# The Full Guide to Operating Systems

What an operating system is, which one you have,
and how to install Linux and Windows yourself

# Contents

Chapter 1
What is an operating system? 3

Chapter 2
What operating system do I have? 9

Chapter 3
How Linux works and how to install it 16

Chapter 4
Linux distros: which one do I choose? 24

Chapter 5
How Windows works and how to install it 31

# Chapter 1: What is an operating system?

## Press the power button

You press the power button. A logo appears, a few seconds pass, and then your desktop is there with your wallpaper, your files and your apps. You click the browser icon and a window opens. You plug in a mouse and it simply works.

None of that happens by itself. A computer fresh from the factory is a box of parts: a processor that does arithmetic very fast, memory chips that hold numbers, a disk that stores them, and a screen that lights up coloured dots. Not one of those parts knows what a file, a window or an app is.

Something has to turn that box of parts into a machine you can use. That something is the operating system. This chapter explains what it is, what it does all day, and where you have already met it.

## What an operating system is

An operating system, or OS, is the main program on a computer. It is the first program to start when the machine turns on and the last to stop when it shuts down. While it runs, it manages everything else: the hardware and every other program.

It helps to look at it from two sides.

- **To the hardware, the OS is the boss.** It decides which program gets the processor, how much memory each one may use, and who is allowed to read or write the disk.
- **To programs, the OS is a helper.** It offers ready-made services such as "save this file", "show this window" and "send this over the internet", so no app has to build them from scratch.

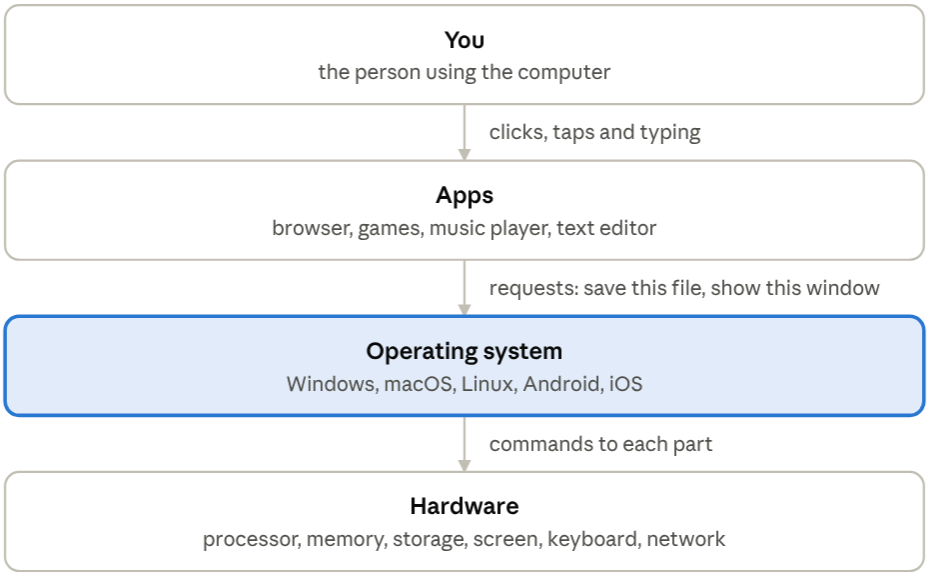
Think of a hotel. Guests come and go, and the staff hand out rooms, keep guests out of each other's rooms, run the lifts and the plumbing, and keep the records at the front desk. Guests never fix the plumbing themselves. They ask the staff.

On a computer the apps are the guests and the operating system is the staff. Rooms are memory, the plumbing is the hardware, and the front desk records are your files.

Computers did not always work this way. The earliest machines of the 1940s and 1950s had no operating system at all. They ran one program at a time, and every program had to control the hardware by itself. Operating systems were invented so that programmers could stop rewriting the same hardware code again and again, and so that one machine could be shared by many programs.

# Where the operating system sits

**Every app reaches the hardware through the operating system**




the four layers of a computer in use

Read the picture from top to bottom. You talk to apps, apps ask the operating system, and only the operating system gives orders to the hardware. Answers travel back up the same way.

## The five jobs of an operating system

Every operating system, from the one in your phone to the one in a supercomputer, does the same five jobs.

### 1. Running programs

A program on the disk is only a file. When you open it, the OS loads it into memory and starts it. A running program is called a process.

You can listen to music, download a file and type a message at the same time. A single processor core can only do one thing at a time, so the OS switches between processes many times every second. The switching is so fast that everything seems to run at once. Deciding who goes next is called scheduling.

### 2. Managing memory

Memory, or RAM, is the fast workspace where running programs keep what they are working on. There is only so much of it, and every process wants some.

The OS gives each process its own share and keeps the shares apart, so a bug in one app cannot scribble over another app's data. When you close an app, the OS takes its memory back and hands it to something else.

### 3. Storing files

A disk on its own stores nothing but millions of numbered blocks of data. It has no idea what a photo or a folder is.

The part of the OS called the file system turns those blocks into named files inside folders. It remembers which blocks belong to which file, how big each file is, and when it was last changed. When you save a document, the file system finds free blocks and records where it put them.

### 4. Talking to devices

Keyboards, screens, printers, cameras, Wi-Fi cards and game controllers all speak different languages. The OS deals with each one through a driver, a small piece of software that translates between the OS and one kind of device.

This is why an app can simply say "print this page" without knowing anything about your printer. It is also why a new device sometimes does nothing until its driver is installed.

### 5. Keeping things safe

The OS decides who is allowed to do what. It asks you to sign in, keeps each user's files private, and stops ordinary apps from changing important system settings without permission.

It also protects programs from each other. If one app crashes, the OS cleans up after it and everything else keeps running.

## See it for yourself

You can watch the OS doing the first two jobs right now.

- **Windows:** press Ctrl + Shift + Esc to open Task Manager.
- **macOS:** open Activity Monitor from the Utilities folder inside Applications.
- **Linux:** open System Monitor, or type `top` in a terminal.

Each tool lists every running process and shows how much processor time and memory it is using. You will see many more processes than the apps you opened. Most of them are parts of the operating system working in the background.

## The kernel and everything around it

The heart of an operating system is the kernel. It is the part that stays in memory the whole time the computer is on, and the only part with full control over the hardware. The five jobs above are, at the lowest level, the kernel's jobs.

Apps run in a restricted mode called user mode, where they cannot touch the hardware. The kernel runs in kernel mode, where it can. When an app needs something only the kernel can do, it makes a system call, which is a formal request to the kernel.

Saving a file shows how this works:

1. You press Save in a text editor.
2. The editor makes a system call that says "write this data to this file".
3. The kernel checks that you are allowed to change that file.
4. The file system picks free blocks on the disk.
5. The disk driver sends the data to the disk.
6. The kernel tells the editor the save worked.

The editor never learns what kind of disk you have. It only knows how to ask.

A kernel alone is not something you can sit down and use. A complete operating system wraps more around it:

- **System services:** background programs that handle things like networking, printing and updates.
- **A user interface:** either graphical, with windows, icons and a pointer, or a command line, where you type commands as text.
- **Basic tools:** a file manager, a settings app, an installer for new software.

The difference matters most with Linux. Strictly speaking, Linux is only a kernel. Ubuntu, Fedora and Linux Mint are complete operating systems built around that kernel, and they are called distributions. Chapter 4 helps you choose one.

You can ask your own kernel for its name. On Linux or macOS, open a terminal and type:

```
uname -sr
```

It answers with the kernel's name and version, something like `Linux 6.8.0` or `Darwin 24.6.0`. On Windows, open Command Prompt and type:

```
ver
```

It answers with something like `Microsoft Windows [Version 10.0.26100]`.

## Operating systems you already use

You have almost certainly used at least two of these six.

| Operating system | Made by | First released | Where you find it |
|---|---|---|---|
| Windows | Microsoft | 1985 | Most desktop and laptop PCs |
| macOS | Apple | 2001, as Mac OS X | Mac computers |
| Linux | Linus Torvalds and thousands of volunteers and companies | 1991 | Servers, supercomputers, the Raspberry Pi, some PCs |
| Android | Google | 2008 | Most phones and tablets, many TVs |
| iOS | Apple | 2007 | iPhones |
| ChromeOS | Google | 2011 | Chromebooks |

Some of these are related. Android and ChromeOS are both built on the Linux kernel. macOS and iOS share the same kernel too.

