KODEKLODE LINUX

What is Linux??

- Linux is **a free, open-source operating system (OS) that's used on many devices, including personal computers, cell phones, and supercomputers**.
- It is known for its security, flexibility

Why we use linux?

Why do we care about operating system ?

Linux is by far the most widely used server operating system in the world, covering all its different versions and flavors like Ubuntu, CentOS, Red Hat. According to recent reports, all 500 of the fastest supercomputers run on Linux, 96.3% of the top websites are powered by Linux, and surprisingly, 86% of all smartphones use Linux-based operating systems. This shows how important and popular Linux is in modern technology

How does this relate to devops?

In the cloud and DevOps world, most new tools are first built for Linux. This is because Linux is widely used in servers and data centers. Later, some tools get support for Windows, but Mac is rarely used for servers.

For example, Docker, which helps run apps in containers, was only available on Linux for many years. Only recently did it start working on Windows.

When it comes to Kubernetes, which manages containers, the main part (called the control plane or master nodes) can only run on Linux. Even if your apps run on Windows servers, you still need Linux for the control plane. There are no plans to make a Kubernetes cluster that works only on Windows.

So, when you learn Kubernetes and set up a cluster, you always use Linux for the control plane. This is why Linux is very important for DevOps and cloud work.

Working in a Windows-dominant environment and switching to Linux can bring some challenges, including:

- The Linux command line is text-based and works differently than Windows, so it can feel confusing.
- Linux organizes files and folders in a different way, so finding things takes some getting used to.
- Text editors like vi or vim are keyboard-focused and not like Windows editors.
- There are many types of Linux (Ubuntu, CentOS, etc.), each with its own tools and commands.
- Installing software can be tricky because Linux uses different package managers (like apt, yum, rpm) and sometimes you get errors with missing files or dependencies.
- Setting up networking between Linux virtual machines can be harder than in Windows.
- Linux has strict permission settings, so you might run into issues with files or folders you can’t access.
- Without regular practice, it’s easy to forget commands and how things work.

What is linux shell?

The linux shell is a program that allows text-based interaction between the user and operating system. The very first directory you are taken to is your home directory.

All types of shells in Linux??

- **Bash (Bourne Again Shell): is the most popular shell in Linux. It is widely used because it has useful features like it suggests commands as you type and lets you create multiple files or folders at once, which are not available in some other shells. Most Linux systems use Bash as the default shell.**
- **Zsh (Z Shell)**:
- **Ksh (Korn Shell):**
- **Csh (C Shell)**:

How can you check which shell you are currently using in Linux?

```
echo $SHELL  #$SHELL  is a environment variable
```

How do you change shell?

First check where the Bourne shell is installed (usually /bin/sh)

```
cat /etc/shells #shows a list of all the available shells on your Linux system.
#The /etc/shells file is a system configuration file that lists all valid login shells on y
our Linux system.
```

Change shell:

```
chsh -s /bin/bash
#Replace /bin/bash with the path of the shell you want (e.g., /bin/zsh for Z shell).
# -s stand for shell
```

Difference between a variable and an environment variable??

- A variable is like a small box where you can store a piece of information, like a name or a number, for use in your current program or terminal. If you close the program, the box disappears.
- An environment variable is a special kind of box that can be seen and used by many different programs and processes. It stays available as long as your computer is running and can be used by any program that needs it.
- So, use a variable for temporary storage in one program, and use an environment variable to share information between different programs.

How to create and share environment variable from one program to another??

On Linux/macOS:

```
export MY_VAR=value
```

If you use `export MY_VAR=value`, the variable works in all your scripts and programs. If you just write `MY_VAR=value`, it only works in the current shell and not in others. To make the

variable work every time you log in, add it to the `.profile` or `.pam_environment` file in your home folder. Then it will be set automatically whenever you start your computer or open a new session.

On Windows:

```
set MY_VAR=value
```

How to check value of environment variable?

```
echo $variablename
```

How to see the list of all environment variable??

```
env
```

How to check path?

```
echo $PATH
```

How to check logname?

```
echo $LOGNAME
```

History command?

To get a list of previously run commands

```
history
```

What “which” command do?

If you just type ”**which**” in the terminal, it will show you where the ”**which**” command itself is located on your system, **like ”/usr/bin/which”**. This tells you the file path of the which command, not any other command. To find where other commands are, you need to type ”**which followed by the command name”, like ”which ls”** . This helps you

know which version of a command your system is using and makes it easy to find programs. If you want to see all possible locations for a command, you can use **which -a command**

How to set the path in “which” command show path is not define?

1. Find the directory where the command is located (for example, `/home/user/bin`).
2. Add this directory to your `PATH` by running:

```
export PATH=$PATH:/path/to/directory
#Replace /path/to/directory with the actual path.
```

To make this change permanent??

```
echo 'export PATH=$PATH:/path/to/directory' >> ~/.profile
```

Uptime command??

This is used to print information about how long the system has been running for since the last reboot (means to restart your computer or server), along with other information

```
uptime
#output: 18:35:49 up 13 min, 1 user, load average: 0.00, 0.00, 0.00
```

The uptime command shows four main things:

- Current time: The time when you ran the command (e.g., 18:35:49).
- System uptime: How long the system has been running (e.g., up 13 min means the system has been on for 13 minutes).
- Number of users: How many users are currently logged in (e.g., 1 user).
- Load average: The average system load over the last 1, 5, and 15 minutes (e.g., 0.00, 0.00, 0.00 means the system is not busy).

  1 minute: Tells you how busy your computer is right now.

  5 minutes: Shows your computer busy for short time

  15 minutes: Shows your computer busy for a long time or not

  This helps you quickly check system activity and performance

Scenario:

You check the load average on your computer and see:

load average: 2.50, 1.20, 0.50

Question:

1. What does each number (1 minute, 5 minutes, 15 minutes) tell you about how busy your computer has been?

Ans: 1 minutes(2.50): Your computer is very busy right now

5 minutes (1.20): Your computer has been busy recently, but not as much as it is now.

