Mawlana Bhashani Science and Technology University

Lab-Report

Report No : 06

Experiment name : Linux command for process.

Course code : ICT-3110

Course title : Operating System Lab.

Date of Performance :

Date of Submission :

Submitted by

Name: Md Sohanur

ID: IT-18011

3rd year 1st semester

Session: 2017- 18

Dept. of ICT

MBSTU.

Submitted To

Nazrul Islam

Assistant Professor

Dept. of ICT

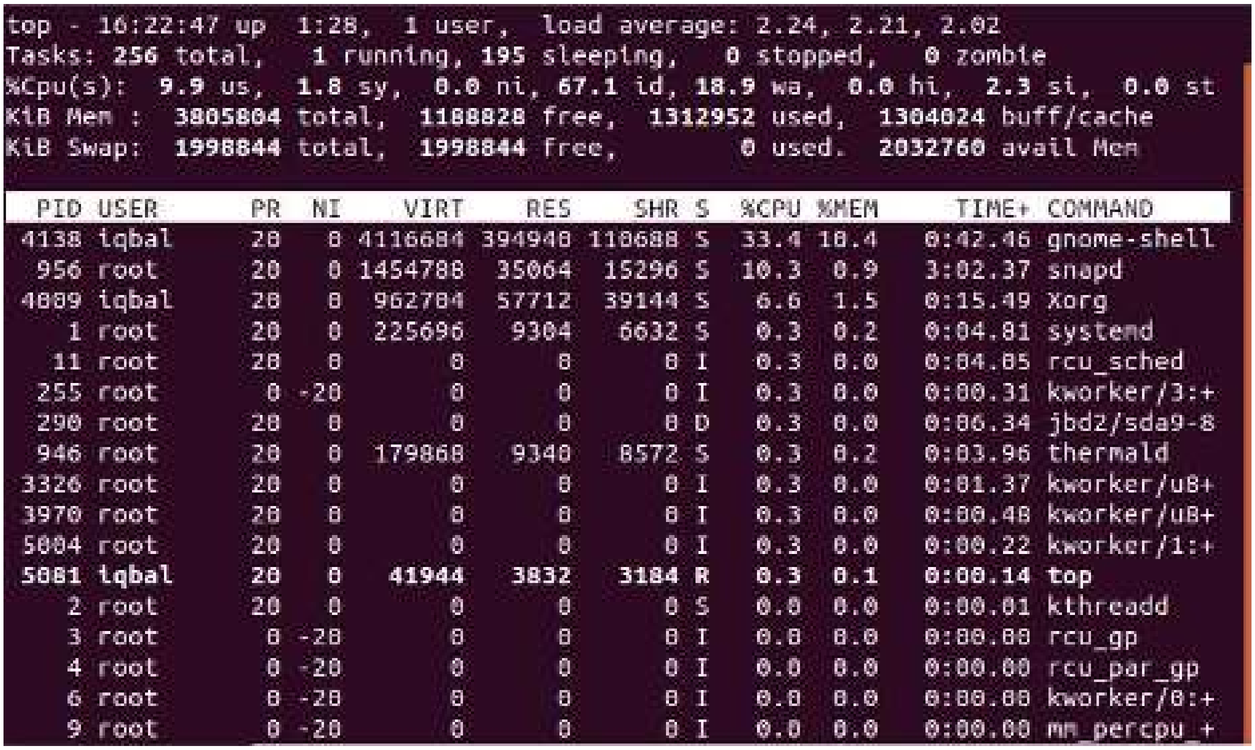
MBSTU.

i) How to Manage Processes from the Linux Terminal?

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.

ii) Run the following process commands in Linux.

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 displays a list of processes, with the ones using the most CPU at the top.

