**ITE213**

**Platform Technologies**

**UNIT 1. Operating System**

1. **General Learning Outcome:**

Discuss the Emerging technologies and current trends in Web and Mobile development.

Specific Learning Outcomes:

* Identified what is operating system and its functions
* Identified the popular operating system
* Differentiate the advantages of operating system to others

1. **Pre-Test**

* What is an OS?
* What are its key functions?
* The evaluation of OS?
* What are the popular types of OS?

1. **Content**

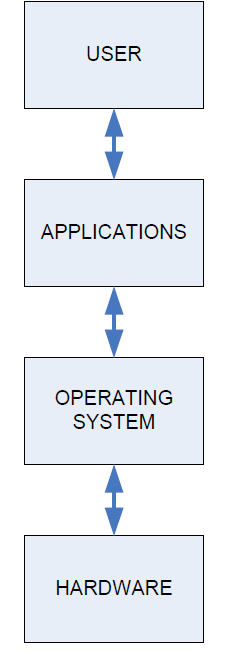
**1 What is an Operating System?**

Computer System = Hardware + Software

Software = Application Software + System Software(OS)

An Operating System is a system Software that acts as an intermediary/interface between a user of a computer and the computer hardware.

* Operating system goals:
  + Execute user programs and make solving user problems easier
  + Make the computer system convenient to use
  + Use the computer hardware in an efficient manner

**The Structure of Computer Systems**

* Accessing computer resources is divided into *layers*.
* Each layer is isolated and only interacts directly with the layer below or above it.
* If we install a new hardware device
  + No need to change anything about the user/applications.
  + However, you do need to make changes to the operating system.
  + You need to install the device drivers that the operating system will use to control the new device.
* If we install a new software application
  + No need to make any changes to your hardware.
  + But we need to make sure the application is supported by the operating system
  + user will need to learn how to use the new application.
* If we change the operating system
  + Need to make sure that both applications and hardware will compatible with the new operating system.

**Computer Architecture**

Special busses (roads) connecting all input/output devices to motherboard.

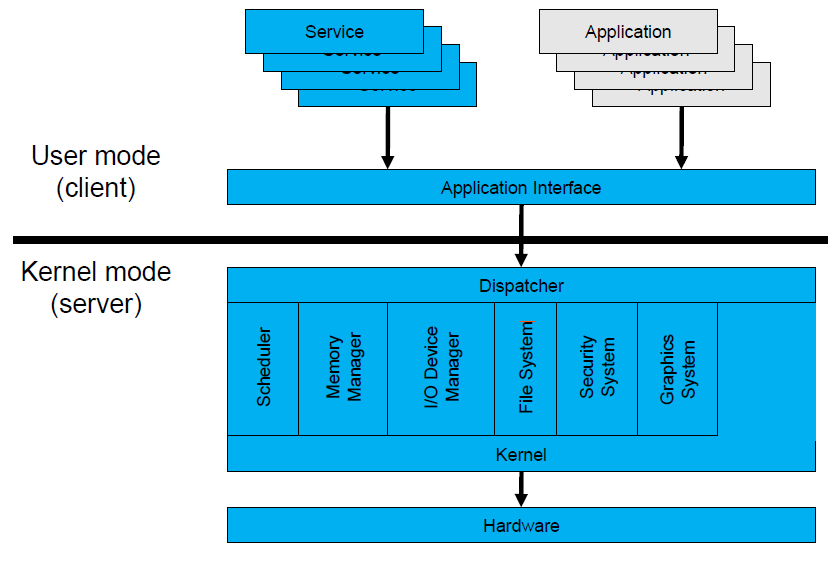
**CPU – Central Processing Unit**

* This is the brain of your computer.
* It performs all of the calculations.
* In order to do its job, the CPU needs commands to perform, and data to work with.
* The instructions and data travel to and from the CPU on the system bus.
* The operating system provides rules for how that information gets back and   
   forth, and how it will be used by the CPU.

**RAM – Random Access Memory**

* This is like a desk, or a workspace, where your computer temporarily stores all of the information (data) and instructions (software or program code) that it is currently using.
* Each RAM chip contains millions of address spaces.
* Each address space is the same size, and has its own unique identifying number (address).
* The operating system provides the rules for using these memory spaces, and controls storage and retrieval of information from RAM.
* Device drivers for RAM chips are included with the operating system.

