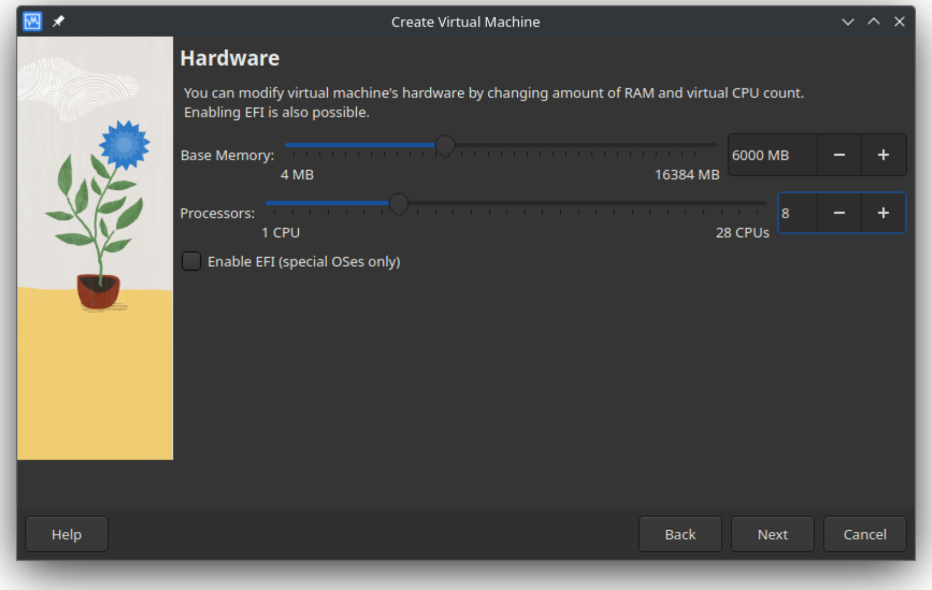
□ Skip Unattended Installation

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

When you are done, click "Next".

[3] The unattended installation feature lets you input several extra settings in VirtualBox and then skip Ubuntu's own install dialogue. Although I have opted not to use this feature in this guide (as it is relatively new), you can try it if you wish.

VirtualBox allocates a certain amount of your computer's memory and CPU resources to the virtual machine when it is running.



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: 6000 MB (4 MB – 16384 MB)

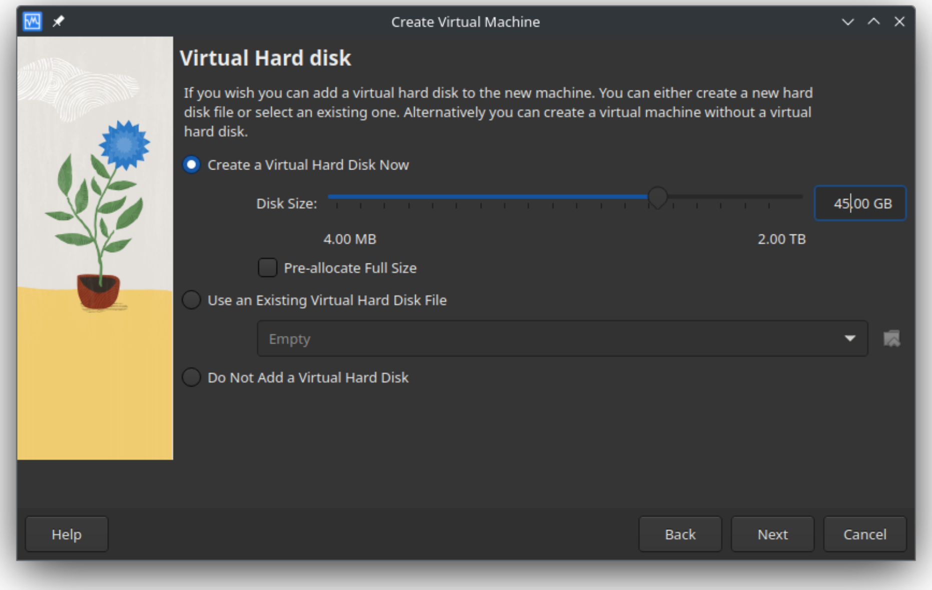
Processors: 8 (1 CPU – 28 CPUs)

□ Enable EFI (special OSes only)

The appropriate amount to choose depends on your hardware; here, I have chosen to allocate 6GB of memory and 8 CPU cores to this VM.

When done, click "Next".

Almost done! Next, we will set the maximum amount of space our Ubuntu install can take up. As Ubuntu takes up much less space than Windows or Mac OS, a small amount of space is fine.



