

# Linux lab4

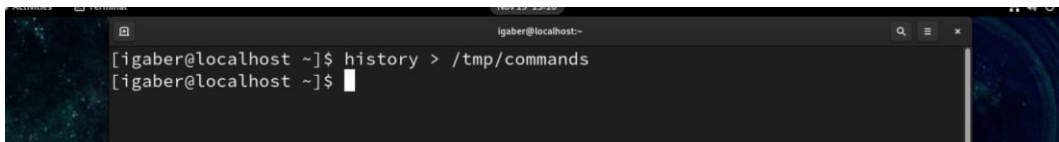
## Name :Israa Mohamed Gaber

1. List the user commands and redirect the output to /tmp/commands.

```
[igaber@localhost ~]$ history > /tmp/commands
```

Correct

```
[igaber@localhost ~]$ ls /usr/bin/ > /tmp/commands
```

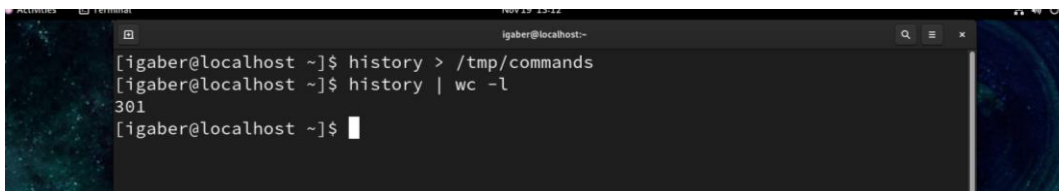
A terminal window with a dark background and a space-themed wallpaper. The prompt is [igaber@localhost ~]. The command history shows [igaber@localhost ~]\$ history > /tmp/commands. The cursor is on the next line.

2. Count the number of user commands

```
[igaber@localhost ~]$ history | wc -l
```

Correct

```
[igaber@localhost ~]$ sudo wc -l /tmp/commands
```

A terminal window showing the command history | wc -l and the output 301. The prompt is [igaber@localhost ~].

3. ....

Start sleep with lower priority (30), then change it back to normal

```
[igaber@localhost ~]$ nice -n 10 sleep 100 &
```

```
[1] 5129
```

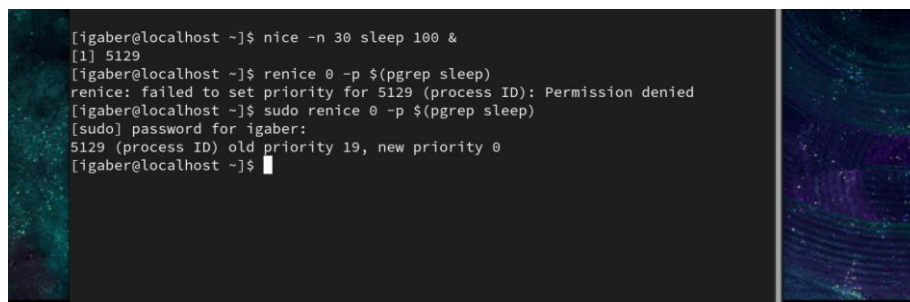
```
[igaber@localhost ~]$ renice 0 -p $(pgrep sleep)
```

renice: failed to set priority for 5129 (process ID): Permission denied

```
[igaber@localhost ~]$ sudo renice 0 -p $(pgrep sleep)
```

```
[sudo] password for igaber:
```

5129 (process ID) old priority 19, new priority 0

A terminal window showing the execution of nice -n 30 sleep 100 &, the process ID [1] 5129, the renice command failing with a permission denied error, and the successful execution of sudo renice 0 -p \$(pgrep sleep) after providing the password. The final output is 5129 (process ID) old priority 19, new priority 0.

4. Get its process ID in 2 ways

```
[igaber@localhost ~]$ pgrep sleep
```

```
[igaber@localhost ~]$ ps -ef | grep sleep
```

Or

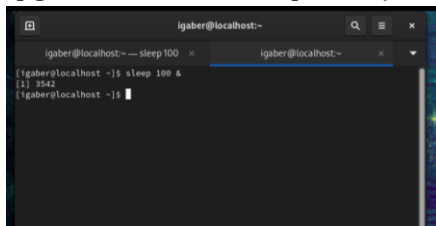
```
[igaber@localhost ~]$ ps -aux | grep sleep
```



```
[igaber@localhost ~]$ pgrep sleep
[igaber@localhost ~]$ ps -ef | grep sleep
igaber    3484    2827    0 13:16 pts/0    00:00:00 grep --color=auto sleep
[igaber@localhost ~]$
```

5. Issue the command sleep 100.

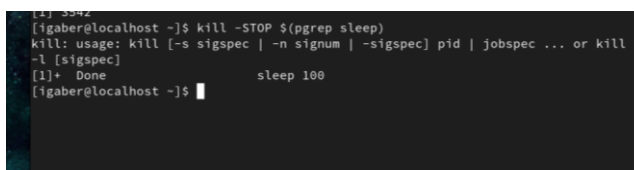
```
[igaber@localhost ~]$ sleep 100 &
```



```
igaber@localhost:~$ sleep 100 &
[igaber@localhost ~]$ sleep 100 &
[1] 3542
[igaber@localhost ~]$
```

