An operating system (OS), in its most general sense, is software that allows a user to run other applications on a computing device. While it is possible for a software application to interface directly with hardware, the vast majority of applications are written for an OS, which allows them to take advantage of common libraries and not worry about specific hardware details.

The operating system manages a computer's hardware resources, including:

* Input devices such as a keyboard and mouse
* Output devices such as display monitors, printers and scanners
* Network devices such as modems, routers and network connections
* Storage devices such as internal and external drives

The OS also provides services to facilitate the efficient execution and management of, and memory allocations for, any additional installed software application programs.

The OS consists of many components and features. Which features are defined as part of the OS varies with each OS. However, the three most easily defined components are:

* Kernel: This provides basic-level control over all of the computer hardware devices. Main roles include reading data from memory and writing data to memory, processing execution orders, determining how data is received and sent by devices such as the monitor, keyboard and mouse, and determining how to interpret data received from networks.
* User Interface: This component allows interaction with the user, which may occur through graphical icons and a desktop or through a command line.
* Application Programming Interfaces: This component allows application developers to write modular code.

There are a few common operating systems available:

* Mac OS X
* Linux
* Windows
* Android (based on Linux)
* iOS

Different operating systems run on different types of hardware and are designed for different types of applications. For example, iOS is designed for iPhones and iPad tablets, while Mac desktops and laptops use macOS. Your computer or smartphone comes equipped with an OS, but you can install another one in some cases.

## Microsoft Windows

Microsoft Windows has existed in one form or another since 1985, and it remains the most popular operating system for home and office computers. Its latest versions, including Windows 10, are also used on some tablets, and the OS is used on some web and number-crunching server computers as well. Computers from a wide variety of manufacturers can use Windows.

* Initial versions of Windows worked with an earlier Microsoft operating system called MS-DOS, providing a modern graphical interface on top of DOS's traditional text-based commands. Signature features of Microsoft Windows's user interface include windows themselves – rectangle-shaped, on-panel screens that represent individual applications. The Windows Start menu has helped generations of users find programs and files on their devices.
* Efforts to use versions of the Windows OS for smartphones have been less successful.

## Apple iOS

* Apple's iOS is one of the most popular smartphone operating systems, second only to Android. It runs on Apple hardware, including iPhones, iPad tablets and iPod Touch media players.
* Signature features of iOS include the App Store where users buy apps and download free software, an emphasis on security including strong encryption to limit what unauthorized users can extract from the phone, and a simple, streamlined interface with minimal hardware buttons.

## Google's Android OS

* Android is the most popular operating system in the world judging by the number of devices installed. Largely developed by Google, it's chiefly used on smartphones and tablets. Unlike iOS, it can be used on devices made by a variety of different manufacturers, and those makers can tweak parts of its interface to suit their own needs.
* Users can download custom versions of the operating system because large portions of it are open source, meaning anyone can legally modify it and publish their own. However, most people prefer to stick with the version that comes on their devices.
* Android, like iOS, comes with an application and media store called the Play Store built by Google. Some phone manufacturers and other organizations also offer their own stores to install software and media.

## Apple macOS

* Apple's macOS, successor to the popular OS X operating system, runs on Apple laptops and desktops. Based in part on the historic family of Unix operating systems dating back to research in the 1960s at AT&T's Bell Labs, macOS shares some features with other Unix-related operating systems including Linux. While the graphical interfaces are different, many of the underlying programming interfaces and command line features are the same.
* Signature elements of macOS include the dock used to find programs and frequently used files, unique keyboard keys including the Command key, and the stoplight-colored buttons used to resize open program windows. MacOS is known for its user-friendly features, which include Siri, a natural-voice personal assistant, and FaceTime, Apple's video-calling application.

## Linux Operating System

* Unlike many other operating systems, development on Linux isn't led by any one company. The operating system was created by Finnish programmer Linus Torvalds in 1991. Nowadays, programmers from all over the world collaborate on its open source code and submit tweaks to the central kernel software and other programs.
* A wide assortment of commercial and open source software is available for Linux, and various Linux distributions provide custom user interfaces and tools for installing software onto machines running the operating system. A favorite of many programmers, Linux is widely used on corporate and scientific servers, including cloud computing environments. Linux can be run on a wide variety of hardware and is available free of charge over the internet.

