**What is Shell Programming?**

Shell programming refers to writing scripts to automate tasks using a command-line interpreter, such as Bash on Linux or PowerShell on Windows. Python provides the ability to execute shell commands, process their outputs, and manage system resources.

### ****Why Use Python for Shell Programming?****

* **Cross-Platform:** Python works seamlessly across different operating systems.
* **Extensive Libraries:** Libraries like os, subprocess, and shutil simplify shell programming.
* **Readable Syntax:** Python’s clean syntax makes it easier to write and maintain scripts.

#### **Key Functions:**

* os.getcwd(): Get the current working directory.
* os.chdir(path): Change the current working directory.
* os.listdir(path): List all files and directories in a path.
* os.remove(file): Remove a file.
* os.mkdir(directory): Create a new directory.
* os.rmdir(directory): Remove an empty directory.

#### **Example:**

import os

print("Current Directory:", os.getcwd()) # Get the current directory

os.chdir("/path/to/directory") # Change to a new directory

print("Directory Changed To:", os.getcwd())

print("Files:", os.listdir(".")) # List files

### ****Running Shell Commands with**** subprocess

#### Writing to a File

* **open('example.txt', 'w')**: Opens (or creates) a file named example.txt in write mode.
* **with statement**: Ensures the file is properly closed after its suite finishes, even if an exception is raised.
* **file.write()**: Writes the specified string to the file.

Reading from a File

* **open('example.txt', 'r')**: Opens the file in read mode.
* **file.read()**: Reads the entire content of the file.
* The content is then printed to the console.