# Exsercise 6: Managing Software and Processes - Part 2-2

## I. Create, monitor and kill process (cont)

1. Login to the ubuntu system with the username and password provided.
2. List all the files and directories existed in your server in foreground using **ls -R /** command.
3. Run the above command in background. Is there any information showed on your screen? Could you utilize the shell?
4. Re-do the step 3 but redirect stdout and stderr of the command to a file named **log**. Could you utilize the shell now?
5. Run the command **sleep 300** in background.
6. List all the background jobs in your system with their process id
7. Kill the job created in step 5
8. Run **sleep 300** command in foreground
9. Move the process created in step 8 to background
10. Move back the above process to foreground
11. Run the **sleep 300** command in background, send the following signal to the process one by one and re-run the process after each. Using **jobs** command after sending the signal to confirm that you sent the correct signal.
    - SIGTERM
    - SIGINT
    - SIGKILL
    - SIGHUP
12. Run the **sleep 300** command again and this time you have to ensure that the process will still be running after you logout. Specify the PPID of the process before and after you log-out.
13. Using the ps command to show the nice value of the running processes
14. Run the **sleep 300** command in background with the nice value **10**. Using ps command to display the nice value of the sleep process.
15. Change the nice value of sleep process to **-5**. Using ps command to display the nice value after changing.

# Exsercise Instructions

## I. Create, monitor and kill process

1. Log int to the Ubuntu system with the user name and password provided: student/lpic1@123
2. List all the files and directories existed in your server in foreground using ls -R / command.
   **$ ls -R /**

3. Run the above command in background. Is there any information showed on your screen? Could you utilize the shell?
   **$ ls -R / &**

4. Re-do the step 3 but redirect stdout and stderr of the command to a file named log. Could you utilize the shell now?
   **$ ls -R / &>log &**

5. Run the command sleep 300 in background.
   **$ sleep 300 &**

```
student@ubuntu:~$ sleep 300 &
[1] 2253
```

6. List all the background jobs in your system with their process id
   **$ jobs -l**

```
student@ubuntu:~$ jobs -l
[1]+  2253 Running                 sleep 300 &
```

7. Kill the job created in step 5
   **$ kill %<job id>**

```
student@ubuntu:~$ kill %1
```

8. Run sleep 300 command in foreground
   **$ sleep 300**

9. Move the process created in step 8 to background
   While sleep process is running type Ctrl+z to suspend the process

```
student@ubuntu:~$ sleep 300
^Z[1]   Terminated              sleep 300

[2]+  Stopped                 sleep 300
```

   **$ jobs**

```
student@ubuntu:~$ jobs
[2]+  Stopped                 sleep 300
```

**$ bg %<job id>**

```
student@ubuntu:~$ bg %2
[2]+ sleep 300 &
```

**$jobs**

```
student@ubuntu:~$ jobs
[2]+  Running                 sleep 300 &
```

10. Move back the above process to foreground

**$ fg %<job id>**

```
student@ubuntu:~$ fg %2
sleep 300
```

11. Run the sleep 300 command in background, send the following signal to the process one by one and re-run the process after each. Using jobs command after sending the signal to confirm that you sent the correct signal.
    - SIGTERM
    - SIGINT
    - SIGKILL
    - SIGHUP

**$ sleep 300 &**
**$ jobs -l**

```
student@ubuntu:~$ jobs -l
[1]+  2255 Running                 sleep 300 &
```

**$ kill <PID>**

```
student@ubuntu:~$ kill 2255
```

**$ jobs**

```
student@ubuntu:~$ jobs
[1]+  Terminated              sleep 300
```

**$ sleep 300 &**

```
student@ubuntu:~$ sleep 300 &
[1] 2257
```

**$ kill -2 <PID>**

```
student@ubuntu:~$ kill -2 2257
```

**$ jobs**

```
student@ubuntu:~$ jobs
[1]+  Interrupt               sleep 300
```

**$ sleep 300 &**
**$ kill -9 <PID>**
**$ jobs**

**$ sleep 300 &**
**$ kill -1 <PID>**
**$ jobs**

12. Run the sleep 300 command again and this time you have to ensure that the process will still be running after you logout. Specify the PPID of the process before and after you log-out.
    **$ nohup sleep 300 &**
    **$ ps -ef |grep sleep**
    **$ logout**

    Re-login to the system with user name and password provided: student/lpic1@123
    **$ ps -ef |grep sleep**

13. Using the ps command to show the nice value of the running processes
    **$ ps -eo user,pid,ppid,ni,comm**

14. Run the sleep 300 command in background with the nice value 10. Using ps command to display the nice value of the sleep process.
    **$ nice -n 10 sleep 300 &**
    **$ ps -eo user,pid,ppid,ni,comm**

15. Change the nice value of sleep process to -5. Using ps command to display the nice value after changing.
    **$ sudo renice -n -5 <PID of sleep process>**