**Objectives of Operating System**

The objectives of the operating system are −

* To make the computer system convenient to use in an efficient manner.
* To hide the details of the hardware resources from the users.
* To provide users a convenient interface to use the computer system.
* To act as an intermediary between the hardware and its users, making it easier for the users to access and use other resources.
* To manage the resources of a computer system.
* To keep track of who is using which resource, granting resource requests, and mediating conflicting requests from different programs and users.
* To provide efficient and fair sharing of resources among users and programs.

**Functions of the operating system**

The operating system performs several key functions:

* **interface** - provides a user interface so it is easy to interact with the computer
* **manages** the CPU - runs applications and executes and cancels processes
* **multi-tasks** - allows multiple applications to run at the same time
* **manages** memory - transfers programs into and out of memory, allocates free space between programs, and keeps track of memory usage
* **manages** peripherals - opens, closes and writes to peripheral devices such as storage attached to the computer
* **organises** - creates a file system to organise files and directories
* **security** - provides security through user accounts and passwords
* **utilities** - provides tools for managing and organising hardware

**User interface**

The OS provides a user interface (UI), an environment for the user to interact with the machine. The UI is either graphical or text-based.

**Graphical user interface (GUI)**

The OS on most computers and smartphones provides an environment with tiles, icons and/or menus. This type of interface is called the graphical user interface (GUI) because the user interacts with images through a mouse, keyboard or touchscreen.

**Command line interface (CLI)**

An OS also provides a method of interaction that is non-graphical, called the command line interface (CLI). This is a text-only service with feedback from the OS appearing in text. Using a CLI requires knowledge of the commands available on a particular machine.

Advantages of using the command line include:

* a **faster** way to get tasks done
* it is **more flexible** than a GUI
* it uses **less** memory

Some games, such as Minecraft, also make use of a command line tool which allows the user to bypass the main interface and alter the game’s mechanics or environment.

**Managing the CPU**

The OS is used to run programs by clicking on an icon, selecting the program from a menu, or typing in an instruction at the command line.

When the OS runs a piece of software it has to find the program files on the storage drive, load them into main memory, and instruct the CPU to start executing the program from the beginning.

In each case, the OS performs the same sequence of steps:

1. the program code is found on the storage drive
2. a section of RAM is reserved for the program and space is allocated for the program's data
3. the program code is copied from storage into the reserved space in the memory
4. the CPU program counter is set to the memory location of the first instruction in the program, and execution begins

**Multitasking**

The OS makes it possible to run several programs at once. Several programs can be stored in RAM at the same time, however only one program at a time is processed by the CPU. Programs can be in one of three states:

* running
* waiting
* runnable

Only one process can be running at any one time. CPUs are extremely fast, so if a program is processed for even a short time it can do quite a lot. The OS decides the best way to swap between running, runnable and waiting processes. It controls which process is being executed by the CPU at any point in time, and shares access to the CPU between processes. The job of working out when to swap processes is known as **scheduling**.

Swapping happens so fast that it appears that all processes are running at the same time. When there are too many processes, or some of them are making the CPU work especially hard, it can look as though some or all of them have stopped.

**Managing memory**

The OS manages how main memory is used. It decides:

* how memory is shared between processes
* what happens when there is not enough main memory to get the job done

Different processes running at the same time must not interfere with one another. This means they have to use different parts of the computer’s memory.

The OS handles the transfer of data between processes. This is done by setting aside areas in memory where data values can be shared.

The OS uses buffering to set aside memory for the temporary storage of data. A process may output data and leave it in the buffer. This means that processes can each get on with their jobs at their own rate. It is only when the buffer is full that processes will need to wait.

Buffers are also used when you stream online content.

**Peripheral devices**

Each peripheral is programmed with its own machine code. Each has its own rules that dictate how it transmits data values between the computer and the device. These rules make up a protocol for controlling and communicating with the device.

The protocol dictates the structure, speed and timing of the messages that are sent and received.

**Drivers**

The OS uses programs called **device drivers** to manage connections with peripherals.

A device driver:

* handles the translation of requests between a device and the computer
* defines where a process must put outgoing data before it can be sent, and where incoming messages will be stored when they are received
* wakes up the device when it is needed and put it back to sleep when it is not

An OS will have generic device drivers to enable it to connect to most common peripherals. Some peripherals, however, will have their own drivers that need to be installed before use.

