**OPERATING SYSTEMS**

**Lab Assignment Sheet-1**

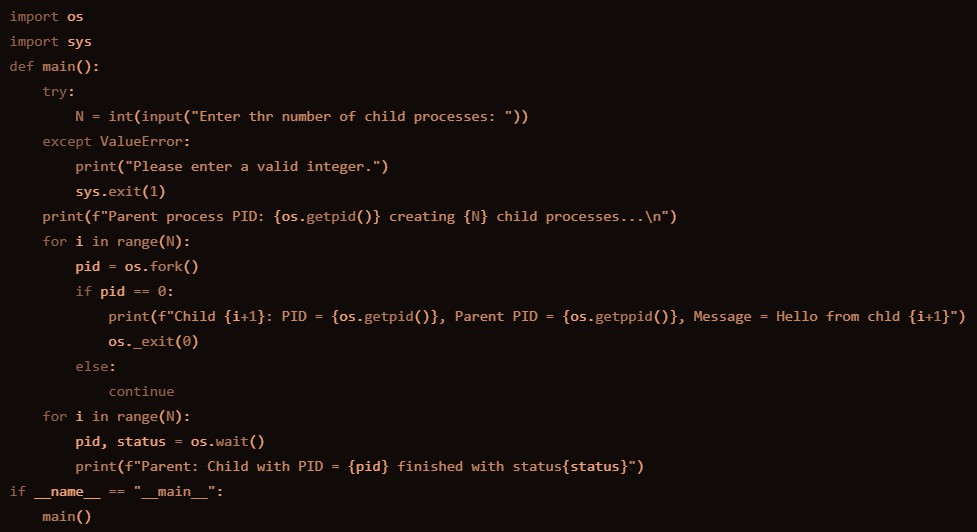
**NAME – Ishika Roll No. – 2301420052**

**B. Tech CSE Data Science**

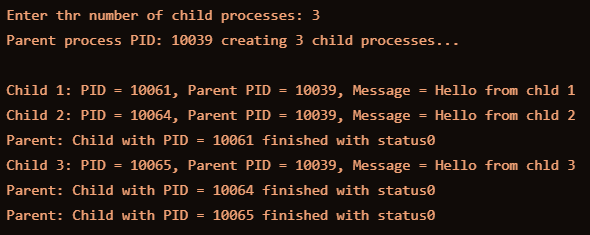
# Experiment Title: Process Creation and Management Using Python OS Module Task 1: Process Creation Utility

Write a Python program that creates N child processes using os.fork(). Each child prints:

* Its PID
* Its Parent PID
* A custom message

The parent should wait for all children using os.wait(). INPUT-

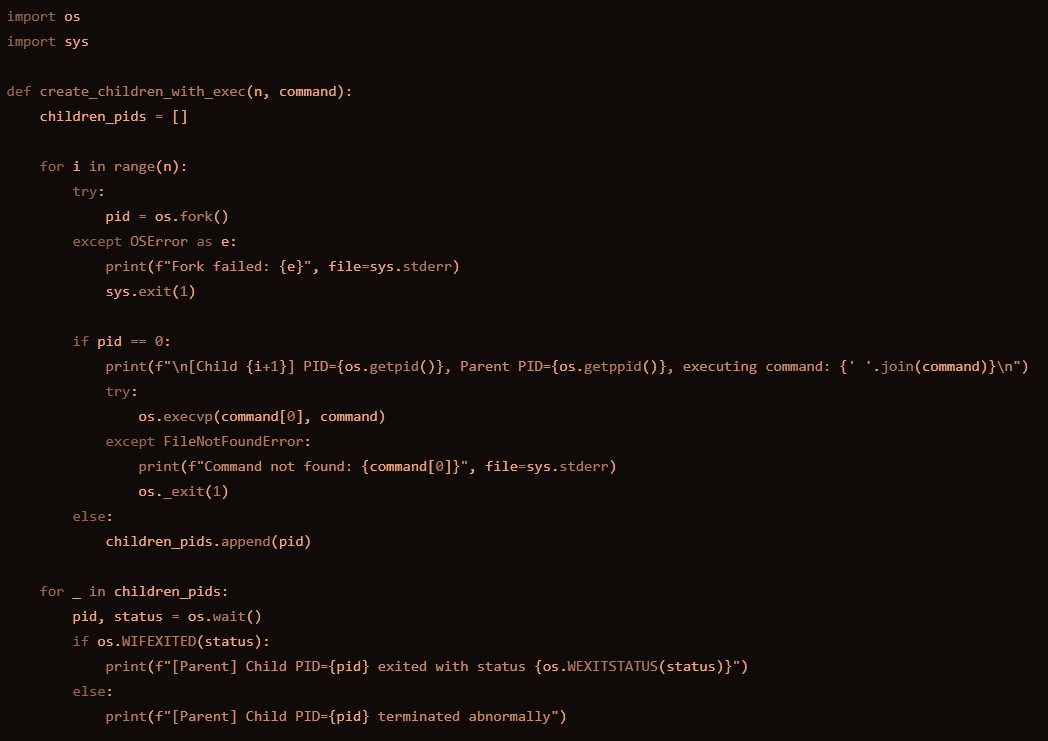
OUTPUT

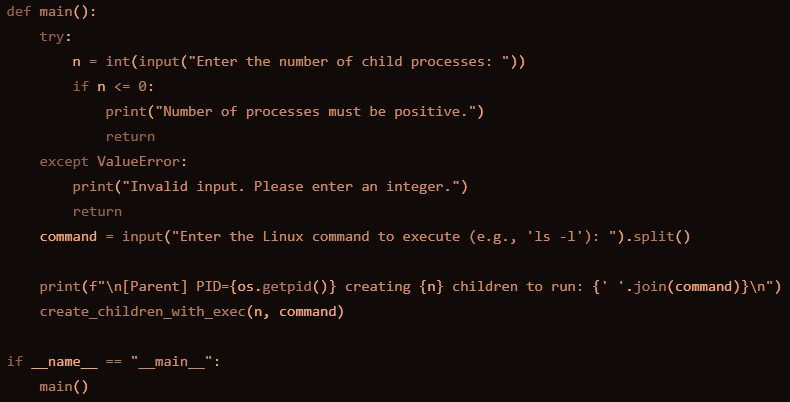


# Task 2: Command Execution Using exec()

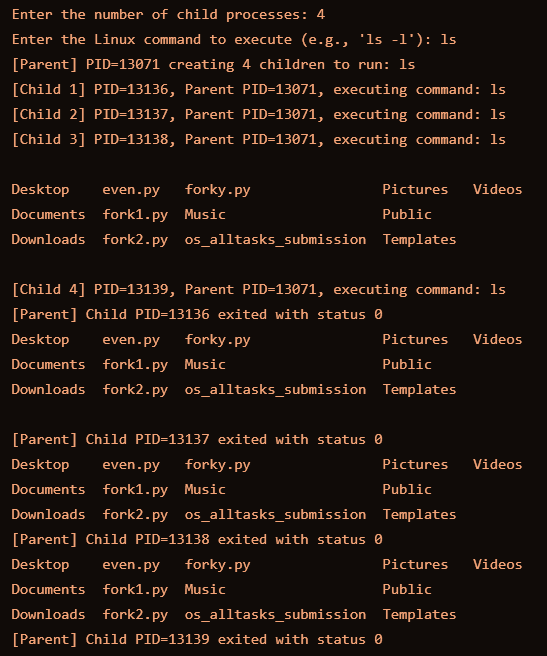
Modify Task 1 so that each child process executes a Linux command (ls, date, ps, etc.) using os.execvp() or subprocess.run().