Operating systems also hide in places you might not expect. A games console, a smart TV, a Wi-Fi router, a smartwatch and the screen in a car's dashboard each run one. So do the servers behind every website you visit, and most of those run Linux.

## Common misunderstandings

**"The operating system is the desktop I see."** The desktop is only the visible surface. Most of the OS is the kernel and background services you never see. Many servers run with no desktop at all.

**"Only computers have an operating system."** Phones, tablets, consoles, TVs and watches all have one. Anything that runs more than one program needs something to manage them.

**"My browser and my office apps are part of the operating system."** They are separate programs that run on top of it. You can remove them or swap them for others, and the OS carries on.

**"Linux is one operating system."** Linux is a kernel. There are hundreds of operating systems built on it, and they can look and feel very different from each other.

**"The operating system is built into the computer and cannot be changed."** The OS is software stored on the disk. On most PCs you can replace it, or install a second one beside it. Chapters 3 and 5 show how.

## Summary

- An operating system is the main program on a computer. It manages the hardware and every other program.
- It does five jobs: running programs, managing memory, storing files, talking to devices and keeping things safe.
- The kernel is the core of the OS and the only part with full control over the hardware.
- Apps cannot touch hardware directly. They ask the kernel through system calls.
- A complete OS is a kernel plus services, a user interface and basic tools.
- Windows, macOS, Linux, Android, iOS and ChromeOS are all operating systems, and many everyday devices run one.

## Exercises

1. Name three devices in your home, other than a computer or a phone, that run an operating system.
2. Open Task Manager, Activity Monitor or System Monitor. How many processes are running? Which one uses the most memory?
3. Explain in your own words the difference between a kernel and an operating system.
4. A music app keeps playing while you browse the web. Which of the five jobs make that possible? There is more than one.
5. Why is it safer for apps to ask the kernel to use the disk than to control the disk themselves?

The next chapter shows how to find out exactly which operating system, and which version of it, you have.

# Chapter 2: What operating system do I have?

## "What are you running?"

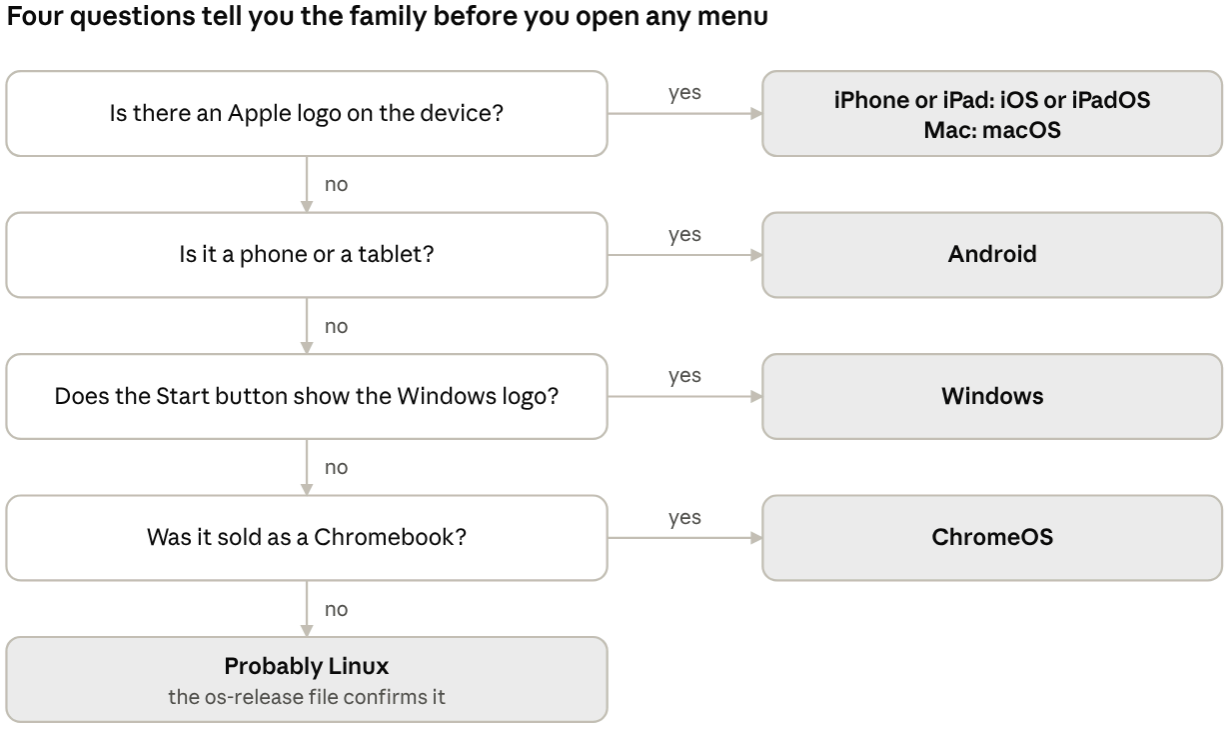
You find a game you want, and the store page says it requires "Windows 10 version 22H2 or later, or macOS 13 or later". You ask for help on a forum, and the first reply is "which OS are you on?". A printer's website offers six different driver downloads and asks you to pick yours.

In each case you need the same three facts: the name of your operating system, its version, and the kind of processor it runs on. Many people know the first one roughly ("it's Windows") and neither of the others.

This chapter shows where every major operating system keeps those facts. Finding them takes less than a minute once you know where to look.

## Spot the family at a glance

Before you open any menu, the device itself gives the first answer.




Identifying the family: four questions, five answers

Start at the top and follow the first "yes". The shaded boxes are the answers.

This only tells you the family, and it can be wrong. A laptop sold with Windows may have had Linux installed on it later. The sections below give the exact name and version, straight from the operating system itself.

## Windows

The fastest way works on every version of Windows.

1. Hold the Windows key and press R. A small Run box opens.
2. Type `winver` and press Enter.

A window appears with the Windows name at the top and a line such as "Version 24H2 (OS Build 26100.4351)". Lower down it names the edition, for example Windows 11 Home.

For more detail, use the Settings app.

1. Open the Start menu and choose Settings.
2. Choose System, then scroll down to About.

The About page has two lists.

- **Device specifications** describes the machine: its processor, its memory, and a line called System type.
- **Windows specifications** describes the OS: Edition, Version and OS build.

System type is the line to note. It says either "x64-based processor" or "ARM-based processor". You will need that in Chapter 5 when you download Windows to install it.

If you prefer typing, open Command Prompt and run:

```
systeminfo | findstr /B /C:"OS Name" /C:"OS Version"
```

It prints two lines, something like:

```
OS Name:     Microsoft Windows 11 Home
OS Version:  10.0.26100 N/A Build 26100
```

The version number still begins with 10.0 on Windows 11. Microsoft kept that internal number when it changed the name, so trust the OS Name line.

If your PC turns out to run Windows 10, note that Microsoft ended regular support for it on 14 October 2025. It still works, but it no longer gets free security fixes.

## macOS

On a Mac, everything is one click away.

1. Click the Apple logo in the top-left corner of the screen.
2. Choose About This Mac.

The window shows the Mac's model, its chip, its memory, and a line such as "macOS Sequoia 15.6". That line holds both the name and the version number.

Every major version of macOS has a name and a number: Sonoma is 14, Sequoia is 15 and Tahoe is 26. The jump is not a mistake. In 2025 Apple skipped from 15 to 26 so that the number follows the year.

The Chip line tells you the processor type. "Apple M1", "M2" and so on mean an ARM processor, often called Apple silicon. Older Macs show a Processor line that names an Intel chip instead.