Peripherals that use the same protocol may be controlled by the same driver. If a number of identical game controllers are plugged in, each device will store its data in a different place so they do not interfere with each other.

**File systems**

The OS organises files on the storage drive.

In order to retrieve data from a file, the computer needs to know:

* which storage device it is held on
* where it is stored on the device
* how files are organised on the device
* how much data is in it
* the protocol needed to communicate with it

It is the job of the OS to maintain this information for other programs, and it does this by providing a file system. The purpose of a file system is to provide programs with a uniform way of storing and retrieving data.

**File management**

The OS manages how data is organised into files. This makes it easier for the user to see files using programs like the Windows File Explorer or Mac OS X Finder. The OS organises where and how files are stored, deleted, read, found and repaired. It detects errors such as missing disks or incorrect file names, and informs the user that errors have occurred.

Each file has a unique name and the OS maintains a set of **look-up** tables that relate file names to locations on storage drives.

**Hierarchies**

File systems work in a similar way to the way that libraries organise books. Folders and directories correspond to different sections of the library. Inside each folder can be other folders (sub-sections within a subject) and files (the books themselves). If you need to access a specific file you just need to know how to look for it in the index which describes where each file is located.

Most file systems are hierarchical and contain directories that contain lists of other files. Hierarchical file systems usually have a special directory at the root. It can be imagined to be similar to a tree - the branch points are directories that lead to other files. Data files are at the ends of branches and these include files containing program code.

File systems can become corrupt if a computer is turned off before a program is copied to a new location.

**Utilities**

The OS uses applications called **utilities** which allow the user to manage the computer. There are many different utility programs and they may vary across operating systems. They are often accessed via a special menu or control panel in the OS.

**Maintenance utilities**

These include:

* **Backup** - This allows the user to restore the system to a previous state which is saved as a backup. This is only usually used if a system malfunctions.
* **Disk cleaner** - The storage drive is divided into a number of clusters. The table of contents serves as an address book, keeping a record of each file and the clusters used to store that file. When a file is deleted, the address to the location on the disk is removed.
* **Disk defragmentation** - When files are deleted, unused clusters become available for reuse. These can end up being distributed across a drive, especially if the original files were small. If a large file is then written to a drive, its data could be spread across different clusters leading to file fragmentation. Defragmentation involves rearranging the information on a disk so that files appear in continuous sequences of clusters. This will improve file access times. Most modern operating systems run this process automatically.
* **Formatting** - Storage drives need to be formatted to be compatible with an OS. The OS usually formats storage media when it is connected to the computer. It is often the case that a storage drive cannot be compatible with both Windows and Mac OS X.

**Security utilities**

These include:

* **user accounts** - allow the user to allocate specific users and protects personal files and programs from unauthorised access.
* encryption - can encrypt data when it is stored, or whenever it is transmitted over a network.
* **anti-virus software** - detects and blocks viruses.
* **firewall** - can be used to filter between trusted and untrusted networks and prevent programs from communicating through the use of ports.

**How to install your operating system**

**Step One: Edit your BIOS:** When you first start up your computer, it'll tell you to press a key to enter setup, usually DEL. This takes you to the setup of your Basic Input/Output System, or BIOS. Here, you can configure some of the lowest-level aspects of your new machine. You may not actually need to edit any of these settings, but it's a good idea to go through, get acquainted with them, and make sure everything's in good order before moving on

Note that the BIOS will be a little bit different on different brands of motherboard

First, make sure that everything's been installed correctly. If you have a System Information page in your BIOS, head there and make sure the amount of RAM listed is the same amount you put in. If it isn't detecting all of your RAM, some of it might not be seated correctly, so go back and fix that before continuing.

Lastly, find the "Boot Order" or "Boot Priority" page. Make sure your DVD drive is the first drive on the list (or your USB drive if you're installing from a flash drive), and that the hard drive you'll be installing to is second.

**Step Two: Install Windows**

Before you install Windows, make sure you have the optimal version for your system. That is, if you have more than 4GB of RAM, you'll want to use 64-bit Windows instead of the standard 32-bit—that will allow your system to take advantage of your entire RAM. You can read more about this in our guide to 64-bit vs. 32-bit operating systems.

Once you've got the right version of Windows, grab the installation DVD (or flash drive, if that be the case) and pop it in. Start up your computer and it should automatically boot into the Windows installer. If you ever get a "Press any key to boot from CD" option, make sure to hit a key on your keyboard to continue.