15 minutes (0.50): Your computer was less busy in the past

2. How you figureout this?

Ans: Your computer is very busy right now because 2.50 is higher than 1.20 and 0.50

3. If your computer has 2 CPU cores, is it overloaded right now?

Ans: Yes, because 2.50 is higher than 2, which means there are more processes waiting than CPU cores available.

4. What is the meaning of 2 CPU cores?

Ans: A "core" in a computer is like a separate mini-processor inside the CPU. Each core can run its own tasks at the same time, allowing your computer to handle two things simultaneously. For example, one core could run a web browser while the other runs a music player, making your system more efficient and faster at multitasking.

Question: After checking load average what we do next step?

- If your computer has 2 CPU cores, a load average above 2 means it’s overloaded right now (2.50 > 2). This can cause slow performance or delays
- Use commands like ”top” to see which programs are using the most CPU. You may need to close or restart some of them.
- If load stays high, upgrade: the Adding more CPU cores and RAM

Question: How to add more cpu and ram?

Step 1: Stop Your Instance

- Go to the AWS Management Console.

- Open the EC2 dashboard.
- Select your running instance.
- Click Actions → Instance State → Stop.

Step 2: Change Instance Type

- With the instance stopped, select it.
- Click Actions → Instance Settings → Change Instance Type.
- Choose a new instance type with more CPU and RAM (for example, switch from t2.micro to t2.medium or t3.large).
- Click Apply.

Step 3: Start Your Instance

- After changing the instance type, select your instance.
- Click Actions → Instance State → Start.

Step 4: Verify Changes

- Once the instance is running, log in.
- Use commands like nproc (for CPU) and free -h (for RAM) to check the new resources.

Question: What nproc and free -h command do?

- The ”nproc” command shows the number of CPU cores (or processing units) available to your system. For example, if you run nproc and get 4, it means your system has 4 CPU cores.
- The ”free -h” command shows how much RAM (memory) your system has and how much is currently being used. The -h flag makes the output easy to read (in GB or MB).

Question: If your computer might get busy in the future, here’s what to do

- Adding more CPU or RAM before it gets busy.
- Keep your software updated and close programs you don’t need.
- Use commands like `uptime` or `top` to keep an eye on system performance and catch problems early

What are type of command in linux?

1. Internal or build-in Commands: Internal or built-in commands are part of the shell itself, so they work right away and don't need extra files. Examples include `cd`, `pwd`, and `echo`. These commands are always available when you use the shell and are very fast.
2. External commands are separate programs stored as files on your system, usually in folders like `/bin` or `/usr/bin`. When you run an external command, the shell has to find and start that file. Examples include `ls`, `mv`, and `date`. These commands might be slower because the shell needs to search for them first.

Note: To check if a command is internal or external, use the type command.

```
type pwd #output: pwd is a shell builtin
type date #output: date is /usr/bin/date
```

what mkdir(make directories) command do?

The mkdir command in Linux is used to create new directories (folders)

```
mkdir directories_name
mkdir snake frog  # This creates two directories: snake and frog (both at the same l
evel), not nested inside each other.
mkdir -p /home/bob/fish/salmon  # it creates all the directories in the path, even if t
he parent directories don’t exist yet.this is nested directories one inside another
#-p stand for parents
mkdir snake/fish #This creates a directory called fish inside the existing snake direc
tory. If snake doesn’t exist, you’ll get an error unless you use -p
```

What ls(Long list) command do??

ls is useful to list the contents of directory in the linux file system

```
ls
ls -a #list all files including hidden and hidden files are those that start with a dot (.)
ls -ltr #to do reverse and print files from oldest to newest, use the
============================================================
============================
```

```
ls -l
#You can tell if something is a file or a directory by looking at the very first characte
r in the output of ls -l:
#- If the first character is a -, it means it's a regular file.
#- If the first character is a d, it means it's a directory.
```

Where is a user's home directory located in Linux, and how is it named?

In Linux, each user has a home directory, which is usually created under the /home folder. The name of the home directory is the same as the username. For example, if the user is named "Bhoomi," their home directory will be /home/bhoomi. Every user has their own unique home directory, so no two users share the same home folder.

Why do we need home directory??

In Linux, the home directory is a special place where each user can store their personal files and folders. Every user has their own unique home directory, and only that user can access it. This means no other user can see or change your files unless you give permission. You can quickly find your home directory by typing `~` (tilde) in the terminal.

Note: Once you are logged into bash shell, the line you see is called the bash prompt

~ = present working directory

$ = User prompt symbol

What is absolute path and relative path?

- An absolute path is the full path to a file or directory, starting from the root directory (/). It always begins with a / and gives the complete address. For example, if you are in /home/bhoomi and want to go to a directory called asia inside your home directory, the absolute path would be /home/bhoomi/asia.
- A relative path is the path in relation to your current directory. It does not start with a / and is based on where you are right now. For example, if you are in /home/bhoomi, you can simply run ”cd asia” to enter the asia directory. Here, asia is a relative path because it is relative to your current location

Note: The alternative of “cd” this is “pushd” command

Difference between dir and pwd?

- The dir command is used to list the contents of a directory, showing files and folders inside it, similar to the ”ls” command in Linux.
- The pwd command stands for "print working directory." It shows the full path of the directory you are currently in. For example, if you are in /home/bhoomi, running pwd will display /home/bhoomi

What is mv Command??

Moves a file or directory from one place to another. The original is removed from its old location and placed in the new location. It can also be used to rename files

This command require two argument

1. The first argument is the source directory, current file location
2. The second argument is the destination directory, Where we want it to be moved

Using absolute path, this can be done easily using the command

```
mv /home/bhoomi/Europe/Morroco /home/bhoomi/africa/
```

or

Using relative path

```
mv Europe/Morroco Africa/
```

How to renamed directory name?