To get the same facts as text, open the Terminal app and type:

```
sw_vers
```

It prints something like:

```
ProductName:    macOS
ProductVersion:  15.6
BuildVersion:    24G84
```

## Linux

Linux comes in many distributions, and each one arranges its menus a little differently. Most desktop versions have an About page.

1. Open Settings.
2. Choose About. On newer versions it sits inside System.

The page shows the distribution's name and version, such as "Ubuntu 24.04.1 LTS", along with the processor and memory. LTS stands for long-term support, meaning that version receives updates for several years.

One method works on almost every distribution, whatever its menus look like. Open a terminal and type:

```
cat /etc/os-release
```

This prints a small text file that every modern distribution includes. The useful lines look like this:

```
PRETTY_NAME="Ubuntu 24.04.1 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
ID=ubuntu
ID_LIKE=debian
```

`PRETTY_NAME` is the answer in one line. `ID_LIKE` names the family the distribution belongs to, which matters when you look for help or software. Chapter 4 explains those families.

Two more commands complete the picture.

- `uname -r` prints the kernel version, such as `6.8.0-45-generic`.
- `uname -m` prints the processor type: `x86_64` for Intel and AMD chips, `aarch64` for ARM.

## Phones, tablets and Chromebooks

These devices have no command to type. The Settings app holds everything.

### Android

1. Open Settings.
2. Scroll to the bottom and tap About phone, or About tablet.
3. Look for Android version. On Samsung devices, tap Software information first.

You will see a number such as 15 or 16. Tap it to see the date of the last security update.

Many phone makers put their own design on top of Android and give it a name, such as One UI on Samsung phones or HyperOS on Xiaomi phones. That name and its version appear on the same page. The operating system underneath is still Android.

### iPhone and iPad

1. Open Settings.
2. Tap General, then About.
3. Read the line called iOS Version, or iPadOS Version on an iPad.

Apple renumbered these in 2025 as well, going from iOS 18 straight to iOS 26.

### Chromebook

1. Click the clock in the bottom-right corner.
2. Click the gear icon to open Settings.

3. Choose About ChromeOS at the bottom of the menu on the left.

The page shows the ChromeOS version number and a button that checks for updates.

## What the details mean

The pages above use a handful of words again and again. Each one answers a different question.

**Name.** Which operating system it is: Windows, macOS, Ubuntu, Android. This tells you which apps can run at all.

**Edition.** Which variety of that system you have. Windows comes as Home and Pro, and Pro adds features aimed at businesses. Most apps do not care about the edition.

**Version.** How new it is. App requirements are written in versions, such as "macOS 13 or later". The version also decides whether you still receive security updates, because makers stop fixing old versions after some years.

**Build.** The exact release, down to the last small update. You rarely need it, but support staff sometimes ask for it because two machines on the same version can be on different builds.

**Processor type.** The family of chip the OS was made for. Two families cover almost everything today.

- **x64** is used by Intel and AMD processors in most PCs. It is also written x86_64 or AMD64. All three names mean the same thing.
- **ARM** is used by every phone and tablet, by Macs with Apple chips, and by some newer Windows laptops. It is also written ARM64 or aarch64.

A program or an operating system built for one family does not run directly on the other. That is why download pages offer both and ask you to choose.

You may also see "64-bit". It describes how much memory the processor can address at once. Nearly every device sold in the last ten years is 64-bit, so the question that matters now is x64 or ARM.

## Quick reference

Keep this table for the next time someone asks what you are running.

| System | Where to click | What to type instead |
| --- | --- | --- |
| Windows | Settings → System → About | `winver` in the Run box |

| System | Where to click | What to type instead |
|---|---|---|
| macOS | Apple menu → About This Mac | `sw_vers` in Terminal |
| Linux | Settings → About | `cat /etc/os-release` in a terminal |
| Android | Settings → About phone | Nothing: use Settings |
| iPhone and iPad | Settings → General → About | Nothing: use Settings |
| ChromeOS | Settings → About ChromeOS | Nothing: use Settings |

## Common misunderstandings

**"I have a Dell, so that is my operating system."** Dell, HP, Lenovo and Asus make the machine. The operating system is the software on it, usually Windows, and it could be replaced with another.

**"I use Chrome, so I have ChromeOS."** Chrome is a web browser, an app that runs on Windows, macOS, Linux and Android. ChromeOS is a separate operating system found only on Chromebooks.

**"My Samsung phone runs One UI, not Android."** One UI is Samsung's design and extra features placed on top of Android. The apps you install are Android apps.

**"Windows 11 is the whole answer."** It is only the name. Two Windows 11 PCs can be on different versions, and an app or a fix may need a particular one.

**"The kernel version is the OS version."** They are different numbers. A PC running Ubuntu 24.04 reports a kernel version such as 6.8, and a Mac running macOS 15 reports a kernel called Darwin 24. Chapter 1 explained why: the kernel is only one part of the operating system.

## Summary

- Three facts describe what you are running: the name of the OS, its version, and the processor type.
- On Windows, type `winver` in the Run box, or open Settings, System, About.
- On a Mac, open the Apple menu and choose About This Mac.
- On Linux, open Settings and About, or type `cat /etc/os-release`.
- On phones, tablets and Chromebooks, look for About in the Settings app.
- Processors come in two main families, x64 and ARM, and software built for one does not run directly on the other.

- The version decides which apps you can run and whether you still get security updates.

## Exercises

1. Find the name, version and build of the operating system on the device you are using now. Write them down.
2. Do the same for one phone or tablet in your home.
3. Is your computer x64 or ARM? Say where you found the answer.
4. Pick an app or game you like and look up its system requirements. Does your device meet them?
5. Visit the website of the company that makes your operating system and find out whether your version still receives security updates.

The next chapter leaves the Settings app behind and looks inside Linux: how it works, and how to install it yourself.

# Chapter 3: How Linux works and how to install it

## The operating system you can install yourself

Somewhere in many homes there is an old laptop that became too slow to use. It takes minutes to start, it can no longer get updates, and nobody wants it.

That laptop can often be made useful again in about an hour, for free. You download an operating system, copy it to a USB stick, and install it yourself. The operating system is Linux.

Linux is not only for rescuing old machines. It runs most of the servers behind the internet, every one of the 500 fastest supercomputers, and, through Android, most of the world's phones. This chapter explains how it works and then walks through installing it on a PC.

## Where Linux came from

In 1991 Linus Torvalds, a 21-year-old student in Helsinki, wrote a kernel for his own PC and shared it on the internet. He described it as "just a hobby, won't be big and professional". Other programmers started sending improvements, and they have never stopped.

