SCENARIO-BASED PRACTICAL QUESTIONS: Process Management

🔹 1. Monitoring System Load

Scenario:
Your system is running slow. You want to check which processes are using high CPU and memory.

✅ Task:

- Use `top` and `htop` to identify the top 5 CPU and memory consuming processes.
- Take a screenshot or copy the process names and their PID.

🔹 2. Kill a Process by PID

Scenario:
A process is consuming 100% CPU and making the system unresponsive. The process is named `stress`.

✅ Task:

- Find the PID of the process using `ps aux | grep stress` or `top`.
- Kill the process using `kill <PID>`.
- If it doesn't stop, forcefully kill using `kill -9 <PID>`.

🔹 3. Background and Foreground Jobs

Scenario:
You start a program (like `ping google.com`) but need to run other commands.

✅ Task:

- Run `ping google.com` and send it to background using `Ctrl+Z` and `bg`.

- Check background jobs using `jobs`.
- Bring it back to foreground using `fg`.

🔹 4. Start Process with Low Priority

Scenario:
You want to run a backup script without affecting system performance.

✅ **Task:**

- Start a long-running process (e.g., `yes > /dev/null`) with low priority using `nice`.
- Use `top` or `ps -eo pid,ni,comm` to check the nice value.

🔹 5. Change Priority of Running Process

Scenario:
You started a process with normal priority, but now you want to reduce its priority.

✅ **Task:**

- Use `ps` to find the PID of the running process (e.g., `yes`).
- Change its priority using `renice +15 <PID>`
- Verify using `top`.

🔹 6. View Tree of Running Processes

Scenario:
You want to see the parent-child relationship of running processes.

✅ **Task:**

- Use `pstree` or `ps -ejH` to view the hierarchy.
- Find the parent process of your terminal or shell.

🔹 7. Track Process Resource Usage Over Time

Scenario:
You want to monitor a specific process over time (e.g., a script).

✅ **Task:**

- Start a process (e.g., `dd if=/dev/zero of=/dev/null`)
- Use `top`, `htop`, or `pidstat -p <PID> 1` to monitor live statistics.

🔹 8. List All Processes for a Specific User

Scenario:
You want to find all processes started by user `john`.

✅ **Task:**

- Use `ps -u john` or `top -u john`
- Filter by `UID` using `ps aux | grep john`.

🔹 9. Find Command Behind a PID

Scenario:
You have a PID, and you want to know which command is running.

✅ **Task:**

- Use `ps -p <PID> -o cmd`

- Or `cat /proc/<PID>/cmdline`