***Problem:*** *If RAM needs an operating system to work, and an operating system needs RAM in order to work, how does your computer activate its RAM to load the operating system?*

**Operating System Mode**

* The *User Mode* is concerned with the actual interface between the user and the system.
* It controls things like running applications and accessing files.
* The *Kernel Mode* is concerned with everything running in the background.
* It controls things like accessing system resources, controlling hardware functions and processing program instructions.
* System calls are used to change mode from User to Kernel.

**Kernel**

* Kernel is a software code that reside in central core of OS. It has complete control over system.
* When operation system boots, kernel is first part of OS to load in main memory.
* Kernel remains in main memory for entire duration of computer session. The kernel code is usually loaded in to protected area of memory.
* Kernel performs it’s task like executing processes and handling interrupts in kernel space.
* User performs it’s task in user area of memory.
* This memory separation is made in order to prevent user data and kernel data from interfering with each other.
* Kernel does not interact directly with user, but it interacts using SHELL and other programs and hardware.

**Kernel includes:-**

1. Scheduler: It allocates the Kernel’s processing time to various processes.
2. Supervisor: It grants permission to use computer system resources to each process.
3. Interrupt handler : It handles all requests from the various hardware devices which compete for kernel services.
4. Memory manager : allocates space in memory for all users of kernel service.

Kernel provides services for process management, file management, I/O management, memory management.

**System calls** are used to provide these type of services.

* **System call** is the programmatic way in which a computer program/user application requests a service from the kernel of the operating system on which it is executed.
* Application program is just a user-process. Due to security reasons , user applications are not given access to privileged resources(the ones controlled by OS).
* When they need to **do any I/O** or have **some more memory** or **spawn a process** or wait for **signal/interrupt,** it requests operating system to facilitate all these. This **request is made through System Call.**
* System calls are also called **software-interrupts.**

  
**Starting an Operating System(Booting)**

* BIOS will then look for a small sector at the very beginning of your primary hard disk called MBR.
* The MBR contains a list, or map, of all of the partitions on your computer‘s hard disk (or disks).
* After the MBR is found the Bootstrap Loader follows basic instructions for starting up the rest of the computer, including the operating system.
* In Early Kernel Initialization stage, a smaller core of the Kernel is activated.
* This core includes the device drivers needed to use computer‘s RAM chips.
* Power On Switch sends electricity to the motherboard on a wire called the *Voltage Good* line.
* If the power supply is good, then the BIOS (Basic Input/Output System) chip takes over.
* In Real Mode, CPU is only capable of using approximately 1 MB of memory built into the motherboard.
* The BIOS will do a Power-On Self Test (POST) to make sure that all hardware are working.

**BIOS -** BIOS firmware was stored in a ROM/EPROM (Erasable Programmable Read-Only Memory) chip known as **firmware** on the PC motherboard.

* BIOS can be accessed during the initial phases of the boot procedure by pressing del, F2 or F10.
* Finally, the firmware code cycles through all storage devices and looks for a boot-loader. (usually located in first sector of a disk which is 512 bytes)
* If the boot-loader is found, then the firmware hands over control of the computer to it.

**UEFI -** UEFI stands for Unified Extensible Firmware Interface. It does the same job as a BIOS, but with one basic difference: it stores all data about initialization and startup in an .efi file, instead of storing it on the firmware.

* This .efi file is stored on a special partition called EFI System Partition (ESP) on the hard disk. This ESP partition also contains the bootloader.
* UEFI was designed to overcome many limitations of the old BIOS, including:
  + UEFI supports drive sizes upto 9 zettabytes, whereas BIOS only supports 2.2 terabytes.
  + UEFI provides faster boot time.
  + UEFI has discrete driver support, while BIOS has drive support stored in its ROM, so updating BIOS firmware is a bit difficult.
  + UEFI offers security like "Secure Boot", which prevents the computer from booting from unauthorized/unsigned applications. This helps in preventing rootkits.
  + UEFI runs in 32bit or 64bit mode, whereas BIOS runs in 16bit mode. So UEFI  is able to provide a GUI (navigation with mouse) as opposed to BIOS which allows navigation only using the keyboard.

**Functions of Operating System**

**A *process* is a program in execution.**

A process needs certain resources, including CPU time, memory, files, and I/O devices to accomplish its task.

