UNIX Lab Assignment 3

Name: Ninad Rao Roll No. 53

Process Management Commands:

Ps - Ps command is used to list the currently running processes and their PIDs along with some other information depends on different options. It reads the process information from the virtual files in /proc file-system. /proc contains virtual files, this is the reason it's referred as a virtual file system.

```
ninad@ninad-VirtualBox:~$ ps
    PID TTY          TIME CMD
   1714 pts/0    00:00:00 bash
   1726 pts/0    00:00:00 ps
ninad@ninad-VirtualBox:~$ ps -a
    PID TTY          TIME CMD
    886 tty2     00:00:01 Xorg
    977 tty2     00:00:00 gnome-session-b
   1727 pts/0    00:00:00 ps
ninad@ninad-VirtualBox:~$ 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
      5 ?        00:00:00 kworker/0:0-cgroup_destroy
      6 ?        00:00:00 kworker/0:0H-kblockd
      7 ?        00:00:00 kworker/0:1-events
      8 ?        00:00:00 kworker/u2:0-ext4-rsv-conversion
      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 kdevtmpfs
     16 ?        00:00:00 netns
     17 ?        00:00:00 rcu_tasks_kthre
     18 ?        00:00:00 rcu_tasks_rude_
     19 ?        00:00:00 rcu_tasks_trace
     20 ?        00:00:00 kauditd
     21 ?        00:00:00 khungtaskd
     22 ?        00:00:00 oom_reaper
     23 ?        00:00:00 writeback
     24 ?        00:00:00 kcompactd0
     25 ?        00:00:00 ksmd
     26 ?        00:00:00 khugepaged
     72 ?        00:00:00 kintegrityd
     73 ?        00:00:00 kblockd
     74 ?        00:00:00 blkcg_punt_bio
     75 ?        00:00:00 tpm_dev_wq
     76 ?        00:00:00 ata_sff
     77 ?        00:00:00 md
     78 ?        00:00:00 edac-poller
     79 ?        00:00:00 devfreq_wq
     80 ?        00:00:00 watchdogd
     81 ?        00:00:00 kworker/u2:1-events_power_efficient
     82 ?        00:00:00 pm_wq
     84 ?        00:00:00 kswapd0
     85 ?        00:00:00 ecryptfs-kthrea
     87 ?        00:00:00 kthrotld
```

```
   1091 ?        00:00:00 ssh-agent
   1113 ?        00:00:00 at-spi-bus-laun
   1118 ?        00:00:00 dbus-daemon
   1134 ?        00:00:00 gnome-session-c
   1141 ?        00:00:00 gnome-session-b
   1155 ?        00:00:07 gnome-shell
   1180 ?        00:00:00 ibus-daemon
   1184 ?        00:00:00 ibus-memconf
   1185 ?        00:00:01 ibus-extension-
   1189 ?        00:00:00 ibus-x11
   1192 ?        00:00:00 ibus-portal
   1202 ?        00:00:00 at-spi2-registr
   1206 ?        00:00:00 xdg-permission-
   1211 ?        00:00:00 gnome-shell-cal
   1217 ?        00:00:00 evolution-sourc
   1225 ?        00:00:00 evolution-calen
   1235 ?        00:00:00 dconf-service
   1242 ?        00:00:00 evolution-addre
   1258 ?        00:00:00 packagekitd
   1267 ?        00:00:00 gvfsd-trash
   1274 ?        00:00:00 gjs
   1291 ?        00:00:00 gsd-a11y-settin
   1293 ?        00:00:00 gsd-color
   1296 ?        00:00:00 gsd-datetime
   1299 ?        00:00:00 gsd-housekeepin
   1307 ?        00:00:00 gsd-keyboard
   1308 ?        00:00:00 gsd-media-keys
   1310 ?        00:00:00 gsd-power
   1312 ?        00:00:00 gsd-print-notif
   1313 ?        00:00:00 gsd-rfkill
   1316 ?        00:00:00 gsd-screensaver
   1321 ?        00:00:00 gsd-sharing
   1324 ?        00:00:00 gsd-smartcard
   1327 ?        00:00:00 gsd-sound
   1329 ?        00:00:00 gsd-disk-utilit
   1334 ?        00:00:00 gsd-usb-protect
   1340 ?        00:00:00 gsd-wacom
   1341 ?        00:00:00 gsd-wwan
   1346 ?        00:00:00 gsd-xsettings
   1368 ?        00:00:00 evolution-alarm
   1409 ?        00:00:00 colord
   1423 ?        00:00:00 gsd-printer
   1428 ?        00:00:00 ibus-engine-sim
   1506 ?        00:00:00 gvfsd-metadata
   1511 ?        00:00:00 update-notifier
   1631 ?        00:00:00 gnome-calendar
   1634 ?        00:00:00 seahorse
   1635 ?        00:00:00 gnome-terminal-
   1714 pts/0    00:00:00 bash
   1729 pts/0    00:00:00 ps
ninad@ninad-VirtualBox:~$
```