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: 45.00 GB

4.00 MB 2.00 TB

□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

Empty

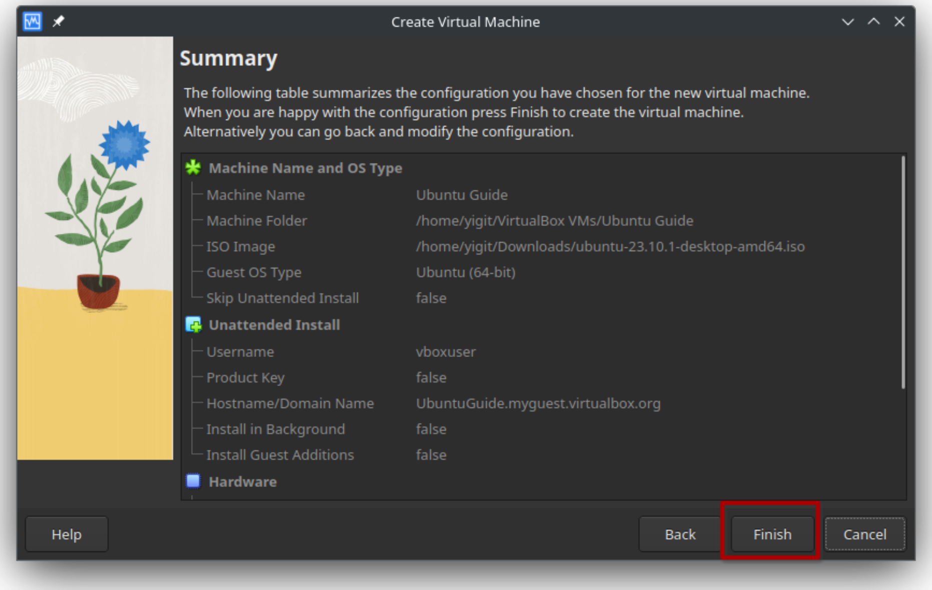
○ Do Not Add a Virtual Hard Disk

Make sure that the “Create a Virtual Hard Disk Now” option is selected and set your desired amount of space. Here, I have chosen to allocate 45 GB of space.

Leave the “Pre-allocate Full Size” box unchecked. This way, ‘empty’ space in the Ubuntu install will not take up space on your actual hard drive.

Click “Next”.

View the summary, ensure all details are correct, and click “Finish” to create the virtual machine[4].



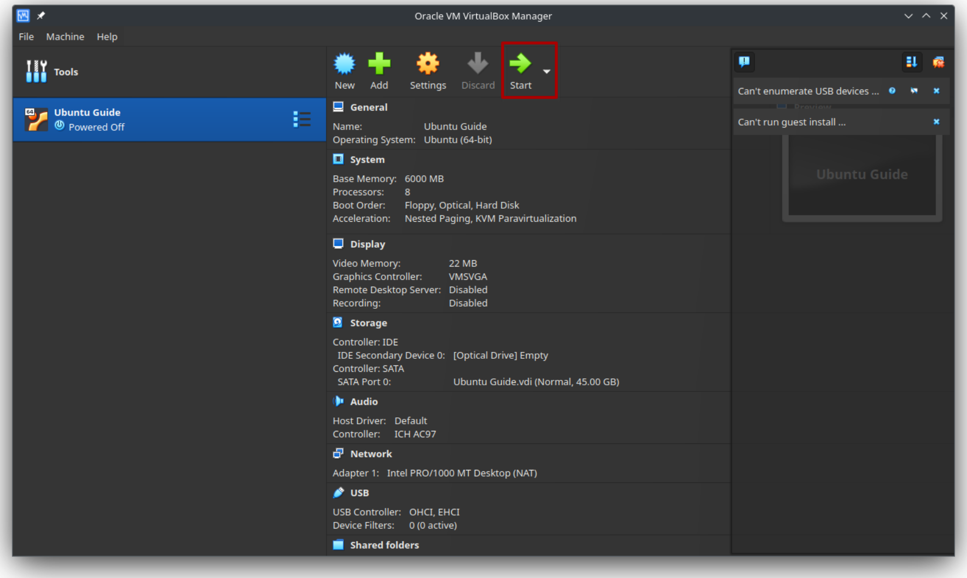
Summary

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 Guide
Machine Folder	/home/yigit/VirtualBox VMs/Ubuntu Guide
ISO Image	/home/yigit/Downloads/ubuntu-23.10.1-desktop-amd64.iso
Guest OS Type	Ubuntu (64-bit)
Skip Unattended Install	false
Unattended Install	
Username	vboxuser
Product Key	false
Hostname/Domain Name	UbuntuGuide.myguest.virtualbox.org
Install in Background	false
Install Guest Additions	false
Hardware	

[4] The correct configuration options might differ slightly from this screenshot; namely, you need to set Skip Unattended Install to ‘true’.

If the installation went well, there is now a new entry in the VirtualBox main menu:



```
General
Name:               Ubuntu Guide
Operating System:   Ubuntu (64-bit)

System
Base Memory:   6000 MB
Processors:    8
Boot Order:    Floppy, Optical, Hard Disk
Acceleration:  Nested Paging, KVM Paravirtualization

Display
Video Memory:            22 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 Guide.vdi (Normal, 45.00 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
```

