# Operating System Labs: Instructor Solutions

## Lab 1: View and Inspect Running Processes

* Commands:

1. ps aux
2. ps -o pid,ppid,ni,stat,cmd
3. top

* Expected Output:

1. Ps output:  
   USER PID PPID NI STAT CMD  
   user 2945 2891 0 Ss bash  
   root 1 0 0 Ss /sbin/init  
   user 3120 2945 0 R+ top
2. Top output:  
   PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND  
   3120 user 20 0 162m 12m 5m R 3.0 0.2 0:01.23 top

## Lab 2: Foreground and Background Jobs

* Commands:

1. sleep 120 &
2. jobs
3. fg %1
4. # (press CTRL+Z to suspend)
5. bg %1

* Expected Output:

1. [1] 3355
2. [1]+ Running sleep 120 &
3. [1]+ Stopped sleep 120
4. [1]+ sleep 120 &

## Lab 3: Kill and Signals

* Commands:

1. sleep 200 &
2. ps aux | grep sleep
3. kill -15 <PID>
4. kill -9 <PID>
5. killall sleep

* Expected Output:

1. kill -15: process exits cleanly (preferred)
2. kill -9: kills even stuck processes
3. killall: kills all sleep processes

## Lab 4: Nice and Renice

* Commands:

1. nice -n 10 sleep 300 &
2. ps -o pid,ni,cmd
3. renice 5 <PID>

* Expected Output:

1. 3333: old priority 10, new priority 5
2. renice: failed to set priority for 3333: Permission denied

## Lab 5: Trap Signals in a Script

* Commands:

1. #!/bin/bash
2. trap "echo 'SIGINT received. Exiting...'; exit" SIGINT
3. while true; do
4. echo "Running..."
5. sleep 2
6. done

* Expected Output:

1. Running...
2. Running...
3. ^C
4. SIGINT received. Exiting...

## Lab 6: Create a Custom systemd Service

* Commands:

1. #!/bin/bash
2. while true; do
3. echo "$(date)" >> /tmp/log.txt
4. sleep 10
5. done
6. chmod +x /usr/local/bin/timerlog.sh
7. [Unit]
8. Description=Log time every 10 seconds
9. [Service]
10. ExecStart=/usr/local/bin/timerlog.sh
11. Restart=always
12. [Install]
13. WantedBy=multi-user.target
14. sudo systemctl status timerlog
15. tail -f /tmp/log.txt