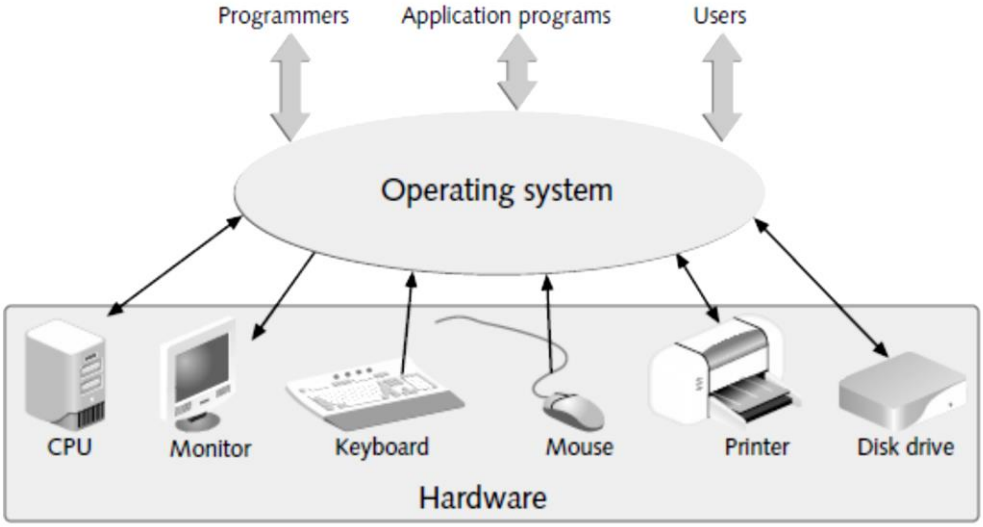
Unit-1 Introduction to Linux OS

What is an Operating System?

- An **operating system (OS)** is system software that acts as interface between user and Hardware.
- Because of system software, operating system capable to communicate with hardware's directly.
- Operating System is a master control program (basic s/w) which monitors and manages the internal functions of computer system and resources (i.e. CPU, Memory, I/O Device etc.).

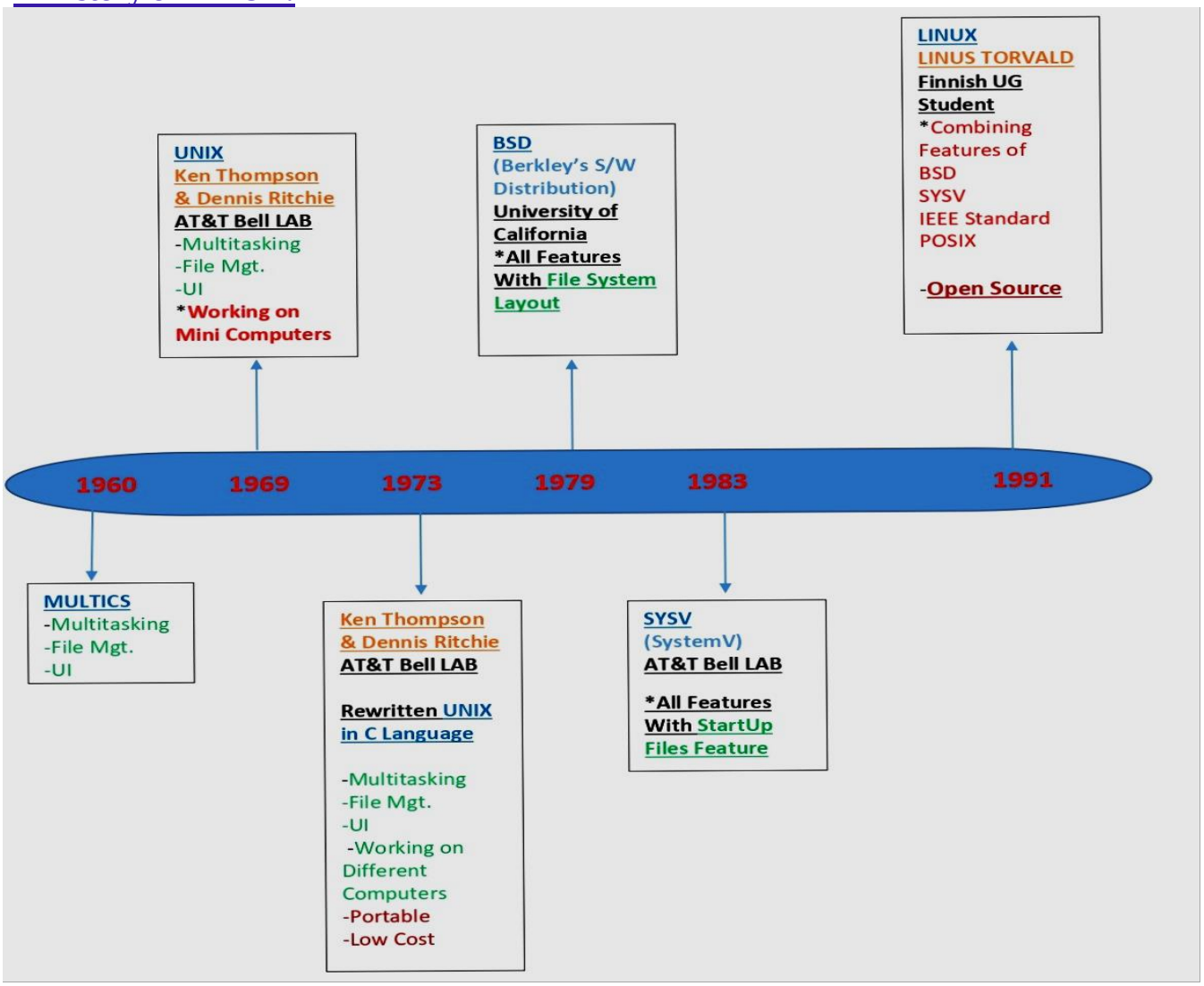


Eg. Examples of operating systems are Microsoft windows, Apple Mac OS, iOS, Android etc. Just like that, Linux is also an operating system basically used in very large servers like mainframe etc.

Introduction to Linux Operating Systems:

- Linux is open source, multiuser, multitasking operating system having built-in networking functions.
- Linux is also portable operating system means it can be carried on a physical drive and it can easily installed on varieties of hardware platforms.
- Linux operating system is one of the popular versions of the UNIX operating system, which is designed to offer a free and low-cost operating system for personal computer users.
- Basically Linux systems is used to provide security for communications over a network, including FTP applications and other network applications including Telnet (Linux program used on machine to perform networking task).
- The open source nature of Linux means that the source code of the Linux kernel is freely available to anyone and hence we can add features and correct deficiencies as per requirements.
- Many organizations choose UNIX and Linux because-
  - It is well suited for network environments.
  - It offers many security options.
  - It has thousands of applications for both commercial and personal use for free.
  - It is stable, reliable and versatile.
  - It enable employees to work on a wide range of computers over the network.

A History of LINUX:



- In the late 1960's when there was an idea to develop new operating system technique occurred. And special work on an experimental operating system called MULTICS (Multiplexed Information and Computing Service) is developed. It incorporated many new concepts like Multitasking, File Management and User Interaction.
- In 1969, Ken Thompson, Dennis Ritchie and AT&T Bell Labs researchers' a group of programmers developed UNIX. It includes features of MULTICS. It runs on minicomputer.
- In 1973 Ken Thompson &Dennis Ritchie rewrite the programming code for UNIX in 'C' language. Users Attracted by its portability and low cost, universities began to modify the UNIX code to make it work on different machines.
- Two standard versions of UNIX evolved one is BSD Berkeley Software Distribution (BSD) in 1979 at the University of California and in 1983 AT&T Bell Labs produced SYSV
- In year 1991, Linux is a free open source UNIX OS for PCs was developed by Linus Torvalds [A Finnish (Finland) undergraduate student of university of Helsinki].
- Linux has some features from SYSV (startup files) and BSD (File system layout) and aims to conform a set of IEEE standards called POSIX (Portable Operating System Interface). To maximize code portability, it typically supports SYSV, BSD and POSIX system.
- Because of open source nature of Linux, source code for the Linux kernel is freely available so that anyone can add features and correct deficiencies. This approach of Linux becomes very successful and one person's project has now turned into a collaboration of hundreds of developers around the globe. The open source approach has not just successfully been applied to kernel code, but also to application programs for Linux.