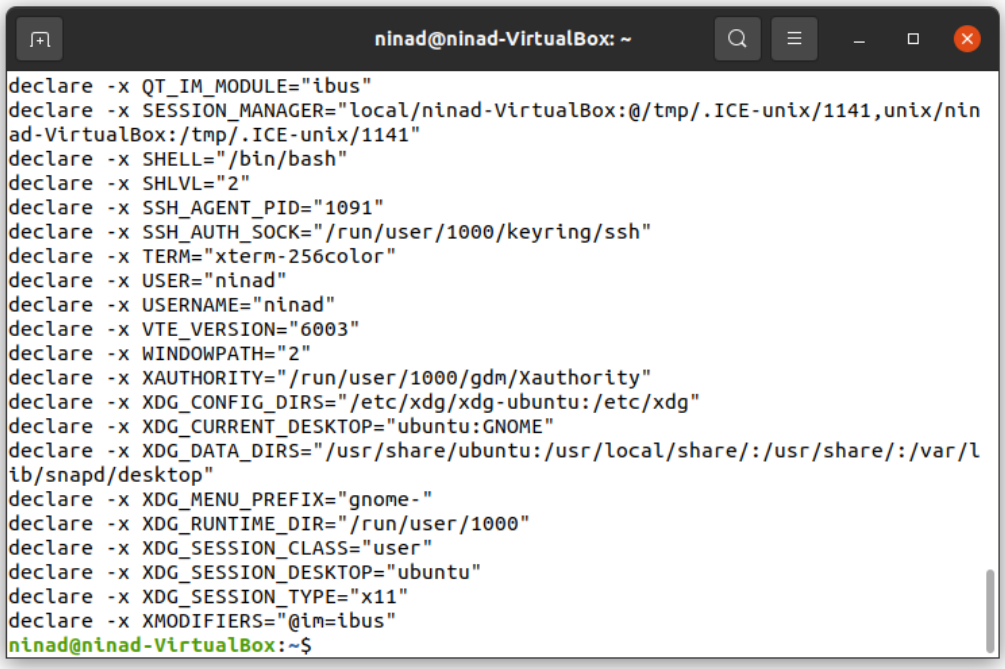
Pstree - Pstree command in Unix that shows the running processes as a tree which is a more convenient way to display the processes hierarchy and makes the output more visually appealing. The root of the tree is either init or the process with the given pid. Pstree can also be installed in other Unix systems.

```
ninad@ninad-VirtualBox:~$ pstree
systemd─┬─ModemManager───2*[{ModemManager}]
        ├─NetworkManager───2*[{NetworkManager}]
        ├─3*[VBoxClient───VBoxClient───2*[{VBoxClient}]]
        ├─VBoxClient───VBoxClient───3*[{VBoxClient}]
        ├─VBoxService───8*[{VBoxService}]
        ├─accounts-daemon───2*[{accounts-daemon}]
        ├─acpid
        ├─avahi-daemon───avahi-daemon
        ├─colord───2*[{colord}]
        ├─cron
        ├─cups-browsed───2*[{cups-browsed}]
        ├─cupsd
        ├─dbus-daemon
        ├─gdm3─┬─gdm-session-wor─┬─gdm-x-session─┬─Xorg───5*[{Xorg}]
        │      │                 │               ├─gnome-session-b─┬─ssh-agent
        │      │                 │               │                 └─2*[{gnome-session-b}]
        │      │                 │               └─2*[{gdm-x-session}]
        │      │                 └─2*[{gdm-session-wor}]
        │      └─2*[{gdm3}]
        ├─gnome-keyring-d───3*[{gnome-keyring-d}]
        ├─2*[kerneloops]
        ├─networkd-dispat
        ├─polkitd───2*[{polkitd}]
        ├─rsyslogd───3*[{rsyslogd}]
        ├─rtkit-daemon───2*[{rtkit-daemon}]
        ├─snapd───7*[{snapd}]
        ├─switcheroo-cont───2*[{switcheroo-cont}]
        ├─systemd─┬─(sd-pam)
        │         ├─at-spi-bus-laun─┬─dbus-daemon
        │         │                 └─3*[{at-spi-bus-laun}]
        │         ├─at-spi2-registr───2*[{at-spi2-registr}]
        │         ├─dbus-daemon
        │         ├─dconf-service───2*[{dconf-service}]
        │         ├─evolution-addre───5*[{evolution-addre}]
        │         ├─evolution-calen───8*[{evolution-calen}]
        │         ├─evolution-sourc───3*[{evolution-sourc}]
        │         ├─gjs───4*[{gjs}]
        │         ├─gnome-calendar───5*[{gnome-calendar}]
        │         ├─gnome-session-b─┬─evolution-alarm───5*[{evolution-alarm}]
        │         │                 ├─gsd-disk-utilit───2*[{gsd-disk-utilit}]
        │         │                 ├─update-notifier───3*[{update-notifier}]
        │         │                 └─3*[{gnome-session-b}]
        │         ├─gnome-session-c───{gnome-session-c}
        │         ├─gnome-shell─┬─ibus-daemon─┬─ibus-engine-sim───2*[{ibus-engine-sim}]
        │         │              │             ├─ibus-extension-───3*[{ibus-extension-}]
