The **File Hierarchy Standard (FHS)** in Linux defines the structure and organization of directories and files in a Linux system. It provides a standard layout that allows different Linux distributions to be consistent and compatible with each other.

Linux File System Hierarchy (FHS):

Directory	Purpose
`/`	Root directory. The top of the hierarchy.
`/bin`	Essential binaries (commands like `ls`, `cp`, `mv`, etc.).
`/sbin`	System binaries (commands for system administration like `shutdown`, `fdisk`, etc.).
`/boot`	Boot loader files (Kernel and GRUB).
`/etc`	Configuration files for the system and services.
`/home`	Home directories for users (e.g., `/home/user`).
`/lib`	Shared libraries and kernel modules.
`/lib64`	64-bit libraries.
`/usr`	User programs and utilities (e.g., `/usr/bin`, `/usr/local`).
`/var`	Variable data (like logs, cache, and databases).
`/tmp`	Temporary files (deleted on reboot).
`/opt`	Optional software packages.
`/dev`	Device files (like `/dev/sda`, `/dev/null`).
`/mnt`	Mount point for external devices (like USB drives).
`/media`	Mount point for removable media (CD/DVD).
`/proc`	Virtual filesystem for kernel and process information.
`/sys`	Virtual filesystem for hardware and system information.
`/root`	Home directory for the root user.
`/srv`	Data for services like web servers.
`/run`	Runtime data for processes and services.

Real-life example:

If you want to install a program like Nginx:

- Configuration file: `/etc/nginx/nginx.conf`
- Binary file: `/usr/sbin/nginx`
- Log files: `/var/log/nginx/access.log`

Why FHS is important?

- Helps maintain **consistency across all Linux distributions**.
- Makes it easier for **developers and system administrators** to navigate the system.
- Improves **security and performance** by separating user files, system files, and logs

Complete Explanation of Each Directory in Linux File System Hierarchy (FHS):

1. `/` - Root Directory

- The **top-most directory in the Linux file system**.
- All other files and directories are located under this.
- Only the **root user** has full access to this directory.

2. `/bin` - Binaries

- Contains **essential commands and programs** that are used by both normal users and the system during booting.
- Examples: `ls`, `cp`, `mv`, `cat`, `echo`, etc.

☑ **Think of it as**: Basic tools for survival

3. `/sbin` - System Binaries

- Contains **system-level commands** that can only be run by the **root user (administrator)**.
- Used for **system maintenance** and configuration.

Examples:

- `shutdown`
- `fdisk` (Disk Partitioning)
- `ifconfig` (Network Configuration)

☑ **Think of it as**: Tools for controlling the system.

4. `/boot` - Boot Loader Files

- Contains **all files needed to boot the Linux system**.
- Includes the **Linux kernel**, **GRUB bootloader**, and **initial RAM disk (initramfs)**.

Files inside `/boot`:

- `vmlinuz`: Linux kernel file.
- `initrd.img`: Initial RAM disk.
- `grub/`: GRUB bootloader files.

☑ **Think of it as**: The heart of Linux startup.

5. `/etc` - Configuration Files

- Contains **all configuration files for system and services**.
- Almost every service like Apache, MySQL, and SSH has its configuration stored here.

Examples:

- `/etc/passwd`: User information.
- `/etc/fstab`: Filesystem mounting information.
- `/etc/nginx/nginx.conf`: Nginx configuration.

6. `/home` - User Home Directory

- Contains **personal files and settings of users**.
- Each user has a separate directory:
  `/home/user1`, `/home/user2`, `/home/ali`, etc.

Files inside `/home`:

- Documents
- Downloads
- Videos
- Hidden configuration files (like `.bashrc`).

☑ **Think of it as**: Your personal workspace.

7. `/usr` - User Programs

- Contains **user-installed programs and libraries**.
- Most of the **installed software and applications are stored here**.

Subdirectories inside `/usr`:

Subdirectory	Purpose
`/usr/bin`	User commands and programs
`/usr/sbin`	System commands
`/usr/local`	Locally installed software
`/usr/lib`	Libraries

☑ **Think of it as**: The storage for installed software.

`/lib` & `/lib64` - Libraries

- Contains **shared libraries (similar to Windows DLL files)** needed by system programs.
- `/lib64` is for **64-bit libraries**.

Examples:

- `libc.so`: C Standard Library.
- `libm.so`: Math Library.

☑ **Think of it as**: Brains for the programs to function.

9. `/var` - Variable Files

- Stores **data that changes frequently**.

Files inside `/var`:

Folder	Purpose
`/var/log`	System logs (e.g., `/var/log/syslog`)
`/var/cache`	Cache files
`/var/www`	Website files
`/var/mail`	Email storage

☑ **Think of it as**: A dynamic storage room.

10. `/tmp` - Temporary Files

- Stores **temporary files created by programs and users**.
- Automatically **deleted after reboot**.

☑ **Think of it as**: A trash bin that cleans itself.

11. `/dev` - Device Files

- Contains **special files that represent hardware devices** (like USB, hard disk, and keyboard).

Examples:

- `/dev/sda` → Main hard disk.
- `/dev/null` → Discard data.
- `/dev/tty` → Terminal.

☑ **Think of it as**: A bridge between software and hardware.

12. `/media` - Removable Media

- Mount point for **USB drives, CDs, and external drives**.

☑ **Think of it as**: A plug-and-play area.

13. `/mnt` - Mount Directory

- Used to **manually mount file systems or network drives**.

☑ **Think of it as**: A temporary storage point for other filesystems.

14. `/proc` - Process Files

- A **virtual file system** that stores information about **running processes and system information**.

Examples:

- `/proc/cpuinfo`: CPU details.
- `/proc/meminfo`: Memory information.

☑ **Think of it as**: A live monitor of the system.

15. `/sys` - System Information

- Another virtual file system that contains **information about hardware and kernel modules**.

☑ **Think of it as**: A control center for hardware.

16. `/root` - Root User's Home

- The **home directory for the root user (admin)**.

☑ **Think of it as**: The king's personal room.

17. `/opt` - Optional Software

- Used to install **third-party software and packages manually**.

☑ **Think of it as**: A playground for additional software.

18. `/run` - Runtime Data

- Stores **temporary data related to processes and services** that disappears after a reboot.

☑ **Think of it as**: A storage room for current system status.

19. `/srv` - Service Data

- Contains **data for services like web servers and FTP servers**.

☑ **Think of it as**: The warehouse for online services.

PRO TIP TO REMEMBER IT ALL:

"BEHT U V M D P S R"

B -> bin

E -> etc

H -> home

T -> tmp

U -> usr

V -> var

M -> mnt/media

D -> dev

P -> proc

S -> sys

R -> root