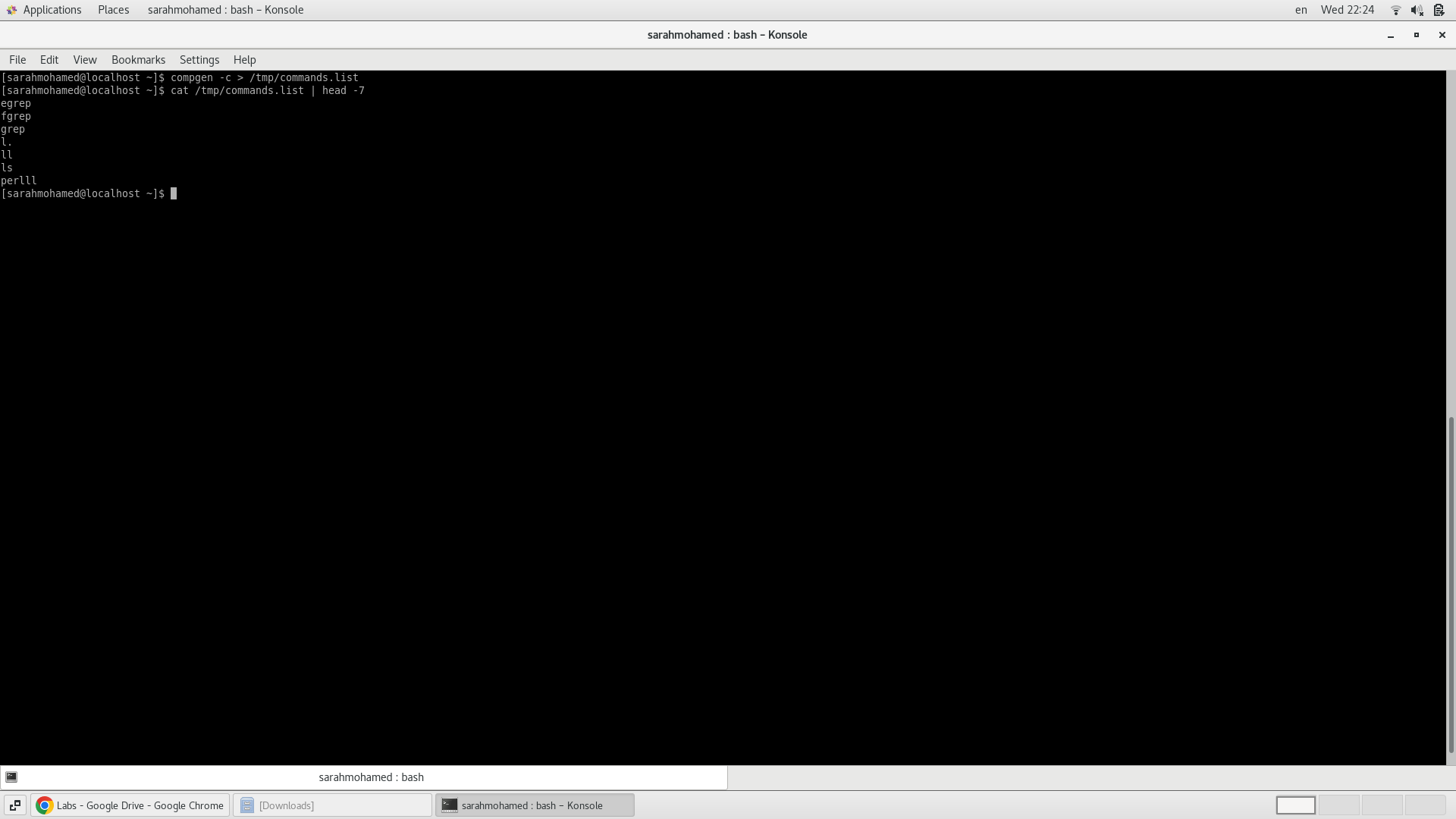