INPUT





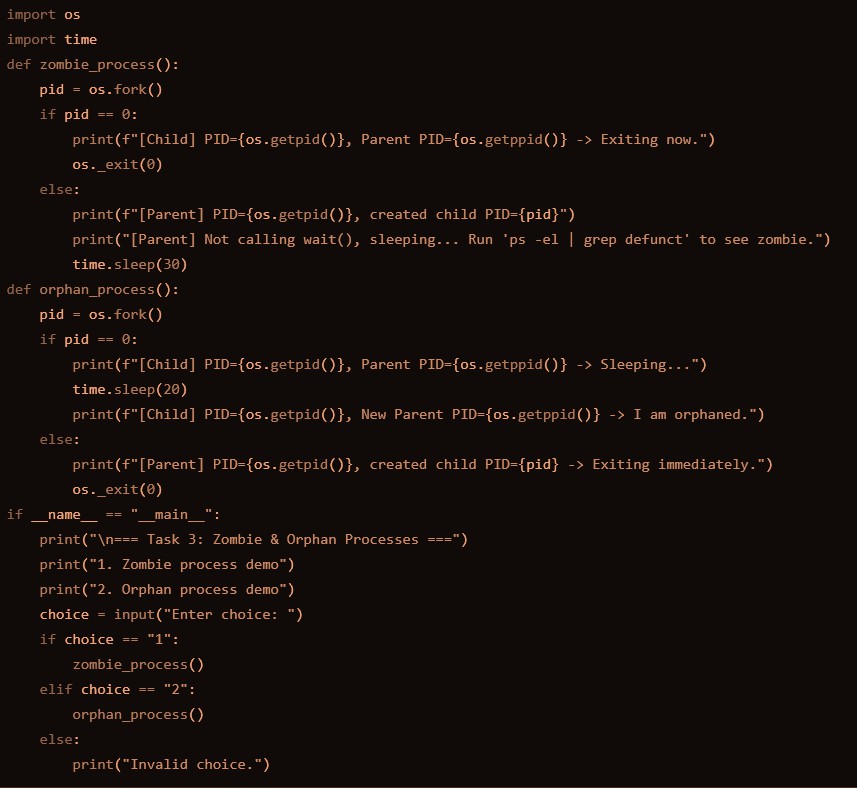
OUTPUT



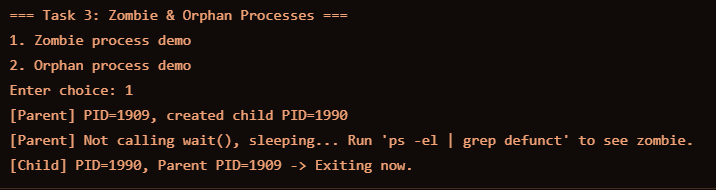
# Task 3: Zombie & Orphan Processes

Zombie: Fork a child and skip wait() in the parent. Orphan: Parent exits before the child finishes.