        │         │              │             ├─ibus-memconf───2*[{ibus-memconf}]
        │         │              │             └─2*[{ibus-daemon}]
        │         │              └─9*[{gnome-shell}]
        │         └─gnome-shell-cal───5*[{gnome-shell-cal}]
```

```
        │           ├─gnome-terminal-─┬─bash───pstree
        │           │                 └─4*[{gnome-terminal-}]
        │           ├─goa-daemon───3*[{goa-daemon}]
        │           ├─goa-identity-se───2*[{goa-identity-se}]
        │           ├─gsd-a11y-settin───3*[{gsd-a11y-settin}]
        │           ├─gsd-color───3*[{gsd-color}]
        │           ├─gsd-datetime───3*[{gsd-datetime}]
        │           ├─gsd-housekeepin───3*[{gsd-housekeepin}]
        │           ├─gsd-keyboard───3*[{gsd-keyboard}]
        │           ├─gsd-media-keys───3*[{gsd-media-keys}]
        │           ├─gsd-power───3*[{gsd-power}]
        │           ├─gsd-print-notif───2*[{gsd-print-notif}]
        │           ├─gsd-printer───2*[{gsd-printer}]
        │           ├─gsd-rfkill───2*[{gsd-rfkill}]
        │           ├─gsd-screensaver───2*[{gsd-screensaver}]
        │           ├─gsd-sharing───3*[{gsd-sharing}]
        │           ├─gsd-smartcard───4*[{gsd-smartcard}]
        │           ├─gsd-sound───3*[{gsd-sound}]
        │           ├─gsd-usb-protect───3*[{gsd-usb-protect}]
        │           ├─gsd-wacom───2*[{gsd-wacom}]
        │           ├─gsd-wwan───3*[{gsd-wwan}]
        │           ├─gsd-xsettings───3*[{gsd-xsettings}]
        │           ├─gvfs-afc-volume───3*[{gvfs-afc-volume}]
        │           ├─gvfs-goa-volume───2*[{gvfs-goa-volume}]
        │           ├─gvfs-gphoto2-vo───2*[{gvfs-gphoto2-vo}]
        │           ├─gvfs-mtp-volume───2*[{gvfs-mtp-volume}]
        │           ├─gvfs-udisks2-vo───3*[{gvfs-udisks2-vo}]
        │           ├─gvfsd─┬─gvfsd-trash───2*[{gvfsd-trash}]
        │           │       └─2*[{gvfsd}]
        │           ├─gvfsd-fuse───5*[{gvfsd-fuse}]
        │           ├─gvfsd-metadata───2*[{gvfsd-metadata}]
        │           ├─ibus-portal───2*[{ibus-portal}]
        │           ├─ibus-x11───2*[{ibus-x11}]
        │           ├─pulseaudio───3*[{pulseaudio}]
        │           ├─seahorse───3*[{seahorse}]
        │           ├─tracker-miner-f───4*[{tracker-miner-f}]
        │           └─xdg-permission-───2*[{xdg-permission-}]
        ├─systemd-journal
        ├─systemd-logind
        ├─systemd-resolve
        ├─systemd-timesyn───{systemd-timesyn}
        ├─systemd-udevd
        ├─udisksd───4*[{udisksd}]
        ├─unattended-upgr───{unattended-upgr}
        ├─upowerd───2*[{upowerd}]
        ├─whoopsie───2*[{whoopsie}]
        └─wpa_supplicant
ninad@ninad-VirtualBox:~$
```