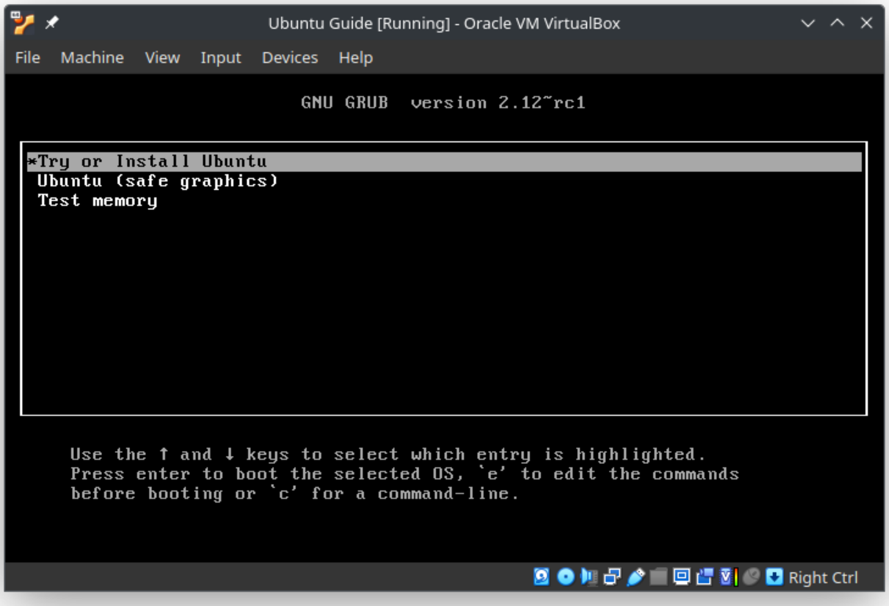
Can't enumerate USB devices ...

Can't run guest install ...

Click "Start" while the new VM is selected to start the Ubuntu install process[5].

[5] If you see a message saying that the virtual machine failed to boot, point the "DVD:" field in the pop-up box to the Ubuntu installation ISO. You can do this by clicking the drop-down arrow on the right-hand side of the field and choosing the ISO file from the menu. If the ISO file is not listed, click "Other..." and then navigate to the ISO file in the file picker dialog.

On the next screen (titled "GNU GRUB"), make sure the "Try or install Ubuntu" option is highlighted and press ENTER on your keyboard.



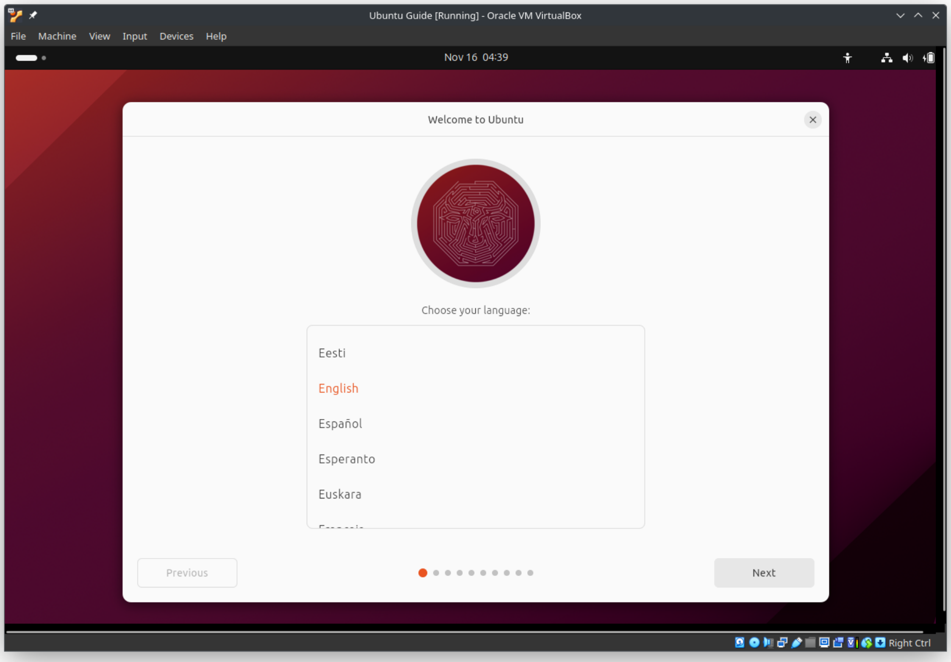
```
                    GNU GRUB  version 2.12~rc1

*Try or Install Ubuntu
 Ubuntu (safe graphics)
 Test memory

     Use the ↑ and ↓ keys to select which entry is highlighted.
     Press enter to boot the selected OS, `e' to edit the commands
     before booting or `c' for a command-line.
```

A loading screen will appear. It will take a bit of time for the installation medium to boot up.

If all goes well, the Ubuntu installation image will boot and you will see the Ubuntu setup dialog.



In the following screens, do the following:

1. Select a language of your choosing. In this guide, we will select English.
2. Select "Install Ubuntu". Since we are installing Ubuntu into a virtual machine, we do not need to worry about data loss[6] and thus can simply install Ubuntu without needing to try it out first.
3. Select your keyboard layout as desired.
4. Select "Use wired connection".
5. You can skip the prompt to update the Ubuntu installer.

[6] If you are installing Ubuntu directly onto a hard drive, the "Try Ubuntu" option will let you try out Ubuntu within the installation medium, but you will lose all changes you made when you reboot.

6. On the “Applications and updates” screen, select “Default installation” (you will figure out how to install additional apps soon enough). I would recommend checking the two boxes in the “Other options” section.

7. On the following screen, select “Erase disk and install Ubuntu” - this will wipe the (currently empty) virtual machine drive and install Ubuntu into it[7].

8. On the “Ready to install” screen, check that the text under “Devices” is `VBOX HARDDISK sda` and then click “Install”.