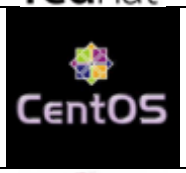
Feature /Characteristics/ Properties of LINUX:

- **Multi user** – Linux O.S is capable to handle multiple users at a same time that means multiple users can access system resources like memory, RAM, application programs at same time in linux operating system.
- **Multitasking** – In Linux, we can run multiple tasks or application at same time.
- **Portability** – Linux is portable means it can be carried on a physical drive and it can easily installed on varieties of hardware platforms.
- **Open Source** – Linux O.S. is free to use. Anyone has freedom to access, modify OS (Kernel) code and modified it in our own manner.
- **Hierarchical File System**– Linux provides a standard file structure in which system files and user files are arranged. Each file have access permission such as read, write and execute according to role of user.
- **Shell (Command Interpreter)** – Linux provides a special interpreter program called "Shell" which can be used to execute commands of the operating system. It can be used to do various types of operations like making directory, listing files and directories, sending message etc.
- **Security** –Linux O.S. provides security by assign login name and password to individual user. Also, at the file level it gives permissions to read, write, modify, and execute a file depending on role of user. Also, Linux has a file encryption utility that encodes your file into an unreadable format, if someone open it but file and data are safe because firstly you want to decrypt those files or data.
- **Communication** – Linux O.S. widely used over the network for sending mails, data, programs and also passing information, messages to one another.
- **Low cost:** We don't need to buy license of Linux because it come with the GNU (General Public License). We can freely download & use it easily from the internet.
- **Stability:** Linux doesn't need to be updated or restarted periodically to maintain performance levels because it is stable.
- **Performance:** Linux provides constant high performance on networks. It can handle large numbers of users simultaneously.
- **Network friendliness:** Linux has strong support for network functionality & client-server systems. It can performs networking task easily such as taking network backups faster and more consistently etc.

Linux Distributions:

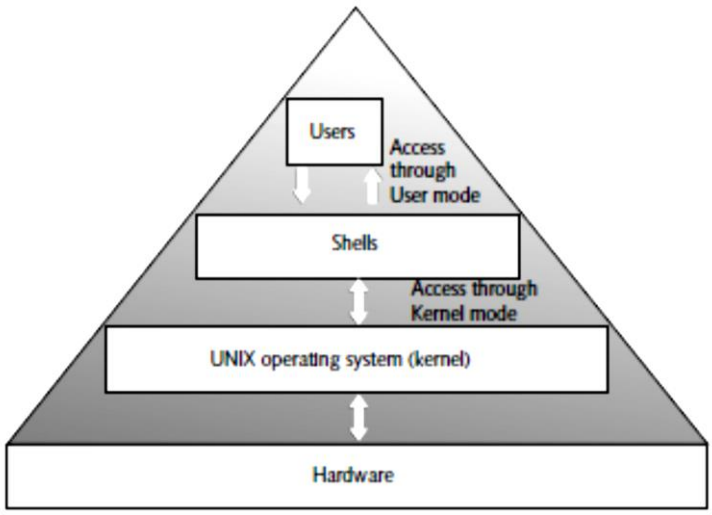
- A Linux distribution basically called as 'distro' is an operating system made from a software collection that includes the Linux kernel and other package management apps.
- A typical Linux distribution comprises of Linux kernel, GNU tools, libraries, additional software, documentation, a window system (X Window System, Wayland etc.), a window manager, and a desktop environment.
- Linux has different distributions because of its open source nature.
- The most common Linux distributions explained in following table-

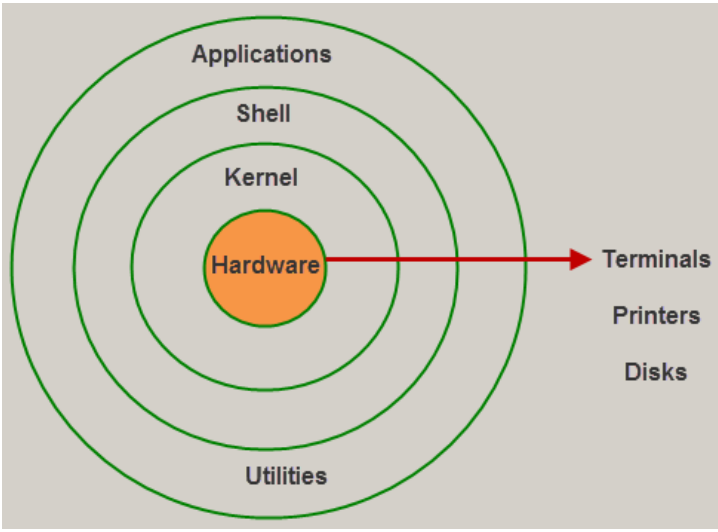
Linux Distribution	Name	Description
	**Arch**	This Linux Distro is popular among Developers. It is an independently developed system. It is designed for users who believes for do-it-yourself approach.
	**RedHat**	Red Hat is another Linux Distribution popular in networking area.

	**RedHat enterprise**	Red Hat Enterprise is an enterprise based Linux Distribution. It has evolved from Red Hat Linux which was discontinued in 2004. It is a commercial Linux Distro and very popular among its clienttele communication.
	**CentOS**	It is most used Linux Distribution for enterprise & web servers. It is a free enterprise class Operating system and is based heavily on Red Hat enterprise Distro.
	**Debian**	Debian is a stable & popular non-commercial Linux distribution. It is widely used as a desktop Linux Distro and user-oriented. It strictly acts within the Linux protocols.
	**Fedora**	Fedora is supported by the Fedora project, it is an attempt by Red Hat. It is popular among desktop users. Its versions are known for their short life cycle.
	**Gentoo**	It is a source based Distribution which means that you need to configure the code on your system before you can install it. It is not for Linux beginners, but it is sure fun for experienced users.
	**LinuxMint**	It is one of the most popular Desktop Distributions launched in year 2006 and is now considered to be the fourth most used Operating system in the computing world.
	**OpenSUSE**	It is an easy to use and a good alternative to MS Windows. It can be easily set up and can also run on small computers with obsolete configurations.
	**Slackware**	Slackware is one of the oldest Linux kernel based OS's. It is another easy desktop Distribution. It aims at being a 'Unix like' OS with minimal changes to its kernel.
	**Ubuntu**	This is the third most popular desktop operating system after Microsoft Windows and Apple Mac OS. It is based on the Debian Linux Distribution, and it is famous for its desktop environment.

Linux Architecture

- We know that, computer's operating system acts as an interface between computer hardware and user. Also, number of software applications are run on operating systems to manage hardware resources on a computer.
- LINUX is layered operating system that has several layers. And each Layer of operating system protects its dynamic hardware, software components and to manage the core system and its users.
- Following diagram shows Linux architecture:





The architecture of Linux divide into three major components, viz-

- Kernel
- Environment
- File Structure

Let's see these components in details-

1) Kernel:

- The **kernel** is the base or core component of OS, it interacts directly with the hardware, software services, application programs, and user script files (A files containing commands to execute).
- Kernel is heart of Linux O/S. It is the innermost layer of the OS that provides interface between hardware and shell. It manages resource of Linux OS.
- When the machine starts, the Boot Loader passes the control to kernel. It is responsible for managing the I/O devices such as VDU, mouse, keyboard etc. It also deals with memory management and process management. The kernel does not directly deal with the user; instead it creates a new startup, interactive process for the user via shell.
- Kernel decides who will use particular resource, how much time and when. It runs your programs & transfers all data between the file system, H/W & also memory.
- Kernel manages various scheduling of running programs/processes in memory & allocation of CPU time to all running programs.
- Kernel also handles any interrupts issues, as it deals with H/W directly.
- It performs following tasks **(Functions of kernel):-**
  - I/O management
  - Process management
  - Memory management
  - Disk and File System Management
  - Networking
  - Security
  - Graphical User Interface (GUI)
  - Device Driver Management

2) Environment:

- We know that, to work with hardware there is necessary of environment. With the help of environment user can communicate with hardware via. Operating system. And Linux operating provides several kind of environment as follow:

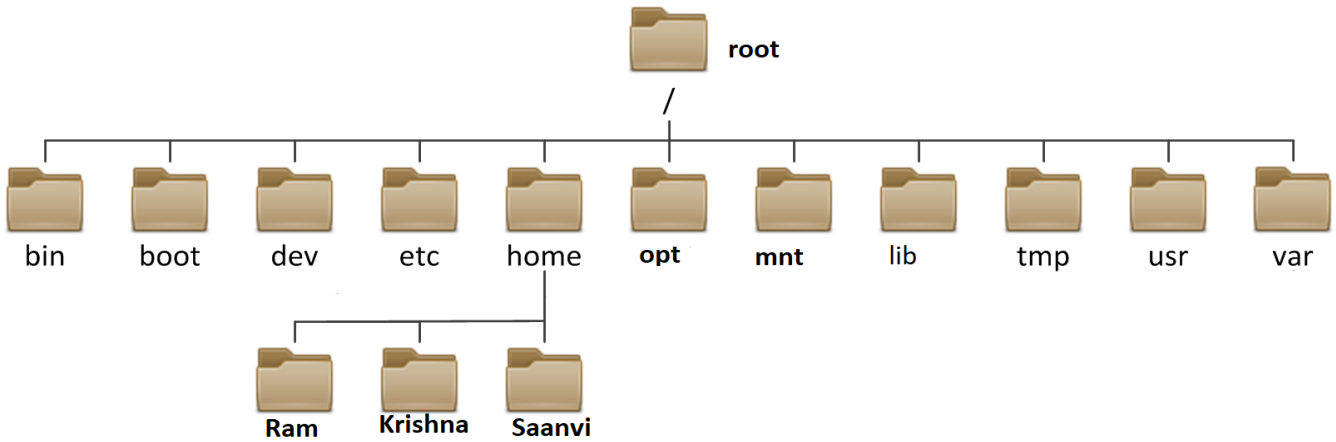
1. Desktops:

- Desktop is the best environment for the user to work with O.S. because it supports for GUI (Graphical User Interface) as like Windows operating system. With the help of Linux GUI, user can easily give commands to operating system and then O.S. communicate with appropriate hardware.