Nice - Nice command in Unix helps in execution of a program/process with modified scheduling priority. It launches a process with a user-defined scheduling priority. In this, if we give a process a higher priority, then Kernel will allocate more CPU time to that process.

```
ninad@ninad-VirtualBox:~$ ps -el | grep terminal
0 S  1000    1635     869  0  80   0 - 205477 poll_s ?       00:00:01 gnome-terminal-
ninad@ninad-VirtualBox:~$ nice -10 gnome-terminal
ninad@ninad-VirtualBox:~$
```



Kill - Kill command in Unix (located in /bin/kill), is a built-in command which is used to terminate processes manually. kill command sends a signal to a process which terminates the process. If the user doesn't specify any signal which is to be sent along with the kill command then default TERM signal is sent that terminates the process.

```
ninad@ninad-VirtualBox:~$ ps
    PID TTY          TIME CMD
   1714 pts/0    00:00:00 bash
   1760 pts/0    00:00:00 ps
ninad@ninad-VirtualBox:~$ kill 1760
bash: kill: (1760) - No such process
ninad@ninad-VirtualBox:~$ 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
ninad@ninad-VirtualBox:~$
```

Pkill - Pkill is a command-line utility that sends signals to the processes of a running program based on given criteria. The processes can be specified by their full or partial names, a user running the process, or other attributes.

```
ninad@ninad-VirtualBox:~$ pkill --help

Usage:
 pkill [options] <pattern>

Options:
 -<sig>, --signal <sig>    signal to send (either number or name)
 -e, --echo                display what is killed
 -c, --count               count of matching processes
 -f, --full                use full process name to match
 -g, --pgroup <PGID,...>   match listed process group IDs
 -G, --group <GID,...>     match real group IDs
 -i, --ignore-case         match case insensitively
 -n, --newest              select most recently started
 -o, --oldest              select least recently started
 -P, --parent <PPID,...>   match only child processes of the given parent
 -s, --session <SID,...>   match session IDs
 -t, --terminal <tty,...>  match by controlling terminal
 -u, --euid <ID,...>       match by effective IDs
 -U, --uid <ID,...>        match by real IDs
 -x, --exact               match exactly with the command name
 -F, --pidfile <file>      read PIDs from file
 -L, --logpidfile          fail if PID file is not locked
 -r, --runstates <state>   match runstates [D,S,Z,...]
 --ns <PID>                match the processes that belong to the same
                           namespace as <pid>
 --nslist <ns,...>         list which namespaces will be considered for
                           the --ns option.
                           Available namespaces: ipc, mnt, net, pid, user, uts

 -h, --help     display this help and exit
 -V, --version  output version information and exit

For more details see pgrep(1).
ninad@ninad-VirtualBox:~$
```

Killall - Killall sends a signal to all processes running any of the specified commands. If no signal name is specified, SIGTERM is sent. Killall returns a zero return code if at least one process was killed for each listed command, or no commands were listed and at least one process matched the -u and -Z search criteria. Killall returns non-zero otherwise. A killall process never kills itself, but may kill other killall processes.

```
ninad@ninad-VirtualBox:~$ killall --help
Usage: killall [ -Z CONTEXT ] [ -u USER ] [ -y TIME ] [ -o TIME ] [ -eIgiqrvw ]
               [ -s SIGNAL | -SIGNAL ] NAME...
       killall -l, --list
       killall -V, --version

  -e,--exact          require exact match for very long names
  -I,--ignore-case    case insensitive process name match
  -g,--process-group  kill process group instead of process
  -y,--younger-than   kill processes younger than TIME
  -o,--older-than     kill processes older than TIME
  -i,--interactive    ask for confirmation before killing
  -l,--list           list all known signal names
  -q,--quiet          don't print complaints
  -r,--regexp         interpret NAME as an extended regular expression
  -s,--signal SIGNAL  send this signal instead of SIGTERM
  -u,--user USER      kill only process(es) running as USER
  -v,--verbose        report if the signal was successfully sent
  -V,--version        display version information
  -w,--wait           wait for processes to die
  -n,--ns PID         match processes that belong to the same namespaces
                      as PID
  -Z,--context REGEXP kill only process(es) having context
                      (must precede other arguments)

ninad@ninad-VirtualBox:~$
```