We use “mv” command

```
# mv source directory destination directory
mv Asia/India/Mumbai Asia/India/delhi
```

what is command, option, flags, switch ,argument??

```
cp -r folder1 folder2
```

```
#Option/Flag/Switch: Extra settings you add to change how the command works.
#Argument: The specific file, folder, or value the command works on.
```

- `cp` is the command (copy).
- `r` is the option/flag/switch
- `folder1` and `folder2` are the arguments (the folders to copy from and to).

So, options/flags/switches are extra instructions, and arguments are the things the command acts on.

What is cp command do??

Makes a duplicate of a file or directory. The original stays in place, and a copy is created in the new location.

```
# cp source directory destination directory
cp Asia/India/delhi/city.txt Africa/Egypt/Cairo
```

The main difference between cp and mv in Linux is

- **cp (copy)**: Makes a duplicate of a file or directory. The original stays in place, and a copy is created in the new location.
- **mv (move)**: Moves a file or directory from one place to another. The original is removed from its old location and placed in the new location. It can also be used to rename files

What rm command do??

```
rm Cairo
```

This will delete the file named Cairo from your current directory.

Note: The rm command deletes files permanently, so be careful. If the file is protected, you may need to confirm before deleting it

What cat command do?

The cat command is used to display the contents of a file. For example, cat filename shows what's inside that file

```
cat file.txt
cat Asia/India/Mumbai/City.txt
```

What is "cat > filename" command do?

```
cat > Asia/Egypt/Cairo/City.txt
# > redirection symbol. with help of this we change the text for the file city.txt within single line
#After writing text press ctrl+d
```

The cat > command is used to create a new file or overwrite an existing file with new content.

For example, cat > filename lets you type text, and when you press Ctrl+D, it saves that text in the file. If the file already exists, its content will be replaced

What touch command do?

The touch command in Linux is used to create empty files

```
touch myfile.txt
```

what vim command do?

The vim command opens the Vim text editor, which lets you create, edit, and manage text files directly from the termina

```
vim file.txt
```

Some basic vim commands:

- "i" to enter insert mode and start typing.
- "Esc" to exit insert mode and return to command mode.
- ":wq" to save changes and quit.
- ":q!" to quit without saving.

- “dd” to delete a line.
- “u” to undo the last change.
- “yy” to copy (yank) a line, p to paste it.
- :w= only save the file
- :q: quit

What “more” or “less” command do?

The ”more” command is the older, more basic pager.

- It displays content page by page. Users can move forward through the file by pressing the Spacebar (next page) or Enter
- Limitation: The original implementation of more did not allow scrolling backward. Once you moved to the next page, you could not go back to the previous one within the utility itself.
- Efficiency: It reads the entire file into memory before displaying it, which can make it slower for very large files compared to less

```
more file.txt
#[space]- scrolls the display, one screenful of data at a time
#[Enter]- scroll the display one line
#[b]- scrolls the display backwards one screenful of data
#[/]- search text
#use Q key to exit out
```

What “Less” command do?

The less command was created later as an improvement on more, providing more features and flexibility.

- Functionality: It allows for both forward and backward navigation. Users can use the Spacebar or PageDown to move to the next page, and the b key or PageUp to go back to the previous one. The arrow keys also work for line-by-line scrolling.
- Efficiency: It does not read the entire file into memory at once, only enough to fill the current screen. This makes it much faster for viewing large files (gigabytes in size).
- Other Features: It includes powerful search capabilities (using / followed by the search pattern), allows for real-time file monitoring (similar to tail -f), and offers a

variety of customization options.

```
less file.txt
```

Why is the Linux command-line interface important compared to the graphical interface?

The Linux command-line interface (also called the shell) lets you work on your Linux computer more effectively. While the graphical interface looks easier and more attractive for new users, it has limits in what it can do. The command-line is more powerful and gives you more control to perform tasks and manage your system.

What is graphical interface?

A graphical interface, also called a GUI (Graphical User Interface), is a way to interact with a computer using visual elements like icons, buttons, windows, and menus instead of typing commands. You can click on these elements with a mouse or touchpad to open files, run programs, or change settings, making it easier for beginners to use computers. Most operating systems like Windows, macOS have a graphical interface to help users work without needing to remember complex commands

why graphical interface not good?

A graphical interface (GUI) is not always good because it can be slow and doesn't let you do everything you need, especially on servers. It uses more computer power, which is not good for systems that need to run fast and smoothly.

Linux has a GUI to help beginners use the system easily by clicking icons and menus. But for advanced work, servers, and cloud environments, Linux mostly uses the command-line (CLI) because it is faster, uses less power, and gives more control over the system

What is operating system??

An operating system (OS) is the main software that makes your computer work. It manages everything like running programs, storing files, and connecting to devices. Examples are Windows, Linux, and macOS. The OS helps you use your computer easily by handling all the basic tasks behind the scenes.

What is linux kernel?

The Linux kernel is the main part of the Linux operating system. It connects your computer's hardware (like the CPU and memory) with the software (like apps and programs) so they can work together. It manages resources, runs programs, handles communication, and controls devices, letting applications work smoothly without needing to know the details of the hardware. You can think of it as the helper that makes sure everything runs properly

The kernel is responsible like

- Memory management- keeps track of how much memory is used to store what and where
- Process management- determine which processes can use the CPU, when and for how long
- Device Drivers- act as a mediator or an interpreter between the hardware and processes

```
uname #which kernel you use
uname -r #which kernel version you use
#output: 6.1.158-180.294.amzn2023.x86_64
#You are running a Linux kernel based on version 6.1.
#The numbers after (158-180.294) are patches and updates made by Amazon for A
mazon Linux 2023.
#amzn2023 means this is Amazon Linux 2023.
#x86_64 indicates it's for 64-bit Intel/AMD processors.
```

Memory is divided into two areas

- **Kernel space:** This is the protected area of memory where the kernel (the core of the operating system) runs and manages hardware, processes, and system resources. Only the kernel can access this space, which helps keep the system secure and stable. it is used by the operating system kernel like managing hardware, memory, and security.
- **User space:** This is the area where user programs and applications run. Each program gets its own part of user space, and cannot access kernel space or the memory of other programs. This separation keeps programs safe and prevents them from interfering with the system or each other. **User space** is used by regular

programs and applications, like web browsers, text editors. user space applications include programs that are written in java, python or any other programming language

- Where this data live: Most commonly, it is stored in memory and on disk. user programs get access to data by making special requests to the kernel called system calls.

why we use sudo?