6. Stop the last command.

```
[igaber@localhost ~]$ kill -STOP $(pgrep sleep)
```



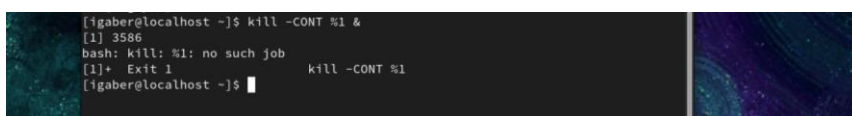
```
[1] 3542
[igaber@localhost ~]$ kill -STOP $(pgrep sleep)
kill: usage: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill
-l [sigspec]
[1]+  Done                  sleep 100
[igaber@localhost ~]$
```

7. Resume the last command in the background

```
[igaber@localhost ~]$ kill -CONT %1 &
```

Or

```
bg
```



```
[igaber@localhost ~]$ kill -CONT %1 &
[1] 3586
bash: kill: %1: no such job
[1]+  Exit 1                  kill -CONT %1
[igaber@localhost ~]$
```

8. Issue the jobs command and see its output.

```
[igaber@localhost ~]$ sleep 60 &
```

```
[1] 3621
```

```
[igaber@localhost ~]$ jobs
```

```
[1]+  Running                sleep 60 &
```

```
[igaber@localhost ~]$ sleep 60 &
[1] 3621
[igaber@localhost ~]$ jobs
[1]+  Running                  sleep 60 &
[igaber@localhost ~]$
```

9. Send the sleep command to the foreground and send it again to the background.

```
[igaber@localhost ~]$ fg %1
bash: fg: job has terminated
[1]+  Done                    sleep 60
[igaber@localhost ~]$ bg %1
bash: bg: %1: no such job
or
fg
bg
```

```
[igaber@localhost ~]$ fg %1
bash: fg: job has terminated
[1]+  Done                    sleep 60
[igaber@localhost ~]$ bg %1
bash: bg: %1: no such job
[igaber@localhost ~]$
```

10. Kill the sleep command.

```
[igaber@localhost ~]$ kill %1
Or
[igaber@localhost ~]$ sleep 1000&
[1] 3422
[igaber@localhost ~]$ kill %1
[1]+  Terminated            sleep 1000
```

```
[igaber@localhost ~]$ bg %1
bash: bg: %1: no such job
[igaber@localhost ~]$ kill %1
bash: kill: %1: no such job
[igaber@localhost ~]$
```

11. Display your processes only

```
[igaber@localhost ~]$ ps -u $USER
```

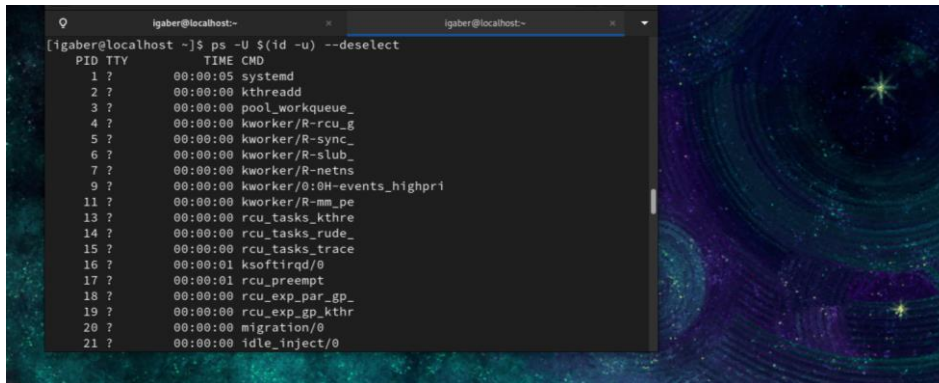
```
Nov 19 13:47
igaber@localhost:~
bash: kill: %1: no such job
[igaber@localhost ~]$ ps -u $USER
  PID TTY          TIME CMD
 1667 ?        00:00:01 systemd
 1669 ?        00:00:00 (sd-pam)
 1984 ?        00:00:00 gnome-keyring-d
 1989 tty2    00:00:00 gdm-wayland-ses
 1992 ?        00:00:00 dbus-broker-lau
 1996 ?        00:00:00 dbus-broker
 1999 tty2    00:00:00 gnome-session-b
 2030 ?        00:00:00 gnome-session-c
 2032 ?        00:00:00 gnome-session-b
 2051 ?        00:01:12 gnome-shell
 2063 ?        00:00:00 gvfsd
 2068 ?        00:00:00 gvfsd-fuse
 2076 ?        00:00:00 at-spi-bus-laun
 2081 ?        00:00:00 dbus-broker-lau
 2082 ?        00:00:00 dbus-broker
 2101 ?        00:00:00 xdg-permission-
 2105 ?        00:00:00 gnome-shell-cal
```

