# Phase 1: System update & Directory setup

## describe the Linux architecture layers (hardware -> kernel -> shell -> user space). Where do system calls fit?

1. Hardware layer:

* The physical layer: CPU, RAM, Disk, GPU, etc.

1. Kernel Layer:

* Its the base of the operating system
* It makes all the processes
* system call lives here

1. Shell Layer:

* is the command-line interface
* translates human commands to instructions

1. User space:

* all the apps outside kernal run there

## Explain the purpose of these directories: /, /bin, /sbin, /usr, /etc, /var.

1. /:

* the root directory
* contain all the system files

1. /bin:

* the binary directory
* contains all the command-line programs

1. /sbin:

* system binaries directory
* contain system commands

1. /usr:

* user system directory
* contain all the programs, libraries that not essential

1. /etc:

* system configuration files directory

1. /var:

* variable date directory
* contain files that change frequently

## Why does Linux treat everything as a file?

1. Uniform interface: The kernel doesn’t need separate mechanisms for different resources. Everything can be read, written, or modified using the same file operations
2. Abstraction: Applications don’t care whether data comes from a disk, keyboard, or network — they all use the same interface.
3. Flexibility: Enables redirection, piping , and device management without special APIs.

## 4. Explain the difference between a program and a process.

1. Program:

* program is just a file on disk that contains compiled instructions
* It is not running, just stored.

1. Process:

* process is an instance of a program that is running in memory.
* It has CPU state, memory space, and resources.
* Each process gets a unique PID (Process ID).