* Simultaneous execution leads to multiple processes. Hence creation, execution and termination of a process are the most basic functionality of an OS
* If processes are dependent, than they may try to share same resources. thus task of process synchronization comes to the picture.
* If processes are independent, than a due care needs to be taken to avoid their overlapping in memory area.
* Based on priority, it is important to allow more important processes to execute first than others.

**Memory management**

**Memory** is a large array of words or bytes, each with its own address.

It is a repository of quickly accessible data shared by the CPU and I/O devices. Main memory is a **volatile** storage device. When the computer made turn off everything stored in RAM will be erased automatically.

In addition to the physical RAM installed in your computer, most modern operating systems allow your computer to use a *virtual memory system. Virtual memory allows your computer to use part of a permanent storage device (such as a hard disk) as extra memory.*

* The operating system is responsible for the following activities in connections with memory management:
  + Keep track of which parts of memory are currently being used and by whom.
  + Decide which processes to load when memory space becomes available.
  + Allocate and de-allocate memory space as needed.

**File Management**

A file is a collection of related information defined by its creator. *File systems provide the conventions for the encoding, storage and management of data on a storage device such as a hard disk.*

* + FAT12 (floppy disks)
  + FAT16 (DOS and older versions of Windows)
  + FAT32 (older versions of Windows)
  + NTFS (newer versions of Windows)
  + EXT3 (Unix/Linux)
  + HFS+ (Max OS X)

The operating system is responsible for the following activities in connections with file management:

1. File creation and deletion.
2. Directory creation and deletion.
3. Support of primitives for manipulating files and directories.
4. Mapping files onto secondary storage.
5. File backup on stable (nonvolatile) storage media.

**Device Management or I/O Management**

*Device controllers are components on the motherboard (or on expansion cards) that act as an interface between the CPU and the actual device.*

*Device drivers, which are the operating system software components that interact with the devices controllers.*

* A special device (inside CPU) called the **Interrupt Controller** handles the task of receiving interrupt requests and prioritizes them to be forwarded to the processor.
* **Deadlocks** can occur when two (or more) processes have control of different I/O resources that are needed by the other processes, and they are unwilling to give up control of the device.
* It performs the following activities for device management.
  + Keeps tracks of all devices connected to system.
  + Designates a program responsible for every device known as Input/output controller.
  + Decides which process gets access to a certain device and for how long.
  + Allocates devices in an effective and efficient way.
  + Deallocates devices when they are no longer required.

**Security & Protection**

The operating system uses password protection to protect user data and similar other techniques. It also prevents unauthorized access to programs and user data by assigning access right permission to files and directories. The owners of information stored in a multiuser or networked computer system may want to control use of that information, concurrent processes should not interfere with each other.

**User Interface Mechanism**

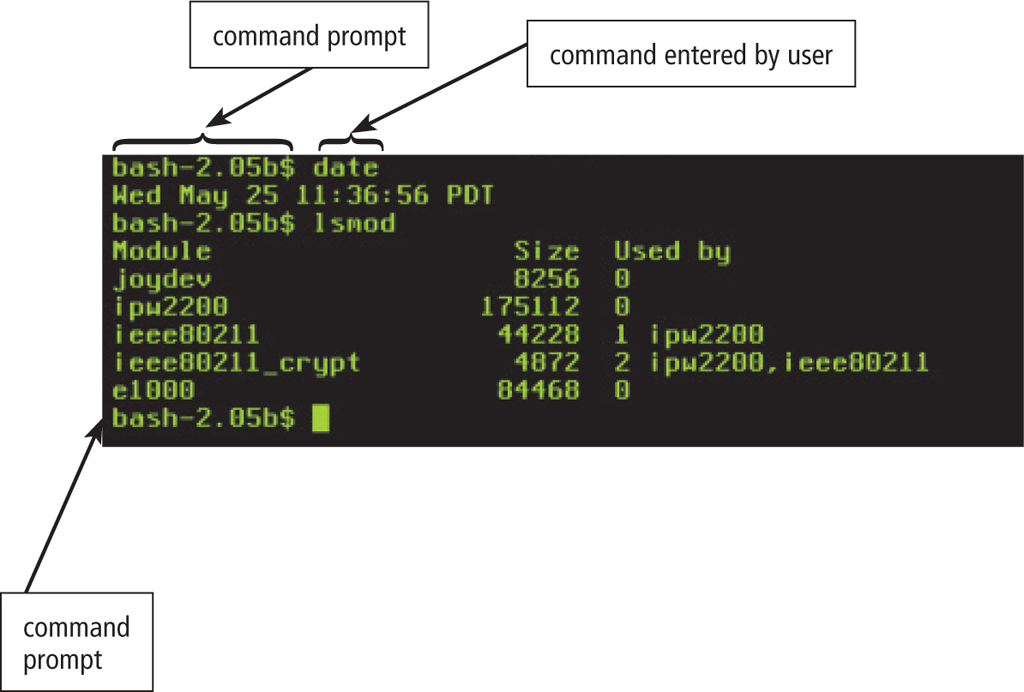
A **user interface** (**UI**) controls how you enter data and instructions and how information is displayed on the screen