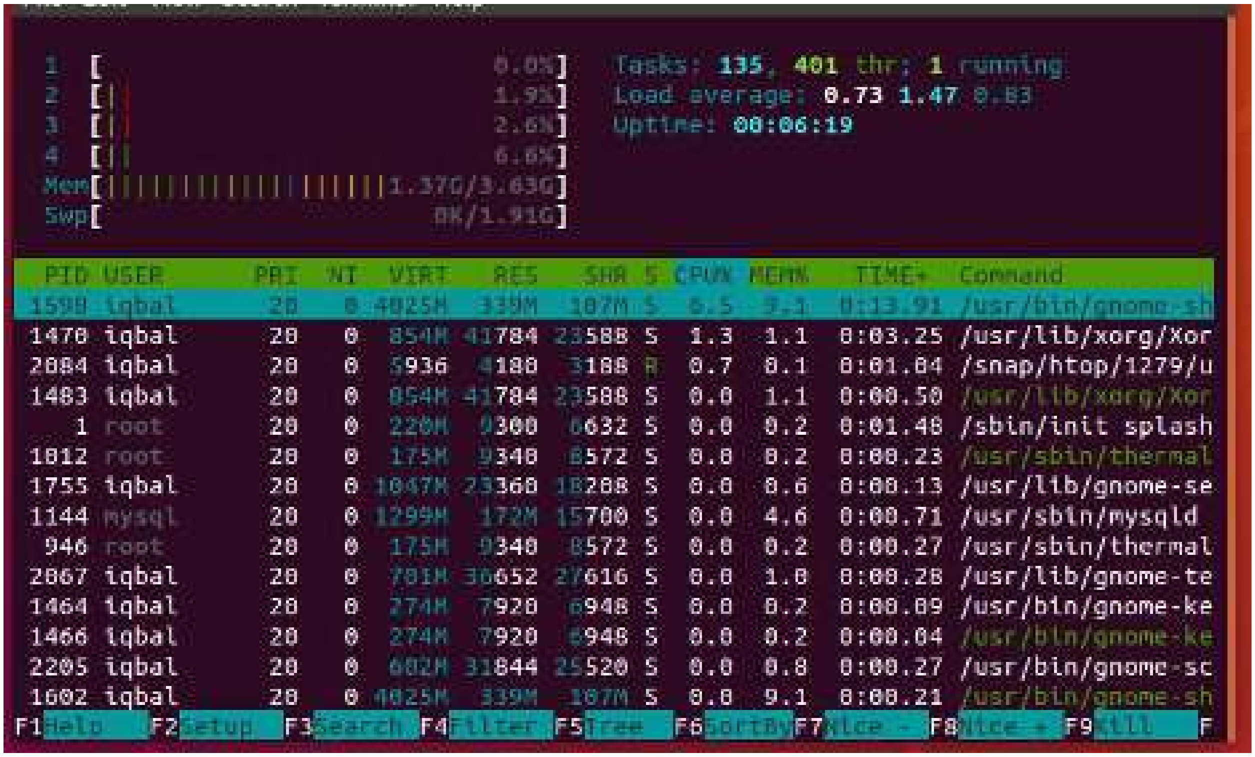


```
top - 16:22:47 up  1:28,  1 user,  load average: 2.24, 2.21, 2.02
Tasks: 256 total,   1 running, 195 sleeping,   0 stopped,   0 zombie
%Cpu(s):  9.9 us,  1.8 sy,  0.0 ni, 67.1 id, 18.9 wa,  0.0 hi,  2.3 si,  0.0 st
KiB Mem :  3805804 total,  1188828 free,  1312952 used,  1304024 buff/cache
KiB Swap:  1998844 total,  1998844 free,        0 used.  2032760 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 4138 iqbal     20   0 4116684 394940 110688 S  33.4 10.4   0:42.46 gnome-shell
  956 root      20   0 1454788  35064  15296 S  10.3  0.9   3:02.37 snapd
 4809 iqbal     20   0  962784  57712  39144 S   6.6  1.5   0:15.49 Xorg
    1 root      20   0  225696   9304   6632 S   0.3  0.2   0:04.81 systemd
   11 root      20   0       0      0      0 I   0.3  0.0   0:04.85 rcu_sched
  255 root       0 -20       0      0      0 I   0.3  0.0   0:00.31 kworker/3:+
  290 root      20   0       0      0      0 D   0.3  0.0   0:06.34 jbd2/sda9-8
  946 root      20   0  179868   9340   8572 S   0.3  0.2   0:03.96 thermald
 3326 root      20   0       0      0      0 I   0.3  0.0   0:01.37 kworker/u8+
 3970 root      20   0       0      0      0 I   0.3  0.0   0:00.48 kworker/u8+
 5004 root      20   0       0      0      0 I   0.3  0.0   0:00.22 kworker/1:+
 5081 iqbal     20   0   41944   3832   3184 R   0.3  0.1   0:00.14 top
    2 root      20   0       0      0      0 S   0.0  0.0   0:00.01 kthreadd
    3 root       0 -20       0      0      0 I   0.0  0.0   0:00.00 rcu_gp
    4 root       0 -20       0      0      0 I   0.0  0.0   0:00.00 rcu_par_gp
    6 root       0 -20       0      0      0 I   0.0  0.0   0:00.00 kworker/0:+
    9 root       0 -20       0      0      0 I   0.0  0.0   0:00.00 mm_percpu_+
```