We use sudo to do important tasks in Linux, like installing software or changing system settings. Regular users don’t have permission to do these things for security reasons. When you use sudo, you temporarily get extra permissions to do these tasks safely. This helps protect the system from mistakes or harmful changes

What is the use of sudo su?

sudo su is a Linux command that switches you to the root user (full admin) using your own user’s sudo permissions.

Note: After sudo su your prompt changes (often from $ to #) and every command you type has full power.

How to helps protect the system from mistakes or harmful changes??

Using sudo helps protect the system by making sure only authorized users can do important tasks. When you use sudo, you must enter your password, which proves you are allowed to do that action. This stops accidental or harmful changes, like deleting important files or changing system settings, because regular users can't do these things without permission.

Difference between sudo and sudo su?

1. Sudo <command>: says you are root only for that one command. After it finishes, you are back to normal user

How sudo command is safer?

- You add sudo in front of one command, for example:

```
sudo apt install nginx
```

- That command runs as root, then you are back to normal user.
- If that one command is wrong, it can still break things, but:
- Only that command has root power. After that you are back to the normal user
- Each sudo command is written to the logs with the exact user and command, so you can later check “who ran what”.

2. Sudo su : Says Opens a full root shell; you “become” root until you type exit.

How sudo su is riskier??

- When you run: sudo su
- you become root (superuser), and:
- Every command you type after that has full power.
- If you mistype a command, you can delete important files or break permissions for the whole system.
- You stay root until you type: exit , So the **whole session** is risky, not just one command.
- In the logs it mostly looks like just root doing things. It is harder to see which real user ran which dangerous command.

Linux commands for working with hardware

How to check how much space is used and how much is free in computer storage?

```
df -h
```

```
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           481M     0  481M   0% /dev/shm
tmpfs           193M  444K  192M   1% /run
/dev/xvda1      8.0G  1.6G  6.4G  21% /
tmpfs           481M     0  481M   0% /tmp
```

```
/dev/xvda128     10M  1.3M  8.7M  13% /boot/efi
tmpfs            97M     0   97M   0% /run/user/1000
```

- **Filesystem**: The device or special filesystem (like `devtmpfs`, `tmpfs`, or `/dev/xvda1`).
- **Size**: Total size of the filesystem.
- **Used**: Space currently used.
- **Avail**: Space still available.
- **Use%**: Percentage of space used.
- **Mounted on**: Where the filesystem is mounted (e.g., `/`, `/boot/efi`).

simple scenarios where you would use the df -h command

- If your computer is slow or you can't save new files, use `df -h` to check which drive is full.
- Before deleting old files, run `df -h` to see which folders are using the most space.
- Before installing a new program, check if you have enough free space using `df -h`.
- Regularly run `df -h` to monitor storage and avoid problems before they happen.

How can you find which files or folders are using the most space on your system?

```
du -h /path/to/folder #to see the size of a specific folder in a readable format (like MB or GB)
du -sh /path/to/folder # to get a summary of the total space used by that folder
du -a  #to list the size of every file and subdirectory, helping you find large files or f olders to clean up.
```

Difference between du and df?

- df (disk free): Shows the overall disk space usage for entire filesystems. It gives you a big-picture view of how much space is used and available on each drive.
- du (disk usage): Shows how much space individual files and directories are using. It helps you find which specific folders or files are taking up space

On which command we see the memory being used?

The `free -m` command shows how much memory (RAM) is being used and how much is free on your Linux system, with all values displayed in megabytes (MB). This helps you quickly see if your system has enough memory available for running programs or if it's running low

What “top” command do??

See the live list of running process, check how much cpu and memory each process is using

What is boot process?

The boot process is what happens when you turn on your computer.

There are four type of boot process

1. **BIOS POST (Power-On Self-Test):**

   When you turn on your computer, the BIOS checks if all hardware is working. If something is wrong, the computer stops and shows an error.

2. **Bootloader:**

   After the hardware check, the BIOS loads a small program called the bootloader (like GRUB2) from the hard disk. The bootloader lets you choose which operating system to start and then loads the Linux kernel into memory.

3. **Kernel Initialization:**

   The kernel (the core of Linux) is loaded and starts running. It sets up hardware, manages memory, and prepares the system for running programs. Once ready, it looks for the next step.

4. **Init Process:**

   The kernel starts the init process, which is usually systemd these days. Systemd starts up all the services and programs needed for you to use the system. In the past, a different process called SysV-Init was used, but systemd is faster because it starts services at the same time.

5. **Login Prompt:** Once everything is ready, you get a screen to log in and use your computer.

What is systemd?

Systemd is the main program that starts up your Linux computer. It loads all the necessary services and programs so your system is ready to use. It is faster than older systems because it starts everything at once, not one by one. Most Linux computers now use systemd

What systemd command do?

Used to manage services and the system

```
#Use this when you install something like Nginx, MySQL, Docker, etc., and you want to turn it on or off.
sudo systemctl start servicename
sudo systemctl stop servicename
sudo systemctl restart servicename
#If it’s failed, you read the status + logs to see why and then fix config and restart.
systemctl status servicename
#Use this when you want a service to start automatically after reboot (typical for we b servers, databases, monitoring agents).
sudo systemctl enable servicename  #
sudo systemctl disable servicename #
systemctl list-units --type=service  #Check what services are currently active.
```

What is the difference between start the service and enable the service?