Xkill - Command xkill is used to kill a process on X server without passing process name or PID. It forces the X server to close the communication with its clients, which ultimately kills its clients by its X resource. In short, xkill instructs X server to terminate the client. Using xkill command when you want to kill a process, type xkill on the terminal. Your cursor will change in the shape of x, click on the window which you want to kill using x cursor.

```
ninad@ninad-VirtualBox:~$ xkill --help
xkill: unrecognized argument --help

usage:  xkill [-option ...]
where options include:
    -display displayname    X server to contact
    -id resource            resource whose client is to be killed
    -frame                  don't ignore window manager frames
    -button number          specific button to be pressed to select window
    -all                    kill all clients with top level windows
    -version                print version and exit

ninad@ninad-VirtualBox:~$
```

Fg - Fg command in Unix used to put a background job in the foreground.

Bg - Bg command in Unix is used to place foreground jobs in the background.

```
ninad@ninad-VirtualBox:~$ jobs
ninad@ninad-VirtualBox:~$ sleep 500
^Z
[1]+  Stopped                 sleep 500
ninad@ninad-VirtualBox:~$ jobs
[1]+  Stopped                 sleep 500
ninad@ninad-VirtualBox:~$ bg %1
[1]+ sleep 500 &
ninad@ninad-VirtualBox:~$ jobs
[1]+  Running                 sleep 500 &
ninad@ninad-VirtualBox:~$ fg %1
sleep 500
^Z
[1]+  Stopped                 sleep 500
ninad@ninad-VirtualBox:~$ fg --help
fg: fg [job_spec]
    Move job to the foreground.

    Place the job identified by JOB_SPEC in the foreground, making it the
    current job.  If JOB_SPEC is not present, the shell's notion of the
    current job is used.

    Exit Status:
    Status of command placed in foreground, or failure if an error occurs.
ninad@ninad-VirtualBox:~$ bg --help
bg: bg [job_spec ...]
    Move jobs to the background.

    Place the jobs identified by each JOB_SPEC in the background, as if they
    had been started with `&'.  If JOB_SPEC is not present, the shell's notion
    of the current job is used.

    Exit Status:
    Returns success unless job control is not enabled or an error occurs.
ninad@ninad-VirtualBox:~$
```

Pgrep - Pgrep looks through the currently running processes and lists the process IDs which matches the selection criteria to stdout. All the criteria have to match. It is a command-line utility that allows you to find the process IDs of a running program based on given criteria. It can be a full or partial process name, a user running the process, or other attributes.

```
ninad@ninad-VirtualBox:~$ pgrep --help

Usage:
 pgrep [options] <pattern>

Options:
 -d, --delimiter <string>  specify output delimiter
 -l, --list-name           list PID and process name
 -a, --list-full           list PID and full command line
 -v, --inverse             negates the matching
 -w, --lightweight         list all TID
 -c, --count               count of matching processes
 -f, --full                use full process name to match
 -g, --pgroup <PGID,...>   match listed process group IDs
 -G, --group <GID,...>     match real group IDs
 -i, --ignore-case         match case insensitively
 -n, --newest              select most recently started
 -o, --oldest              select least recently started
 -P, --parent <PPID,...>   match only child processes of the given parent
 -s, --session <SID,...>   match session IDs
 -t, --terminal <tty,...>  match by controlling terminal
 -u, --euid <ID,...>       match by effective IDs
 -U, --uid <ID,...>        match by real IDs
 -x, --exact               match exactly with the command name
 -F, --pidfile <file>      read PIDs from file
 -L, --logpidfile          fail if PID file is not locked
 -r, --runstates <state>   match runstates [D,S,Z,...]
 --ns <PID>                match the processes that belong to the same
                           namespace as <pid>
 --nslist <ns,...>         list which namespaces will be considered for
                           the --ns option.
                           Available namespaces: ipc, mnt, net, pid, user, uts

 -h, --help     display this help and exit
 -V, --version  output version information and exit

For more details see pgrep(1).
ninad@ninad-VirtualBox:~$
```

