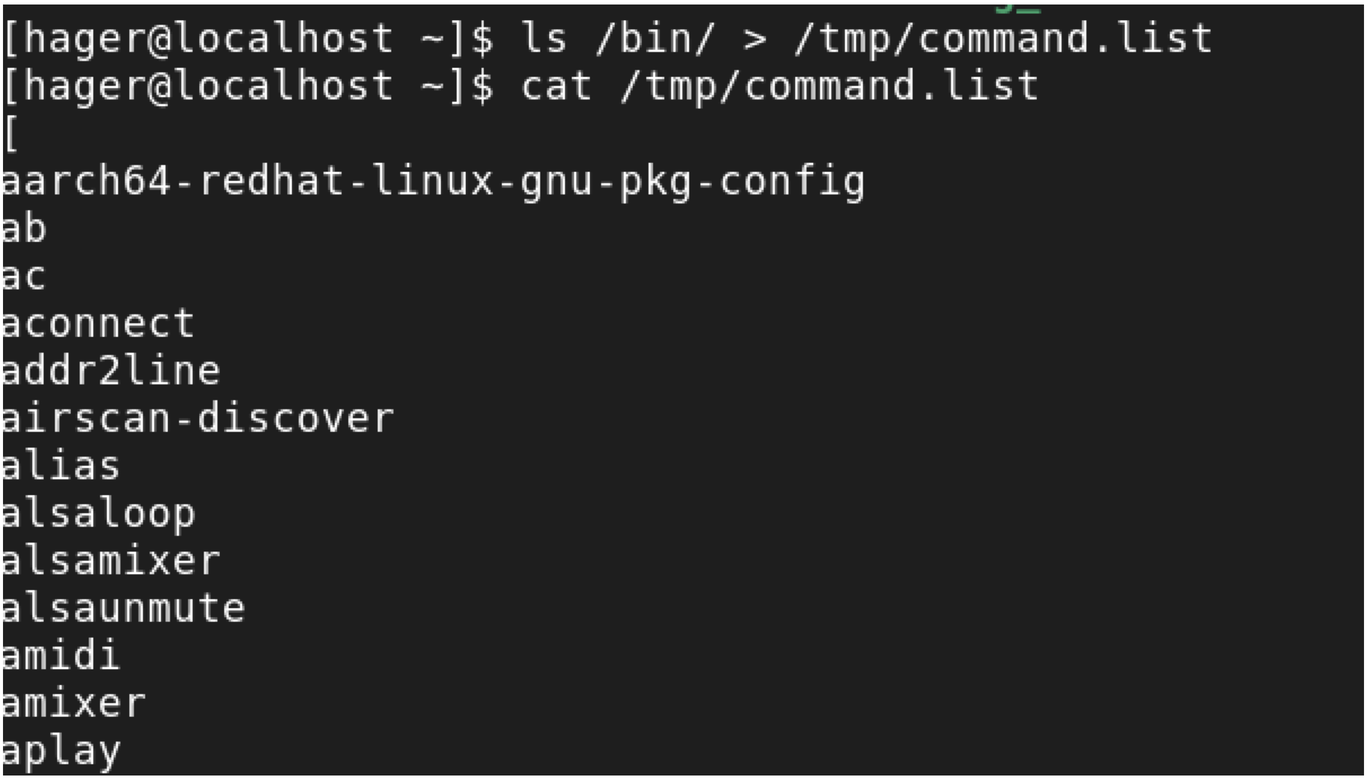
lab4

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

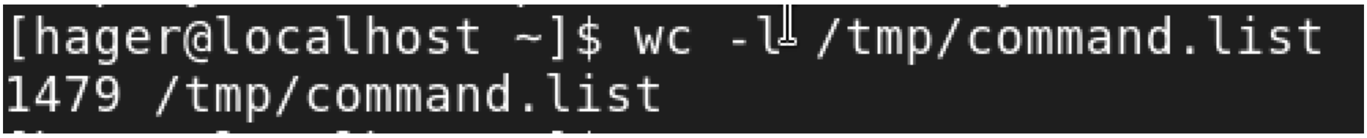
📸



```
[hager@localhost ~]$ ls /bin/ > /tmp/command.list
[hager@localhost ~]$ cat /tmp/command.list
[
aarch64-redhat-linux-gnu-pkg-config
ab
ac
aconnect
addr2line
airscan-discover
alias
alsaloop
alsamixer
alsaunmute
amidi
amixer
aplay
```