To exit top or htop, use the Ctrl-C keyboard shortcut. This keyboard shortcut usually kills the currently running process in the terminal.

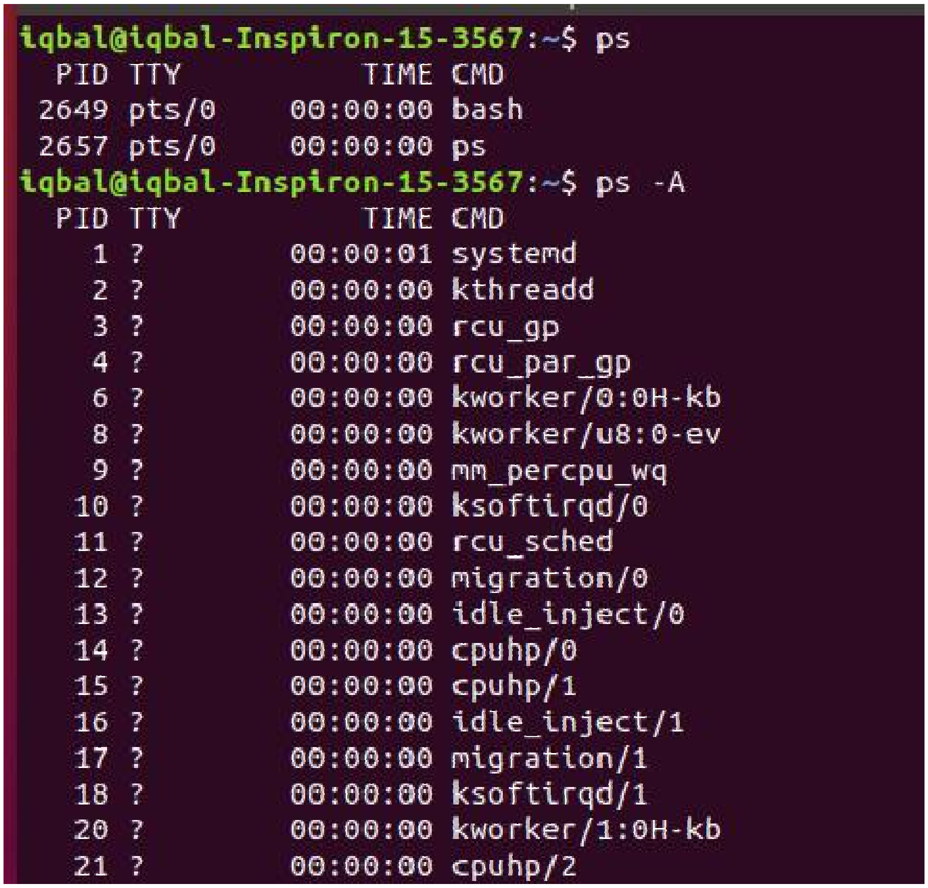
2) htop: The **htop** command is an improved top. It’s not installed by default on most Linux distributions — here’s the command you’ll need to install it on Ubuntu:

** sudo apt-get install htop.**



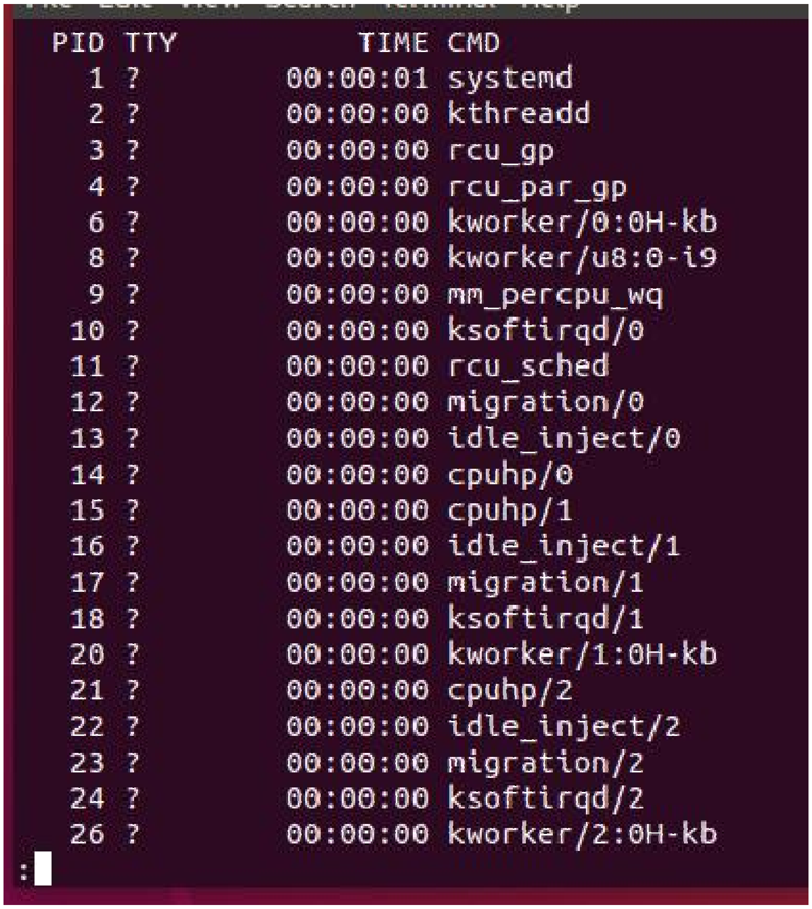
```
  1  [                          0.0%]   Tasks: 135, 401 thr; 1 running
  2  [|                         1.9%]   Load average: 0.73 1.47 0.83
  3  [|                         2.6%]   Uptime: 00:06:19
  4  [|                         6.6%]
  Mem[|||||||||||||||||||1.37G/3.63G]
  Swp[                      0K/1.91G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
 1598 iqbal      20   0 4025M  339M  107M S  6.5  9.1  0:13.91 /usr/bin/gnome-sh
 1470 iqbal      20   0  854M 41784 23588 S  1.3  1.1  0:03.25 /usr/lib/xorg/Xor
 2084 iqbal      20   0  5936  4180  3188 R  0.7  0.1  0:01.04 /snap/htop/1279/u
 1483 iqbal      20   0  854M 41784 23588 S  0.0  1.1  0:00.50 /usr/lib/xorg/Xor
    1 root       20   0  220M  9300  6632 S  0.0  0.2  0:01.48 /sbin/init splash
 1012 root       20   0  175M  9340  8572 S  0.0  0.2  0:00.23 /usr/sbin/thermal
 1755 iqbal      20   0 1047M 23360 18288 S  0.0  0.6  0:00.13 /usr/lib/gnome-se
 1144 mysql      20   0 1299M  172M 15700 S  0.0  4.6  0:00.71 /usr/sbin/mysqld
  946 root       20   0  175M  9340  8572 S  0.0  0.2  0:00.27 /usr/sbin/thermal
 2067 iqbal      20   0  701M 30652 27616 S  0.0  1.0  0:00.28 /usr/lib/gnome-te
 1464 iqbal      20   0  274M  7920  6948 S  0.0  0.2  0:00.09 /usr/bin/gnome-ke
 1466 iqbal      20   0  274M  7920  6948 S  0.0  0.2  0:00.04 /usr/bin/gnome-ke
 2205 iqbal      20   0  602M 31844 25520 S  0.0  0.8  0:00.27 /usr/bin/gnome-sc
 1602 iqbal      20   0 4025M  339M  107M S  0.0  9.1  0:00.21 /usr/bin/gnome-sh
F1Help  F2Setup F3Search F4Filter F5Tree  F6SortByF7Nice - F8Nice + F9Kill  F
```