9. Select your timezone.

10. Fill the fields in the next step however you wish. The first field corresponds to the name you will see on the lock screen and so on. I have decided to use `Ubuntu Guide`.

11. Decide on your computer’s name. “Your computer’s name” is also called the **hostname,** and it is the name that your computer will have on the local network (e.g. over Bluetooth). Your hostname should have no spaces. I chose `UbuntuGuide`.

12. On the same screen, decide on a username - for convenience, this should be a short and all-lowercase name. You will use this username repeatedly in commands. I chose `ubuntuguide`.

13. Afterward, on the same screen set a password, confirm the password, and ensure that “Require password to log in” is enabled. Move on to the next screen.

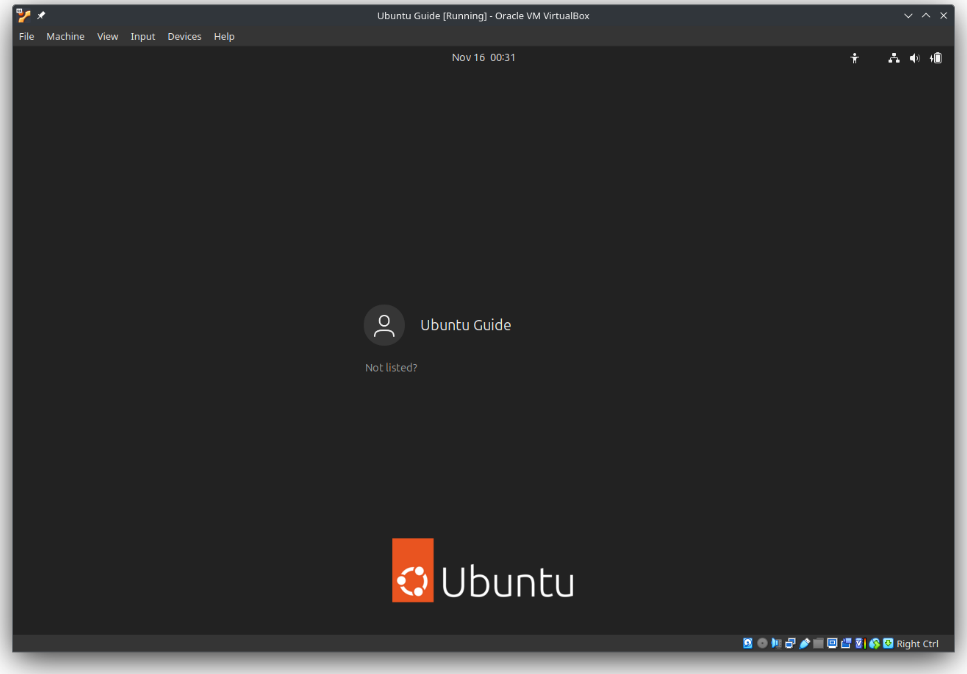
14. Choose a theme on the next screen.

15. Ubuntu will now be installed into the virtual machine you have created. Once the confirmation screen appears, click “Restart now” to reboot the VM into Ubuntu proper.

16. Remove the install medium (from VirtualBox menu, navigate to Devices - Optical Drives, uncheck the Ubuntu ISO file) when prompted, and press ENTER.

[7] The “Manual partitioning” option allows you to customize your install, install Ubuntu alongside Mac OS or Windows, and so on. However, there are a lot of configuration options involved, so skip this step unless you are trying to install Ubuntu on bare metal.

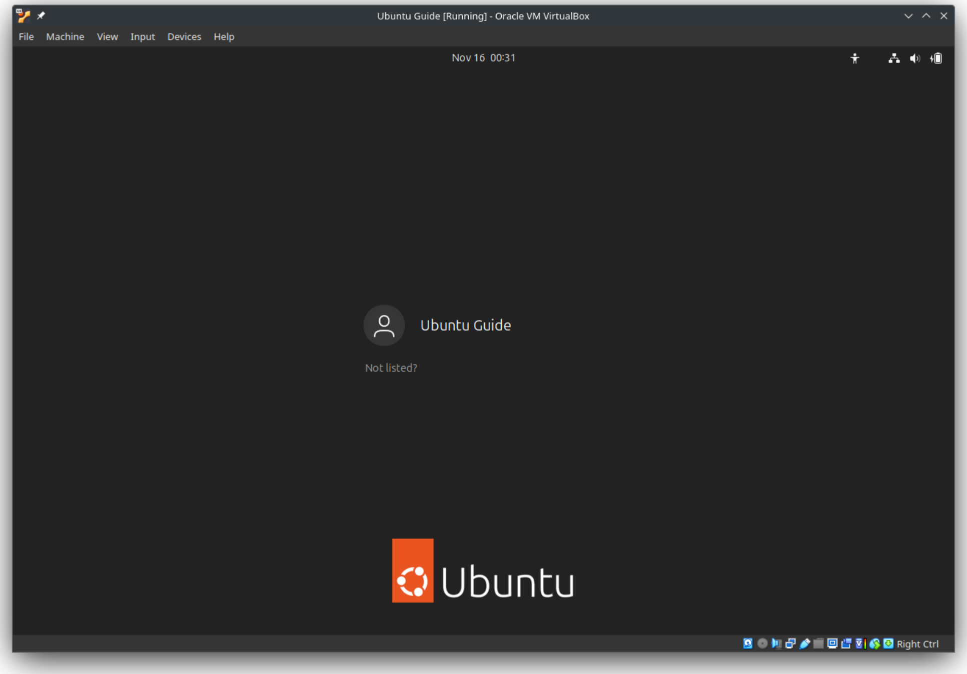
17. Wait until the VM reboots into the Ubuntu login screen.