2. Shells (Command Interpreter):
   - Linux also provide command line interpreter called 'Shell' to communicate with operating system.
   - To use commands in Linux, it has a special window, called a 'Terminal window', and using this terminal window, we can communicate with the Linux OS through a shell interpreter.
   - Most versions of Linux supports for 'Terminal window' which is a powerful feature because it is literally your window to use commands.

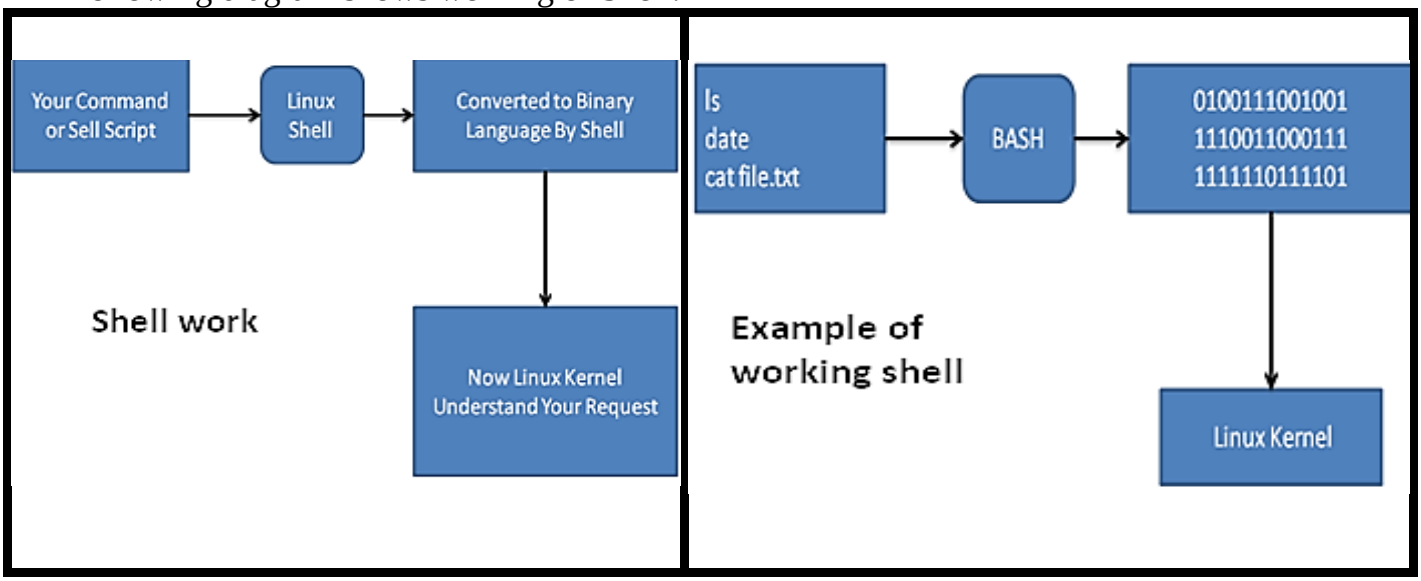
3) File Structure:

- We know that, File is collection of large amount of data or information which is permanently stored on memory devices for future use.
- In Linux, everything is configured as a file. This includes not only text files, information, data, music files, movie, images and compiled programs (executable files), but also directories, partitions and hardware device drivers.
- The LINUX file system provides a logical method for organizing, storing, retrieving, manipulating, and managing information.
- The entire Linux file system is one large interconnected set of directories, further each containing files, some directories etc.
- In Linux, some directories are reserved for system use. Also, we can create our own directories for our own files as well as we can easily move files from one directory to another.
- Also, we can move and share entire directories with other user. And we can set the permissions to directories, files & data.
- *The entire Linux file structure is of hierarchical tree structure beginning with initial **'Root' directory**.*
- *Following diagram shows hierarchical file structure of Linux:*



Shell and Types of shell: (Command-line Environment)

- In Linux, a shell is a command-line interface/interpreter that allows users to interact with the operating system (Kernel) by typing commands. A shell is a program that provides a text-based interface for accessing the system's resources and executing programs.
- Or we can say that, Shell is a command interpreter that interprets the commands given by the user via input devices (Keyboard, mouse etc.) or from a file and handed over this interpreted result (Binary format Data) to kernel and then kernel interact with appropriate hardware or resources. Thus, In Linux, Shell acts as an interface between User and Kernel.
- The shell is an important tool in Linux used by developers and system administrators, to automate tasks and manage systems from the command line.
- The most commonly used shell in Linux is the BASH (Bourne-Again Shell) shell. Other popular shells include the Korn shell (ksh) and the Z shell (zsh).
- Each shell has its own set of features and capabilities depending upon linux distro., but they all share a common syntax for executing commands and managing files and directories.
- Following diagram shows working of Shell:



Features of Shell:

The shell is a program that provides a command-line interface to interact with the operating system in Linux. Here are some of the features of the shell in Linux:

- Command-line interface: The shell provides a command-line interface that allows users to enter commands and receive output from the system.
- Scripting: The shell allows for the creation of scripts, which are sequences of commands that can be executed as a single command.
- Command history: The shell keeps a history of previously executed commands, which can be accessed using the up and down arrow keys.
- Environment variables: The shell allows for the creation and manipulation of environment variables, which are used to store system-wide settings and preferences.
- Pipes and redirection: The shell allows for the use of pipes and redirection to send output from one command to another or to redirect input/output from/to files.
- Job control: The shell allows for the control of running processes, including the ability to start and stop jobs in the background.
- Command substitution: The shell allows for the substitution of the output of one command into another command, using the backtick (`) or dollar sign ($) notation.
- Aliases: The shell allows for the creation of aliases, which are user-defined shortcuts for commonly used commands or command sequences.

- Tab completion: The shell provides tab completion, which allows users to quickly complete command and file names by typing the first few characters and then pressing the tab key.

These are some of the features of the shell in Linux, which make it a powerful tool for interacting with the operating system.

Types of Shell:

- In Linux, a shell is a program that provides a command-line interface for interacting with the operating system. The shell interprets the user's commands and executes them by interacting with the system's kernel.
- There are several types of shell available in Linux, each with its own set of features and capabilities. Here are some of the most commonly used ones:

1) Bourne Shell (sh):

- The Bourne shell is originally developed for UNIX by Stephen Bourne at AT&T's Bell Labs in 1977.
- This is the oldest shell. Its feature set is sufficient for most programming needs.
- It is the favorite shell for shell programming because of its neatness and speed.

Features of sh:

- It keeps the record of command history.
- Easily make Input and output redirection. (Reading/Writing data from file)
- Ability of shell scripting with string and integer variables.
- Supports for condition testing and looping.

Drawbacks:

- It is hard to use interactively.
- No option to re-edit previous commands or not support for command editing.
- Difficulty in executing multiple background processes or jobs.
- Unable to handle built-in arithmetic and logical expressions.

2) csh (C Shell):

- The C shell was developed by Bill Joy for the Berkeley Software Distribution in 1978.
- The C shell syntax is taken from the C programming language. It is used primarily for interactive terminal use.
- It is less frequently used for scripting and operating system control.
- Basically it is used as interactive login shell and a shell script command processor.
- For e.g. login and write program or update system configuration.

Features of Csh:

- It has history mechanism to store commands.
- It have Job control facilities.
- C shell have many interactive commands to deal with kernel.

3) tcsh: (TC Shell)

- tcsh is an enhanced and compatible version of the C shell.
- It has a number of enhancements and features than the bash shell.
- It is a command language interpreter usable both as an interactive login shell and a shell script command processor.

Features of tcsh:

- It also have command edit facility.
- It has programmable word completion.
- It has spelling correction facility.
- It also have history mechanism that stores executed commands.

4) Korn shell (ksh):

- Korn shell was developed in 1983 by David at AT&T Bell Lab.
- ksh compatible with sh and bash.
- It includes more powerful programming features like built-in arithmetic and C-like arrays, functions, and string-manipulation facilities.
- It is faster than the C shell.
- Ksh shell works well across a network using a physical terminal.
- ksh merge the features of shells like the C shell, TC shell, and Bourne shell.
- It allows the developers to create new shell commands when needed.

Features of ksh-

- It have command history & Command-line editing facility.
- It have file name completion and Alias command facility.
- This shell have job control functionality.
- It have integrated programming features & Control structures.
- It also have advanced I/O features, Security features
- It allowed regular expressions & Debugging facility.

5) Bourne-again Shell (BASH)

- Bash shell is default shell in most of Linux distributions. It was developed by Brian Fox for the GNU project Released in 1989.
- The Bash shell is a combination of features of the Bourne Shell, C Shell and Korn shell.

Features of BASH:

- It has Command editing facility.
- It has good control over the job.
- It has many useful environment variables.
- It also have unlimited size command history.
- It provides Shell Functions and Aliases
- It supports arithmetic expression handling.
- It also supports for regular expressions and associative arrays.