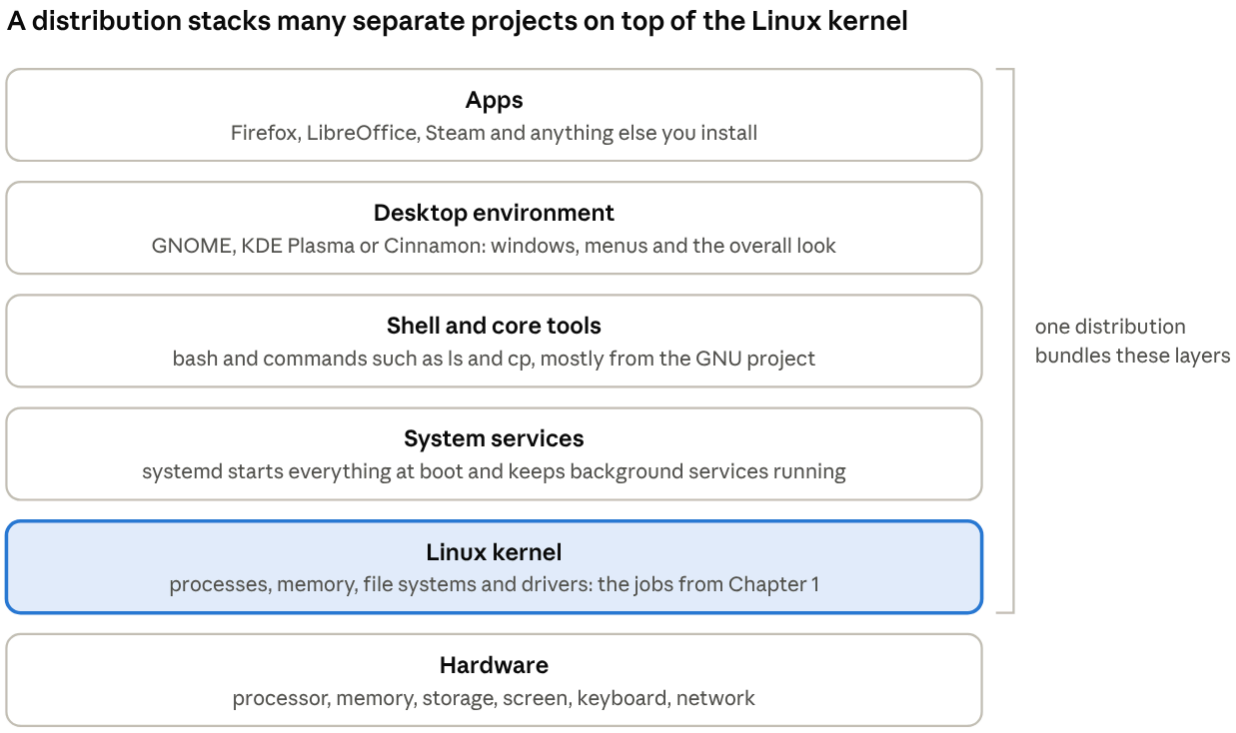
A kernel was the missing piece of a larger effort. Since 1983 the GNU project, started by Richard Stallman, had been writing free versions of all the tools an operating system needs, but it had no working kernel. Linux plus the GNU tools made a complete system.

Linux is open source. Anyone may read its code, change it and share it, as long as they share their changes under the same terms. Two things follow from that.

- **It costs nothing.** You can download it, install it on as many machines as you like, and give copies away.
- **Nobody owns it alone.** Thousands of volunteers and companies improve it together, and anyone can build a new operating system from it. That is why so many distributions exist.

## How a Linux system is put together

Chapter 1 said that a complete operating system is a kernel with more wrapped around it. On Linux each of those parts is a separate project, made by a different group of people. A distribution chooses one of each and bundles them.




The layers of a Linux system, from apps down to hardware

Read the picture from the bottom up. The kernel controls the hardware, the system services start and run in the background, the shell and tools let you give commands, and the desktop environment draws what you see.

Because the parts are separate, they can be swapped. Two distributions can share the same kernel and still look nothing alike, simply because they chose different desktop environments.

## Four ideas you meet on day one

Linux does the same five jobs as every operating system. Four of its habits feel different from Windows, and knowing them makes everything else easier.

### 1. One tree of folders

Windows gives each disk a letter, such as C: and D:. Linux has no drive letters. Every file on every disk lives in a single tree that starts at one folder called the root, written as a lone slash: `/`.

| Folder | What it holds |
| --- | --- |
| `/home` | One folder per user, with your documents, downloads and settings |

| Folder | What it holds |
|---|---|
| `/etc` | System settings, stored as plain text files |
| `/usr` | Installed programs |
| `/var` | Data that changes as the system runs, such as logs |
| `/dev` | Devices, each shown as a file |
| `/tmp` | Temporary files, cleared regularly |

Your own files live in `/home/yourname`. A USB stick you plug in appears as a folder inside the tree instead of getting a letter.

## 2. Ordinary users and root

Linux separates everyday work from system changes. Your account can read and change your own files but not the system's. One special account, called root, can change anything.

You do not sign in as root. When a task needs its power, you put the word `sudo` in front of a command and type your password. This habit means a mistake or a bad program in your account cannot easily damage the whole system.

## 3. The terminal

A modern Linux desktop has windows, menus and a mouse pointer, and you can use it for years without typing a command. The terminal is still worth learning, because it is fast and it works the same on every distribution.

The program that reads what you type is called the shell. Seven commands cover most of what a beginner needs.

| Command | What it does |
|---|---|
| `pwd` | Shows which folder you are in |
| `ls` | Lists the files in that folder |
| `cd Documents` | Moves into the Documents folder |
| `mkdir notes` | Makes a new folder called notes |
| `cp a.txt b.txt` | Copies a file |
| `mv a.txt notes/` | Moves a file, or renames it |

| Command | What it does |
| --- | --- |
| `man ls` | Shows the manual page for a command |

One warning. The command `rm` deletes files at once, with no recycle bin. Check what you typed before pressing Enter.

### 4. Software comes from a package manager

On Windows you usually find a program on a website, download an installer and run it. On Linux, most software comes from a repository, a large library of tested programs kept by the people who make your distribution.

A tool called the package manager fetches programs from the repository, installs them and keeps them updated. On Ubuntu and its relatives the tool is `apt`.

```
sudo apt update
sudo apt install vlc
```

The first line refreshes the list of available software. The second installs the VLC media player. One more command, `sudo apt upgrade`, updates every program on the system at once, not only the operating system.

If you prefer clicking, every desktop distribution includes an app store that does the same job.

## Four ways to run Linux

You do not have to give up your current system to try Linux. The four options below run from the safest to the most committed.

| Way | What happens | Risk to your files | Good for |
| --- | --- | --- | --- |
| Live USB | Linux runs straight from a USB stick and nothing is installed | None | A first look, and testing that your hardware works |
| Virtual machine | Linux runs in a window inside Windows or macOS, using a free program such as VirtualBox | None | Learning the terminal and trying several distributions |

| Way | What happens | Risk to your files | Good for |
|---|---|---|---|
| Dual boot | Linux is installed beside Windows and you pick one each time the PC starts | Some, because the disk is divided in two | Keeping Windows for a few programs |
| Full install | Linux replaces everything on the disk | Everything on the disk is erased | An old or spare PC |

Start at the top of the table. A live USB costs you nothing but ten minutes, and the same stick is what you use to install later.

The steps in this chapter are for ordinary PCs. On a Mac with an Apple chip, use a virtual machine instead.

## Before you install

Gather these first. The example distribution in this chapter is Ubuntu, and the numbers are its stated minimums.

- [ ] A PC with an x64 processor, which Chapter 2 showed you how to check
- [ ] At least 4 GB of memory and 25 GB of free disk space
- [ ] A USB stick of 8 GB or more, which will be wiped
- [ ] A backup of every file you care about, on another drive or in the cloud
- [ ] About an hour, and the charger plugged in if it is a laptop

The backup is the one item you must not skip. Installing an operating system changes how the disk is divided, and one wrong click can erase it.

If you plan to dual boot with Windows, check two more things inside Windows.

- **Find your BitLocker recovery key.** Many Windows 11 PCs encrypt the disk automatically. Changing how the PC starts can make Windows ask for a long recovery key, which is normally saved in your Microsoft account. Make sure you can reach it before you begin.
- **Turn off Fast Startup.** This Windows feature leaves the disk half asleep when you shut down, and Linux then cannot use it safely. You will find it in Control Panel under Power Options, in "Choose what the power buttons do".

## Installing Linux, step by step

These steps use Ubuntu. Other distributions follow nearly the same path, and Chapter 4 helps you choose between them.

1. **Download the installer.** Go to the Ubuntu website and download Ubuntu Desktop. You get one large file ending in `.iso`, about 6 GB in size. An ISO file is a complete copy of an installation disc.
2. **Write it to the USB stick.** Dragging the file onto the stick does not work. Use a free tool such as balenaEtcher, or Rufus on Windows: choose the ISO, choose the stick, and press the button to write. This erases the stick.
3. **Start the PC from the stick.** Plug it into the PC you want to install on and switch the PC on. Straight away, press the key that opens the boot menu. It is often F12, F10, F2, Esc or Del, and it usually flashes on screen for a moment. Pick the USB stick from the menu.
4. **Try before you install.** A menu offers "Try or Install Ubuntu". Choose to try it. You are now running Linux from the stick, and nothing on the disk has changed. Check that the Wi-Fi, sound, screen and touchpad all work.
5. **Start the installer.** Open the Install Ubuntu icon on the desktop. It asks for your language, your keyboard layout and your Wi-Fi network.
6. **Choose where to install.** This is the one step that can destroy data, so read the screen twice. "Install alongside Windows" sets up a dual boot and lets you decide how much disk each system gets. "Erase disk and install" removes everything and gives the whole disk to Linux.
7. **Create your account.** Type your name, a name for the computer and a password. Then pick your time zone.
8. **Wait.** Copying the files takes between 10 and 30 minutes on most machines.
9. **Restart.** Pull out the USB stick when the screen tells you to. The PC starts from its own disk, and you sign in with the password you chose.