Congratulations! You have installed Ubuntu.

2 Ubuntu Basics: Getting Our Bearings

And we are off!



The Ubuntu 23.10 login screen.

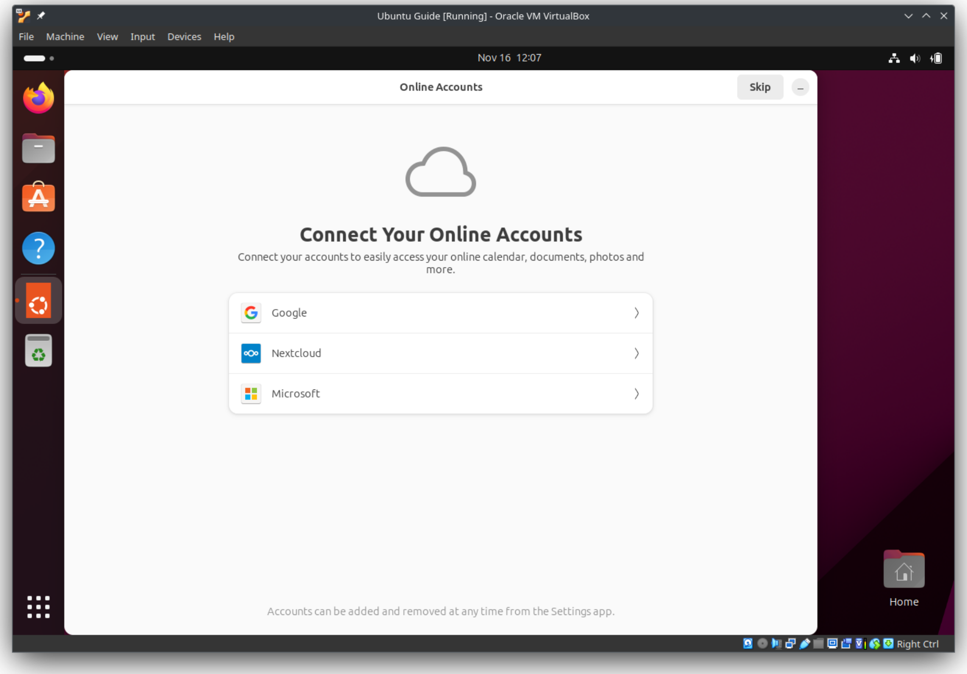
The lock screen in (default) Ubuntu is called the GNOME Display Manager, or GDM for short.

Click your name, input your password and press ENTER.

You will now log in to Ubuntu.

2.1 Where am I? - The GNOME Desktop Environment

After a successful login, you will come across this screen:



Connect Your Online Accounts

Connect your accounts to easily access your online calendar, documents, photos and more.

Accounts can be added and removed at any time from the Settings app.

Close the white window by clicking "Skip" on the upper right corner of the window, then "Next", and finally "Done".

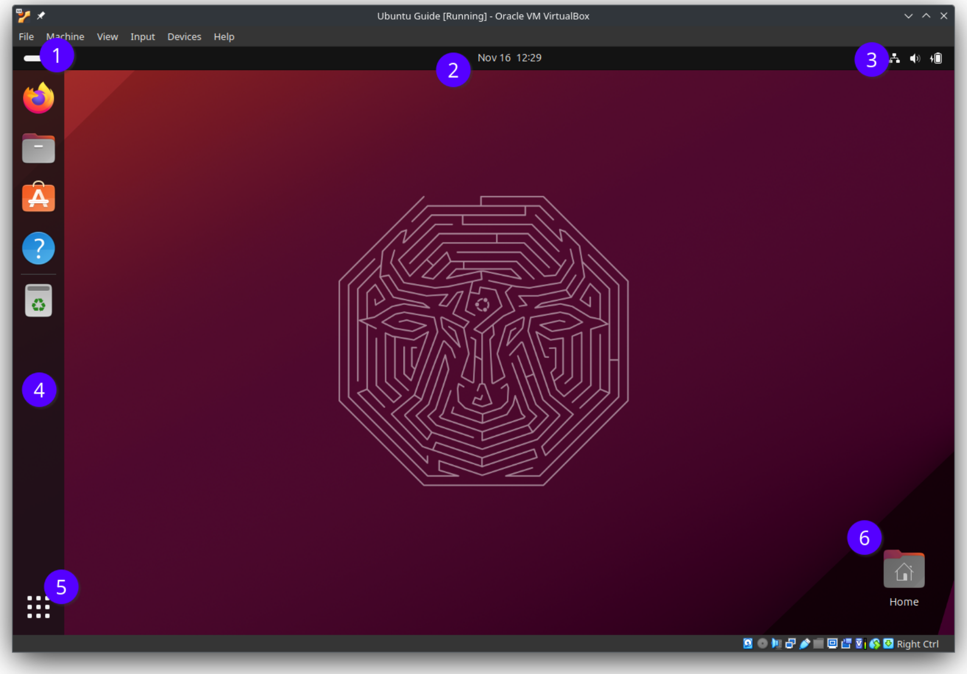
What you are looking at is called the **GNOME Desktop Environment**, or **GNOME** for short.

