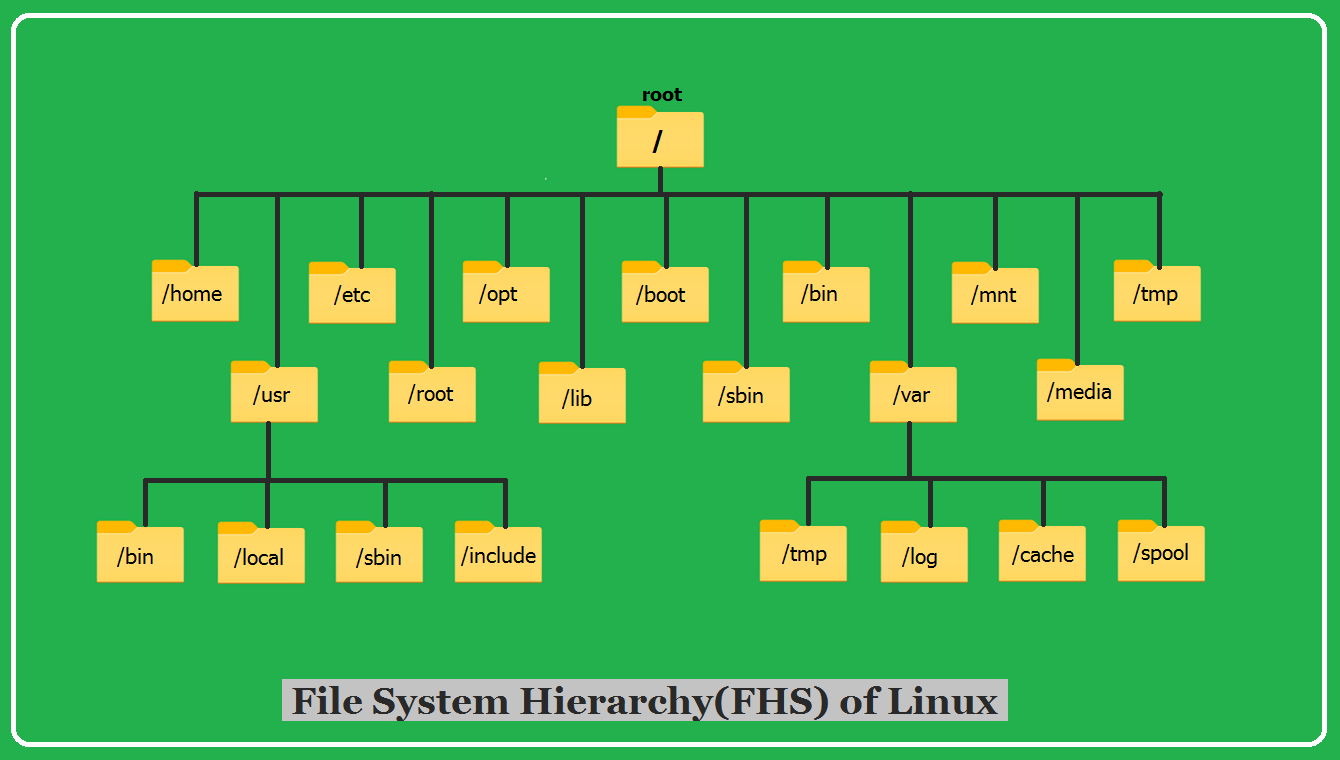
**1. Linux file system hierarchy**



**/ (Root)**

The top-level directory in the filesystem hierarchy. All other directories and files are subdirectories or files of the root directory.

**/bin (Binaries)**

Essential binary executables required by the system to boot, run, and repair the system.

**/boot (Boot Loader Files)**

Contains files needed for the boot process, such as the Linux kernel, initial RAM disk (initrd), and boot loader configurations.

**/dev (Device Files)**

Contains device files used to interact with hardware devices on the system.

**/etc (Configuration Files)**

Contains system-wide configuration files controlling various aspects of the system.

**/home (Home Directories)**

Contains user's personal files and settings.

**/lib (Libraries)**

Shared libraries needed by the executables in /bin and /sbin.

**/mnt (Mount Point)**

Used as a mount point for temporary filesystems, such as USB drives or network shares.

**/opt (Optional)**

Used for installing optional software packages.

**/proc (Process Information)**

A virtual filesystem providing information about running processes and system resources.

**/root (Root Home Directory)**

Home directory for the root user, containing personal files and settings.

**/sbin (System Binaries)**

Essential system administration binaries used for system maintenance and repair.

**/tmp (Temporary Files)**

Used for storing temporary files that are only needed temporarily.

**/usr (User)**

Contains user-related programs, libraries, documentation, and other files not required for basic system operation.

**/var (Variable)**

Contains variable data files, such as log files, spool files, and temporary files expected to grow in size.

**2. LAMP installation**

Setting up a LAMP (Linux, Apache, MySQL, PHP) stack on a Linux system involves installing and configuring each component to work together. Here's a step-by-step tutorial for installing and configuring a basic LAMP stack on a Linux system:

Install Apache:

Update the package index:

sudo apt update

Install Apache:

sudo apt install apache2

Start Apache:

sudo systemctl start apache2

Enable Apache to start on boot:

sudo systemctl enable apache2

Verify Apache is running by visiting your server's IP address in a web browser. You should see the Apache default page.

Install MySQL:

Install MySQL server:

sudo apt install mysql-server

Run the MySQL security script to improve the security of your MySQL installation: sudo mysql\_secure\_installation

Follow the on-screen instructions to set up a root password, remove anonymous users, disallow root login remotely, and remove the test database.

Install PHP:

Install PHP and the PHP module for Apache:

sudo apt install php libapache2-mod-php php-mysql

Restart Apache for the changes to take effect:

sudo systemctl restart apache2

Test PHP by creating a PHP info file:

echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/info.php

Visit http://your\_server\_ip/info.php in a web browser. You should see a page with detailed information about your PHP installation.

Optional:

Install PHPMyAdmin (for database management):

Install PHPMyAdmin:

sudo apt install phpmyadmin

During installation, choose Apache as the web server to configure PHPMyAdmin to work with Apache.

Enable the PHPMyAdmin Apache configuration: sudo ln -s /etc/phpmyadmin/apache.conf /etc/apache2/conf-available/phpmyadmin.conf

Enable the PHPMyAdmin configuration:

sudo a2enconf phpmyadmin

Restart Apache:

sudo systemctl restart apache2

Access PHPMyAdmin by visiting http://your\_server\_ip/phpmyadmin in a web browser.

Configure Firewall (if enabled):

If you have a firewall enabled (e.g., UFW), allow HTTP and HTTPS traffic:

arduino

Copy code

sudo ufw allow 'Apache'

sudo ufw allow 'Apache Full'

Optional: Configure Virtual Hosts:

If you're hosting multiple websites, set up Apache virtual hosts to serve each site from its own directory.

That's it! You've now installed and configured a basic LAMP stack on your Linux system. You can now start developing and hosting PHP web applications.