Difference Between Linux and Windows:

Linux	Windows
1) Linux is Open Source operating system developed by the Linus Torvalds.	1) Windows is not open source operating system developed in Microsoft organization.
2) It comes under GNU (General Public license) which is Free.	2) It not comes in GNU therefore we have to purchase it. (Not Free)
3) Linux kernel code is easily accessible to everyone.	3) Windows kernel code is not accessible to everyone.
4) Linux have two core GUIs one is KDE & other is Gnome. But there are number of alternatives such as LXDE, Xface, Mate etc.	4) The Windows GUI is an integral component of the OS and is not replaceable or it has no alternatives.
5) Bourne Again Shell (BASH) is the Linux default command line interpreter.	5) MS-DOS i.e. command prompt is the Windows default command line interpreter.
6) Linux is more secure operating system than windows.	6) Windows is less secure than Linux operating system.
7) Threat (risk) detection and solution is very fast, several developers start working on it from different place of the world immediately.	7) Threat (risk) detection and solution is slow. Microsoft releases a patch that can fix the problem and it can take more than 2/3 months.

8) Linux file name is case-sensitive. Thus, 'sample' and 'SAMPLE' are 2 different files in Linux	8) Windows file name is case- insensitive. (not case sensitive)
9) Linux uses monolithic kernel.	9) Windows uses micro kernel.
10) Linux is widely used in hacking purpose based systems.	10) Windows does not provide much efficiency in hacking.
11) Easily installed and configured but on limited range of hardware platforms.	11) Easily installed and configured on wide range of hardware platforms.
12) In Linux, forward slash is used for separating the directories.	12) In Windows, backward slash is used for separating the directories.
13) Linux have 3 types of built in user accounts– (1) Regular , (2) Root , (3) Service account	13) Windows have 4 types of built in user accounts– (1) Administrator , (2) Standard , (3) Child , (4) Guest
14) In Linux, Root user is the super user and has all administrative privileges.	14) In Windows, 'Administrator' user has all administrative privileges.
15) Ubuntu, Fedora, Red Hat, Debian, etc. are the examples of Linux OS.	15) Windows 8, 8.1, 7, Vista, XP etc. are the examples of windows OS.
16) Most of the work is done through Terminal (a console window), and if a problem arises, a GUI is rarely usable to fix them.	16) Everything can be controlled through a GUI and incompatibility problems are rare.
17) EXT,EXT2,EXT3,EXT4,XFS,JFS File Systems supported by linux	17) FAT,FAT32,NTFS and ReFS File Systems supported by windows.

Linux Working environments: (Desktop Environment)

- We know that, to work or deal with kernel (OS) there is necessary of environment. And linux OS provides two environments one is Shell (Command line interface) and other is desktop (GUI).
- In previous point, we had already seen Shell and its types. In addition to the command-line interface (Shell), Linux also provides a graphical user interface (GUI) that allows users to interact with the operating system using a mouse and keyboard.
- This Desktop environment (GUI) provides access to a range of applications and tools, including text editors, web browsers, file managers, and more.
- The Linux working environment also includes a range of programming and development tools, including compilers, debuggers, and other tools that developers can use to create and test software applications.
- GNOME, KDE and Xfce4 are the most popular desktop environments of the Linux. Let's see these Linux GUI environment in details.

1) GNOME:

- GNOME **G**NU **N**etwork **O**bject **M**odel **E**nvironment (**GNOME)** is free and open-source Linux desktop environment developed in August 1997.
- GNOME is a free and open-source desktop environment and graphical user interface (GUI) for Unix-based operating systems, such as Linux.
- It is one of the most popular desktop environments available for Linux, and it is the default desktop environment for several Linux distributions, including Fedora, Debian, and Ubuntu.

Features/Characteristics/Advantages of GNOME:

- User-friendly interface: GNOME has a modern and user-friendly interface that is easy to navigate, even for new users.

- Customizable: GNOME is highly customizable, with a range of themes, extensions, and plugins available to change the look and feel of the interface.
- Integrated applications: GNOME comes with a range of integrated applications, such as a file manager, web browser, and email client, which can be convenient for users who want a complete desktop environment out of the box.
- Accessibility: GNOME includes a range of accessibility features, such as support for screen readers, high contrast mode, and keyboard shortcuts, making it more accessible for users with disabilities.
- The core components of the GNOME desktop consist of a panel for starting programs that use buttons, menus, windows and desktop functionality. Other components such as a file manager, a web browser, and a window manager are provided by GNOME-compliant applications.
- The official file manager of the GNOME desktop is Nautilus.
- The GNOME desktop does not have its own window manager. The **Metacity** window manager is used by GNOME distribution.
- The standards used in GNOME is the **Common Object Request Broker Architecture (CORBA)** developed by the Object Model Group for use on UNIX systems. With such framework/architecture, GNOME applications and clients can directly communicate with each other.

Disadvantages of GNOME:

1) Resource-intensive: GNOME can be resource-intensive, requiring a relatively powerful computer to run smoothly.
2) Limited customization: While GNOME is highly customizable, there are some limitations to what can be customized, and some users may find it difficult to modify the interface to their liking.
3) Steep learning curve: While the interface is user-friendly, GNOME can have a steep learning curve for users who are new to Linux or desktop environments in general. (Steep learning curve means-The knowledge that takes longer time to learn.)
4) Lack of features: Some users may find that GNOME lacks certain features that they want or need, such as advanced file management or customization options.

2) KDE: (K Desktop Environment) (Kool Desktop Environment)

- The KDE is popular Linux working environment which was developed & distributed under the KDE Project in October 1996.

Features/Characteristics/Advantages of KDE:

- Desktop and Panels: The KDE desktop is organized into panels, which are similar to taskbars in other operating systems. We can add applets, widgets, launchers, and system indicators to these panels to customize them according to our preferences.
- Kicker Menu: The Kicker menu is the main application launcher in KDE. It is accessible from the panel and provides quick access to frequently used applications, system settings, and other features.
- Dolphin File Manager: KDE uses Dolphin as its default file manager. It is a highly customizable and user-friendly file manager that provides features such as split views, tabs, and thumbnail previews.
- Window Management: KDE provides a range of window management options, including virtual desktops, window snapping, and a variety of window effects.
- Activities: Activities are a unique feature in KDE that allow you to group together applications, files, or specific tasks or projects.
- KDE Applications: KDE provides a range of powerful and customizable applications, including web browsers, email clients, media players, and office suites. These applications are designed to integrate seamlessly with the KDE desktop environment.

- On a system administration level, KDE provides several tools for configuring your system with KUser, we can manage user accounts, adding or removing etc.
- KDE is an Internet aware system that includes a full set of integrated network/Internet applications, including a mailer, a newsreader and a web browser etc.
- Multiple applications are integrated in KDE such as editors, photo-paint applications, KWord or KMail application, calculators, console windows, notepad & S/W package managers etc.
- Customizability: KDE is highly customizable and allows users to configure nearly every aspect of the desktop environment, including the look and feel, desktop effects, and behavior.
- User-Friendly Interface: KDE is designed to be user-friendly and provides a straightforward interface that is easy to navigate.
- Feature-Rich Applications: KDE applications are feature-rich and provide a wide range of functions, including web browsing, email, document editing, and multimedia playback.
- Active Community: KDE has an active community of developers and users who provide support, bug fixes, and new features.
- Compatibility: KDE is compatible with a wide range of Linux distributions and runs on various hardware configurations.

Disadvantages KDE:

1) Resource-Intensive: KDE is also resource-intensive, requires a relatively powerful computer to run smoothly.
2) Complexity: KDE's high degree of customization and feature-rich applications can also make it more complex and harder to use for some users.
3) Learning Curve: For users who are not familiar with KDE, there may be a learning curve when navigating the desktop environment.
4) Stability: While KDE is generally stable, it may occasionally encounter stability issues or crashes, particularly with third-party applications.
5) Installation: Installing KDE can require a significant amount of disk space and may require additional dependencies that need to be installed.

3) Xface4 or Xfce:

- Xface4 is a desktop environment based on the GTK+ toolkit used by GNOME.

Features/Characteristics/Advantages of using Xfce:

- Lightweight: Xfce is designed to be lightweight and fast, making it ideal for use on older or low-spec hardware.
- Customizable: Xfce provides a high degree of customization, allowing users to configure various aspects of the desktop environment to suit their preferences.
- Stability: Xfce is known for its stability and reliability, with few reported crashes or stability issues.
- Easy to Use: Xfce is designed to be user-friendly and straightforward, with a simple interface that is easy to navigate.
- Low System Requirements: Xfce has low system requirements, which means it can run on a wide range of hardware configurations.
- It is fully configurable, has a main panel with menus, applets, and application launchers, provides a file manager and sound manager, and is theme able.
- Since it is fast, light, and efficient, it is ideal for older or slower machines with memory limitations.
- Xfce provides a traditional desktop interface, including a panel at the bottom of the screen for launching applications, switching between workspaces, and displaying system indicators. It also provides a customizable application menu, a file manager, and basic desktop setting.
- Overall, Xfce is a lightweight, stable, and customizable desktop environment that is ideal for users who value simplicity and efficiency.

Disadvantages of using Xfce:

1) Limited Features: Xfce provides a basic set of features, which may be insufficient for users who require more advanced functionality.
2) Outdated Appearance: Some users may find the default appearance of Xfce to be outdated compared to other desktop environments.
3) Lack of Community Support: Xfce has a smaller community compared to other desktop environments like GNOME or KDE, which may make it harder to find support or resources.