Every visual element you see in the picture (besides the VirtualBox window borders) is a part of GNOME; from the taskbar on the left, to the desktop, to the thin bar at the top.

Sidebar: There is an important but unintuitive - and uniquely Linux! - fact to take note of here: *GNOME is actually distinct from Ubuntu!* Ubuntu itself is more so the *backend* of your computer, GNOME is the thing determining what Ubuntu *looks like.* Other distros can also use GNOME as their desktop environment, in which case they look very similar to this screenshot;

conversely, Ubuntu can use other desktop environments and look completely different [8].

Let's look at the primary UI elements on this screen.



1. **Activities button:** Click this corner of the screen to go to the Activities view, where you can view all workspaces at once. The ellipse shows you which workspace you are currently viewing (circles denote other workspaces). Pressing the **SUPER key** (Windows key/ Option key) has the same effect.

2. **Top bar:** This top bar features a clock, the Activities button (left), and the system status area (right).

[8] Ubuntu supplies the GNOME desktop environment (DE) by default, which is why we are using GNOME right now. However, there are versions of Ubuntu that ship with *different* desktop environments - Kubuntu ships with KDE Plasma, Ubuntu MATE ships with the MATE desktop environment, and so on.

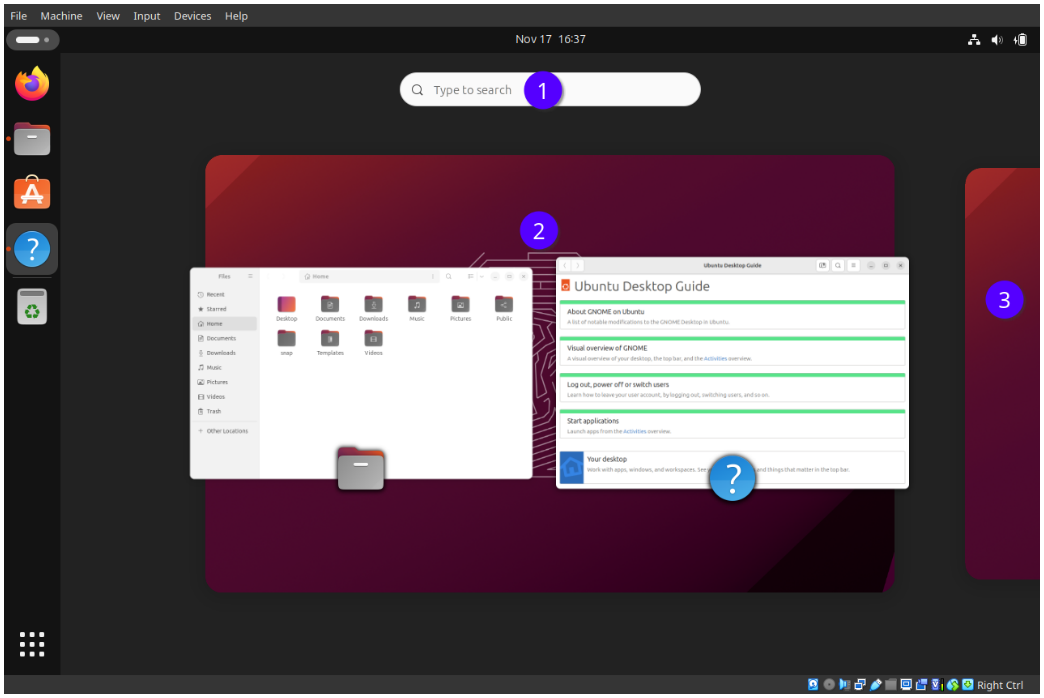
3. **System status area:** View system information (e.g. battery level) here. You can access the **user menu** by clicking the system status area; the user menu has options for common system settings like Wi-fi, Bluetooth, and power-off options.

4. **Dash:** This is where you can view the currently open applications and launch them, usual desktop fare.

5. **Application launcher:** Click this button to bring up the application launcher (a grid of desktop applications).

6. **Desktop icons:** As usual, the desktop houses shortcuts to various folders and programs. You do not really need to use this often in GNOME, though; the application launcher and Activities overview are better-integrated tools for finding files, folders or apps.

If you press the SUPER key (that's the Windows key on Windows computers or the "option" key on Mac) or click the upper-left corner on this main screen, you will get to the Activities overview.



Ubuntu Desktop Guide

About GNOME on Ubuntu
A list of notable modifications to the GNOME Desktop in Ubuntu.

Visual overview of GNOME
A visual overview of your desktop, the top bar, and the Activities overview.

Log out, power off or switch users
Learn how to leave your user account, by logging out, switching users, and so on.

Start applications
Launch apps from the Activities overview.

Your desktop
Work with apps, windows, and workspaces. See and things that matter in the top bar.

I opened a couple windows ("Files" and "Help" from the Dash on the left side) to illustrate how this screen looks when you have a few windows open.