Renice - Renice command allows you to change and modify the scheduling priority of an already running process. Unix Kernel schedules the process and allocates CPU time accordingly for each of them. The renice command modifies the priority of running processes. It is similar to the nice command, but is used for processes that are already running.

```
ninad@ninad-VirtualBox:~$ ps -el | grep terminal
0 S  1000    1533     890  0  80   0 - 206076 poll_s ?       00:00:01 gnome-terminal-
ninad@ninad-VirtualBox:~$ renice -n 15 -p 1533
1533 (process ID) old priority 0, new priority 15
ninad@ninad-VirtualBox:~$ renice -n 10 -g 1
renice: failed to set priority for 1 (process group ID): Operation not permitted
ninad@ninad-VirtualBox:~$ renice --help

Usage:
 renice [-n] <priority> [-p|--pid] <pid>...
 renice [-n] <priority>  -g|--pgrp <pgid>...
 renice [-n] <priority>  -u|--user <user>...

Alter the priority of running processes.

Options:
 -n, --priority <num>   specify the nice increment value
 -p, --pid <id>         interpret argument as process ID (default)
 -g, --pgrp <id>        interpret argument as process group ID
 -u, --user <name>|<id> interpret argument as username or user ID

 -h, --help             display this help
 -V, --version          display version

For more details see renice(1).
ninad@ninad-VirtualBox:~$
```

Top - Top command is used to show the Unix processes. It provides a dynamic real-time view of the running system. Usually, this command shows the summary information of the system and the list of processes or threads which are currently managed by the Unix Kernel.
As soon as you run this command it will open an interactive command mode where the top half portion will contain the statistics of processes and resource usage. And the lower half contains a list of the currently running processes. Pressing q will simply exit the command mode.

```
top - 10:33:43 up  1:30,  1 user,  load average: 0.00, 0.00, 0.00
Tasks: 175 total,   1 running, 174 sleeping,   0 stopped,   0 zombie
%Cpu(s):  4.8 us,  0.7 sy,  0.3 ni, 93.8 id,  0.3 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   1987.3 total,    347.8 free,    687.6 used,    951.9 buff/cache
MiB Swap:    448.5 total,    448.5 free,      0.0 used.   1120.8 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   1177 ninad     20   0 3759604 379500 133464 S   5.3  18.6   0:40.35 gnome-shell
    907 ninad     20   0  572332  96056  51704 S   1.7   4.7   0:07.99 Xorg
   1533 ninad     35  15  825404  52540  38740 S   1.7   2.6   0:02.62 gnome-terminal-
      1 root      20   0  102064  11776   8540 S   0.0   0.6   0:02.78 systemd
      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
      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/0:0H-kblockd
      9 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
     10 root      20   0       0      0      0 S   0.0   0.0   0:00.06 ksoftirqd/0
     11 root      20   0       0      0      0 I   0.0   0.0   0:00.38 rcu_sched
     12 root      rt   0       0      0      0 S   0.0   0.0   0:00.03 migration/0
     13 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/0
     14 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
     15 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kdevtmpfs
     16 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 netns
     17 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_kthre
     18 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_rude_
     19 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_trace
     20 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kauditd
     21 root      20   0       0      0      0 S   0.0   0.0   0:00.00 khungtaskd
     22 root      20   0       0      0      0 S   0.0   0.0   0:00.00 oom_reaper
     23 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 writeback
     24 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kcompactd0
     25 root      25   5       0      0      0 S   0.0   0.0   0:00.00 ksmd
     26 root      39  19       0      0      0 S   0.0   0.0   0:00.00 khugepaged
     72 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kintegrityd
     73 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kblockd
     74 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 blkcg_punt_bio
     75 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 tpm_dev_wq
     76 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 ata_sff
     77 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 md
     78 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 edac-poller
     79 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 devfreq_wq
     80 root      rt   0       0      0      0 S   0.0   0.0   0:00.00 watchdogd
     82 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 pm_wq
     86 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kswapd0
     87 root      20   0       0      0      0 S   0.0   0.0   0:00.00 ecryptfs-kthrea
     89 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kthrotld
     90 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 acpi_thermal_pm
     91 root      20   0       0      0      0 S   0.0   0.0   0:00.01 scsi_eh_0
     92 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 scsi_tmf_0
     93 root      20   0       0      0      0 S   0.0   0.0   0:00.00 scsi_eh_1
     94 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 scsi_tmf_1
```