Linux installation and configuration:

Linux installation and configuration refer to the process of installing the Linux operating system on a computer and configuring it to meet the user's needs. Here are the general steps involved in Linux installation and configuration:

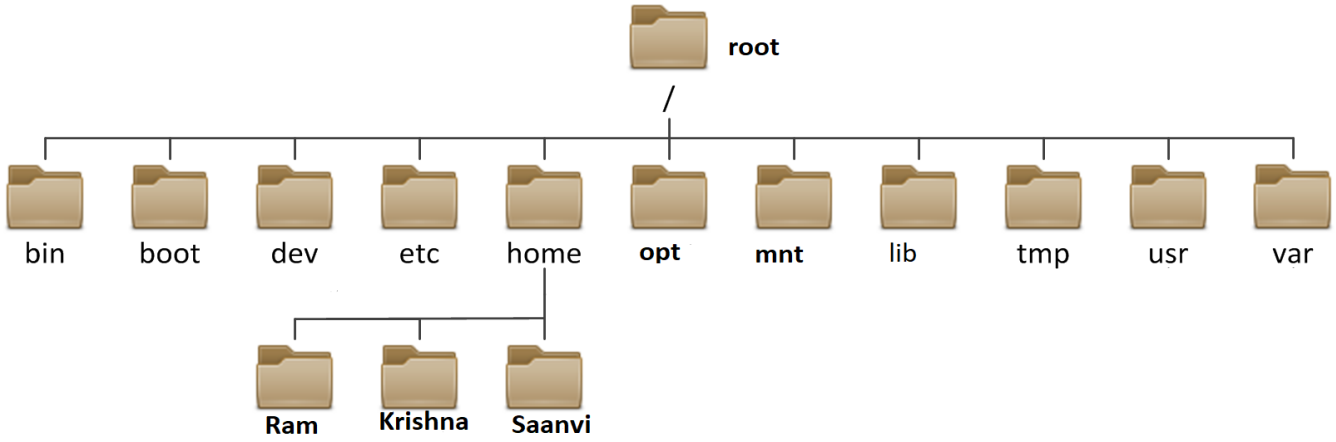
- Choose a Linux distribution: There are many different Linux distributions available, each with its own set of features, applications, and user interfaces. Choose a distribution that best suits your needs and experience level.
- Download the installation image: Once you have chosen a distribution, download the installation image from the distribution's website. The image can be burned to a DVD or USB drive for installation.
- Boot from the installation media: Insert the installation media into the computer and boot from it. This will start the Linux installer.
- Follow the installation wizard: The installer will guide you through the installation process, prompting you to make various choices such as partitioning the hard drive, selecting a language, and configuring network settings.
- Customize the installation: Depending on the distribution, you may be able to customize the installation by selecting which applications and packages to install, as well as setting up user accounts and passwords.
- Finish the installation: Once you have completed the installation, the computer will reboot into the newly installed Linux operating system.
- Configure the system: After installation, you may need to configure various aspects of the system, such as setting up network connections, installing additional applications or drivers, and customizing the desktop environment.

The exact steps involved in Linux installation and configuration can vary depending on the distribution and the user's specific needs.

Linux File System

Linux File System:

- A Linux file system is a structured collection of files on a disk drive or a partition. A partition is a segment of memory and contains some specific data. In our machine, there can be various partitions of the memory. Generally, every partition contains a file system.
- Or we can also say that File System is nothing but a logical collection of files on disk.
- The Linux file system is the structure that organizes and manages files, directories, and other data on a Linux-based operating system.
- The Linux file system follows a hierarchical tree-like structure, with the root directory (represented by the symbol "/") at the top, followed by a series of subdirectories and files which is shown as follow:



Some commonly used directories in the Linux file system are:

- /root:
  This is the home directory of the system administrator.
- /bin:
  Contains essential binary executable (programs) that are required for basic system functions.
  This directory contains several useful commands that are of use to both the System Admin (i.e. root user) as well as non-privileged users.
- /etc:
  Contains configuration files for system applications and services.
  This is the nerve center of the system. It contains all the system related configuration files here or in its sub-directories.
- /home:
  Contains the home directories of individual users.
  Each & every user have their own home directory having name as its username.
  For General user, home directory will be /home/username.
  For root user, home directory is /root.
- /lib:
  Contains shared library files that are required by system programs.
  Contains Shared Libraries and Kernel Modules.
- /proc:
  Provides access to system information and process data.
  Contains all processes marked as a file by process id.
- /tmp:
  Contains temporary files and directories.
  Contains temporary files created by O.S/ users which are automatically deleted when system is shutdown or restart.
- /boot
  This directory contains everything required for the boot process (i.e. boot loader programs) except for the configuration files not needed at the boot time.
- /dev
  This is the location of the special files or the device files.
- /mnt
  This is a generic mount point under which you mount your file system for External storage devices.
- /opt
  This directory is reserved all the software & add-on packages that are not the part of default installation.

- /usr
  Used for storing files separately for each user when multiple users are working on system.
- /var
  Contains Variable length files on a system. Data that will change as the system is running (Log files, backups, print files, etc.)

Some key aspects of the Linux file system:

- Root Directory: The root directory ("/") is the highest level of the file system hierarchy and contains all other directories and files.
- Directories: Directories are used to organize files and other directories. Each directory has a name, and can contain zero or more files and subdirectories.
- Files: Files are individual data objects that are stored within directories. Files can contain text, binary data, or other types of data.
- File Permissions: Linux file permissions determine who can access and modify files and directories. Permissions can be set for three categories of users: owner, group, and others.
- Mount Points: Mount points are locations on the file system where external storage devices or network drives can be connected and accessed.
- Virtual File Systems: Linux also uses virtual file systems to manage certain types of data, such as system information and device drivers.

Linux file system parts:

- We know that, A file system is a logical collection of files on a partition or disk where partition is a container for information and can span an entire hard drive.
- We also know that, Linux file system is a hierarchical directory structure used by Linux operating systems to organize and store files and directories.
- The Linux file system follows a tree-like structure with a root directory at the top of the tree.
- The Linux file systems consist of several following parts:
  1) Boot blocks
  2) Superblock
  3) Inode block
  4) Data blocks
  5) Directory
  6) Journal
  7) Mount point
  8) Virtual file system (VFS)

Let's see these parts in details-

1) Boot blocks:

- Boot block is the beginning or first part of the Linux file system.
- The boot block contains information required for booting the operating system.
- This block contains the program called 'bootstrap loader' and it is executed when we start the host machine.
- Although only one boot block is needed to start up the system, all file systems contain one (possible empty) boot block.

2) Super blocks:

- The superblock contains memory information about the file system, including the size of the file system, the block size, the number of inodes, and the location of the inode table.
- It describes the state of file system and state of File System contains information such as how large it is? (Capacity), how many files are accommodated? (Allocated), how many more files can be created (free space) etc.

3) Inode block (Inode table):

- The inode table contains general information about each file in the file system, such as ownership, permissions, and file type etc.

- The file specific information of all files (not the contents) are stored in Inode Table on the disk. And for each file there is an inode entry in the table.
- File specific information is made up of 64 bytes and contains relevant information for that file with following details:
  - Owner of the file.
  - Group to which owner belongs.
  - Type of file
  - File access permissions
  - Date and Time of last access
  - Date and Time of last modification
  - Number of links to the file.
  - Size of file
  - Address of blocks where file is physically stored

4) Data block:

- The data block contains the actual file data, organized into blocks of a fixed size.
- In short we can say that these blocks contains actual file contents i.e. data in a file.
- An allocated block can belongs to only one file in the file system
- This block can NOT be used for storing any other files contents unless the file to which it originally belonged is deleted.

5) Directory:

- The directory is a special part of file that contains a list of file names and their corresponding inodes.

6) Journal:

- Some file systems, such as ext3 and ext4, use a journal to record or stores changes to the file system before they are committed to disk. This helps to prevent data corruption in the event of a system crash or power failure.

7) Mount point:

- The mount point is the location in the file system hierarchy where other file system is mounted. All files and directories within the file system are accessed relative to the mount point.

8) Virtual file system (VFS):

- The virtual file system is an abstraction layer that provides a unified or virtual view of entire file system hierarchy, regardless of the underlying file system implementation. This allows different types of file systems, such as ext4, NFS, and FAT32, to be used interchangeably.