On a dual-boot PC, a short menu now appears every time you switch on. It lists Ubuntu and Windows, and you choose with the arrow keys.

## If something goes wrong

- **The PC ignores the stick and starts Windows.** You missed the boot-menu key. Restart and press it sooner, or try another USB port.
- **The stick is not in the boot menu.** Write the ISO to the stick again, or use a different stick.
- **The screen goes black after you choose Ubuntu.** Restart from the stick and choose the "safe graphics" option in its menu.
- **A message mentions Secure Boot.** Secure Boot is a firmware feature that only starts approved systems. Ubuntu is approved, but some distributions need it switched off in the PC's firmware settings.

## Your first hour

A fresh install is ready to use, but five small jobs make it properly yours.

1. **Update everything.** The installer is months old by the time you use it. Open Software Updater, or type the two commands below in a terminal.
2. **Install your apps.** Open the App Center and search. A web browser is already installed. LibreOffice opens Word and Excel files, VLC plays almost any video, and Steam runs many Windows games.
3. **Check for extra drivers.** Open the Additional Drivers tool. It offers better drivers for some graphics cards and Wi-Fi chips.
4. **Look around the file tree.** Open the Files app, or type `ls /` in a terminal, and find the folders from the table earlier in this chapter.
5. **Learn where to get help.** Your distribution has official documentation and a community forum. When something fails, search for the exact words of the error message.

```
sudo apt update
sudo apt upgrade
```

## Common misunderstandings

**"Linux is only for programmers."** A modern desktop distribution has an app store, a settings app and a file manager, and everyday tasks need no commands. The terminal is an extra tool, not an entry requirement.

**"Installing Linux means losing Windows."** Only a full install erases the disk. A live USB and a virtual machine change nothing, and a dual boot keeps both systems.

**"None of my programs will work."** Many popular apps have Linux versions, including Firefox, Chrome, VLC, Spotify, Discord and Steam. Some do not, such as the desktop versions of Microsoft Office and Adobe Photoshop, and some online games block Linux. Check the programs you depend on before a full install.

**"Linux cannot get viruses."** It is attacked far less often than Windows, and its separation of users and root limits the damage. It is not immune. Keep it updated and install software from the repository.

**"Free means low quality."** The same kernel runs the servers of the largest companies in the world. It is free because of how it is licensed, not because it is unfinished.

## Summary

- Linux is a free, open-source kernel first released by Linus Torvalds in 1991. A distribution adds everything else needed to make a complete operating system.
- All files live in one tree that starts at `/`, and your own files are in `/home`.
- Everyday work happens in an ordinary account. System changes need root, borrowed for one command with `sudo`.
- Software comes from a repository through a package manager such as `apt`.
- You can run Linux four ways: live USB, virtual machine, dual boot or full install.
- Installing takes a USB stick, an ISO file and about an hour. Back up first, and read the disk step twice.

## Exercises

1. Draw the layers of a Linux system from memory, from the hardware up to the apps.
2. Which folder holds your personal files? Which one holds system settings?
3. Make a live USB and start a PC from it, with the owner's permission. Do the Wi-Fi and sound work?
4. In a live session or a virtual machine, open a terminal. Use `pwd`, `ls`, `mkdir` and `cd` to make a folder called practice and move into it.
5. Explain the difference between a dual boot and a full install, and say when you would choose each.
6. Why does Linux ask for your password when you install software?

This chapter used Ubuntu as its example. The next chapter compares the main distributions and helps you pick the one that suits you.

# Chapter 4: Linux distros: which one do I choose?

## Hundreds of answers to one question

When you decide to try Windows, you download Windows. When you decide to try Linux, you search for a download and find hundreds of them: Ubuntu, Mint, Fedora, Debian, Arch, Manjaro, Pop!_OS, Zorin and many more, each with fans who say theirs is the best.

This stops many beginners before they start. It should not. All of these systems share the same Linux kernel and run mostly the same software. The differences are real but few, and only a handful of distributions matter for a first choice.

This chapter explains what the differences are, compares five well-known distributions, and ends with a simple way to pick one.

## What actually differs

Chapter 3 showed that a distribution is a bundle: the Linux kernel plus system services, tools, a desktop and apps. Distributions differ in what they put in the bundle and how they keep it updated. Four choices account for nearly everything you will notice.

**The package manager.** Each family of distributions has its own tool for installing software. Ubuntu and its relatives use `apt`, Fedora uses `dnf`, and Arch uses `pacman`. They do the same job with different commands, so guides written for one family do not always work on another.

**The desktop environment.** This decides how the screen looks and behaves: where the menu is, how windows are arranged, how much you can customise. It is the difference you see first, and a later section compares the main ones.

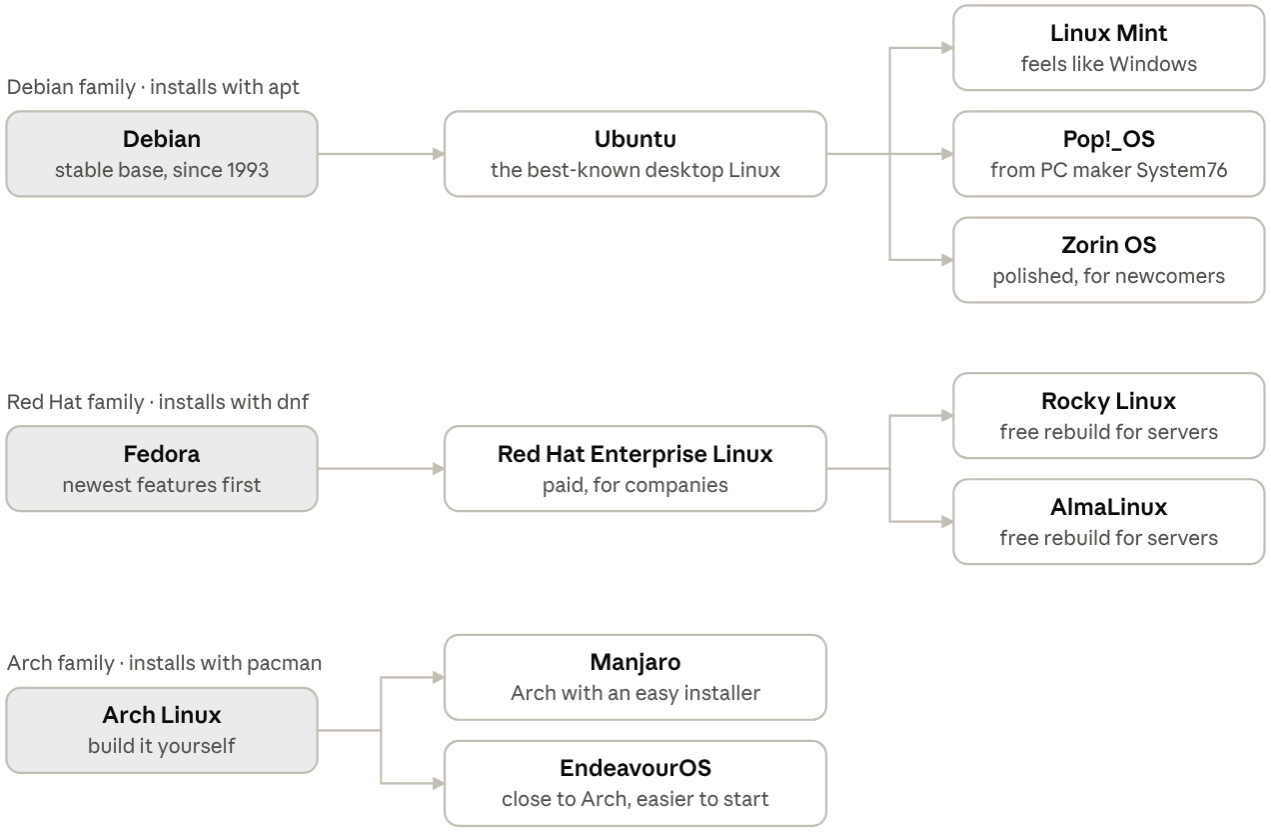
**The release style.** Some distributions publish a new version on a schedule and change little in between. This is called a fixed release, and it favours stability. Others update every program continuously as soon as a new version is ready. This is called a rolling release, and it favours having the newest software.

