Lab No	: 06
Name of the Lab	: Linux Command for processs
ID	: IT-17005

Objectives:

i)   How to manage processes from the Linux Terminal?

ii)   Run the following process commands in Linux:
Top, htop, Ps, pstree, kill, pgrep, pkill ,killall, renice, xkill.

Answer no (i):

An instance of a program is called a Process. In simple terms, any command that you give to your Linux machine starts a new process. The Linux terminal has a number of useful commands that can display running processes, kill them, and change their priority level. This post lists the classic, traditional commands, as well as some more useful, modern ones. Many of the commands here perform a single function and can be combined — that's the Unix philosophy of designing programs.

Answer no (ii):

1) "top": The **top** command is the traditional way to view your system's resource usage and see the processes that are taking up the most system resources.

```
top - 06:55:44 up 42 min,  1 user,  load average: 0.44, 0.37, 0.39
Tasks: 227 total,   1 running, 226 sleeping,   0 stopped,   0 zombie
%Cpu(s):  5.7 us,  1.6 sy,  0.0 ni, 91.4 id,  0.3 wa,  0.0 hi,  0.9 si,  0.0 st
MiB Mem :   3858.4 total,    135.2 free,   1753.2 used,   1970.0 buff/cache
MiB Swap:   1024.0 total,   1024.0 free,      0.0 used.   1458.4 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 1626 ruhan     20   0 2888584 225924 112436 S  18.5   5.7   3:01.46 gnome-she+
 1194 ruhan     20   0 1055492 113960  86452 S   6.3   2.9   2:20.19 Xorg
 3465 ruhan     20   0 2481212 156844 119168 S   0.7   4.0   0:25.11 Web Conte+
 3575 ruhan     20   0 2586912 201080 104652 S   0.7   5.1   0:22.69 Web Conte+
 4381 ruhan     20   0   20860   4036   3340 R   0.7   0.1   0:00.10 top
    1 root      20   0  167592  11932   7560 S   0.3   0.3   0:05.35 systemd
   10 root      20   0       0      0      0 I   0.3   0.0   0:03.35 rcu_sched
  344 root      20   0       0      0      0 I   0.3   0.0   0:02.48 kworker/3+
  553 root      -2   0       0      0      0 S   0.3   0.0   0:00.82 i915/sign+
 2966 ruhan     20   0 3789636 369100 206684 S   0.3   9.3   1:55.16 firefox
 3017 ruhan     20   0 2976144 205312 132820 S   0.3   5.2   1:18.01 Web Conte+
 3131 ruhan     20   0 2473764 161364 119152 S   0.3   4.1   0:12.92 Web Conte+
 3503 ruhan     20   0 2621676 183852 126524 S   0.3   4.7   0:28.08 Web Conte+
 3816 ruhan     20   0 1356832 274948 143368 S   0.3   7.0   0:46.37 soffice.b+
 4250 root      20   0       0      0      0 I   0.3   0.0   0:00.13 kworker/u+
    2 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kthreadd
    3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_gp
```

2) "htop": The **htop** command is an improved top. It's not installed by default on most Linux distributions — here's the command you'll need to install it on Ubuntu: sudo apt-get install htop.

```
  1  [||                                    4.5%]   Tasks: 121, 560 thr; 1 running
  2  [||                                    1.3%]   Load average: 0.21 0.31 0.37
  3  [||                                    1.9%]   Uptime: 00:41:18
  4  [||                                    2.7%]
  Mem[|||||||||||||||||||||2.06G/3.77G]
  Swp[                          0K/1024M]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%    TIME+  Command
 1098 ruhan      20   0 20572  4472     0 S  0.0  0.1  0:00.00 (sd-pam)
 1096 ruhan      20   0 16844  9472  7520 S  0.0  0.2  0:00.35 /lib/systemd/sys
  315 root       19  -1 66088 21520 19968 S  0.0  0.5  0:00.89 /lib/systemd/sys
  924 root       20   0 14732  7124  6164 S  0.0  0.2  0:00.45 /lib/systemd/sys
  847 systemd-r  20   0 22624  9596  8340 S  0.0  0.2  0:00.51 /lib/systemd/sys
  863 systemd-t  20   0 90264  6124  5348 S  0.0  0.2  0:00.00 /lib/systemd/sys
  846 systemd-t  20   0 90264  6124  5348 S  0.0  0.2  0:00.33 /lib/systemd/sys
  380 root       20   0 19444  5364  3004 S  0.0  0.1  0:02.57 /lib/systemd/sys
 1167 root       20   0  9512  5940  4676 S  0.0  0.2  0:00.01 /sbin/dhclient -
    1 root       20   0  163M 11932  7560 S  0.7  0.3  0:05.21 /sbin/init splas
 3284 root       20   0 18456  2652  1840 S  0.0  0.1  0:00.35 /sbin/mount.ntfs
  931 root       20   0 11408  7584  6808 S  0.0  0.2  0:00.53 /sbin/wpa_suppli
 1606 ruhan      20   0  7268  4060  3608 S  0.0  0.1  0:00.20 /usr/bin/dbus-da
F1Help  F2Setup F3Search F4Filter F5Tree  F6SortBy F7Nice - F8Nice + F9Kill  F10Quit
```