- Starting a service means running it right now, but it won’t start automatically next time you restart your computer.
- Enabling a service means setting it up to start automatically every time your computer boots. Enabling does not start the service immediately; it just makes sure it will run when the system starts.

What is files?

In Linux, "files" are storage used to save information like text, programs

Types of files

1. Regular files: These store data like text, images, scripts, or programs. They are shown with a" -" in the ls -l output.
2. Directories: These are folders that contain other files and directories. They are shown with a "d" in the ls -l output.
3. Special files-

- Symbolic links: These are shortcuts to other files or directories. They are shown with an "l" in the ls -l output.
- Character device files: These represent devices that handle data byte by byte, like keyboards or mice. They are shown with a "c" in the ls -l output.
- Block device files: These represent devices that handle data in blocks, like hard disks. They are shown with a "b" in the ls -l output.
- Named pipes (FIFO): These allow processes to exchange data. They are shown with a "p" in the ls -l output.
- Sockets: These are used for inter-process communication. They are shown with an "s" in the ls -l output

Linux File System Structure

Folder	What It Stores?
`/` **(Root)**	The **main** directory (everything starts here).
`/home`	Stores **user files** (documents, downloads).
`/bin`	Basic **system commands** (`ls` , `cp` , `cat`).
`/sbin`	Commands for **admin users** (`shutdown` , `fdisk`).
`/etc`	Holds system configuration files. ex: User, network , system app
`/var`	Where variable data such as logs and databases are stored
`/usr`	User-installed **programs and apps**.
`/lib`	/lib is a folder that stores important system files to help Linux run commands and programs correctly
`/tmp`	Temporary files created by system and users
`/dev`	/dev is a special folder where Linux stores files that control hardware devices, like hard drives, USBs, and keyboards!

`/mnt` & `/media`	**External drives** (USB, CD).
`/boot`	Files needed to **start the system** (Linux Kernel, GRUB).
`/proc`	**System and process info**.
`/opt`	Third-party **software and applications**.
`/root`	Home folder for the **admin (root) user**.

Why is the File System Important?

- **Keeps files organized**
- **Controls file access (security)**
- **Manages storage devices (HDD, SSD, USB)**

File Path Format in Linux vs Windows ??

- **Linux paths** use `/` and start from `/` (root).
- Windows paths use \ and start from C:\ (drive letter).

Types of file system?

1. ext4(Fourth Extended File System): Default Linux file system
2. **XFS(Extended File System) :** Good for large files
3. Btrfs(B-Tree File System) : Supports backups & recovery
4. **FAT32(File Allocation Table 32-bit) :** Works with Linux, Windows & USB drives
5. exFAT(Extended File Allocation Table) : Best for USB drives & external storage

Note: For high-throughput and extensive parallel processing capabilities , XFS is typically more effective than ext4

How to check your linux Filesystem ??

- lsblk: Shows **file system type** for all storage devices.
- df -Th: Shows **file system type and disk usage**.
- cat/etc/fstab: Displays **file system details** used when the system starts.

==========

What is package manager?

A package manager is a tool that makes it easy to install, update, and remove software

- **apt-get (Advanced Package Tool):** Used in Ubuntu
- **yum (Yellowdog Updater, Modified):** Used in amazon
- **npm** (Node Package Manager) is a tool used in JavaScript and Node.js to install, update, and manage software packages and libraries

Difference between apt and apt-get?

apt and apt-get are both package managers used in Debian-based Linux systems (like Ubuntu) to install, update, and remove software. The main differences are:

- apt is a newer, user-friendly and already installed by default command. It provides a progress bar, and gives more helpful output during tasks. It’s meant for everyday users.
- apt-get is an older, low-level command. It’s more predictable and is often used in scripts or by system administrators because it remains backward compatible. It doesn’t show a progress bar and has less user-friendly output.

what is redhat?

Red Hat is a company that creates Linux-based operating systems for businesses and organizations.

what is Redis?

Redis is an open-source, in-memory data structure server, often used as a database, cache, or message broker. It stores data in memory, which makes it extremely fast for reading and writing. Redis supports various data types like strings, lists, hashes, sets, and more. It's widely used for real-time applications, caching, session storage, and messaging because of its speed and flexibility.

What is telnet?

Telnet is a simple tool that lets you connect to a remote computer over the network. Once connected, you can use keyboard commands to control the remote system, just like if you were sitting right in front of it. You can run programs, create or delete files, manage folders, start or stop services, and check if ports are open or closed. This way, you can do almost anything on a remote server, even if you are far away

What is nginx?

Nginx is a lightweight and fast web server that can handle many users at the same time. It is designed to serve static content (like images, HTML, CSS, and JavaScript) very quickly and efficiently. Nginx can also work as a reverse proxy, meaning it can forward requests from users to other servers and send back the response. This helps in managing high traffic and makes websites faster and more reliable. Many big websites use Nginx because it uses less memory and resources, scales easily(means that Nginx can handle more users or traffic by simply adding more servers or resources, without major changes to the system) , and provides a smooth experience even during high traffic.

Nginx Features?

- **Scalability:**

  Nginx can handle many users at once and can easily add more servers to manage increased traffic, making it great for high-traffic websites.

- **Load Balancing:**

  Nginx distributes incoming traffic across multiple servers, preventing any one server from getting overloaded and improving reliability.

- **Caching:**

  Nginx stores copies of frequently requested files, so it can serve them quickly without reprocessing, making websites faster.

- **Compression:**

  Nginx compresses files before sending them to users, reducing file size and speeding up page loads

At a time nginx is connected to many server??

Yes, Nginx can connect to many servers at the same time. This is especially useful for load balancing, where Nginx distributes incoming traffic across multiple backend servers. Nginx can also serve multiple websites or applications on the same machine by using server blocks (virtual hosts). This means you can run many services or websites behind a single Nginx instance, making it efficient and scalable

What is web server ?

