The key basic functions of any operating system are as following

1. Coordinate hardware components: An OS enables coordination of hardware components. Each hardware device speaks a different language, but the operating system can talk to them through the specific translational software called device drivers. These drivers interact between the other software and the hardware.
2. Provide an interface between user and hardware: An OS provides an interface between the user and machine. This interface can be a graphical user interface (GUI) in which users click on screen elements to interact with the OS or a command-line interface (CLI) in which users enter commands to tell the OS to do things.
3. Provide an environment for software to function: An OS provides the environment for application software to function. In GUI operating systems such as Windows and MacOS, applications run within a consistent, graphical desktop environment.
4. Provide a structure for data management: An OS displays structure/directories for data management. We can view file and folder listings and manipulate on those files and folders such as move, copy, rename, delete, and many others.
5. Monitor system health and functionality: OS monitors the health of our system’s hardware; giving us an idea of how well (or not), it is performing. We can see how busy our CPU is, or how quickly our hard drives retrieve data, or how much data our network card is sending etc. and it also monitors system activity for malware.

Within the broad family of operating systems, there are generally four types, categorized based on the types of computers they control and the sort of applications they support. The categories are:

* **Real-time operating system** (RTOS) - Real-time operating systems are used to control machinery, scientific instruments and industrial systems. An RTOS typically has very little user-interface capability, and no end-user utilities, since the system will be a "sealed box" when delivered for use. A very important part of an RTOS is managing the resources of the computer so that a particular operation executes in precisely the same amount of time, every time it occurs.
* **Single-user, single task** - As the name implies, this operating system is designed to manage the computer so that one user can effectively do one thing at a time. The Palm OS for Palm handheld computers is a good example of a modern single-user, single-task operating system.
* **Single-user, multi-tasking** - This is the most commonly used type of operating system on desktop & laptop computers today. Microsoft's Windows and Apple's MacOS platforms are both examples of operating systems that will let a single user have several programs in operation at the same time. For example, it is entirely possible for a Windows user to be writing a note in a word processor while downloading a file from the Internet.
* **Multi-user** - A multi-user operating system allows many different users to take advantage of the computer's resources simultaneously. The operating system must make sure that the requirements of the various users are balanced, and that each of the programs they are using has sufficient and separate resources so that a problem with one user does not affect the entire community of users. UNIX, VMS and mainframe operating systems, such as *MVS*, are examples of multi-user operating systems.

Application software is a program or group of programs designed for end users. While system software consists of programs that interact with computers at a basic level, application software resides above system software and includes applications such as database programs, word processors and spreadsheets. Application software may be bundled with system software or published alone.

The most common application software programs used by millions every day and include:

* Microsoft suite of products (Office, Excel, Word, PowerPoint, Outlook, etc.)
* Internet browsers like Firefox, Safari, and Chrome