**The defaults.** Distributions decide which apps are installed from the start, how much is set up for you, and whether a company or a volunteer community stands behind the project. Beginner-friendly distributions make more of these decisions for you.

# Three families

Distributions are rarely built from nothing. Most start from an existing one and change it, so they form families.

**Three families produce most of the distributions you will meet**




Family tree of twelve well-known distributions

Read each row from left to right. The shaded box is the root of the family, and each arrow points to a distribution built from the one before it.

The family matters because relatives share a package manager and most of their commands. A guide written for Ubuntu usually works on Linux Mint, and almost never on Fedora or Arch.

# Five distributions compared

These five cover the range from easiest to most demanding. Any other distribution you meet will resemble one of them.

| Distribution | Family | Default desktop | Release style | Suits |
|---|---|---|---|---|
| Linux Mint | Debian, through Ubuntu | Cinnamon | Fixed, following Ubuntu's long-term versions | People coming from Windows, and older PCs |
| Ubuntu | Debian | GNOME | Fixed, every 6 months, with a long-term version every 2 years | Beginners who want the most guides and help |
| Fedora | Red Hat | GNOME | Fixed, about every 6 months | People who want recent software, and future IT workers |
| Debian | Debian | Your choice at install | Fixed, about every 2 years | Anyone who values stability above all, and servers |
| Arch Linux | Arch | None until you pick one | Rolling | Learners who want to build the system themselves |

## Linux Mint

Mint takes Ubuntu and reshapes it for people who have only used Windows. The Cinnamon desktop has a menu in the bottom-left corner, a taskbar and a system tray, all where a Windows user expects them. Media formats and common drivers work from the first start.

Choose it if you want the gentlest landing, or if the PC is several years old.

## Ubuntu

Ubuntu, first released in 2004 by the company Canonical, is the best-known desktop Linux. That popularity is its main advantage: almost every guide, forum answer and software download page covers Ubuntu first.

Its long-term support versions, marked LTS, appear every two years and receive five years of updates. Choose Ubuntu if you want the path most people have walked, and pick an LTS version.

### Fedora

Fedora is backed by Red Hat and brings new technology to its users early while staying reliable. Each version is supported for about 13 months, so you upgrade roughly once a year.

Red Hat's paid system for companies grows out of Fedora. Choose Fedora if you want recent software, or if you may one day work with the Linux servers that businesses use.

### Debian

Debian, founded in 1993, is one of the oldest distributions and the base that Ubuntu and Mint are built on. It is run entirely by volunteers. Its software is tested for a long time before release, so it is very stable and somewhat older.

Choose Debian for a machine that must simply keep working, such as a home server.

### Arch Linux

Arch gives you a minimal system and expects you to add everything else, including the desktop. Installing it means typing commands and reading its documentation, which is among the best anywhere.

That effort is the point. People choose Arch to understand how a Linux system fits together. Choose it as a second distribution, after you are comfortable with one of the others.

## Desktop environments

The desktop environment shapes your daily experience more than the distribution's name does. Four of them cover most installations.

| Desktop | How it feels | Demand on the PC |
|---|---|---|
| GNOME | Its own style: a clean screen, large controls, and an overview that shows every open window | Higher |
| KDE Plasma | Similar to Windows, with a taskbar and start menu, and almost every detail adjustable | Medium |
| Cinnamon | A classic Windows-like layout with few surprises | Medium |
| Xfce | Plain and traditional, built to stay fast | Low, so it suits old PCs |

You are not locked to a distribution's default. Most distributions publish separate editions with other desktops. Ubuntu with KDE Plasma is called Kubuntu, Ubuntu with Xfce is called Xubuntu, and Linux Mint offers an Xfce edition for older machines.

So the choice has two parts. First pick a distribution for its family, release style and community. Then pick the desktop you like looking at.

## How to choose

Find the row that sounds most like you.

| Your situation | Start with |
|---|---|
| This is my first time and I want it to just work | Linux Mint or Ubuntu |
| I have always used Windows | Linux Mint |
| My PC is old or has 4 GB of memory or less | Linux Mint Xfce edition, or Xubuntu |
| I mostly want to play games | Linux Mint, Ubuntu or Fedora, with Steam installed |
| I want the newest software | Fedora |
| I am aiming for a job in IT | Fedora or Ubuntu |
| I am setting up a home server | Debian |
| I want to learn how Linux is built | Arch Linux, as a second distribution |

If no row fits, or two fit equally, choose Linux Mint. It is the answer least likely to disappoint a beginner.

Three rules of thumb help with any distribution not listed here.

- **Prefer the popular.** A widely used distribution has more guides, more answered questions and more tested hardware.
- **Prefer a long-term version.** On your first system, stability matters more than having the newest release.
- **Stay in one family while you learn.** The commands you practise on Mint work on Ubuntu and Debian too.

## Try before you commit

You do not have to decide from a table. Chapter 3 described the live USB, which runs Linux without installing anything. Use it to compare.

1. Pick two distributions from the table above.
2. Download both ISO files and write the first to your USB stick.
3. Start the PC from the stick and spend twenty minutes in it.
4. Write the second ISO to the stick and do the same.

While you are in each one, check four things.

- [ ] Do the Wi-Fi, sound, screen brightness and touchpad work?
- [ ] Can you find the settings and the file manager without help?
- [ ] Does the app store list the programs you need?
- [ ] Do you like looking at it?

The one that passes more checks is your distribution. If both pass, take the one you enjoyed more.

The decision is also not permanent. Your documents, pictures and music are ordinary files. Back them up, install a different distribution, and copy them back.

## Common misunderstandings

**"There is one best distribution."** There is a best one for a particular person and machine. Someone who says otherwise is describing their own taste.

**"Each distribution runs different apps."** Almost all Linux software runs on all of them. What changes is the command that installs it. App formats called Flatpak and Snap go further and work the same on every family.

**"Beginner distributions are weaker."** Mint and Ubuntu use the same kernel and the same tools as Arch. They only arrive already set up. Professionals use them every day.

**"If I choose wrong, I am stuck."** Switching costs an hour and a backup. Many Linux users try several distributions in their first year.

**"A rarer distribution makes me a more serious user."** A small project has fewer people testing it and fewer people to ask when something breaks. Popularity is a practical advantage.

## Summary

- Every distribution shares the Linux kernel. They differ in package manager, desktop environment, release style and defaults.
- Most distributions belong to one of three families: Debian, Red Hat or Arch.
- Linux Mint and Ubuntu are the easiest starting points. Fedora offers newer software, Debian offers stability, and Arch teaches how the system is built.
- The desktop environment is a separate choice, and most distributions offer several.
- A fixed release changes little between versions. A rolling release updates continuously.
- Test two distributions from a live USB before installing. The choice can always be changed later.

## Exercises

1. Name the family of each of these: Linux Mint, Fedora, Manjaro, Ubuntu.
2. What is the difference between a fixed release and a rolling release? Which would you put on a relative's PC, and why?
3. Use the table in "How to choose" to pick a distribution for yourself. Write one sentence explaining your choice.
4. Visit the websites of two distributions. Which desktop environments does each one offer?
5. Start one distribution from a live USB and go through the four checks in this chapter. Did it pass?
6. A friend says Arch is the only real Linux. How would you answer?