A web server is **a system (hardware, software, or both) that stores, processes, and delivers web content like webpages, images, and videos to users over the internet via their browsers using the HTTP protocol**

Difference between Apache and nginx??

- Apache is flexible(creates a new process for each request) and good for dynamic websites. It uses more memory and CPU, especially with many users
- Nginx is fast and efficient for static websites and high traffic.

Dynamic website and Static websites?

Dynamic websites are sites that change content based on user actions, time, or other factors. Examples include:

- **Amazon:** Shows personalized product recommendations and updates prices and availability in real time.
- **Instagram:** Updates your news feed and shows new posts as they happen.

Static websites are sites that show the same content to every visitor and do not change unless manually updated. Examples include:

- **Personal blogs**
- **Company or portfolio websites**
- **Image or video galleries**

How we can check the size of a linux?

The “du” command, which stand for disk usage, is a popular command to inspect the size of the files.

```
du -sh test.img
#du: Stands for "disk usage" and is used to estimate the space used by files or dire
ctories.
#-s: Shows only the total size of the specified file or directory, not individual subdir
ectories or files.
#-h: Formats the output in a human-readable way (e.g., KB, MB, GB), making it easi
er to understand the size.
```

What is the use of “tar” command?

Tar is used to group multiple files or directories into s single file, it is especially useful for archiving data.

```
tar -cf test.tar file1 file2 file3
#-c: Stands for "create" — tells tar to create a new archive
#-f: Specifies the filename for the archive. In this case, it's test.tar.
#test.tar: The name of the output archive file.
#file1 file2 file3: The list of files to include in the archive.
```

```
tar -tf test.tar
#-t: Stands for "list" — tells tar to show the files inside the archive.
#-f: Specifies the archive filename, in this case test.tar.
```

```
tar -xf test.tar
#-x: Stands for "extract" — tells tar to unpack the contents of the archive.
#-f: Specifies the filename of the archive to extract, in this case test.tar.
```

```
tar -zcf test.tar file1 file2 file3
#tar: The command used to create, extract, or manage archive files.
#-z: Tells tar to compress the archive using gzip, resulting in a compressed .tar.gz fi
le
#-c: Stands for "create" — tells tar to create a new archive.
#-f: Specifies the filename for the archive, in this case test.tar.
```

What is compression?

Compression is the technique used to reduce the size by file

```
bzip2 test.img
gzip test1.img
xz test2.img
gunzip test.img
```

What locate command work?

The locate command is used to quickly find files and directories by name

```
locate city.txt
```

```
#example
sudo find /etc -type f -name "dummy.service" > /home/bob/dummy-service
#sudo: Runs the command with administrative privileges, ensuring access to all dire
ctories under /etc, even those restricted to regular users.
#find: The command used to search for files and directories.
##/etc: The starting directory for the search.
#-type f: Limits the search to regular files (not directories or links).
#-name "dummy.service": Specifies the filename to search for.
#>: Redirects the output (the list of matching file paths) to a file instead of displayin
g it on the screen.
```

What is the use to grep command?

The grep command is used to search for specific text patterns within files, such as words, phrases.

```
grep word_name file_name
grep second file.txt



#-i: Case-insensitive search.
#-n: Shows line numbers of matches.
#-r: Searches recursively in directories.
#-w: Matches whole words only.
#-v: Shows lines that do not match the pattern.
```

Redirect the output

The > operator redirects standard output (stdout) to a file, overwriting its contents if the file exists.

```
echo $SHELL > shell.txt #overwriting any existing content in the file
#output: /bin/bash
```

The >> operator appends stdout to a file, adding new content at the end.

```
echo "this is the bash shell" >> shell.txt
#appends the string "this is the bash shell" to the end of shell.txt without deleting w
hat was already there.
```

What is Command line pipes (|) ??

Command line pipes (|) are used to connect multiple commands together, so that the output of one command becomes the input for the next. command1 | command2 | command3, and each will process the data passed from the previous one

```
ls | grep ".txt"  #lists all files in the current directory and then filters only those with
a .txt extension
ps aux | grep "nginx" | awk '{print $2}' #shows all processes, filters for those name
d "nginx", and then prints only the process IDs
```

Network command

What ping command do?

The ping command tests network connectivity between your system and a target host (like google.com). It works by sending small packets. The size of each reply packet (e.g., 64 bytes). The ping command also show latency and Latency information helps you understand how fast your network connection is to a specific server or website. All values are between 1.46 ms and 1.58 ms, which means the network connection is fast and stable.

```
ping google.com
#Shows for each reply: bytes, time (latency), and TTL (time to live).
```

Question: If your network slow then?

- **Restart your network** : sudo systemctl restart NetworkManager

- If you’re using Wi-Fi, make sure there are no big objects blocking the signal
- If the problem continues, your provider might issues on their end.
- If your plan is too slow, ask your provider about faster options.
- Make sure no one is downloading large files or streaming videos, as this can slow down the network for everyone.

What telnet command do??

- The telnet command is used to check if a specific port on a remote server is open and responding.
- If the connection is successful, it means the port is open and the service is running. If it fails, the port might be closed or blocked by a firewall

```
telnet ip_address
telnet hostname port
#telnet devapp01-web 80
```

What netstat command do?

It helps you see what services are running, which ports are being used, and who is connected to your system

```
netstat -tuln
#-t: Show TCP connections.
#-u: Show UDP connections.
#-l: Show only listening ports (services waiting for connections).
#-n: Show numerical addresses remote IP and port (usually 0.0.0.0:* or :::* for listen
ing services).)
#:::*: The service listens on all IPv6 addresses, allowing connections from any IPv6
address.
#0.0.0.0:*: The service listens on all IPv4 addresses,allowing connections from any
IPv4 address.
```

Example of TCP and UDP?