📌2-Count the number of user commands

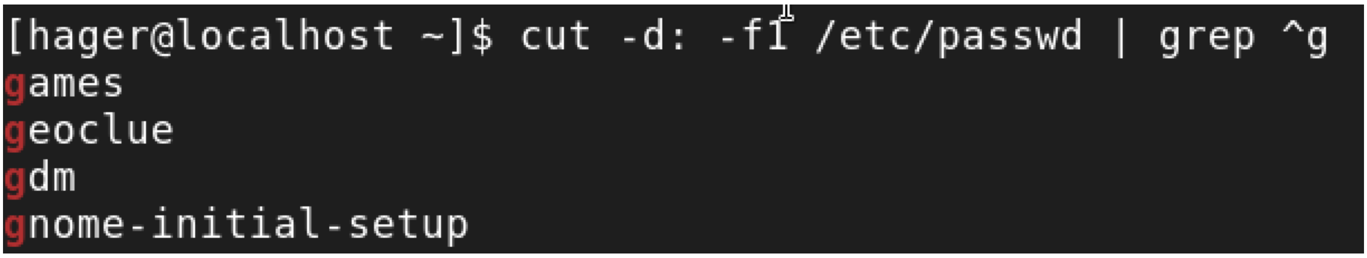
📸



```
[hager@localhost ~]$ wc -l /tmp/command.list
1479 /tmp/command.list
```

📌3-Get all the users names whose first character in their login is ‘g’.

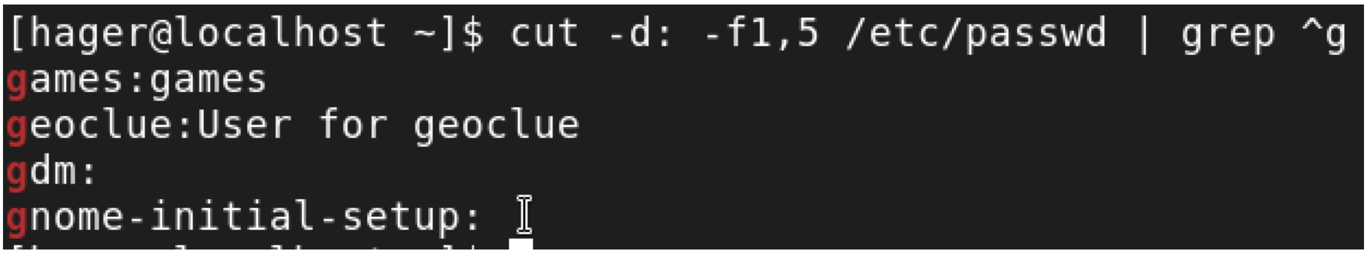
📸



```
[hager@localhost ~]$ cut -d: -f1 /etc/passwd | grep ^g
games
geoclue
gdm
gnome-initial-setup
```

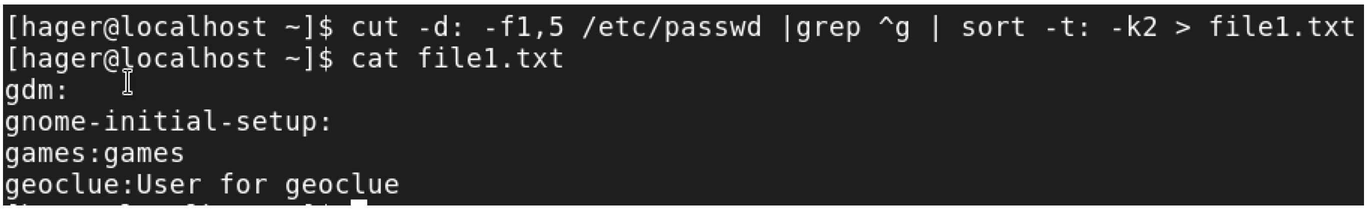
📌4-Get the logins name and full names (comment) of logins starts with “g”.

