UNIX

Foundation Exercises

Module 09 - Process Control

© FDM Group Ltd 2022. All Rights Reserved.

Any unauthorised reproduction or distribution in part  
or in whole will constitute an infringement of copyright.

**Perform the following tasks**

1. Run the **sleep** command for 1000 seconds in the background.:
   1. sleep 1000 &
2. Use the **ps** command to identify the PID of the **sleep** process.
   1. ps
3. Using the **kill** command with the appropriate signal and the PID, stop (pause) the **sleep** process.
   1. kill -19 15628
4. Identify the job ID of the **sleep** process.
   1. Jobs -l
5. Using the **kill** command with the appropriate signal and the job ID, restart the **sleep** process.
   1. kill -18 15628
6. Display the current status of the sleep process.
   1. ps -f | grep sleep
7. Terminate the **sleep** process.
   1. kill -15 15628
8. Check that the termination was successful.
   1. jobs -l

Open a second terminal session. We will refer to this as *Session B*

1. In the first session window (*Session A*), run the following command:

**ping google.com**

(This will continuously send a data packet to *google.com* every second and print the response on the screen. The process will initally be running in the foreground.)

1. Stop (pause) the **ping** process.
   1. Press "Ctrl"+"C"
2. Restart the **ping** process in the background.
   1. ping google.com
   2. Press “ctrl” + “Z”
3. Can you stop (pause) the **ping** process with **Ctrl+z**? Why not?
   1. Ctrl+Z will make the process run in the background

In your *Session B* window:

1. Run the **ps** command and the **jobs** command.
2. Do you see the **ping** process from *Session A* listed? Why not?
   1. Ps command does show the processes, but the jobs -l command did not show anything.
   2. Don’t know why
3. How would you modify the ps command to show the processes running in Session A as well as Session B?
4. Using the **kill** command with the appropriate signal and the PID, stop (pause) the **ping** process running in *Session A*.
5. In *Session A*, display the status of the **ping** process.
   1. Jobs -l
6. Can you **kill** the **ping** process paused in *Session A* by referring to its job ID in *Session B*?
7. Using the **kill** command and the PID, kill the **ping** process running in *Session A*.
   1. Kill -9 19725
8. In *Session A*, display the status of the **ping** process.
   1. Jobs -l
9. Why are you able to control *Session A*'s **ping** process from *Session B* by using the PID, but not the job ID?