12. Display all processes except yours

```
[igaber@localhost ~]$ ps -U $(id -u) --deselect
```

Or

```
[igaber@localhost ~]$ ps aux |pgrep -v igaber
```



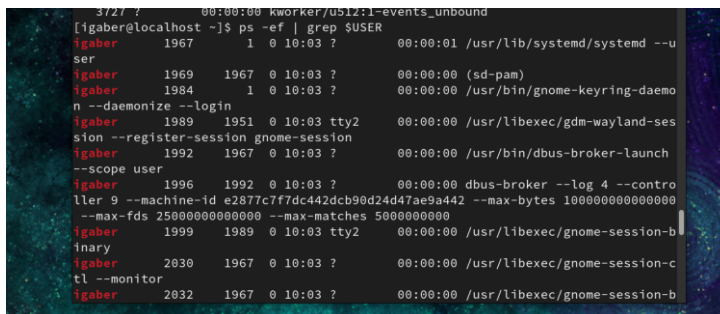
```
igaber@localhost:~$ ps -U $(id -u) --deselect
PID TTY          TIME CMD
  1 ?            00:00:05 systemd
  2 ?            00:00:00 kthreadd
  3 ?            00:00:00 pool_workqueue_
  4 ?            00:00:00 kworker/R-rcu_g
  5 ?            00:00:00 kworker/R-sync_
  6 ?            00:00:00 kworker/R-slub_
  7 ?            00:00:00 kworker/R-netns
  9 ?            00:00:00 kworker/0:0H-events_highpri
 11 ?            00:00:00 kworker/R-mm_pe
 13 ?            00:00:00 rcu_tasks_kthre
 14 ?            00:00:00 rcu_tasks_rude_
 15 ?            00:00:00 rcu_tasks_trace
 16 ?            00:00:01 ksoftirqd/0
 17 ?            00:00:01 rcu_preempt
 18 ?            00:00:00 rcu_exp_par_gp_
 19 ?            00:00:00 rcu_exp_gp_kthr
 20 ?            00:00:00 migration/0
 21 ?            00:00:00 idle_inject/0
```

13. Use the grep command to list your processes only

```
[igaber@localhost ~]$ ps -ef | grep $USER
```

Or

```
[igaber@localhost ~]$ pgrep -l -u igaber
```



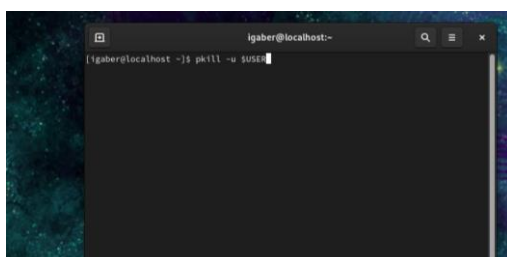
```
igaber@localhost:~$ ps -ef | grep $USER
igaber 1967      1  0 10:03 ?        00:00:01 /usr/lib/systemd/systemd --u
ser
igaber 1969    1967  0 10:03 ?        00:00:00 (sd-pam)
igaber 1984      1  0 10:03 ?        00:00:00 /usr/bin/gnome-keyring-daemo
n --daemonize --login
igaber 1989    1951  0 10:03 tty2      00:00:00 /usr/libexec/gdm-wayland-ses
sion --register-session gnome-session
igaber 1992    1967  0 10:03 ?        00:00:00 /usr/bin/dbus-broker-launch
--scope user
igaber 1996    1992  0 10:03 ?        00:00:00 dbus-broker --log 4 --contro
ller 9 --machine-id e2877c7f7dc442dcb90d24d47ae9a442 --max-bytes 1000000000000000
--max-fds 2500000000000000 --max-matches 5000000000000000
igaber 1999    1989  0 10:03 tty2      00:00:00 /usr/libexec/gnome-session-b
inary
igaber 2030    1967  0 10:03 ?        00:00:00 /usr/libexec/gnome-session-c
tl --monitor
igaber 2032    1967  0 10:03 ?        00:00:00 /usr/libexec/gnome-session-b
```

14. Kill your processes only.

```
[igaber@localhost ~]$ Pkill -u $USER
```

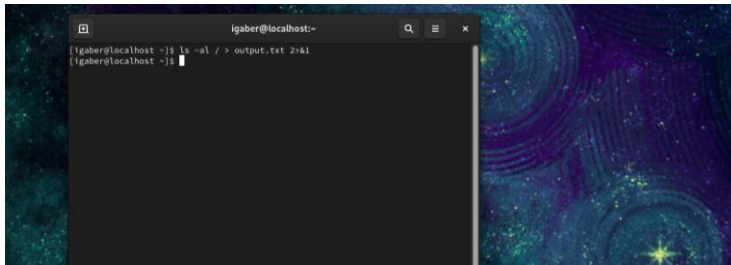
Or

```
[igaber@localhost ~]$ pkill -u igaber
```



```
igaber@localhost:~$ pkill -u $USER
```

15. Redirect the output and the error of the command: "ls -al/" to a file.  
[igaber@localhost ~]\$ ls -al / > output.txt 2>&1



16.

A) cat filename1 | cat filename2

READ SECOND FILE

B) ls | rm

ERROR

The ls command outputs a list of files, which is not a valid input for the rm command.

C) ls /etc/passwd | wc -l

1