3) ps -A : The **ps** command lists running processes. The following command lists all processes running on your system:



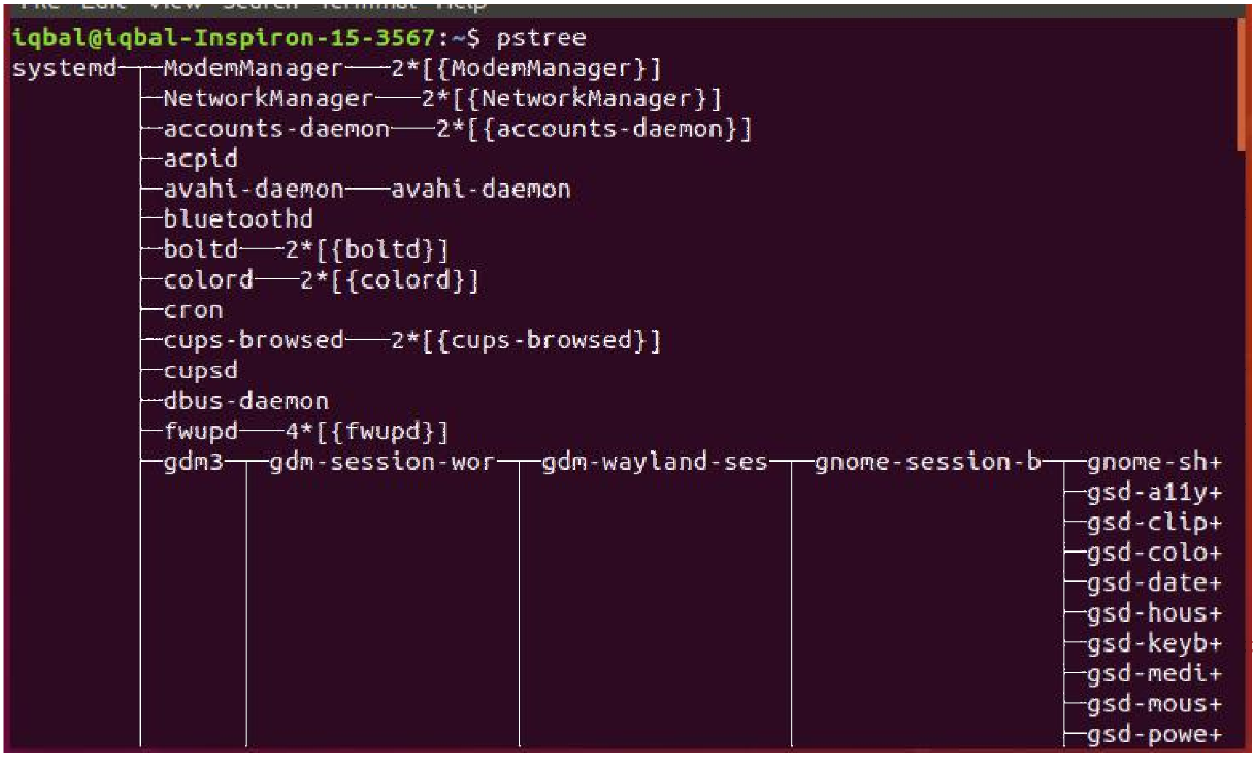
```
iqbal@iqbal-Inspiron-15-3567:~$ ps
  PID TTY          TIME CMD
 2649 pts/0    00:00:00 bash
 2657 pts/0    00:00:00 ps
iqbal@iqbal-Inspiron-15-3567:~$ ps -A
  PID TTY          TIME CMD
    1 ?        00:00:01 systemd
    2 ?        00:00:00 kthreadd
    3 ?        00:00:00 rcu_gp
    4 ?        00:00:00 rcu_par_gp
    6 ?        00:00:00 kworker/0:0H-kb
    8 ?        00:00:00 kworker/u8:0-ev
    9 ?        00:00:00 mm_percpu_wq
   10 ?        00:00:00 ksoftirqd/0
   11 ?        00:00:00 rcu_sched
   12 ?        00:00:00 migration/0
   13 ?        00:00:00 idle_inject/0
   14 ?        00:00:00 cpuhp/0
   15 ?        00:00:00 cpuhp/1
   16 ?        00:00:00 idle_inject/1
   17 ?        00:00:00 migration/1
   18 ?        00:00:00 ksoftirqd/1
   20 ?        00:00:00 kworker/1:0H-kb
   21 ?        00:00:00 cpuhp/2
```