Here are the key features of this screen:

1. **Search bar:** You can launch applications quickly from here by starting to type their name - a list of results will show up as you type. You do not need to click the search bar to start a search - you can just start typing.

2. **Window overview:** You will see all windows open in the current **workspace** in the middle of the screen. Generally, you will use workspaces to group various windows, as they act like different desktops you can quickly switch between. Scroll with your mouse or touchpad to focus on different workspaces.

3. **Next workspace:** You can see the (currently empty) second workspace on the right-hand side. If you are working with multiple workspaces, there might also be another workspace on the left. If you drag and drop a window into this area, it will move to this workspace.

The overview is important because it will be a key part of your workflow in GNOME. The best way to figure out how it might be useful for you is to spend some time working in it.

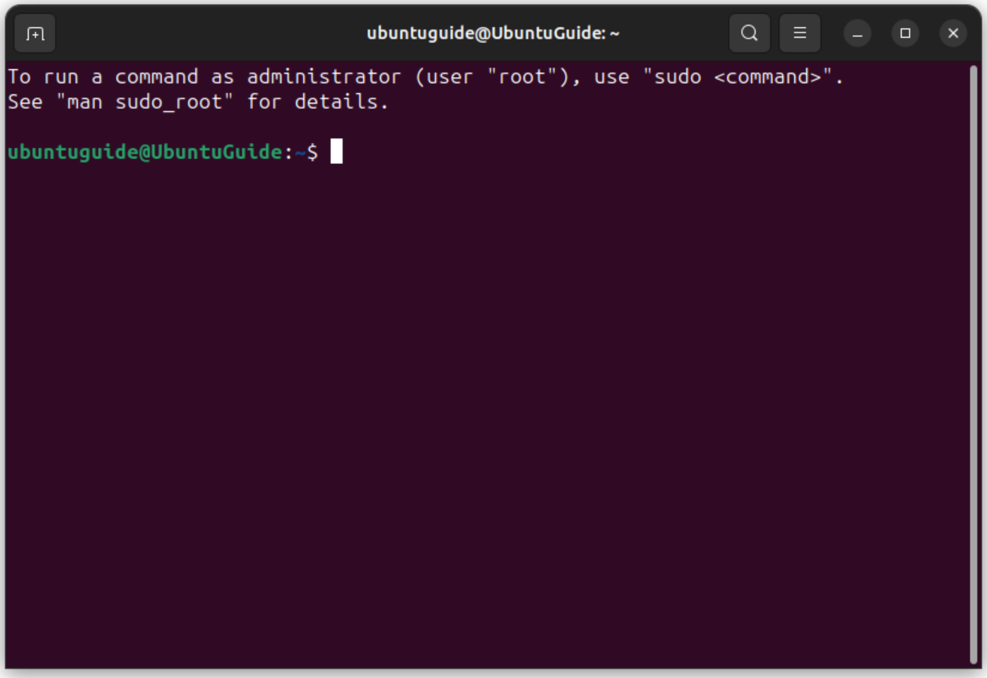
It is worth noting that the overview deliberately discourages relying on desktop icons, unlike Windows (and arguably Mac). Once you adapt, it'll be quicker and easier to launch apps by going into the overview and typing their name, compared to looking for a desktop shortcut each time.

2.2 The Terminal

In order to understand how to effectively use Linux, you will need to learn how to use the terminal.

From the GNOME desktop, open the application menu or the overview, and search for 'Terminal'. This is the most important program in Ubuntu.

Launch the terminal by clicking the Terminal icon, or pressing ENTER.



```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntuguide@UbuntuGuide:~$ 
```

This is the terminal. The central idea is simple- you give the computer a text command, and the computer does what you say.

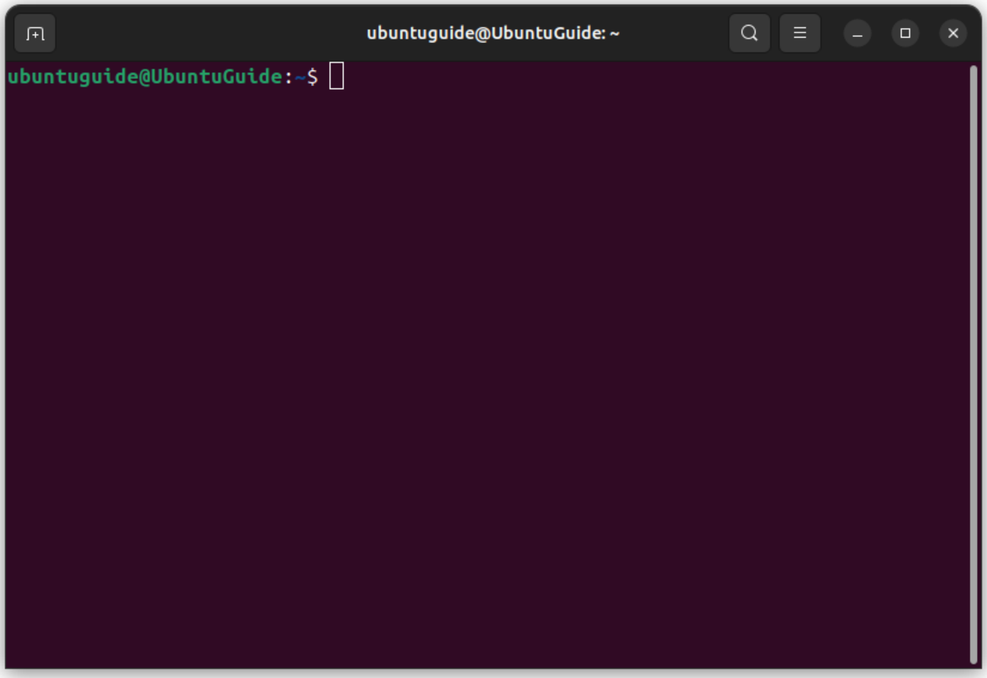
You can use the terminal to do *just about anything*- you can code, manipulate files, install or start programs, change your computer's settings, play music... Essentially, the terminal is a whole new way to use your computer. In the rest of this section, I will introduce you to some basic concepts about the terminal and show you a few basic terminal commands.