**There are two types of user interfaces**

1. Command Line Interface
2. Graphical user Interface

**Command-line interface**

In a command-line interface, a user types commands represented by short keywords or abbreviations or presses special keys on the keyboard to enter data and instructions

****

**Graphical User Interface**

With a graphical user interface (GUI), you interact with menus and visual images



**History of Operating System**

**The First Generation (1940's to early 1950's)**

* + No Operating System
  + All programming was done in absolute machine language, often by wiring up plug-boards to control the machine’s basic functions.

**The Second Generation (1955-1965)**

* + First operating system was introduced in the early 1950's.It was called GMOS
  + Created by General Motors for IBM's machine the 701.
  + Single-stream batch processing systems

**The Third Generation (1965-1980)**

* + Introduction of multiprogramming
  + Development of Minicomputer

**The Fourth Generation (1980-Present Day)**

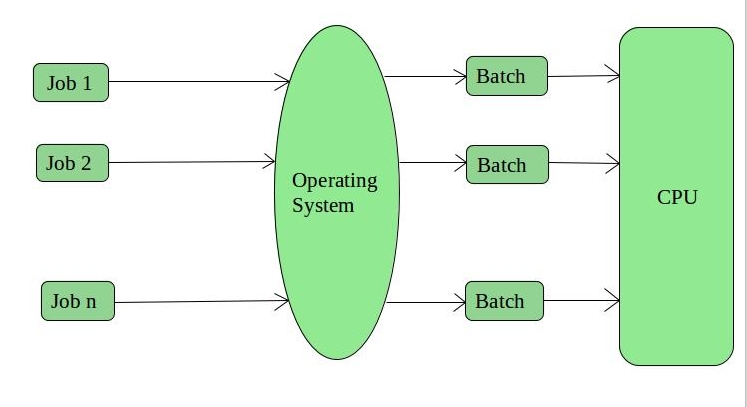
* + Development of PCs
  + Birth of Windows/MaC OS

**Types of Operating Systems**

1. Batch Operating System
2. Multiprogramming Operating System
3. Time-Sharing OS
4. Multiprocessing OS
5. Distributed OS
6. Network OS
7. Real Time OS
8. Embedded OS

**1. Batch Operating System**

* The users of this type of operating system does not interact with the computer directly.
* Each user prepares his job on an off-line device like punch cards and submits it to the computer operator
* There is an operator which takes similar jobs having the same requirement and group them into batches.



**Advantages of Batch Operating System:**

* + Processors of the batch systems know how long the job would be when it is in queue
  + Multiple users can share the batch systems
  + The idle time for the batch system is very less
  + It is easy to manage large work repeatedly in batch systems

**Disadvantages of Batch Operating System:**

* + The computer operators should be well known with batch systems
  + Batch systems are hard to debug
  + It is sometimes costly
  + The other jobs will have to wait for an unknown time if any job fails

**Examples of Batch based Operating System:**

IBM's MVS

**2. Multiprogramming Operating System:**

This type of OS is used to execute more than one jobs simultaneously by a single processor.

It increases CPU utilization by organizing jobs so that the CPU always has one job to execute.

Multiprogramming operating systems use the mechanism of job scheduling and CPU scheduling.