Df - The df command is used to display information related to file systems about total space and available space. Unix df command is used to display the disk space used in the file system. The 'df' stands for "disk filesystem." It defines the number of blocks used, the number of blocks available, and the directory where the file system is mounted.

```
ninad@ninad-VirtualBox:~$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
udev              987940       0    987940   0% /dev
tmpfs             203500    1324    202176   1% /run
/dev/sda5        9736500 7792380   1429816  85% /
tmpfs            1017492       0   1017492   0% /dev/shm
tmpfs               5120       4      5116   1% /run/lock
tmpfs            1017492       0   1017492   0% /sys/fs/cgroup
/dev/loop0         56832   56832         0 100% /snap/core18/1988
/dev/loop2         52352   52352         0 100% /snap/snap-store/518
/dev/loop3        224256  224256         0 100% /snap/gnome-3-34-1804/66
/dev/loop1         56832   56832         0 100% /snap/core18/1997
/dev/loop4         31872   31872         0 100% /snap/snapd/11036
/dev/loop5         66432   66432         0 100% /snap/gtk-common-themes/1514
/dev/loop6         33152   33152         0 100% /snap/snapd/11402
/dev/sda1         523248       4    523244   1% /boot/efi
tmpfs             203496      20    203476   1% /run/user/1000
/dev/sr0           59724   59724         0 100% /media/ninad/VBox_GAs_6.1.18
ninad@ninad-VirtualBox:~$ df --help
Usage: df [OPTION]... [FILE]...
Show information about the file system on which each FILE resides,
or all file systems by default.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all             include pseudo, duplicate, inaccessible file systems
  -B, --block-size=SIZE  scale sizes by SIZE before printing them; e.g.,
                           '-BM' prints sizes in units of 1,048,576 bytes;
                           see SIZE format below
  -h, --human-readable  print sizes in powers of 1024 (e.g., 1023M)
  -H, --si              print sizes in powers of 1000 (e.g., 1.1G)
  -i, --inodes          list inode information instead of block usage
  -k                    like --block-size=1K
  -l, --local           limit listing to local file systems
      --no-sync         do not invoke sync before getting usage info (default)
      --output[=FIELD_LIST]  use the output format defined by FIELD_LIST,
                               or print all fields if FIELD_LIST is omitted.
  -P, --portability     use the POSIX output format
      --sync            invoke sync before getting usage info
      --total           elide all entries insignificant to available space,
                          and produce a grand total
  -t, --type=TYPE       limit listing to file systems of type TYPE
  -T, --print-type      print file system type
  -x, --exclude-type=TYPE   limit listing to file systems not of type TYPE
  -v                    (ignored)
      --help     display this help and exit
      --version  output version information and exit

Display values are in units of the first available SIZE from --block-size,
and the DF_BLOCK_SIZE, BLOCK_SIZE and BLOCKSIZE environment variables.
Otherwise, units default to 1024 bytes (or 512 if POSIXLY_CORRECT is set).

The SIZE argument is an integer and optional unit (example: 10K is 10*1024).
Units are K,M,G,T,P,E,Z,Y (powers of 1024) or KB,MB,... (powers of 1000).

FIELD_LIST is a comma-separated list of columns to be included.  Valid
field names are: 'source', 'fstype', 'itotal', 'iused', 'iavail', 'ipcent',
'size', 'used', 'avail', 'pcent', 'file' and 'target' (see info page).

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Full documentation at: <https://www.gnu.org/software/coreutils/df>
or available locally via: info '(coreutils) df invocation'
```