Use ps -el | grep defunct to identify zombies. INPUT



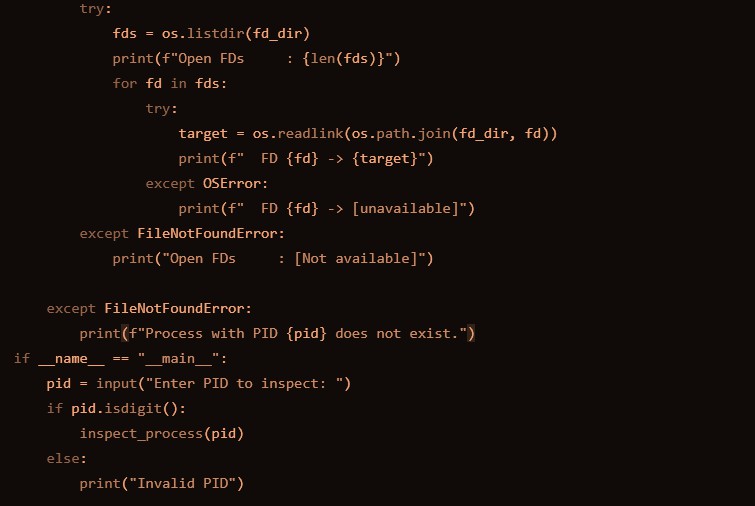
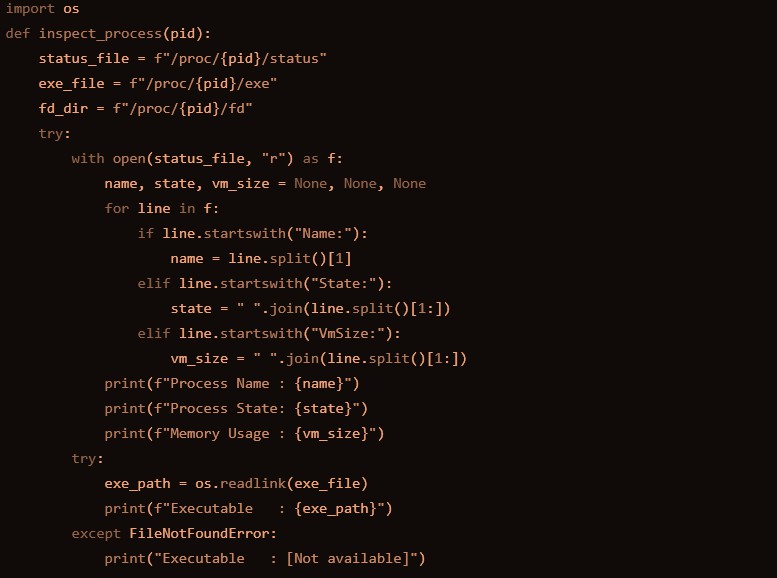
OUTPUT



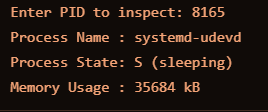
# Task 4: Inspecting Process Info from /proc

Take a PID as input. Read and print:

* Process name, state, memory usage from /proc/[pid]/status
* Executable path from /proc/[pid]/exe
* Open file descriptors from /proc/[pid]/fd INPUT



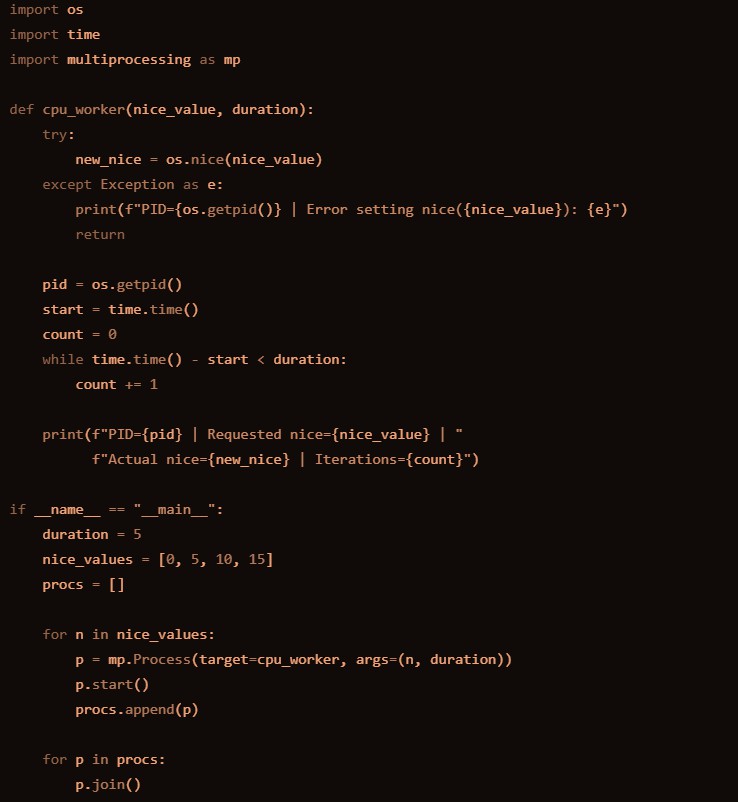
OUTPUT



# Task 5: Process Prioritization

Create multiple CPU-intensive child processes. Assign different nice() values. Observe and log execution order to show scheduler impact.

INPUT



OUTPUT

