RIPHAH INTERNATIONAL UNIVERSITY, ISLAMABAD

Lab#12

Bachelors of Computer Science – 5th Semester

Subject: Operating System

Submitted to: Ms. Kausar Nasreen Khattak

Submitted by: Tabinda Hassan

SAP-46374

Date of Submission: 11-11-2024

Lab Tasks:

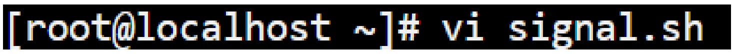
Q1: Which command would you use to find the process ID (PID) of a process named OSLab without running it. After obtaining the PID, which command would you use to kill the process?

To find the PID of a process named `OSLab`: **pgrep OSLab**

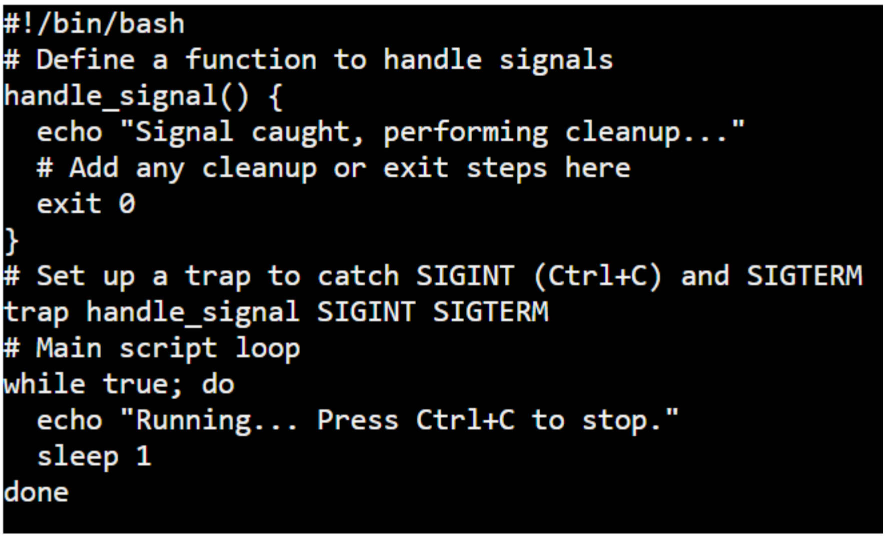
To kill the process: **kill <PID>**

Or forcefully: **kill -9 <PID>**

Q2: How would you write a script that uses a signal trap to handle specific signals, and what is the purpose of a signal trap in such a script?



```
[root@localhost ~]# vi signal.sh
```



```
#!/bin/bash
# Define a function to handle signals
handle_signal() {
  echo "Signal caught, performing cleanup..."
  # Add any cleanup or exit steps here
  exit 0
}
# Set up a trap to catch SIGINT (Ctrl+C) and SIGTERM
trap handle_signal SIGINT SIGTERM
# Main script loop
while true; do
  echo "Running... Press Ctrl+C to stop."
  sleep 1
done
```

Esc

Shift+:

wq

Enter

The `**handle_signal**` function handles cleanup tasks and prints a message before exiting with status 0.

The `trap handle_signal SIGINT SIGTERM` command directs the script to call `handle_signal` when it receives SIGINT (Ctrl+C) oIGTERM.

An infinite loop then prints "Running... Press Ctrl+C to stop." every second until interrupted.

Purpose: Signal traps allow a script to handle interruptions gracefully by executing specific tasks (e.g., saving data, closing files) before exiting.