The next chapter turns to the operating system most PCs are sold with: how Windows works, and how to install it.

# Chapter 5: How Windows works and how to install it

## The operating system that comes in the box

Most people never install Windows. It is already on the PC when they buy it, and it stays there until the PC is replaced.

Sooner or later, though, many people need to. The PC has grown slow and cluttered after years of use. A new, faster disk has replaced the old one. A PC has been built from parts and has nothing on it at all. Or the PC is being sold, and everything personal must be wiped.

Installing Windows yourself takes about an hour and uses the same method you learned for Linux: a USB stick and a boot menu. This chapter first explains how Windows works inside, then walks through a clean install.

## Where Windows came from

The first Windows, released in 1985, was not a full operating system. It was a program that drew windows and menus on top of MS-DOS, an older Microsoft system controlled entirely by typed commands. Through Windows 95 and Windows 98, DOS was still underneath.

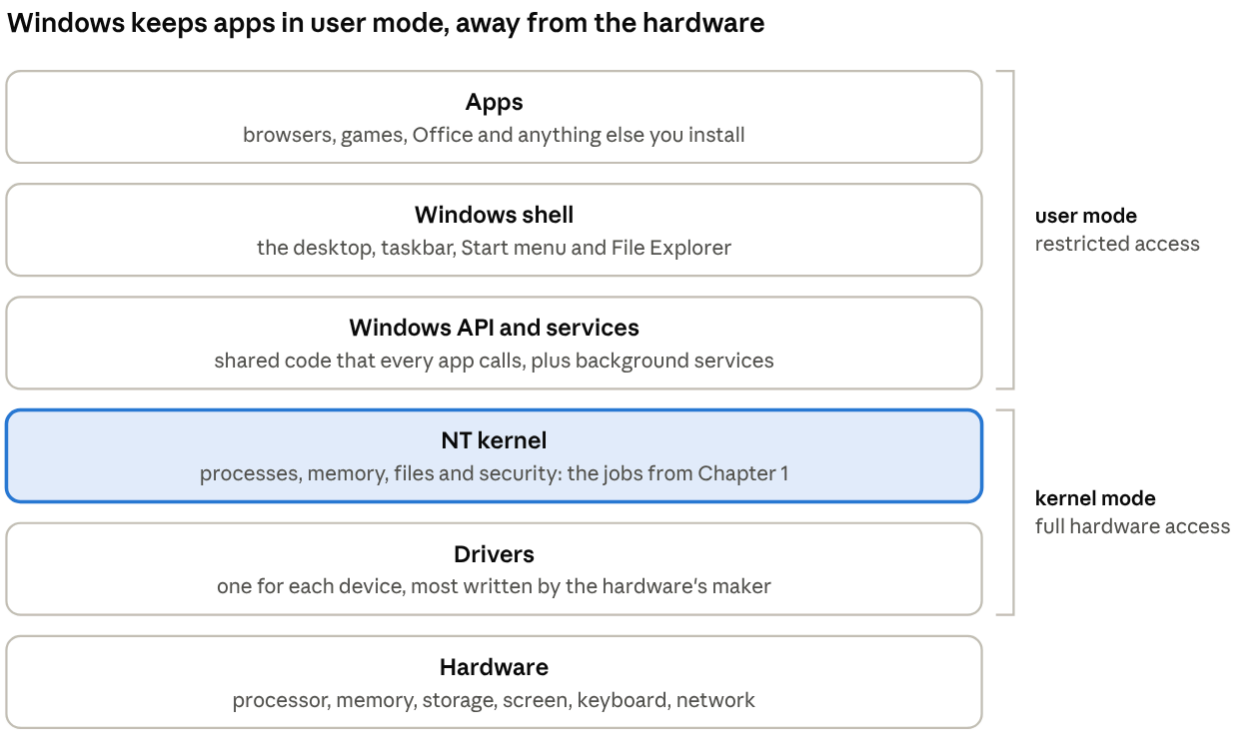
In 1993 Microsoft released a second, separate system called Windows NT, built around a completely new kernel by a team led by Dave Cutler. NT was designed for reliability and security, and at first it was sold to businesses.

Windows XP, in 2001, moved home users onto the NT kernel as well. Every version since, including Windows 10 in 2015 and Windows 11 in 2021, is a descendant of NT. That is why Chapter 2 found the number 10.0 hiding inside Windows 11: under the new look, it is the same family.

Unlike Linux, Windows is closed source. Only Microsoft can read and change its code, and you pay for a licence to use it.

## How Windows is put together

Windows has the same layers as any operating system, and all of them come from one company. The key line in the picture is the one between user mode and kernel mode, which Chapter 1 introduced.




The layers of Windows, split into user mode and kernel mode

Everything you see and start yourself runs in user mode, where a crash harms only that one program. The kernel and the drivers run in kernel mode, where a fault can stop the whole PC. The well-known blue error screen appears when something in kernel mode fails, and a faulty driver is the usual cause.

Unlike on Linux, the parts cannot be swapped. There is one shell, one set of system services and one kernel, and they are designed, tested and updated together.

## Five ideas you meet on day one

Windows does the same five jobs as Linux, but it organises them in its own way. These five habits explain most of what you see.

### 1. Drive letters

Windows gives every disk, and every part of a disk, a letter. The disk that holds Windows is almost always `C:`. A USB stick might appear as `D:` or `E:`. A path starts with the letter and uses backslashes.

| Folder | What it holds |
|---|---|
| `C:\Users` | One folder per user, with Documents, Downloads, Pictures and settings |

| Folder | What it holds |
|---|---|
| `C:\Windows` | The operating system itself |
| `C:\Program Files` | Installed programs |
| `C:\ProgramData` | Settings and data shared by all users |

Your own files live in `C:\Users\yourname`. Leave the Windows folder alone. Deleting files there to save space is a common way to break a PC.

## 2. The registry

Linux keeps its settings in text files. Windows keeps most of them in one large database called the registry. It stores thousands of settings for Windows itself, for hardware and for installed programs.

You change the registry all the time without knowing it, because the Settings app writes to it for you. A tool called Registry Editor lets you edit it by hand. Use it only when a trusted guide tells you exactly what to change, because a wrong value can stop Windows from starting.

## 3. Accounts and the permission prompt

Windows has two kinds of account. A standard account can use the PC but cannot change the system. An administrator account can.

Even an administrator works with limited rights most of the time. When a program tries to change the system, the screen dims and a box asks "Do you want to allow this app to make changes to your device?". This is User Account Control, and it plays the same role as `sudo` on Linux. Read the name of the program before you click Yes.

## 4. Installing software

There are three ways to add a program.

- **An installer from a website.** You download a file ending in `.exe` or `.msi` and run it. This is the traditional way. Download only from the maker's own site.
- **The Microsoft Store.** An app store built into Windows, with automatic updates.
- **The winget command.** A package manager included in Windows 10 and 11. Typing `winget install VideoLAN.VLC` in a terminal installs the VLC player, much as `apt` does on Linux.

### 5. Windows Update and drivers

Windows Update delivers security fixes, usually on the second Tuesday of each month, and a larger feature update about once a year. It also fetches most drivers automatically.

When a device does not work, open Device Manager. It lists every piece of hardware and marks any device that lacks a driver with a yellow warning sign.

### Windows and Linux side by side

| Idea | Linux | Windows |
|---|---|---|
| Kernel | Linux | NT |
| Where files start | `/` | `C:\` |
| Your files | `/home/yourname` | `C:\Users\yourname` |
| System settings | Text files in `/etc` | The registry |
| Permission to change the system | `sudo` | User Account Control |
| Installing software | A package manager such as `apt` | Installers, the Store or `winget` |
| Listing files in a terminal | `ls` | `dir` |
| Source code | Open | Closed |

## Editions, requirements and the licence

Before installing, you need to know three things: which edition you are entitled to, whether the PC can run it, and how Windows will be activated.

**Editions.** Windows 11 comes mainly as Home and Pro. Home is what most PCs are sold with. Pro adds tools for businesses, such as full control of disk encryption, the ability to be controlled remotely, and built-in virtual machines. For home use, Home is enough.