3) "ps": The **ps** command lists running processes.

You could also pipe the output through **grep** to search for a specific process without using any other commands. The following command would search for the Codeblocks process:

```
ruhan@ruhan-HP-Notebook:~$ ps -A | grep firefox
 2966 tty2     00:02:05 firefox
ruhan@ruhan-HP-Notebook:~$
```

4) "pstree": The **pstree** command is another way of visualizing processes. It displays them in tree format.

```
ruhan@ruhan-HP-Notebook:~$ pstree
systemd──ModemManager───2*[{ModemManager}]
        ├─NetworkManager──┬─dhclient
        │                 └─2*[{NetworkManager}]
        ├─accounts-daemon───2*[{accounts-daemon}]
        ├─acpid
        ├─avahi-daemon───avahi-daemon
        ├─bluetoothd
        ├─boltd───2*[{boltd}]
        ├─colord───2*[{colord}]
        ├─cron
        ├─cups-browsed───2*[{cups-browsed}]
        ├─cupsd
        ├─dbus-daemon
        ├─firefox──┬─Web Content───26*[{Web Content}]
        │          ├─2*[Web Content───25*[{Web Content}]]
        │          ├─Web Content───27*[{Web Content}]
        │          ├─2*[Web Content───28*[{Web Content}]]
        │          ├─Web Content───20*[{Web Content}]
        │          ├─WebExtensions───24*[{WebExtensions}]
        │          └─64*[{firefox}]
        ├─fwupd───4*[{fwupd}]
```

5) "kill": The **kill** command can kill a process, given its process ID.

```
ruhan@ruhan-HP-Notebook:~$ kill rhythmbox
bash: kill: rhythmbox: arguments must be process or job IDs
ruhan@ruhan-HP-Notebook:~$ ps -A | grep rhythmbox
 4831 tty2     00:00:01 rhythmbox
ruhan@ruhan-HP-Notebook:~$ kill 3841
bash: kill: (3841) - No such process
ruhan@ruhan-HP-Notebook:~$ kill 4841
bash: kill: (4841) - No such process
ruhan@ruhan-HP-Notebook:~$ kill 4831
ruhan@ruhan-HP-Notebook:~$
```

6) "pgrep": **pgrep** returns the process IDs that match it.

```
ruhan@ruhan-HP-Notebook:~$ pgrep rhythmbox
4917
ruhan@ruhan-HP-Notebook:~$ kill $(pgrep rhythmbox)
ruhan@ruhan-HP-Notebook:~$
```

7) " pkill & killall": The **pkill** and **killall** commands can kill a process, given its name.

```
ruhan@ruhan-HP-Notebook:~$ pkill rhythmbox
```

```
ruhan@ruhan-HP-Notebook:~$ killall rhythmbox
ruhan@ruhan-HP-Notebook:~$
```

8) "renice": The **renice** command changes the nice value of an already running process. The nice value determines what priority the process runs with. A value of **-19** is very high priority, while a value of **19** is very low priority. A value of **0** is the default priority.

```
ruhan@ruhan-HP-Notebook:~$ pgrep rhythmbox
5247
ruhan@ruhan-HP-Notebook:~$ renice 19 5247
5247 (process ID) old priority 0, new priority 19
ruhan@ruhan-HP-Notebook:~$
```

9) "xkill": The **xkill** command is a way of easily killing graphical programs. Run it and your cursor will turn into an **x** sign. Click a program's window to kill that program. If you don't want to kill a program, you can back out of xkill by right-clicking instead.

```
ruhan@ruhan-HP-Notebook:~$ xkill
Select the window whose client you wish to kill with button 1....
```