Linux File types:

Linux supports different types of files, which are discussed as follow:

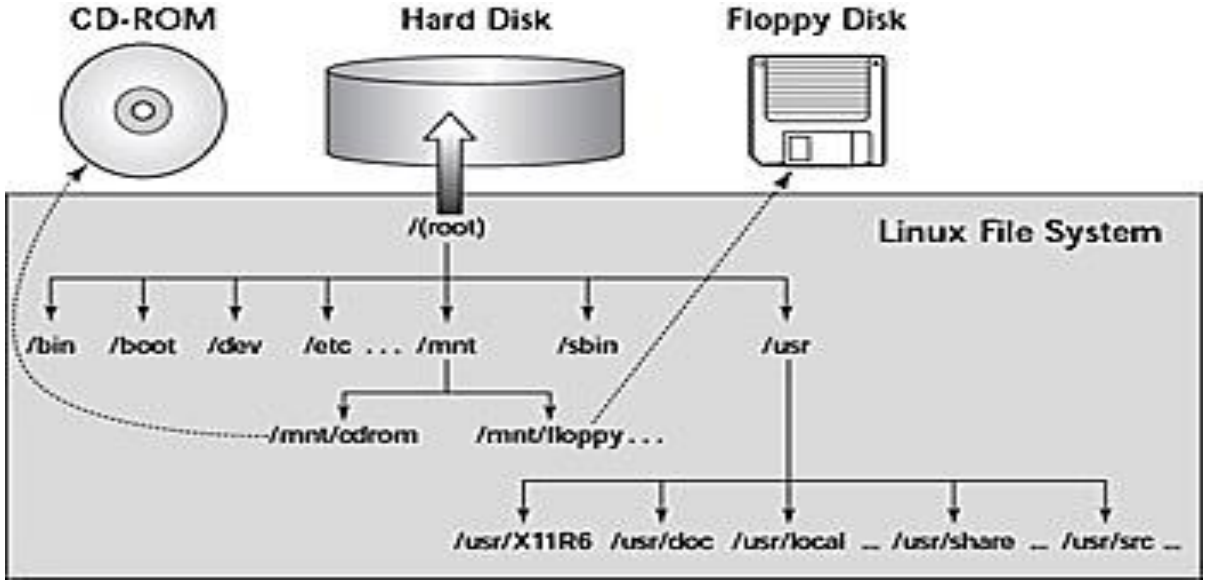
1) **Regular files:** These are the most common type of files in Linux. They contain data, such as text, images, videos, or program instructions. Regular files can be read, written to, and executed by users and programs.
2) **Directories:** Directories are used to organize files and other directories into a hierarchical structure. They can contain other directories and files, and they can be nested inside other directories.
3) **Symbolic links:** Symbolic links, also known as soft links, are files that point to other files or directories. They act as shortcuts to the original files, and they can be used to create complex directory structures and simplify file management.
4) **Device files:** Device files represents to hardware devices, such as hard disks, CD-ROM drives, and printers. They allow programs to access and communicate with the hardware.
5) **Named pipes:** Named pipes, also known as FIFOs, are special files used for interprocess communication between programs. They allow programs to pass data between each other, without the need for a shared memory or a network connection.

6) **Sockets:** Sockets are files used for interprocess communication between programs running on different computers, over a network. They allow programs to send and receive data over the network, using a standardized protocol.
7) **Special files:** Special files include block devices, which are used for random access data storage, and character devices, which are used for sequential data streams.

In addition to these file types, Linux also supports various file system types, including ext4, ext3, NTFS, FAT32, and others. Each file system type has its own characteristics and performance characteristics.

Mounting Devices in Linux:

- Mounting devices in Linux is the process of making a file system available for use by the system and users. The file system can be on a hard disk, USB drive, CD-ROM, or any other type of storage device.
- Mounting: Attaching a file system on a Physical storage device to our main directory tree is called as 'mounting the device'. And the File system is mounted to an empty directory on the main directory tree.
- The Directory in the file structure to which the new file system is attached is called as "MountPoint".
- For example, to access files on a CD/DVD, we have to first mount its file system to a particular directory in directory tree.
- Once a file system of a storage device is mounted then its entry is made by O.S. in **/etc/mstab** File.
- Here in this **/etc/mstab** file we will find list of all file systems which are currently mounted.
- The overall linux file system with **mounting of CD-ROM and Floppy Disk** is shown as follow:



Linux command for mounting and unmounting devices:

In linux, to mount and unmount devices following command are used-

mount Command:

The mount command takes two arguments viz

I) storage device through which Linux accesses the file system where the special device file that connects your system to the hardware device.

II) mount point is the directory on your main directory tree where you want attach the storage device. If the directory does not yet exist, you have to create it.

Syntax:	**# mount**	**device**	**mount point**
Example:	# mount	/dev/hdc2	/mydir

NOTE:
File system can be mounted on hard disk partition only by root user, where CD- ROM, floppies and any other devices can be mounted by any user.

umount Command

umount command can be used to unmount the mounted devices file system from the mount point.
umount command can takes argument either a device name or the directory where it was mounted.
Syntax: **# umount device or mount point**
Example: # umount /dev/hdc2 **OR** # umount /mydir
NOTE:
We Can Not unmount a file system from directory which we are currently working.

List Of mounting Devices:

We can mount and unmounts following devices in Linux-
1. File System
2. CD/DVD Disk
3. Hard Disk Partition
4. USB

1) Mounting/Unmounting File system:

- Mounting a file system simply means making the particular file system accessible at a certain point in the Linux directory tree.

We can mount file System as follow:
Syntax: # mount –t FileSystem **devicename** **mountpoint**

Example: # mount –t vfat /dev/sdc1 /media/dosdir
We can unmount file System as follow:
#umount /media/dosdir **OR** # umount /dev/sdc1

2) Mounting/Unmounting CD/DVD ROMs:

- To mount CD/DVD we have to insert it into drive.
- HAL (H/W Abstraction Layer) will detect it and mount it automatically in **/media/disk** directory.
- If we want to manually mount the drive from command line with mount command we have to first directory to mount on it.
- The /media/disk directory is created dynamically when a disk is inserted and deleted when disk is removed.

To mount disk we use command like: # mount /dev/cdrom /mnt/cddir
To unmount disk we use command like: # umount /mnt/cddir **OR** # umount /dev/cdrom

3) Mounting/Unmounting Hard drive partitions:

- Linux partitions on hard disks which are created during installation are already automatically mounted for us.
- We can also mount a window partition and directly access files on it.

To mount hard drive partition we can use command:
mount –t ext3 /dev/hda4 /mnt/mydir
To unmount hard drive partition we can use command:
umount # umount /dev/hda1

4) Mounting/Unmounting USB drive:

- Whenever we plug in our USB device into USB port, O.S. will add new block device into /dev directory. At this stage we can NOT use this device as USB file system because it must first be mounted.

- To find out what usb name assigned to our block device file by linux we can run #fdisk command as-

#fdisk -l

Here, we will get information about USB device as follow:

DeviceBoot	Start	End	Blocks	Id	System
/dev/sda1	1	7301	58645251	bW95	FAT32

After that by using #mount command we can mount USB device as follow:

#mount /dev/sda1 /mnt/myusb

To unmount usb we can use following command:

#umount /mnt/myusb **OR** #umount /dev/sda1

User and group management in Linux:

- In Linux, user and group management is an important aspect of system administration.
- One of the major work of a System Administrator is User & Group Management. Here, user and group management is nothing but creating new users & new groups, delete existing users and groups, providing users access to a group or folder, assigning specific permission to user and group etc.

User Accounts:

- A user is another name of an account capable of logging into a computer.
- A user is an individual who uses the available hardware and software resources.
- A user account is a set of information that defines a user's identity on the system. Each user has a unique username and a password.
- Every user has its own home directory, for user 'root' it's '/root/' & for 'general' users it located in '/home/'.
- In Linux, **user accounts are stored in the "/etc/passwd" file**. That is **/etc/passwd** stores all detail information of all linux user.
- In Linux, **user password are stored in the "/etc/shadow" file**. That is **/etc/shadow** stores all detail information of all linux user.
- In Linux, all user has its own userID. By default 0 to 999 User ID (UID) are reserved for system purpose, when we create new user account then it got user id from 1000+ onwards.
- **Type of Users in Linux:**
- Linux systems have two types of users, one is general i.e. normal user and another is system user which is root (super user).
- **System User:**
  - It is created by OS automatically.
  - 'root' users have access to anything & everything on the Linux system.
  - System user account by default have user id between 0 to 999.
  - Eg.: 'root' is system user account which is automatically created at the time of Linux OS Installation. It having by default **userID 0** (zero) and home directory **/root/**
- **Normal User (General User):**
  - It is created by Administrator user i.e. by the 'root' user.
  - General users have limited access to the Linux system.
  - Normal user account by default got user id 1000+ onwards.
  - Eg: Student, Sachin, Ajay, Sangola etc.

Linux command to manage user accounts:

Commnad	Use
useradd	It is used to add or create new user account
usermod	It is used to modify existing user account
userdel	It is used to delete existing user account
passwd	It is use to create or change user account password

In Linux, all user account details by default stored in following file:

User account Properties: /etc/passwd

User Password Properties: /etc/shadow

Let's see some user management commands in details:

1) useradd command:

It is used to add or create new user account.

Syntax: useradd userName

Example: useradd ajay

2) grep command:

It is used to check user account properties.

Syntax: grep username Userdirectory

Example: **grep ajay /etc/passwd**

In above example, 'grep' search 'ajay' string into 'passwd' file where stored the user account details and display the whole line regarding to 'ajay' user as follow:

ajay:x:1001:100: :/home/ajay:/bin/bash

Here,

ajay – It is user name (user login name)
x – It shows mask (encrypted) password
1001 – It shows user account ID (UID)
100 – It shows user account primary group ID (GID)
:: – It shows user comment or full name (by default empty)
/home/ajay – It shows user home directory
/bin/bash – It shows user login shell

3) passwd command:

It is used for Create/change user account password.