📸



```
[hager@localhost ~]$ cut -d: -f1,5 /etc/passwd | grep ^g
games:games
geoclue:User for geoclue
gdm:
gnome-initial-setup:
```

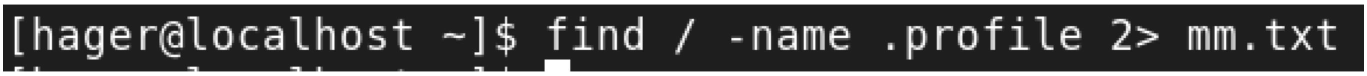
📌5-Save the output of the last command sorted by their full names in a file.



```
[hager@localhost ~]$ cut -d: -f1,5 /etc/passwd |grep ^g | sort -t: -k2 > file1.txt
[hager@localhost ~]$ cat file1.txt
gdm:
gnome-initial-setup:
games:games
geoclue:User for geoclue
```

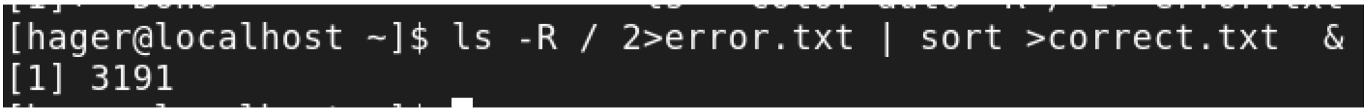
📌6-Write two commands: first: to search for all files on the system that named .bash_profile. Second: sorts the output of ls command on / recursively, Saving their output and error in 2 different files and sending them to the background.

first



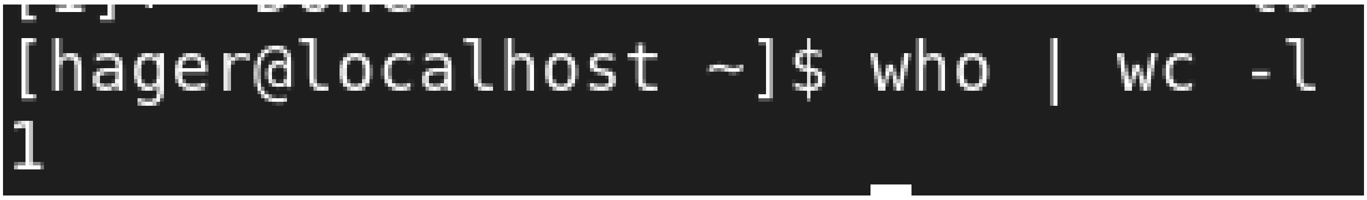
```
[hager@localhost ~]$ find / -name .profile 2> mm.txt
```

Second



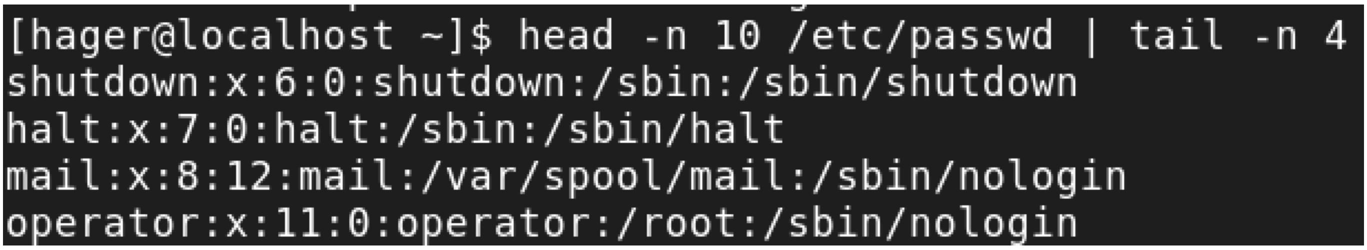
```
[hager@localhost ~]$ ls -R / 2>error.txt | sort >correct.txt  &
[1] 3191
```

📌7-Display the number of users who is logged now to the system.



```
[hager@localhost ~]$ who | wc -l
1
```

📌8-Display lines 7 to line 10 of /etc/passwd file



```
[hager@localhost ~]$ head -n 10 /etc/passwd | tail -n 4
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
```

📌9-What happens if you execute:

- cat filename1 | cat filename2 >> display only the second cat
- ls | rm >> waiting arrgument for rm
- ls /etc/passwd | wc –l >> list only the file /etc/passwd so it only count it

📌10-Issue the command sleep 100.

run in forground so cant use the terminal

📌11- Stop the last command.

control + z

📌12- Resume the last command in the background

bg

📌13-Issue the jobs command and see its output.