**3. Time-Sharing Operating Systems cont..**

**Advantages of Time-Sharing OS:**

* + Each task gets an equal opportunity
  + Fewer chances of duplication of software
  + CPU idle time can be reduced

**Disadvantages of Time-Sharing OS:**

* + Reliability problem
  + One must have to take care of the security and integrity of user programs and data
  + Data communication problem

**Examples of Time-Sharing Oss**

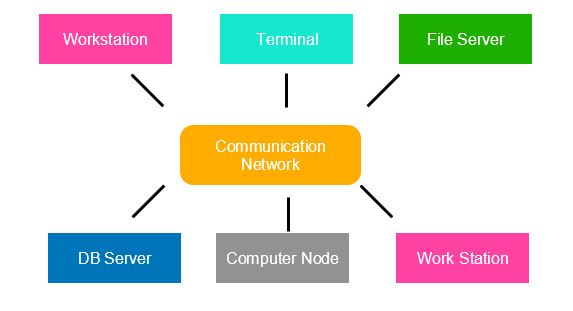
Multics, Unix, etc.

**4. Multiprocessor operating systems**

* Multiprocessor operating systems are also known as parallel OS or tightly coupled OS.
* Such operating systems have more than one processor in close communication that sharing the computer bus, the clock and sometimes memory and peripheral devices.
* It executes multiple jobs at the same time and makes the processing faster.
* It supports large physical address space and larger virtual address space.
* If one processor fails then other processor should retrieve the interrupted process state so execution of process can continue.
* Inter-processes communication mechanism is provided and implemented in hardware.

**5. Distributed Operating System**

* Various autonomous interconnected computers communicate with each other using a shared communication network.
* Independent systems possess their own memory unit and CPU.
* These are referred to as **loosely coupled systems**.
* Examples:- Locus, DYSEAC



**6. Network Operating System**

* These systems run on a server and provide the capability to manage data, users, groups, security, applications, and other networking functions.
* These types of operating systems allow shared access of files, printers, security, applications, and other networking functions over a small private network.
* The “ other" computers arc called client computers, and each computer that connects to a network server must be running client software designed to request a specific service.
* popularly known as **tightly coupled systems**.

**Advantages of Network Operating System:**

* Highly stable centralized servers
* Security concerns are handled through servers
* New technologies and hardware up-gradation are easily integrated into the system
* Server access is possible remotely from different locations and types of systems

**Disadvantages of Network Operating System:**

* Servers are costly
* User has to depend on a central location for most operations
* Maintenance and updates are required regularly

**Examples of Network Operating System are:**

Microsoft Windows Server 2003/2008/2012, UNIX, Linux, Mac OS X, Novell NetWare, and BSD, etc.

**7. Real-Time Operating System**

* These types of OSs serve real-time systems.
* The time interval required to process and respond to inputs is very small.
* This time interval is called **response time**.
* **Real-time systems** are used when there are time requirements that are very strict like
  + missile systems,
  + air traffic control systems,
  + robots, etc.

**8. Embaded Operating System**

An embedded operating system is one that is built into the circuitry of an electronic device. Embedded operating systems are now found in automobiles, bar-code scanners, cell phones, medical equipment, and personal digital assistants.

The most popular embedded operating systems for consumer products, such as PDAs, include the following:

* + Windows XP Embedded
  + Windows CE .NET:- it supports wireless communications, multimedia and Web browsing. It also allows for the use of smaller versions of Microsoft Word, Excel, and Outlook.
  + Palm OS:- It is the standard operating system for Palm-brand PDAs as well as other proprietary handheld devices.
  + Symbian:- OS found in “ smart” cell phones from Nokia and Sony Ericsson

**Popular types of OS**

Desktop Class

* 1. Windows
  2. OS X
  3. Unix/Linux
  4. Chrome OS
* Server Class
  1. Windows Server
  2. Mac OS X Server
  3. Unix/Linux
* Mobile Class
  1. Android
  2. iOS
  3. Windows Phone

**Desktop Class Operating Systems:-**

**Platform:** the hardware required to run a particular operating system

* + Intel platform (IBM-compatible)
    - Windows
    - DOS
    - UNIX
    - Linux
  + Macintosh platform
    - Mac OS
  + iPad and iPhone platform
    - iOS

**MS-DOS**

* Single User Single Tasking OS.
* It had no built-in support for networking, and users had to manually install drivers any time they added a new hardware component to their PC.
* DOS supports only 16-bit programs.
* Command line user interface.
* So, why is DOS still in use? Two reasons are its size and simplicity. It does not require much memory or storage space for the system, and it docs not require a powerful computer.

**Microsoft Windows**

* The graphical Microsoft operating system designed for Intel-platform desktop and notebook computers.
* Best known, greatest selection of applications available.
* Current editions include Windows 7, 8, 8.1 and 10.