Syntax: passwd userName

Example: passwd ajay

Note: After running this command, type a password which we want to set for user.

4) grep command:

grep command is also used to check user password properties.

Syntax: grep username PasswordDirectory

Example: grep ajay /etc/shadow

Aftr running above command, user password properties shows in following format, each filed is separated by : (colon)

ajay: !6$^6ek23u%78 : 19302:0:99999:7: : :

Here,

ajay – It shows user name (login name)
!6$^6ek23u%78 – It shows user encrypted password
19302 – It shows number of days since 1969.
0 – It shows minimum life of password.
99999 – It shows maximum life of password.
7 – It shows warning days before expire password.
::: – It show password expiry days or inactive period of password.

5) su command:
This command is used for switch from one user account to another user account:
Syntax: su userName
Example: su ajay
The above command switches from current user to 'ajay' user.

6) exit command:
It is used to logout from currently login user account.
Syntax: exit
or
press "ctrl+d" shortcut key for logout.

7) userdel command:
It is used for delete user account.
Syntax: userdel userName
Example: userdel ajay
Note: With above command we can delete user account but its home directory still remain, for delete user account with home directory also, we can use following command.
userdel -r ajay

Modifying existing user account properties, Linux provide following commands: (usermod command with all options)

I) usermod –l command:
It is used to change user name (user login name).
Syntax: usermod -l New_Name Old_Name
Example: usermod -l vijay ajay

II) usermod -u command:
It is used to change user ID.
Sysntax: usermod -u NewID UserName
Example: usermod -u 2025 vijay

III) usermod -c command:
It is used to set comment/fullname to user account.
Syntax: usermod -c "comment" UserName
Example: usermod -c "Junior Software Developer" vijay

IV) usermod -d command:
It is used to set new home directory for user.
First we create directory by running commnad: mkdir -p /india/pune
Syntax: usermod -d NewHomeDirectory UserName
Example: usermod -d /india/pune vijay

V) usermod -s command:
It is used to change login shell.
Syntax: usermod -s shell userName
Example: usermod -s /sbin/nologin vijay
Note: If we set /etc/nologin shell to user account then user can't login into bash shell and no other command run properly.

VI) usermod –e command:
It is used to set expiry date to password.
Syntax: usermod -e "Date" userName
Example: usermod -e "2021-12-30" vijay

VI) usermod -L command:
It is used to lock user account password. And **place or add** exclamation mark ("!") in the second field. It means user now locked.
Syntax: usermod -L userName
Example: usermod -L vijay

VIII) usermod -U command:
It is used to *unlock user account password*. And **removes** exclamation mark ("!") in the second field. It means user now locked.
Syntax: usermod -U userName
Example: usermod -U vijay

Groups:

- A group is a collection of user accounts.
- Groups can be used to simplify permissions management, allowing multiple users to share access to files or directories.
- Group is collection of user accounts which is very useful to administrator for manage and apply permission on number of users.
- **Records for all the groups i.e. group account details** are kept in **/etc/group** file.
- **Note:** Whenever a new user is created or added then by default it added to group '**users**'

• Linux command to manage groups:

Let's see some Group management commands in details:

1) groups command:
It is used to display groups for a user.
The groups command prints the available groups of a user.
Syntax: groups
root dialout video users adm cdrom sudo dip plugdev
Without any arguments, it prints the groups a user belong to.
We can also specify the user name(s) to display groups of specific users as follow:
Example: groups abhishek prakash
The above command shows output as follow showing username with their groups:
abhishek : abhishek adm cdrom sudo dip plugdev lpadmin sambashare docker
prakash : prakash sudo

2) groupadd command:
It is used to add or create new Group account.
Syntax: groupadd groupName
Example: groupadd developergrp
Also, we can also check group account properties using following command:
Syntax: grep groupName /etc/group
Example: grep developergrp /etc/group
It shows output as follow:
developergrp:!:1001:ajay,vijay,hary
Here,
developergrp – It is name of group account.
! – It shows mask value (permission) of group

1001 – It shows group ID
ajay,vijay,harry – These are the names of group member.

3) sg command:
It is used for switch to another group account.
Syntax: sg groupName
Example: sg developergrp
The above command switches from current working group into 'developergrp'
Note: Before switching to created group, if we run 'groups' command then it shows all groups belongs to user, except that newly created group. To show newly group, we have to switch to newly created group using 'sg' command then it will shows all groups including newly created group also.

4) useradd -G command:
This command creates new user and then add it in mentioned group.
We use 'useradd' command as follow:
Syntax: useradd -G groupName username
Example: useradd -G developergrp college
The above command adds user 'college' to 'developergrp' group.
Here, -G used for supplementary group.

5) groupmod -g command:
It is used to modify group ID.
Syntax: groupmod -g NewID groupName
Example: groupmod -g 6523 developergrp

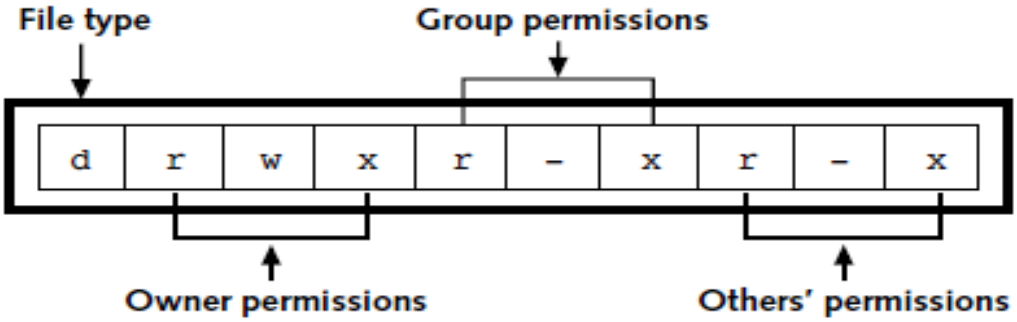
6) groupmod -n command:
This command is used to modify group name.
Syntax: groupmod -n new_groupname old_groupname
Example: groupmod -n computer developergrp
The above command modifies 'developergrp' name to 'computer'.

7) groupdel command:
It is used to delete group account.
Syntax: groupdel groupName
Example: groupdel computer

Assigning Permissions to user and group:

- In Linux, permissions can be assigned to users and groups to control access to files and directories.
- There are three types of permissions that can be assigned: read (r), write (w), and execute (x).
- Here are the steps to assign permissions to users and groups in Linux:

1) Identify the file or directory for which you want to assign permissions.
2) Use the **ls -l** command to display the current permissions for the file or directory.
3) The permissions are displayed in the following format: **-rwxrwxrwx**



Here,

- The first character indicates the type of file (e.g., **-** for a regular file, **d** for a directory, **l** for a symbolic link, etc.), and the remaining nine characters represent the permissions for the owner, group, and others.
- The Next three characters (rwx) represent the permissions for the owner.
- Then the next three characters (rwx) represent the permissions for the group.
- And the last three characters (rwx) represent the permissions for others.

4) To change the permissions **for the owner**, use the **chmod** command followed by the desired permissions and the filename.
   For example, to give the owner of a file read, write, and execute permissions, use the command
   Example: chmod **u**+rwx filename
5) To change the permissions **for the group**, use the **chmod** command followed by the desired permissions and the filename, but **replace u with g**.
   For example, to give the group read and execute permissions, use the command
   Syntax: chmod **g**+rx filename
6) To change the permissions **for other users**, use the chmod command followed by the desired permissions and the filename, but **replace u with o**.
   For example, to remove all permissions for others, use the command
   Syntax: chmod **o**-rwx filename.
7) You can also use numeric values to set permissions. **Each permission is assigned a numeric value: read (4), write (2), and execute (1).** The sum of the numeric values is also used to set the permissions.
   For example, to give the owner read and write permissions (4+2=6), and the group and others read permissions (4), use the command chmod 644 filename
8) Following table shows permission symbol, permission type:

Permission Symbol	
+	Add permission
-	Remove permission
=	Assign permission

Symbolic Value	Octal Value	Permission Type
r	4	Read
w	2	Write
x	1	Execute
	0	No permission

Let's see several example of access permissions with their meaning:

Command	Meaning
# chmod u+x bca	Add (+) executable permission for owner or user (u) to directory bca
# chmod ugo+x bca	'ugo' combines all three categories user, group,others & assign execute permission to bca director
# chmod go-rw bca	removes (-) read & write permission of group & others to bca directory
# chmod u=x, go=rw myfile	Assign execute permission for user and read, write permission for group & others to myfile.

# chmod 777 mydata	Set all permissions to a mydata file whereas previous all permissions removed.
# chmod g+rx-w home/sangola/mca	Set read and execute permission and remove write permission for group for the directory home/sangola/mca
# chmod 750 home/sangola/Desktop	It gives the **owner all permissions** (4+2+1=7), and read, execute permissions (4+1=5) **to the group** and **others cannot read,write,execute** permissions (0) to the directory home/sangola/Desktop

File and Directory Permissions: (File and Directory modes)

- In Linux there are set of rules for each file and directory that defines who (owner,group,others) can access file or directory, and how they can access it.
- These rules [i.e. rwx] are called as file permissions or file modes.
- In linux we can set read, write, execute permissions for both directory and file using **chmod,** and we can change owner using **chown** and we can change group using **'chgrp'** command. Let's see these command in details:

1) chmod command: (Change mode)

- chmod is a Linux command that stands for "change mode" and it is used to change the permissions (owner or group or other user's) for the file or directory.
- It allows you to set or modify the read, write, and execute permissions of a file or directory for the owner, group, and other users.
- The syntax for the chmod command is:

chmod mode fileORDirectory

Here,
mode argument specifies the permissions you want to set, and can be expressed in either symbolic or numeric form.
In symbolic form, the permissions are represented by a combination of letters and symbols, where:

u refers to the file owner.
g refers to the file's group.
o refers to all other users.
a refers to all users.