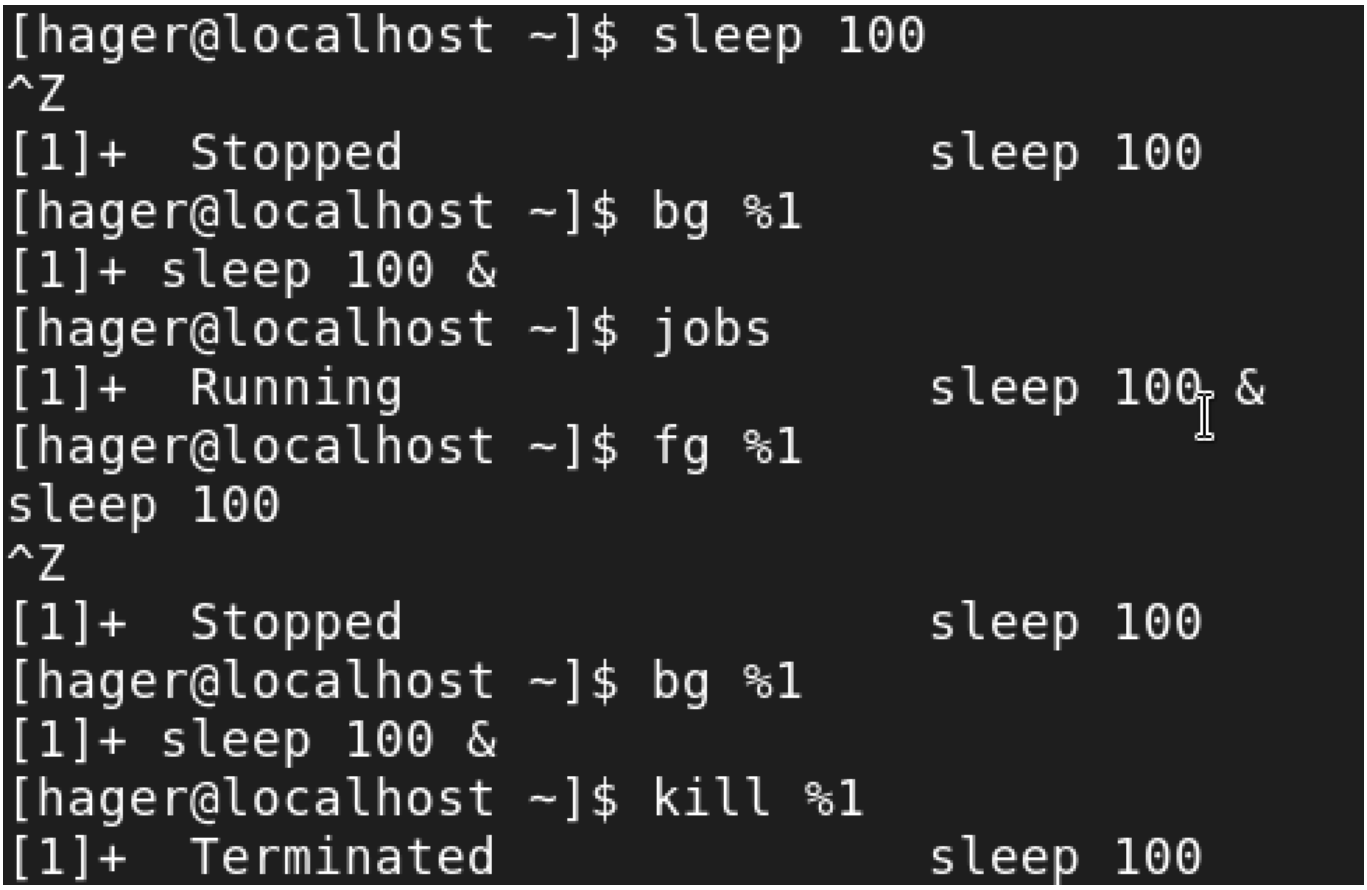
it will show the it running

📌14- Send the sleep command to the foreground and send it again to the background.