4) ps -A | less: ps -A may be too many processes to read at one time, so we can pipe the output through the **less** command to scroll through them at own pace.ps - A | less:



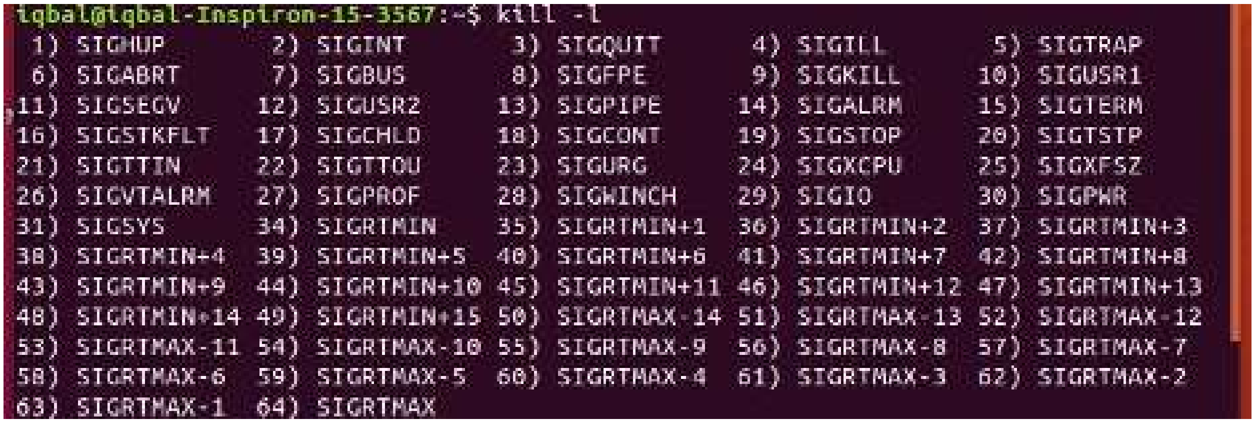
```
  PID TTY          TIME CMD
    1 ?        00:00:01 systemd
    2 ?        00:00:00 kthreadd
    3 ?        00:00:00 rcu_gp
    4 ?        00:00:00 rcu_par_gp
    6 ?        00:00:00 kworker/0:0H-kb
    8 ?        00:00:00 kworker/u8:0-i9
    9 ?        00:00:00 mm_percpu_wq
   10 ?        00:00:00 ksoftirqd/0
   11 ?        00:00:00 rcu_sched
   12 ?        00:00:00 migration/0
   13 ?        00:00:00 idle_inject/0
   14 ?        00:00:00 cpuhp/0
   15 ?        00:00:00 cpuhp/1
   16 ?        00:00:00 idle_inject/1
   17 ?        00:00:00 migration/1
   18 ?        00:00:00 ksoftirqd/1
   20 ?        00:00:00 kworker/1:0H-kb
   21 ?        00:00:00 cpuhp/2
   22 ?        00:00:00 idle_inject/2
   23 ?        00:00:00 migration/2
   24 ?        00:00:00 ksoftirqd/2
   26 ?        00:00:00 kworker/2:0H-kb
:
```