- **TCP (Transmission Control Protocol)** : When you visit a website (like Google), your browser uses TCP to request and receive web pages. TCP service like http ,https, ssh , MySQL

- **UDP (User Datagram Protocol)**: When you watch a live video stream or play an online game, UDP is often used. It sends video or game data quickly, even if some packets are lost, because speed matters more than perfect delivery. Services like YouTube Live, Netflix, or Zoom use UDP.

How would you change ip address into domain name?

Use “cat >> /etc/hosts means” “append what I type next to the /etc/hosts file”.

- cat normally prints what it reads from keyboard (stdin) to the screen (stdout).
- >> /etc/hosts redirects that output and appends it to the file /etc/hosts instead of the screen.

After you run:

```
cat >> /etc/hosts
```

the terminal waits for you to type lines like

```
192.168.1.11   domain_name
#When you finish, press Ctrl + D to end input
```

What is the purpose of the /etc/hosts file?

On Linux and macOS, the file /etc/hosts is used by the system to turn a name (like bhoomi) into an IP address (like 192.168.1.11). then that Linux/macOS machine will understand name as ip address

What is curl command do?

1. Test if a website or API is reachable.

```
curl http://localhost:8080/health
```

2. **Download files**:

```
curl -o bhoomi https://www.google.com
#-o <filename> = save to this file name.
```

```
#This downloads the response and saves it as a file called bhoomi in the current fol
der.
```

The differences between telnet, ping, netstat, curl, and dig??

- **Telnet**: Connects to a remote computer or service to check if a specific port is open
- **Ping**: Checks if a device is reachable over the network by sending a small packet and waiting for a reply
- **Netstat**: Shows all network connections and listening ports on your own computer
- **Curl**: Downloads or sends data from/to a web server (like fetching a web page or sending a file to a website
- **Dig**: Looks up DNS records to find the IP address of a domain name

What are top level domain??

- .com: Dot com for commercial or general purpose (www.google.com)
- .net: for network purpose (infrastructure companies)
- .org: for non-profit organization(charities, communities, foundations, open-source projects, etc.)
- .edu: educational purpose(universities, colleges)

why we use www?

www → optional subdomain (often means “web site”)

- www is just a label in front of the main name.
- It usually means “this is the web site for this domain”.
- Technically you can use anything instead
  of www (like api.example.com, blog.example.com).
- Today many sites work both with and without www and just redirect one to the other.

Record type?

- A(IPV4): maps a domain name to an **IPv4** address.
- AAAA(ipv6): maps a domain name to an **IPv6** address.

- CNAME: maps one domain name (alias) to another domain name (real/original name).. For example, if you want www.example.com to go to example.com, you use a CNAME record. It points one name to another name instead of a number.

Where DNS server IP is stored

```
cat /etc/resolv.conf
```

If your site is not reachable, as a DevOps engineer??

- Verify if the server is running using commands like systemctl status for the web server (e.g., nginx, apache).
- check network connectivity using ”ping”
- **Check web server logs: “ cat /var/log/nginx/error.log”**
- Ensure ports used by your site (like 80, 443) are open and not blocked: using “netstat -tuln”
- Verify if server resources (CPU, RAM, disk space) are exhausted using ”top”,” free -m”, ”df -h”.
- Ensure the web server configuration files are correct and haven't been changed accidentally.
- Restart the web server (sudo systemctl restart nginx) if needed.

what is configuration file and how to check?

A configuration file is a text file that tells a web server (like Apache or Nginx) how to run —what ports to use, which files to serve, security settings, and more. These files are usually named something like httpd.conf (Apache) or nginx.conf (Nginx), and are located in directories such as /etc/apache2/ or /etc/nginx/ .

View the file

```
cat /etc/nginx/nginx.conf
```

Check for changes: Most web servers have a command to check if the config is correct

For Nginx: sudo nginx -t , For Apache: sudo httpd -t

Restart the server: sudo systemctl restart nginx

==========

What is router?

A router helps connect two networks together and enable communication between them and it gets two IPs assigned, one on each network

How to list and modify interfaces on the host

```
ip link
```

How to see the ip addresses assigned to those interface ??

```
ip addr
```

How to set ip addresses on the interface??

```
ip addr add 192.168.1.10/24 dev eth0
```

How to view the routing table?

```
ip route
```

How to add entries into the routing table?

```
ip route add 192.168.1.10/24 via 192.168.2.1
```

We will getting connection timeout error when we try to access this new url

This can happen due to a variety of reasons

- It could be an issue with your local interface not being connected to the network
- it could be your host not resolving the ip address of the hostname you were trying to connect to,

- it could be an issue with the route to the server
- it could be an issue with the server itself. Maybe the server itself has connectivity issues
- the software that hosts the service isn't functioning correctly

How to resolve this problem??

- Check your laptop's network: Run" ip link show enp1s0f1" . If it says UP, your laptop is connected.
- Check if the server's name is correct: Run "nslookup devapp01-web" . If it gives you an IP (like 192.168.2.5), your laptop knows where the server is.
- Try to reach the server: Run "ping 192.168.2.5" . If no reply comes back, the problem is not with your laptop or DNS, but somewhere in the network path.
- Find where the problem is: Run "traceroute 192.168.2.5" . This shows each step between your laptop and the server. If it stops at one step, that device (router, firewall, etc.) is causing the issue.
- Check the server: On the server, run netstat -tuln | grep :80 to check if the web server is running on port 80.
- Check the server's network: Run "ip link show " on the server. If the interface is DOWN, run "sudo ip link set <interface> up " to turn it ON.

Now the web server should be reachable. The main problem was the server's network interface was DOWN, so no one could connect to it.