bg %1 fg%1

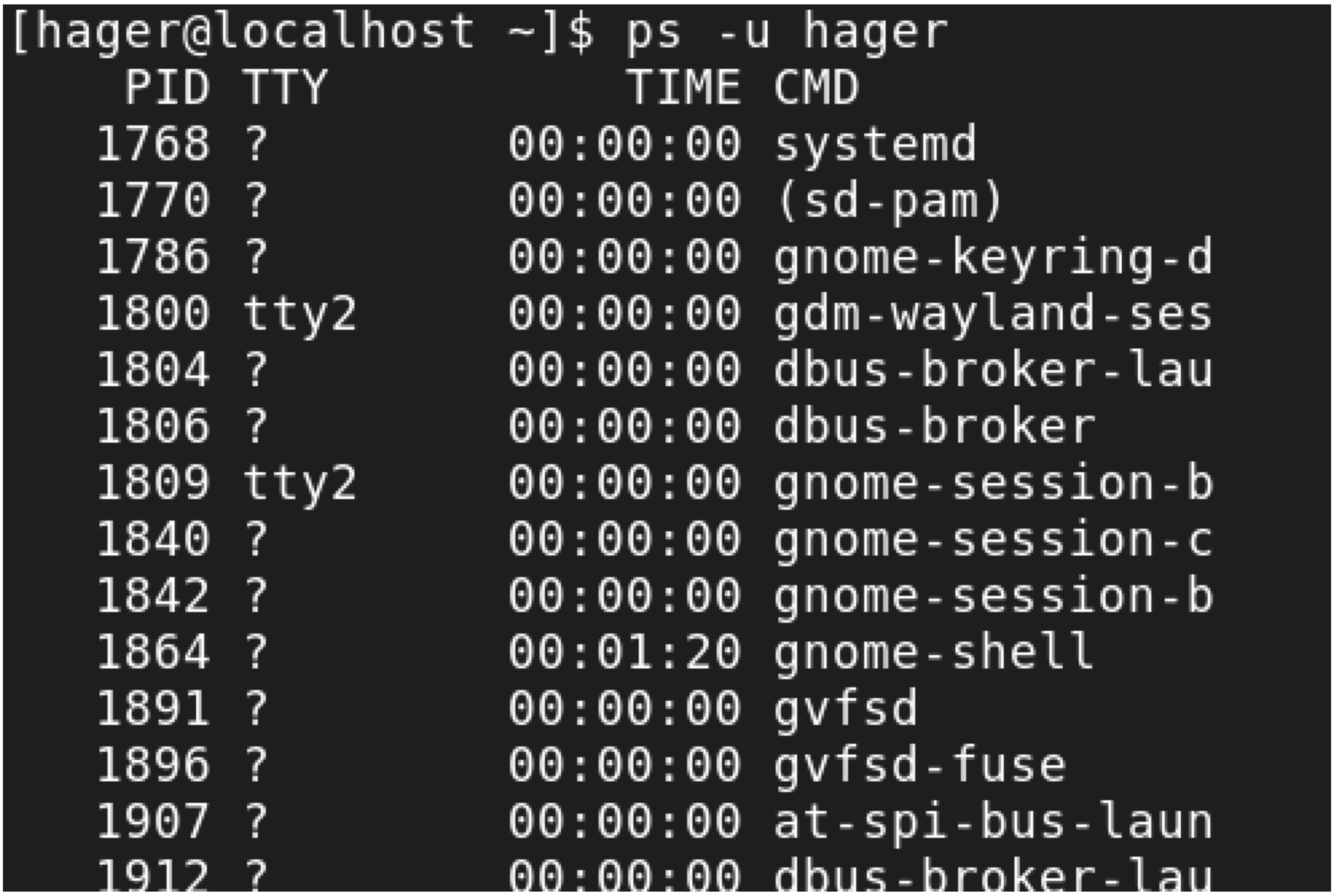
📌15-Kill the sleep command.