**MAC OS**

* User-friendly, runs on Mac hardware. Many applications available.
* Current editions include: Sierra, High Sierra, Mojave, Catalina & Big Sur—Version XI(Released in Nov 2020)



**Linux**

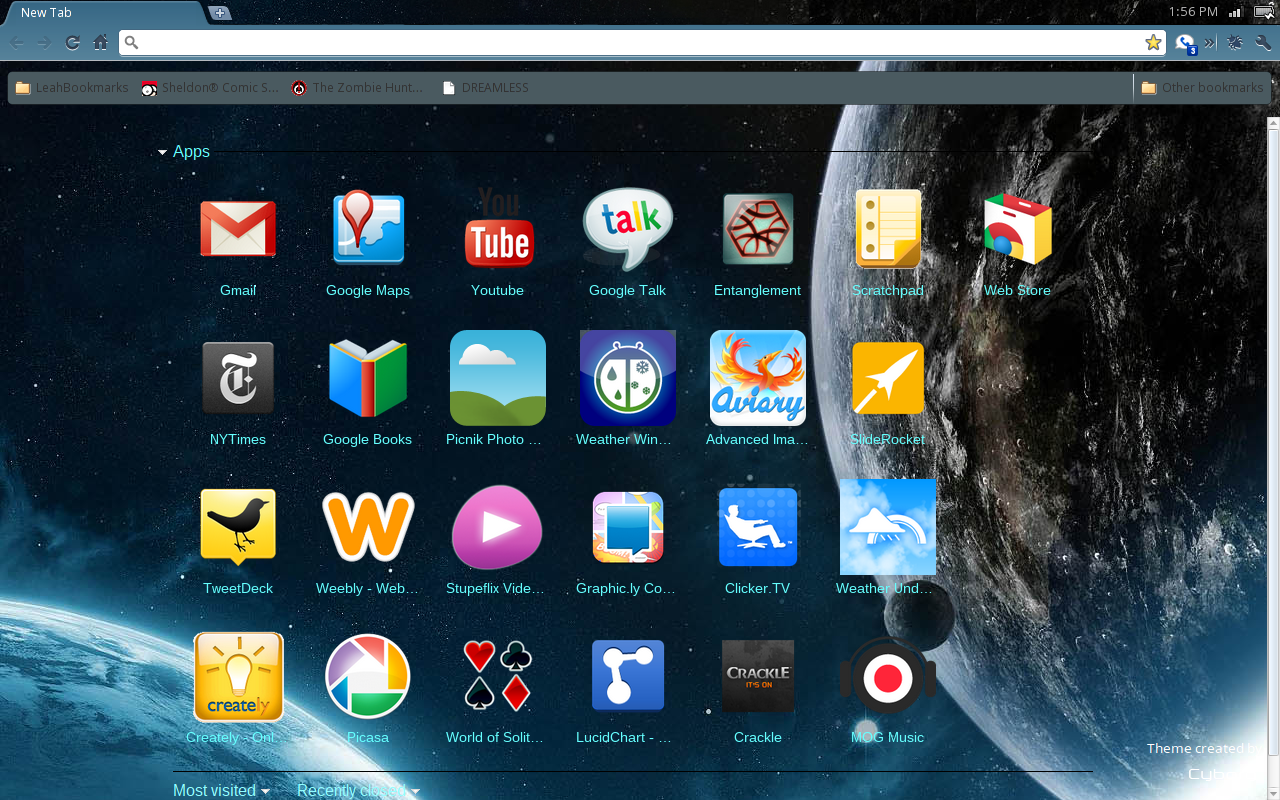
An open-source, cross-platform operating system that runs on desktops, notebooks, tablets, and smartphones.

* The name *Linux* is a combination *Linus* (the first name of the first developer) and *UNIX (another operating system*.
* Users are free to modify the code, improve it, and redistribute it,
* Developers are not allowed to charge money for the Linux kernel itself (the main part of the operating system), but they can charge money for **distributions** (**distros** for short).