Once the installer loads, hit the "Install Now" button, accept the terms of use, and choose "Custom (advanced)" when asked what type of installation you want. Find your primary hard drive (if you have more than one), click on the "Unallocated Space" partition, and hit Next. Windows should start installing.

If you aren't using a brand new drive, you may have to format it first. Click on the currently-used partition, click "Drive options (advanced)", and then hit "Format". It should format the drive to be Windows-compatible, after which you can hit next and let the installation run.

From there, the rest is just a waiting game. Leave your computer alone to do its thing. It'll copy all the necessary files to your disk and reboot a number of times in the process. You'll know you're done when you hear the familiar startup chime and boot into the default Windows 7 desktop.

Windows key ready (after payment)

**Step Three: Install Your Drivers**

The last thing you need to do before you actually use your computer is install your drivers. If your Ethernet or Wi-Fi works out-of-the-box, Windows may find most or all of your drivers for you. If not, you'll need to pop in the CD that came with your motherboard to install the Ethernet or Wi-Fi drivers you need to access the internet.

* Update drivers online
* download the drivers manually from the manufacturer's web site

**Step Four: Install Windows Updates**

The last thing you'll want to do is get Windows up to date. Chances are, you've already gotten a notification from Windows Update at this point, but if not, head into your Start Menu, go to Programs, and hit Windows Update. Install all the updates it gives you, and reboot your computer. Check for updates again and it'll have a whole new slew of them for you. You'll have to do this quite a few times, but eventually it should stop serving you notifications and you'll be all up to date. When you are, you're ready to actually start using your computer.

**Windows booting process**

When you turn on the power to a computer, the first program that runs is usually a set of instructions kept in the computer's read-only memory (ROM). This code examines the system hardware to make sure everything is functioning properly. This **power-on self test** (POST) checks the CPU, memory, and basic input-output systems (BIOS) for errors and stores the result in a special memory location. Once the POST has successfully completed, the software loaded in ROM (sometimes called the BIOS or **firmware**) will begin to activate the computer's disk drives. In most modern computers, when the computer activates the hard disk drive, it finds the first piece of the operating system: the **bootstrap loader**.

The bootstrap loader is a small program that has a single function: It loads the operating system into memory and allows it to begin operation. In the most basic form, the bootstrap loader sets up the small driver programs that interface with and control the various hardware subsystems of the computer. It sets up the divisions of memory that hold the operating system, user information and applications. It establishes the data structures that will hold the myriad signals, flags and semaphores that are used to communicate within and between the subsystems and applications of the computer. Then it turns control of the computer over to the operating system.

**Common symptoms**

**BSOD**

The blue screen of death is a Windows stop error that clearly points onto hardware malfunction or spoilt device drivers. One may decide to start in safe mode or carry out some System Restore.

**Failure to boot**

Failure of one's computer to boot can be as a result of a corrupt operating system or some possible changes in one's system's boot order. One can go to the BIOS setup and try to look at the boot sequence. If that does not work, then one should consider reinstalling the Windows operating system since the problem could be far much serious.

**Improper shutdown**

At times, one's computer may shutdown improperly due to instances of power loss or crushing. In case of such an incidence, restarting the computer might not take one directly to Windows and therefore one should run Windows Error Recovery which automatically checks the file system and drives for any problems. With that, all system files are placed in their right positions and Windows now starts without any problems.

**Spontaneous shutdown/restart**

This is a problem that mostly occurs in instances where one's computer keeps looping over the start-up process where it appears to be starting and then restarts. In such a case, the first step should be to try and establish the occurrence point of the problem either in the course of the BIOS check where the Power on Self-test is undergoing among some other possible reasons. Once that is established, then one can easily decide if the problem is hardware related or related to the Windows configuration.

**RAID not detected during installation**

This is an error message that tends to appear when one are installing an operating system and some of the files required are not found. In such a case, one might open the registry in safe mode to check for the files or restart the installation process with a new installation media.

**Device fails to start**

Failure of one's device can be very frustrating. In most cases, it normally results from a computer crush or some other hardware related problems. In such a case, try to carry out hardware diagnostics so as to try and establish the problem. One can also decide to boot the computer from a Windows installation media to try and get one's computer to restart.