kill %1



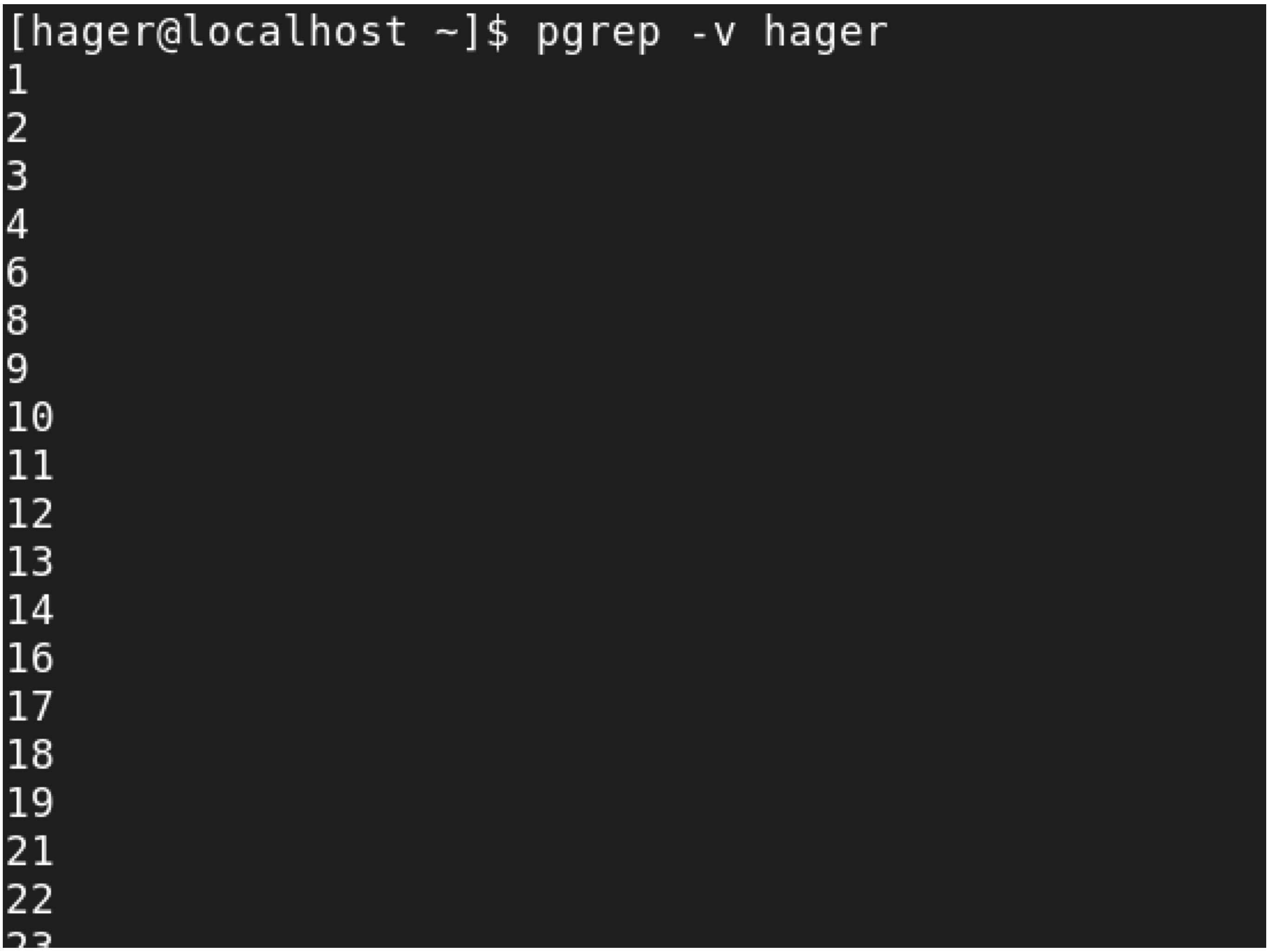
```
[hager@localhost ~]$ sleep 100
^Z
[1]+  Stopped                 sleep 100
[hager@localhost ~]$ bg %1
[1]+ sleep 100 &
[hager@localhost ~]$ jobs
[1]+  Running                 sleep 100 &
[hager@localhost ~]$ fg %1
sleep 100
^Z
[1]+  Stopped                 sleep 100
[hager@localhost ~]$ bg %1
[1]+ sleep 100 &
[hager@localhost ~]$ kill %1
[1]+  Terminated              sleep 100
```