The symbols used are:

+ to add a permission.
- to remove a permission.
= to set the permission exactly as specified.

(**Note:** Using = symbol, previous permissions are removed and specified permissions are assigned)
Also, we can write some integer values corresponding to mode (read, write, execute) as follow:

Mode	Octal Value	Meaning
r	4	Read
w	2	Write
x	1	Execute
	0	No permission

We discussed several examples of chmod command in following table:

Command	Meaning
# chmod u+x bca	Add (+) executable permission for owner or user (u) to directory bca
# chmod ugo+x bca	'ugo' combines all three categories user, group,others & assign execute permission to bca director
# chmod go-rw bca	removes (-) read & write permission of group & others to bca directory

# chmod u=x, go=rw myfile	Assign execute permission for user and read, write permission for group & others to myfile.
# chmod 777 mydata	Set all permissions to a mydata file whereas previous all permissions removed.
# chmod g+rx-w home/sangola/mca	Set read and execute permission and remove write permission for group that uses directory home/sangola/mca directory.
# chmod 750 home/sangola/Desktop	It gives the **owner all permissions** (4+2+1=7), and read, execute permissions (4+1=5) **to the group** and **others cannot read,write,execute** permissions (0) to the directory home/sangola/Desktop

Viewing Permissions of files or directory:

- **ls command:**

- Basically the ls command in Linux is used to list the files and directories in the current working directory or in a specified directory with information such as permissions of the file or directory , owner of file, group of file, size of file in block, date and time of modification and file name etc.
- **ls –l** is a quick and easy way to list a file's or directory permissions with other details.
- For example, to view the permissions of myfile.txt, you could use the command:

ls -l myfile.txt

The output of above command is:

-rwxrw-r-- SY ECS 27 Feb 15 14:35 myfile.txt

The meaning of above output is explained in following table:

-	The first character represents the file type: "**-**" for a regular file, "**d**" for a directory, "**l**" for a symbolic link.
rwx	The next three characters represent the permissions for the **file's owner**: in this case, the owner may **r**ead from, **w**rite to, or e**x**ecute the file.
rw-	The next three characters represent the permissions for members of the file **group**. In this case, any member of the file's owning group may **r**ead from or **w**rite to the file. The final dash is a placeholder; group members do not have permission to execute this file.
r--	The permissions for "**others" (everyone else)**. Others may only **r**ead this file.
SY	The file's owner.
ECS	The group to whom the file belongs.
27	The size of the file in blocks.
Feb 15 14:35	The file's mtime (date and time when the file was last modified).
myfile.txt	The name of the file.

2) chown command: (Change owner)

- The chown command in Linux is used to change the ownership of files or directories.
- The owner of a file or directory determines who has the permissions to read, write, or execute the file or directory.
- The chown commands transfers control over a file to another user. And change the owner of a file assigns full control to the new owner.
- The chown command is a powerful tool for managing file and directory ownership, and can be used to grant or revoke permissions to specific users or groups.
- The syntax for chown command is:

chown new_owner FileOrDirectory

Here,

The new_owner parameter specify the new owner for the file or directory.

Examples:

1) chown Ajay example.txt

The above command changes the owner of a file named ‘example.txt’ to a user named ‘Ajay’.

2) We can also change **owner and group** of file using chown command:
Syntax: chown new_owner: new_group file(s) or directory
E.g. chown Ajay:mygroup myfolder
The above command changes the owner and group of a directory named 'myfolder' to a user named 'Ajay' and a group named 'mygroup'

3) chgrp command: (Change group)

- The chgrp command in Linux is used to change the **group ownership** of files and directories.
- The group ownership of a file or directory determines the group of users that has the permissions to read, write, or execute the file or directory.
- This command used for change the group owner of the files. A user can change the group owner of a file, but only to a group to which he/she also belongs to.
- The chgrp command is also a useful tool for managing file and directory ownership.
- The syntax for chgrp command is:

  chgrp new_group file(s) or directory

Here,
The 'new_group' parameter specifies the new group for the file or directory.
Examples:
1) chgrp mygroup example.txt
The above command changes the group ownership of a file named 'example.txt' to a group named 'mygroup'
2) chgrp mygroup myfolder
The above command changes the group ownership of a directory named 'myfolder' to a group named 'mygroup'

Assignment- I

1) What is Operating system? Give introduction to Linux operating system.
2) List out features/characteristics/properties of Linux
3) Briefly explain architecture of Linux.
4) What is Shell? List out features of shell.
5) Explain different types of shell along with their features.
6) What is environment? What are the working environment supported by linux.
7) What is installation and configuration of linux? List out steps involved in it.
8) What is Linux file system? List out commonly used directories in linux file system with their use.
9) Explain different linux file system parts.
10) Write the difference between windows and linux.
11) What is user account? What are the types of user account?
12) Explain different user management commands with example.
13) What is group? Explain different group management commands with example.
14) Explain format of permission assigned to user or group.
15) Explain following file & directory management commands with example:
    I) chmod II) chown III) chgrp
16) Write short note on following:
    a) History of linux
    b) Linux Distro
    c) GNOME
    d) KDE

e) Xface4

f) Linux file types. g) ls command h) Mounting and Unmounting devices.

Practical Assignment- I

Que 1) Perform following task on linux shell and **write output using grep command**.

1) Create two new user account one is 'RCB' and other is 'CSK'
2) Give appropriate password for 'RCB' and 'CSK' account
3) Show user account details of 'RCB'
4) Show user account details of 'CSK'
5) Show password details of 'RCB'
6) Show password details of 'CSK'
7) Change the user name of 'CSK' to 'MI' and 'RCB' to 'KKR'
8) Change user id of 'MI' to 5678
9) Change user id of 'KKR' to 3456
10) Set comment "Mumbai Indians" to 'MI' account
11) Set comment "Kolkata Knight Riders" to 'KKR' account
12) Set '/cricket/mumbai' home directory to 'MI' account
13) Set '/cricket/Kokata' home directory to 'KKR' account
14) Change login shell of 'MI' account to '/sbin/nologin'
15) Set password expiry date '2024-08-05' to 'MI' account
16) Set password expiry date '2025-06-20' to 'KKR' account
17) Lock user 'MI'
18) Lock user 'KKR'
19) UnLock user 'MI'
20) UnLock user 'KKR'
21) Delete user account 'KKR'

Que 2) Perform following task on linux shell and **write output using grep command**.

1) Show the available groups for user
2) Add or create two groups 'arts' and 'science'
3) Add 'BA-I','BA-II','BA-III' user account to group 'arts'
4) Add 'ECS-I','ECS-II','ECS-III' user account to group 'science'
3) Display group details of 'arts'
4) Display group details of 'science'
7) Display groups of 'BA-II' and 'ECS-III' account
8) Modify groupID of 'arts' group to 4545
9) Modify groupID of 'science' group to 5555
10) Alter the group name of 'arts' to 'NewArts'
11) Alter the group name of 'science' to 'NewScience'
12) Delete group 'NewArts'

Que 3) Perform following task on linux shell and **write output using ls –l command**.

1) Create 'MainFolder' on desktop.
2) Create new two folders 'Fold1' and 'Fold2' inside 'MainFolder'
3) Create new two document files 'Doc1' and 'Doc2' inside 'MainFolder'

4) list the directories and files of 'MainFolder'
5) list the directories and files of 'MainFolder' with permission format and other details
6) Add Read and execute permission to owner of file 'Doc1'
7) Add Read, write and execute permission to group of directory 'Fold1'
8) Add write and execute permission to other users of file 'Doc2'
9) Add Read, write and execute permission to all of directory 'Fold2'
10) Remove Read and execute permission to owner of file 'Doc1'
11) Remove Read, write and execute permission to group of directory 'Fold1'
12) Remove write and execute permission to other users of file 'Doc2'
13) Remove Read, write and execute permission to all of directory 'Fold2'
14) Set or Assign write and execute permission to owner of file 'Doc1'
15) Set or Assign Read only permission to group of directory 'Fold1'
16) Set or Assign Read and execute permission to other users of file 'Doc2'
17) Set no any permission to directory 'Fold2'
18) Set all (rwx) permission to directory 'Fold2'
19) Add owner's read, execute permission, add group's write permission and add other user's read, write, execute permission for the directory 'Fold2'
20) Remove owner's read permission, remove group's write permission and remove other user's write, execute permission for the directory 'Fold2'
21) Change owner of 'Doc1' file to 'Mahendra'
22) Change owner of 'Doc2' file to 'Mahesh' and group to 'NewScience'
23) Change group name of 'Fold1' to 'NewScience'