**Google Chrome OS**

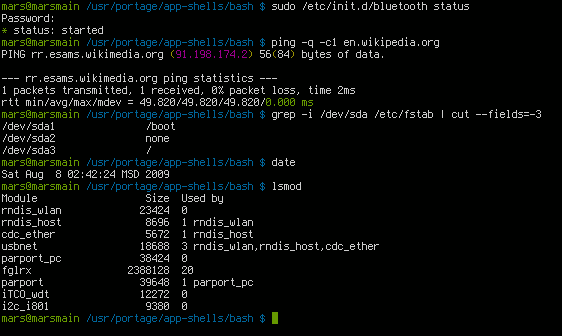
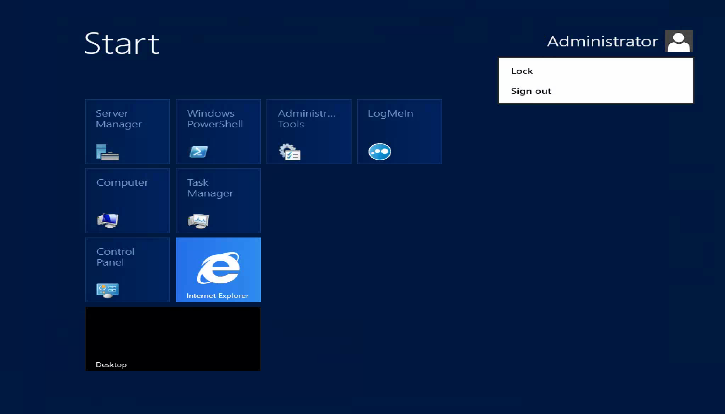
**Chrome OS**. Is a popular thin client operating system.

* **Thin client** A computer with minimal hardware, designed for a specific task. For example, a thin web client is designed for using the Internet.



**Server Operating Systems**

A server operating system, often referred to as a server OS, is a specialized operating system designed to run on servers, which are computers or systems dedicated to providing services, resources, or data to other computers, known as clients, over a network. Server operating systems are tailored to handle the unique demands and requirements of server environments. Here are some popular server operating systems:

* Windows Server
  + Familiar GUI interface for those experienced with Windows
* UNIX
  + Very mature server capabilities, time-tested, large user community, stable
* Linux
  + **Free, customizable, many free services and utilities available

*Windows Server UNIX*

**Tablet and Phone Operating Systems**

**System-on-chip (SoC):** An operating system that comes preinstalled on a chip on a portable device such as a smartphone.

* Popular SoC operating systems:
  + iOS: for iPad, iPhone
  + Android: for a variety of tablets and phones
* Downloadable applications (apps) from an App store, for example:
  + Apple App Store
  + Google Play Store

**iOS on the iPhone and iPad**

* The Apple-created operating system for Apple tablets and phones.
* The current stable version, iOS 17, was released to the public on August 29, 2023.

[](https://www.macstories.net/stories/ios-9-our-complete-overview/).



**Android**

* Android, a popular OS for smartphones and tablets, is based on Linux Kernel.
  + Developed by Google
* Current versions include:
  + Android 8 Oreo
  + Android 9 Pie
  + Android 10
  + Android 11
  + Android 12
  + Android 13 – August 15, 2022

**Advantage of Linux Operating System**

1. **Open Source**

As it is open-source, its source code is easily available.

Anyone having programming knowledge can customize the operating system.

One can contribute, modify, distribute, and enhance the code for any purpose.

1. **Security**

The Linux security feature is the main reason that it is the most favourable option for developers.

It is not completely safe, but it is less vulnerable than others.

Each application needs to authorize by the admin user.

Linux systems do not require any antivirus program.

1. **Free**

Certainly, the biggest advantage of the Linux system is that it is free to use.

We can easily download it, and there is no need to buy the license for it.

It is distributed under GPL (General Public License).

Comparatively, we have to pay a huge amount for the license of the other OS

1. **Lightweight**

The requirements for running Linux are much less than other operating system

In Linux, the memory footprint and disk space are also lower.

Generally, most of the Linux distributions required as little as 128MB of RAM around the same amount for disk space.

1. **Stability**

Linux is more stable than other operating systems.

Linux does not require to reboot the system to maintain performance levels.

It rarely hangs up or slow down. It has big up-times.

1. **Performance**

Linux system provides high performance over different networks.

It is capable of handling a large number of users simultaneously.

1. **Flexibility**

Linux operating system is very flexible.

It can be used for desktop applications, embedded systems, and server applications too.

It also provides various restriction options for specific computers.

We can install only necessary components for a system.

1. **Software Updates**

In Linux, the software updates are in user control.

We can select the required updates.

There a large number of system updates are available. These updates are much faster than other operating systems. So, the system updates can be installed easily without facing any issue.

