Name: Wan Muhammad Rayhan Arwindra

NPM: 1806241210

This week, we learnt about daemon processes. These processes are processes which run on the background, and is detached from the terminal, meaning that it is not under the direct control of the user. The daemon processes have no tie to the terminal, or in other words their parent process is init, with pid of "1". Daemon processes usually start when the OS is first booted.

Daemon processes run on the background, but not all background processes can be classified as a daemon process. A background process still has ties to the current terminal, and can be directly controlled by the user. The user can kill, interrupt, or send other signals to background processes just like any other process. Daemon processes however have no tie to the current terminal, and cannot be directly controlled by the user.

To run a process as a background process, we add the ampersand symbol, or "&" to the end of the execution of that program. We can see the list of background processes on the jobs list, with the command "jobs -l". We can remove the program from the jobs list by using the disown command, which would also prevent the program from being sent a SIGHUP signal. Another way to prevent the program from being sent a SIGHUP signal is by using the nohup command. This would also direct all outputs from the program by default to the file nohup.out.

A process is a part of a group, and a process group is a part of a session. When a daemon process is created, it creates a new session. Usually, the daemon becomes the leader of both the process group, and the session. Also, the daemon is usually the only process within their process group. To create a new session, we can us the command setsid(), this would set the process's parent to init, and detach it from the current terminal/session. Just like a process has an id which is PID, process groups and sessions also have ids. The session id is called SID, and the process group id is called GID.