📌16- Display your processes only



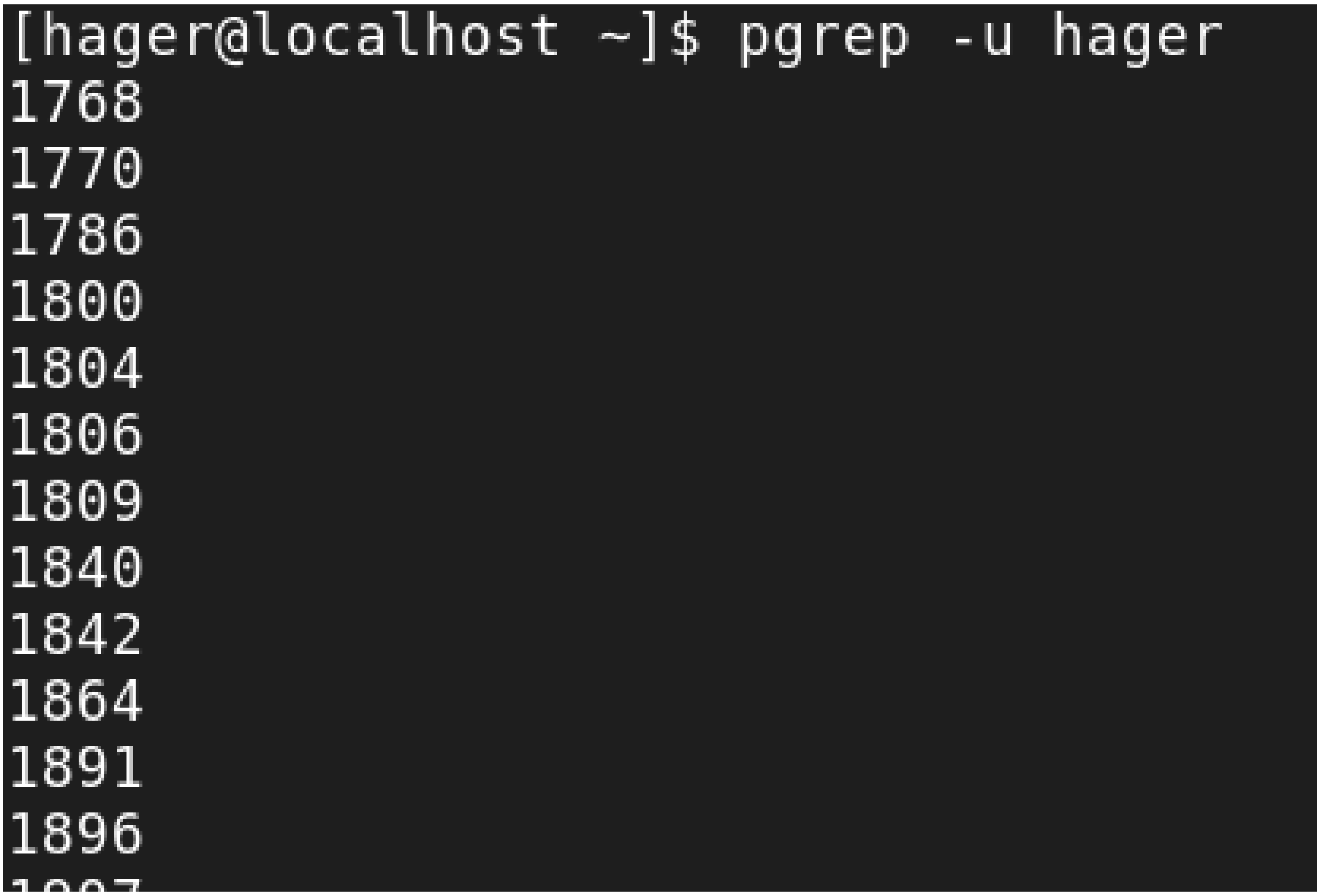
```
[hager@localhost ~]$ ps -u hager
    PID TTY          TIME CMD
   1768 ?        00:00:00 systemd
   1770 ?        00:00:00 (sd-pam)
   1786 ?        00:00:00 gnome-keyring-d
   1800 tty2     00:00:00 gdm-wayland-ses
   1804 ?        00:00:00 dbus-broker-lau
   1806 ?        00:00:00 dbus-broker
   1809 tty2     00:00:00 gnome-session-b
   1840 ?        00:00:00 gnome-session-c
   1842 ?        00:00:00 gnome-session-b
   1864 ?        00:01:20 gnome-shell
   1891 ?        00:00:00 gvfsd
   1896 ?        00:00:00 gvfsd-fuse
   1907 ?        00:00:00 at-spi-bus-laun
   1912 ?        00:00:00 dbus-broker-lau
```

📌17- Display all processes except yours



```
[hager@localhost ~]$ pgrep -v hager
1
2
3
4
6
8
9
10
11
12
13
14
16
17
18
19
21
22
23
```

📌18-Use the pgrep command to list your processes only



```
[hager@localhost ~]$ pgrep -u hager
1768
1770
1786
1800
1804
1806
1809
1840
1842
1864
1891
1896
```