1. **Distributions/ Distros**

There are many Linux distributions available in the market.

It provides various options and flavors of Linux to the users.

We can choose any distros according to our needs.

Some popular distros are **Ubuntu, Fedora, Debian, Linux Mint, Arch Linux,**

For the beginners, Ubuntu and Linux Mint would be useful. Debian and Fedora would be good choices for proficient programmers.

1. **Live CD/USB**

Almost all Linux distributions have **a Live CD/USB** option.

It allows us to try or run the Linux operating system without installing it.

1. **Graphical User Interface**

Linux is a command-line based OS but it provides an interactive user interface like Windows.

1. **Suitable for programmers**

It supports almost all of the most used programming languages such as [C](https://www.javatpoint.com/c-programming-language-tutorial)/C++, Java, Python, Ruby, and more.

Further, it offers a vast range of useful applications for development.

The programmers prefer the Linux terminal over the Windows command line.

The package manager on Linux system helps programmers to understand how things are done.

Bash scripting is also a functional feature for the programmers.

It also provides support for SSH, which helps in managing the servers quickly.

1. **Community Support**

Linux provides large community support.

We can find support from various sources.

There are many forums available on the web to assist users.

Further, developers from the various open source communities are ready to help us.

1. **14. Privacy**

Linux always takes care of user privacy as it never takes much private data from the user. Comparatively, other operating systems ask for the user's private data.

1. **Networking**

Linux facilitates with powerful support for networking. The client-server systems can be easily set to a Linux system. It provides various command-line tools such as ssh, ip, mail, telnet, and more for connectivity with the other systems and servers. Tasks such as network backup are much faster than others.

1. **Compatibility**

Linux is compatible with a large number of file formats as it supports almost all file formats.

1. **Installation**

Linux installation process takes less time than other operating systems such as Windows. Further, its installation process is much easy as it requires less user input. It does not require much more system configuration even it can be easily installed on old machines having less configuration.

1. **Multiple Desktop Support**

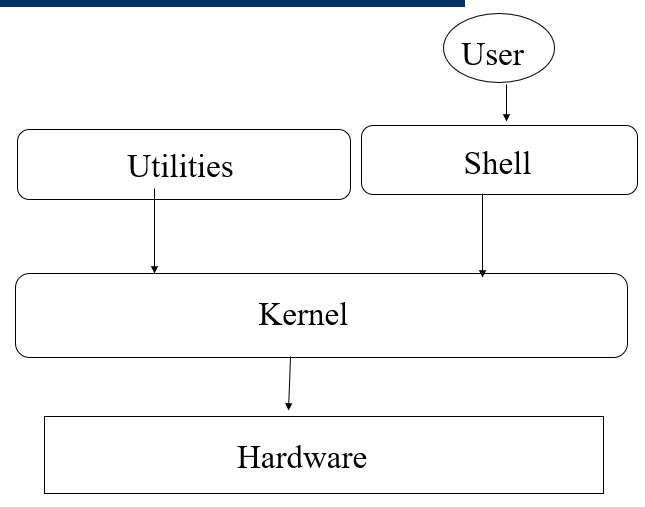
Linux system provides multiple desktop environment support for its enhanced use. The desktop environment option can be selected during installation. We can select any desktop environment such as **GNOME (GNU Network Object Model Environment)** or **KDE (K Desktop Environment)** as both have their specific environment.

1. **Multitasking**

It is a multitasking operating system as it can run multiple tasks simultaneously without affecting the system speed.

1. **Heavily Documented for beginners**

There are many command-line options that provide documentation on commands, libraries, standards such as manual pages and info pages. Also, there are plenty of documents available on the internet in different formats, such as Linux tutorials, Linux documentation project, Serverfault, and more. To help the beginners, several communities are available such as **Ask Ubuntu**, Reddit, and **StackOverflow.**

**UNIX Shell and Utilities**

* The shell used to be in the kernel but now is a utility outside of it.
* Easy to change/debug.
* Many of them (sh, bsh, csh, ksh, tcsh, wsh, bash)
* Possible to switch between them (chsh)

**A very simplified Shell**

**D. Summary**

Operating systems play a critical role in modern computing environments, enabling users to run applications, manage data, and interact with hardware efficiently and securely. They come in various types, including general-purpose desktop operating systems, server operating systems, mobile operating systems, and real-time operating systems, each tailored to specific computing needs.