2.2.1 Terminal basics

The types of programs you are used to running, which create a window and have visual elements you can then interact with (using a mouse and keyboard), are called **Graphical User Interface (GUI)** programs.

In contrast, the programs you run in a terminal by issuing text prompts are called **Command Line Interface (CLI)** programs. These programs take text as input, and often have text as output; however, they can also create, modify and delete files much like GUI programs.

Let's look at an empty terminal.



```
ubuntuguide@UbuntuGuide:~$ 
```

The colorful text at the start, `ubuntuguide@UbuntuGuide:~$`, is called the **command prompt.** The command prompt tells you which user you are logged in as, which machine you are logged in to, and where you are in the file system of this machine. Let's dissect it:

- `ubuntuguide` is our **username.** You will use your username for a lot of things - this is basically the name the Ubuntu knows you by.

- `@` separates the username from the hostname. Conventionally, the username is all lowercase.

- `UbuntuGuide` is our **hostname.** This is the name that the Ubuntu installation will show to other devices on the network. Essentially, this is the name of the machine. Conventionally, the host name is one word but can include capital letters.

- : just separates the hostname from the current directory - see below.

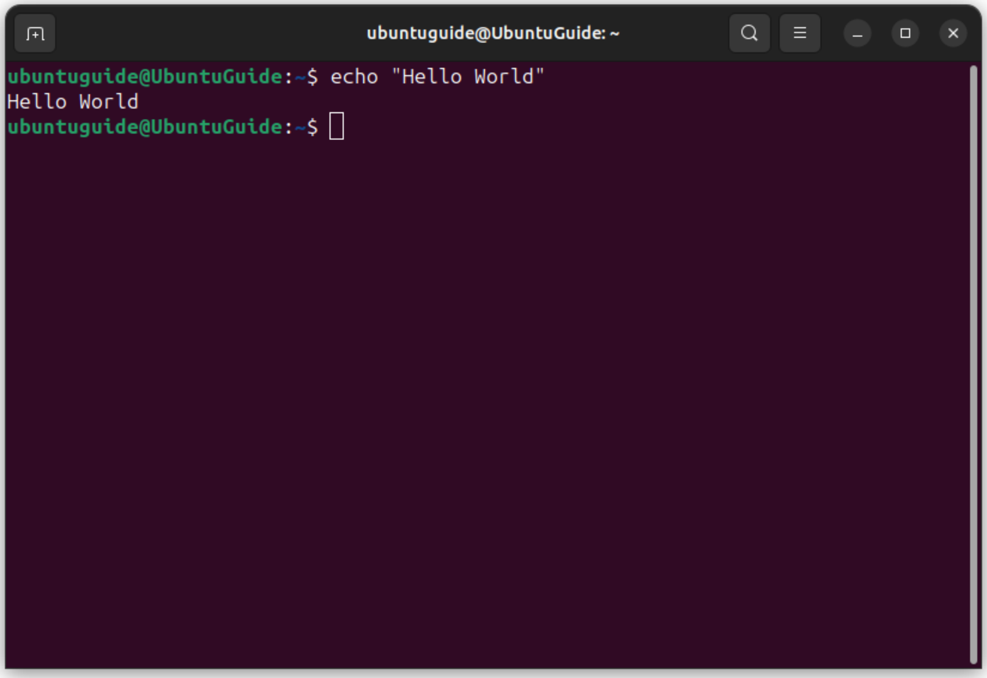
- `~` is the **current working directory.** This is your location in the file system. The tilde (`~`) is shorthand for your **home directory-** that is to say, `~` here is equivalent to `/home/ubuntuguide`. You will learn more about how Linux organizes system files and how to navigate a Linux file system in the terminal later on.

- `$` indicates the end of the command prompt. Furthermore, the fact that this icon is `$` tells us that we are currently logged in as a non-administrator. The administrator account (called the **root account** in Linux parlance) has a slightly different command prompt which ends in `#` instead of `$`.

2.2.2 echo

The first command we will learn about is `echo`. This command basically prints out the text[9] you put in without changing it.

Give it a try: Type `echo "Hello World"` in the terminal, and press ENTER.



```
ubuntuguide@UbuntuGuide:~$ echo "Hello World"
Hello World
ubuntuguide@UbuntuGuide:~$ 
```

Congratulations! You have just run your first terminal program.

[9] Text strings are usually enclosed in single or double quotes. There IS a difference between using double and single quotes, but you do not need to worry about that at present.