LINUX account

- cat /etc/passwd: Displays a list of all user accounts on the system. Each line contains user details like username, user ID (UID), group ID (GID), home directory, and default shell .

```
Output: username:x:1001:1001:User Name:/home/username:/bin/bash
#username: The user's login name.
#x: Placeholder for the password (encrypted passwords are in /etc/shadow).
#1001: User ID (UID).
#1001: Group ID (GID).
#User Name: Full name or description.
```

```
#/home/username: Home directory.
#/bin/bash: Default shell.
```

- cat /etc/group: Displays a list of all groups on the system. Each line shows the group name, group ID (GID), and the users that belong to that group .

```
Output: groupname:x:1001:user1,user2
#groupname: Name of the group.
#x: Placeholder for group password (rarely used).
#1001: Group ID (GID).
#user1,user2: List of users in the group.
```

Account type?

1. User account: Regular people who need access to the system. They can use files and programs but can’t change system settings or other users’ files.
2. Superuser account: The most powerful account (usually "root"). It can do anything on the system, including changing settings, installing software, and accessing all files.
3. System account: Created when the OS or software is installed. These accounts are for running services (like SSH or mail). They usually don’t have a home directory and can’t log in.
4. Service account: Similar to system accounts, but specifically created for services (like NGINX). They let services run safely without giving them full system access.

What “ who “ command do?

Shows who is currently logged into the system, including their username, terminal, and login time .

What “last” command do?

Displays a list of recent logins, including who logged in, when, and from where. It also shows system reboots and shutdowns .

What useradd command do?

The useradd command is used to create a new user account on a Linux system. Only root or users with sudo privileges can run this command.

```
sudo useradd -m -s /bin/bash john #This creates a user named "john" with a home directory and sets their default shell to /bin/bash.
#-m: Creates the user's home directory.
#-s /bin/bash: Sets the user's default shell.
#-e : expiry date
```

How to check if the new user has been created successfully??

```
cat /etc/passwd | grep john
#This command searches for "john" in the /etc/passwd file, which contains all user account information.
```

Alternatively, you can also use:

```
id john
#This will show the user ID, group ID, and other info about the user
```

How to switch to new user?

By default, when you create a user with ”useradd”, no password is set, so you cannot log in until one is assigned.

So first you create password

```
sudo passwd john
```

After setting the password, try switching to john again with

```
su - john
```

How to go back to your original user (for example, ec2-user)

```
exit
```

Difference between “whoami” and pwd?

- whoami: Shows the username of the current user. For example, if you are logged in as john, running whoami will display john.
- pwd: Stands for "print working directory." It shows the full path of your current location in the file system. For example, if you are in /home/john, running pwd will display /home/john.

How to check which group, user part of?

```
group username
```

What “ls -l” command do?

```
ls -l #Lists detailed information for each file and subdirectory inside the current dire
ctory.
#You can tell if something is a file or a directory by looking at the very first characte
r in the output of ls -l:
# If the first character is a -, it means it's a regular file.
#If the first character is a d, it means it's a directory.
```

```
ls -ld
#Shows detailed information (permissions, owner, group, size, modification time) fo
r the directory itself, not its contents.
```

File Permissions:

Linux is a multiuser system. Every file and directory in your account can be protected from other users by changing its access permissions.

Type of permissions

- r - read → 4
- w - write → 2
- x - execute = running a program ⟶ 1
- - means no permission ⟶ 0

Total sum = 1+2+4 = 7 (all permissions)

Each permission (rwx) can be controlled at three levels.

- u - user = yourself
- g - group = can be people in the same Project
- o - other = everyone on the system

NOTE: When a new directory is created in Linux, the default permissions for the owner are typically read, write, and execute (rwx). This means the owner has full control over the directory,

provide full access to owners, group and other

```
chmod 777 <file/dir>
```

- First 7 → for User
- Second 7→ for group
- Third 7 → other group

```
chmod ugo+r-x test-file
#Adds read (r) permission and removes execute (x) permission for user (u), group (g), and others (o).
#If execute was already set, it will be removed; read will be added if not already pre sent.
```

```
chmod o-rwx test-file
#Removes read (r), write (w), and execute (x) permissions for others (o).
#Others will have no access to the file.
```

```
chmod u+rwx,g+r-x,o-rwx test-file
#User (u): Adds read, write, and execute permissions.
#Group (g): Adds read permission and removes execute permission.
#Others (o): Removes all permissions (read, write, execute).
```

How you give permission to files or folders in Linux

To give permission to files or folders in Linux, you use commands
like chmod and chown:

1. Change ownership (if needed):

   Use the chown command to assign a specific user or group as the owner.

Example:

```
sudo chown username:groupname filename
#Replace username and groupname with the actual names
```

2. Set permissions:

   Use the chmod command to set read, write, and execute permissions.

```
chmod u+rwx filename
# This gives the owner (user) read, write, and execute permissions.
```

3. For numeric mode (common for scripts):

```
chmod 755 filename
# Here,7 means read+write+execute for the owner,5 means read+execute for group and others.
```

4. Check permissions:

   Use the " ls -l" filename command to see the current permissions and ownership.

5. Best practices:

- Only give the minimum permissions needed.

```
chmod -R 755 foldername
#chmod changes permissions on files and folders.
#-R means "recursive," so it applies the permission change to all files and subfolder
s inside the specified folder.
#The owner (you) gets read, write, and execute permissions (7).
#Group members and all other users get read and execute permissions (5),but cann
ot write or modify files.
```

This is commonly used for folders because it lets everyone see and open files inside, but only the owner can add, delete, or change files. For example, if you run "chmod -R 755 myfolder", every file and folder inside myfolder will have these permissions.

Question: A new file called soccer has been created under the sports directory. It has full permissions, update the file so that the group and others only have read and execute permissions.

```
chmod 755 soccer
#Owner (user): read, write, execute (rwx)
#Group: read, execute (r-x)
#Others: read, execute (r-x)
```

What chown command do??

The `chown` command in Linux is used to change the owner and/or group of a file or directory.

```
chown [new_owner] file  #this change owner
#chown mercury soccer
chown [new_owner]:[new_group] file #this change owner and group both
#chown [new_owner]:[new_group] file
```