**Requirements.** Windows 11 asks more of a PC than Windows 10 did.

| Part | Minimum for Windows 11 |
|---|---|
| Processor | 64-bit, 2 or more cores, 1 GHz or faster, on Microsoft's supported list |
| Memory | 4 GB |

| Part | Minimum for Windows 11 |
|---|---|
| Storage | 64 GB |
| Firmware | UEFI with Secure Boot |
| Security chip | TPM version 2.0 |

The last two rows stop many older PCs. Microsoft's free PC Health Check app tells you whether a PC passes. A PC that fails is a good candidate for Linux, as Chapter 3 described.

**The licence.** Windows must be activated to prove it is a paid copy. Older PCs came with a 25-character product key. Most PCs today have a digital licence instead, which Microsoft's servers link to that PC's hardware. If Windows was activated on the PC before, the same edition activates again by itself after a reinstall, with no key to type.

## Three ways to install or reinstall

A full install from a USB stick is not always necessary. Start with the gentlest option that solves your problem.

| Way | What happens | What you keep | Good for |
|---|---|---|---|
| Reset this PC | Windows reinstalls itself from inside Settings, with no USB stick | Your files if you choose "Keep my files"; apps are removed | A slow or misbehaving PC that still starts |
| Repair install | You run the Windows installer from inside Windows and it replaces the system files | Files, apps and settings | Fixing damaged system files |
| Clean install | You start the PC from a USB stick, erase the disk and install fresh | Nothing | A new disk, a self-built PC, a PC that will not start, or one being sold |

Reset this PC lives in Settings, under System, then Recovery. Try it first when the PC still works.

The rest of this chapter covers the clean install, because it is the method that works in every situation, including an empty disk.

## Before you install

A clean install erases the disk. Gather everything on this list before you start.

- [ ] A backup of every file you care about, on another drive or in the cloud
- [ ] A USB stick of 8 GB or more, which will be wiped
- [ ] A second, working PC to prepare the stick on, if the target PC does not start
- [ ] The edition of Windows the PC is licensed for, and its product key if it has one
- [ ] Your Microsoft account email and password
- [ ] Your Wi-Fi password
- [ ] A list of the programs you will want to install again
- [ ] About an hour, and the charger plugged in if it is a laptop

If the disk is encrypted with BitLocker, the files on it become unreadable once you erase it. That is expected on a clean install, and it is another reason the backup must be finished first.

## A clean install, step by step

These steps are for Windows 11. The wording on screen changes slightly between versions, but the order stays the same.

1. **Get the Media Creation Tool.** On a working PC, open Microsoft's "Download Windows 11" page and download the tool under "Create Windows 11 Installation Media". Use only Microsoft's own website.
2. **Make the USB stick.** Run the tool, accept the terms, choose "USB flash drive" and pick your stick. The tool downloads Windows and writes it to the stick, which erases the stick. This takes a while on a slow connection.
3. **Start the PC from the stick.** Plug it into the PC you want to install on and switch the PC on. Press the boot-menu key straight away, as in Chapter 3, and pick the USB stick.
4. **Begin Setup.** Choose your language and keyboard, then choose to install Windows.
5. **Deal with the product key.** Type the key if you have one. If Windows was activated on this PC before, choose "I don't have a product key" and it will activate by itself later. If you are asked for an edition, pick the one the PC is licensed for.
6. **Accept the licence terms.**
7. **Choose where to install.** This is the step that erases data, so read the screen twice. Setup lists the disks and the sections of each disk, called partitions. For a clean install, delete the partitions on the target disk until it shows only "Unallocated space", select that space and continue. If the PC has two disks, make sure you are working on the right one.

8. **Wait.** Setup copies files and restarts the PC several times. If it returns to the first Setup screen after a restart, pull out the USB stick and restart again.
9. **Answer the first-run questions.** Windows asks for your region, keyboard and Wi-Fi network, then asks you to sign in with a Microsoft account and create a PIN. It then shows a list of privacy settings. Read them and switch off the ones you do not want.
10. **Reach the desktop.** Windows spends a few minutes preparing, then shows the desktop. The install is done.

### If something goes wrong

- **"This PC can't run Windows 11."** The PC fails a requirement. Check in the firmware settings that Secure Boot and the TPM are switched on. If the PC has neither, it cannot run Windows 11 in the supported way.
- **No disks are listed in step 7.** Setup lacks a driver for the disk controller. Download the storage driver from the PC maker's support page onto the stick and use "Load driver".
- **No Wi-Fi networks appear in step 9.** Setup lacks a driver for the Wi-Fi chip. Connect with a network cable, or load the Wi-Fi driver from the PC maker's support page.
- **Windows says it is not activated.** Check that you installed the same edition the PC was licensed for. Home and Pro licences are not interchangeable.

## Your first hour

A fresh Windows works, but it is out of date and missing your programs. Five jobs finish the work.

1. **Run Windows Update until it finds nothing.** Open Settings and choose Windows Update. Install everything, restart, and check again. Repeat until it reports that you are up to date.
2. **Check the drivers.** Open Device Manager and look for yellow warning signs. Windows Update fixes most of them. For anything left, and for the graphics card, download the driver from the maker's website.
3. **Check activation.** Open Settings, then System, then Activation. It should say that Windows is activated.
4. **Leave the built-in protection on.** Windows Security includes an antivirus and a firewall, and both are on by default. Most home users need nothing more.
5. **Install your programs and restore your files.** Use the Microsoft Store, the makers' own websites, or `winget`. Then copy your files back from the backup.

## Common misunderstandings

**"Reinstalling Windows means buying it again."** The licence stays with the PC. A PC that was activated before activates again after a reinstall of the same edition.

**"Windows 11 is a brand-new operating system."** It is the newest member of the NT family that began in 1993. The look changed far more than the kernel underneath.

**"A slow PC always needs a clean install."** Often it needs fewer programs starting with Windows, more free disk space, or Reset this PC. Try those first, because a clean install costs you every installed program.

**"Registry cleaners make Windows faster."** They make no difference you could measure, and they can remove settings that Windows needs. Leave the registry alone.

**"I need to buy an antivirus."** The protection built into Windows is enough for most home users, provided updates stay switched on and software comes from trustworthy sources.

**"Windows and Linux cannot share a PC."** They can, as a dual boot. Install Windows first and Linux second, as Chapter 3 described, so that the Linux start menu can offer both.

## Summary

- Every modern Windows descends from Windows NT, first released in 1993. Windows is closed source and needs a paid licence.
- Apps run in user mode. The NT kernel and the drivers run in kernel mode and control the hardware.
- Files live under drive letters, and yours are in `C:\Users`. Most settings live in the registry.
- User Account Control asks before a program changes the system. Software comes from installers, the Microsoft Store or `winget`.
- Windows 11 needs a supported 64-bit processor, 4 GB of memory, 64 GB of storage, Secure Boot and a TPM 2.0 chip.
- There are three ways to reinstall: Reset this PC, a repair install and a clean install. Only the clean install erases everything.
- After installing, run Windows Update until it finds nothing, check the drivers and confirm activation.

## Exercises

1. Draw the layers of Windows from memory and mark which run in user mode and which in kernel mode.

2. Open Device Manager on a Windows PC. Does any device show a warning sign?
3. Find the Activation page in Settings. Is the PC activated with a product key or a digital licence?
4. Run the PC Health Check app, or read the requirements table, and decide whether a PC you know could run Windows 11.
5. Fill in the comparison table for a third system you know, such as macOS or Android.
6. A relative's PC is slow but still starts. Which of the three ways to reinstall would you suggest first, and why?

## Where to go from here

You began this book pressing a power button without knowing what happened next. You can now say what an operating system is, find out exactly which one any device runs, explain how Linux and Windows are built, choose a Linux distribution, and install either system yourself.

The best next step is practice. Put Linux on a spare PC or in a virtual machine, break it, and install it again. Nothing teaches how an operating system works faster than setting one up with your own hands.