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



```
iqbal@iqbal-Inspiron-15-3567:~$ pstree
systemd─┬─ModemManager───2*[{ModemManager}]
        ├─NetworkManager───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
        ├─fwupd───4*[{fwupd}]
        ├─gdm3─┬─gdm-session-wor─┬─gdm-wayland-ses─┬─gnome-session-b─┬─gnome-sh+
        │      │                 │                 │                 ├─gsd-a11y+
        │      │                 │                 │                 ├─gsd-clip+
        │      │                 │                 │                 ├─gsd-colo+
        │      │                 │                 │                 ├─gsd-date+
        │      │                 │                 │                 ├─gsd-hous+
        │      │                 │                 │                 ├─gsd-keyb+
        │      │                 │                 │                 ├─gsd-medi+
        │      │                 │                 │                 ├─gsd-mous+
        │      │                 │                 │                 ├─gsd-powe+
```

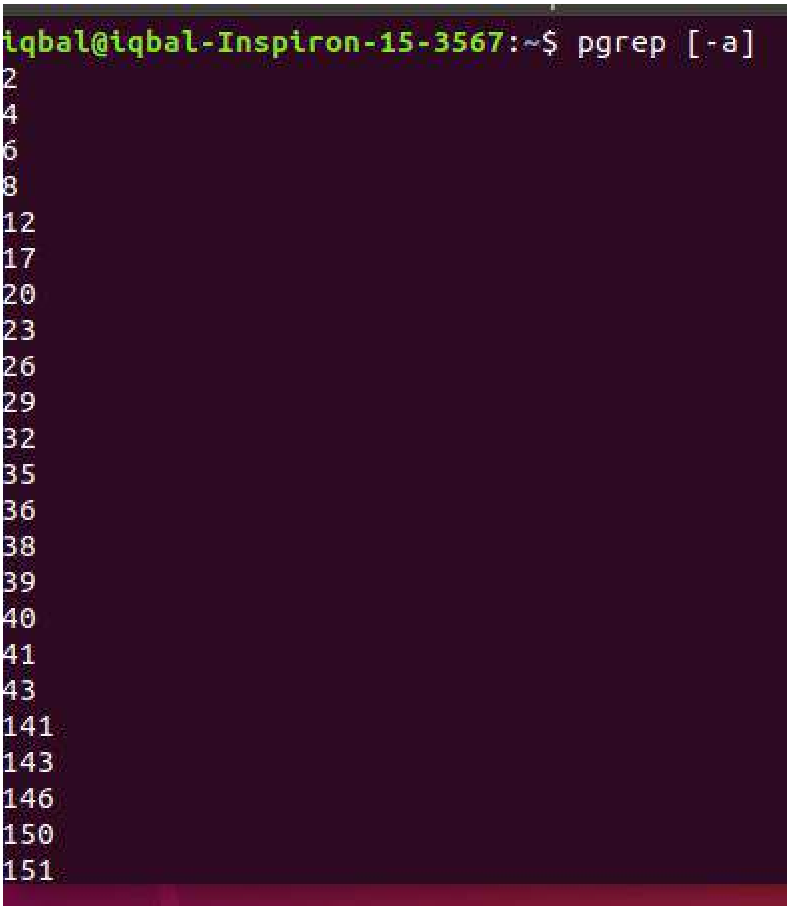
6)kill : "kill [-l]"command List the single names.If arguments follow '-l' they are assumed to be signal numbers for which names should be listed.



```
iqbal@iqbal-Inspiron-15-3567:~$ kill -l
 1) SIGHUP       2) SIGINT       3) SIGQUIT      4) SIGILL       5) SIGTRAP
 6) SIGABRT      7) SIGBUS       8) SIGFPE       9) SIGKILL     10) SIGUSR1
11) SIGSEGV     12) SIGUSR2     13) SIGPIPE     14) SIGALRM     15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
21) SIGTTIN     22) SIGTTOU     23) SIGURG      24) SIGXCPU     25) SIGXFSZ
26) SIGVTALRM   27) SIGPROF     28) SIGWINCH    29) SIGIO       30) SIGPWR
31) SIGSYS      34) SIGRTMIN    35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
```