Free - In UNIX, there exists a command line utility for this and that is free command which displays the total amount of free space available along with the amount of memory used and swap memory in the system, and also the buffers used by the kernel. As free displays the details of the memory related to your system, its syntax doesn't need any arguments to be passed but only options which you can use according to your wish.

```
ninad@ninad-VirtualBox:~$ free
              total        used        free      shared  buff/cache   available
Mem:        2034984      708484      351596       18652      974904     1143296
Swap:        459260           0      459260
ninad@ninad-VirtualBox:~$ free --help

Usage:
 free [options]

Options:
 -b, --bytes         show output in bytes
     --kilo          show output in kilobytes
     --mega          show output in megabytes
     --giga          show output in gigabytes
     --tera          show output in terabytes
     --peta          show output in petabytes
 -k, --kibi          show output in kibibytes
 -m, --mebi          show output in mebibytes
 -g, --gibi          show output in gibibytes
     --tebi          show output in tebibytes
     --pebi          show output in pebibytes
 -h, --human         show human-readable output
     --si            use powers of 1000 not 1024
 -l, --lohi          show detailed low and high memory statistics
 -t, --total         show total for RAM + swap
 -s N, --seconds N   repeat printing every N seconds
 -c N, --count N     repeat printing N times, then exit
 -w, --wide          wide output

     --help     display this help and exit
 -V, --version  output version information and exit

For more details see free(1).
ninad@ninad-VirtualBox:~$
```

Pidof - Pidof command is used to find out the process IDs of a specific running program. It is basically an identification number that is automatically assigned to each process when it is created.To be sure that only the PIDs of the program you are searching for are displayed, use the full pathname to the program as an argument. For example, if you have two running programs with the same name located in two different directories pidof will show PIDs of both running programs.

```
ninad@ninad-VirtualBox:~$ pidof bash
2331
ninad@ninad-VirtualBox:~$ pidof firefox
2529 2475 2456 2422 2367
ninad@ninad-VirtualBox:~$ pidof -s firefox
2529
ninad@ninad-VirtualBox:~$ pidof -x firefox
2529 2475 2456 2422 2367
ninad@ninad-VirtualBox:~$ pidof -c firefox
2529 2475 2456 2422 2367
ninad@ninad-VirtualBox:~$ pidof -o 2529 firefox
2475 2456 2422 2367
ninad@ninad-VirtualBox:~$ pidof -o 2331 bash
ninad@ninad-VirtualBox:~$
```

Sleep - Sleep command is used to create a dummy job. A dummy job helps in delaying the execution. It takes time in seconds by default but a small suffix (s, m, h, d) can be added at the end to convert it into any other format. This command pauses the execution for an amount of time which is defined by NUMBER.

```
ninad@ninad-VirtualBox:~$ sleep --help
Usage: sleep NUMBER[SUFFIX]...
  or:  sleep OPTION
Pause for NUMBER seconds.  SUFFIX may be 's' for seconds (the default),
'm' for minutes, 'h' for hours or 'd' for days.  Unlike most implementations
that require NUMBER be an integer, here NUMBER may be an arbitrary floating
point number.  Given two or more arguments, pause for the amount of time
specified by the sum of their values.

      --help     display this help and exit
      --version  output version information and exit

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Full documentation at: <https://www.gnu.org/software/coreutils/sleep>
or available locally via: info '(coreutils) sleep invocation'
ninad@ninad-VirtualBox:~$ sleep --version
sleep (GNU coreutils) 8.30
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by Jim Meyering and Paul Eggert.
ninad@ninad-VirtualBox:~$
```

Jobs - Jobs command is used to list the jobs that you are running in the background and in the foreground. If the prompt is returned with no information no jobs are present. All shells are not capable of running this command. This command is only available in the csh, bash, tcsh, and ksh shells.

```
ninad@ninad-VirtualBox:~$ jobs --help
jobs: jobs [-lnprs] [jobspec ...] or jobs -x command [args]
    Display status of jobs.

    Lists the active jobs.  JOBSPEC restricts output to that job.
    Without options, the status of all active jobs is displayed.

    Options:
      -l        lists process IDs in addition to the normal information
      -n        lists only processes that have changed status since the last
                notification
      -p        lists process IDs only
      -r        restrict output to running jobs
      -s        restrict output to stopped jobs

    If -x is supplied, COMMAND is run after all job specifications that
    appear in ARGS have been replaced with the process ID of that job's
    process group leader.

    Exit Status:
    Returns success unless an invalid option is given or an error occurs.
    If -x is used, returns the exit status of COMMAND.
ninad@ninad-VirtualBox:~$
```