7) pgrep:

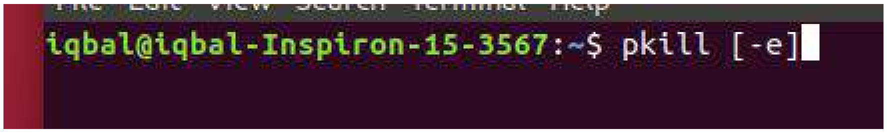
pgrep [-a] : This command list PID and full command line.



```
iqbal@iqbal-Inspiron-15-3567:~$ pgrep [-a]
2
4
6
8
12
17
20
23
26
29
32
35
36
38
39
40
41
43
141
143
146
150
151
```

8) pkill :

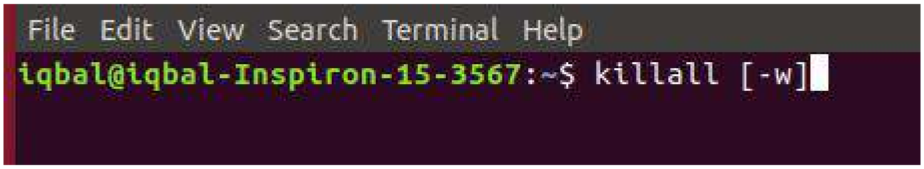
pkill [-e] : Display what is killed.



```
iqbal@iqbal-Inspiron-15-3567:~$ pkill [-e]
```

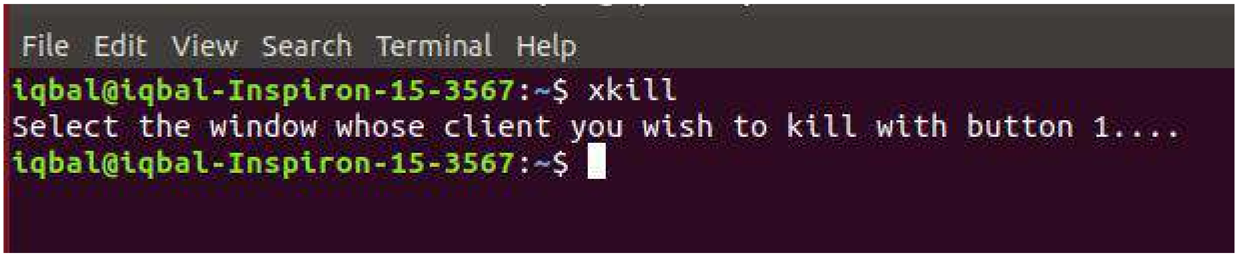
9)killall:

killall [-w] :Wait for process to die.



```
iqbal@iqbal-Inspiron-15-3567:~$ killall [-w]
```

10) “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.



```
iqbal@iqbal-Inspiron-15-3567:~$ xkill
Select the window whose client you wish to kill with button 1....
iqbal@iqbal-Inspiron-15-3567:~$
```

Conclusion: From this lab I learn linux command.now I can use this command where I need.I am so exited to learn this type of command in linux