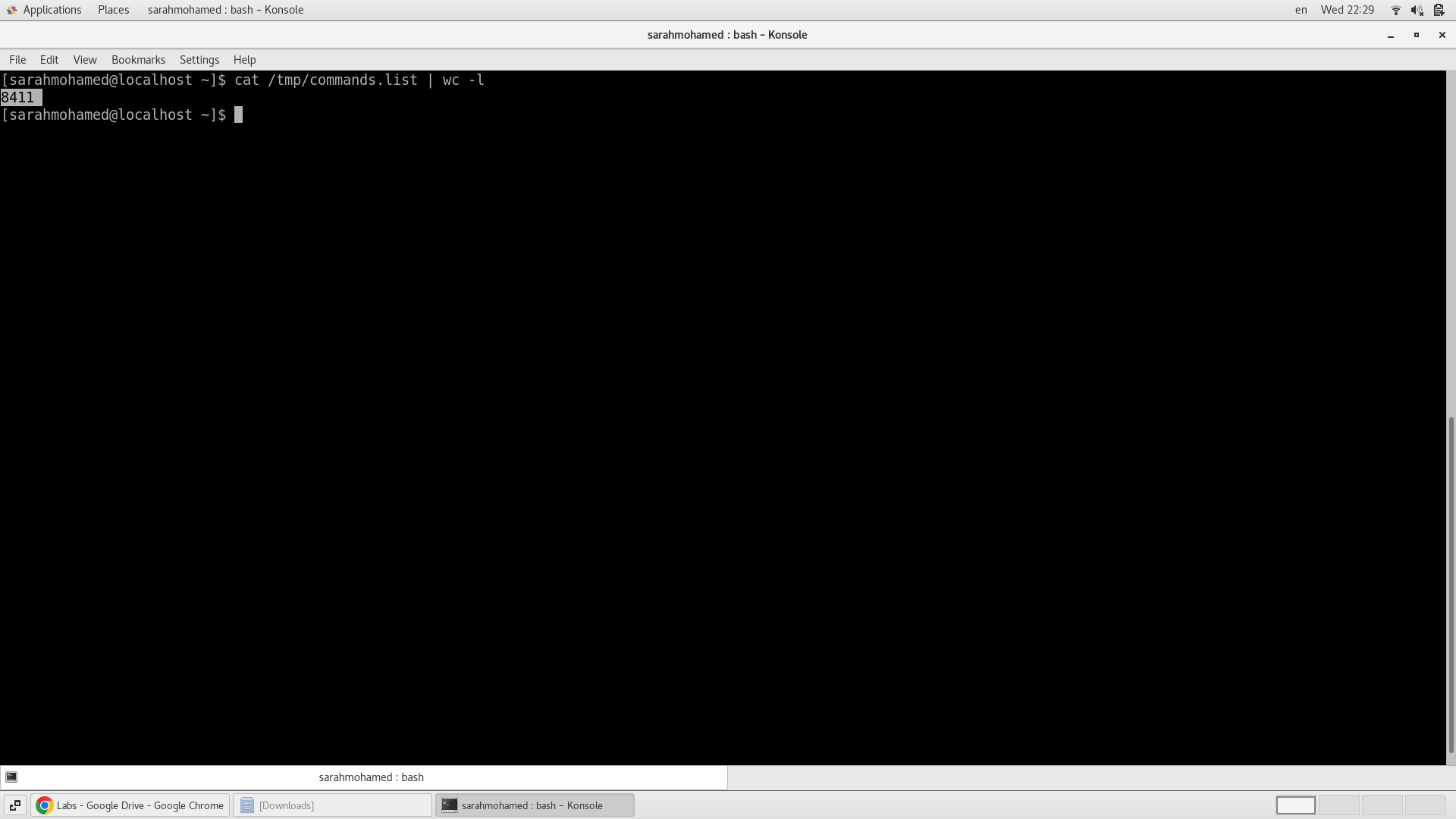
Lab 4

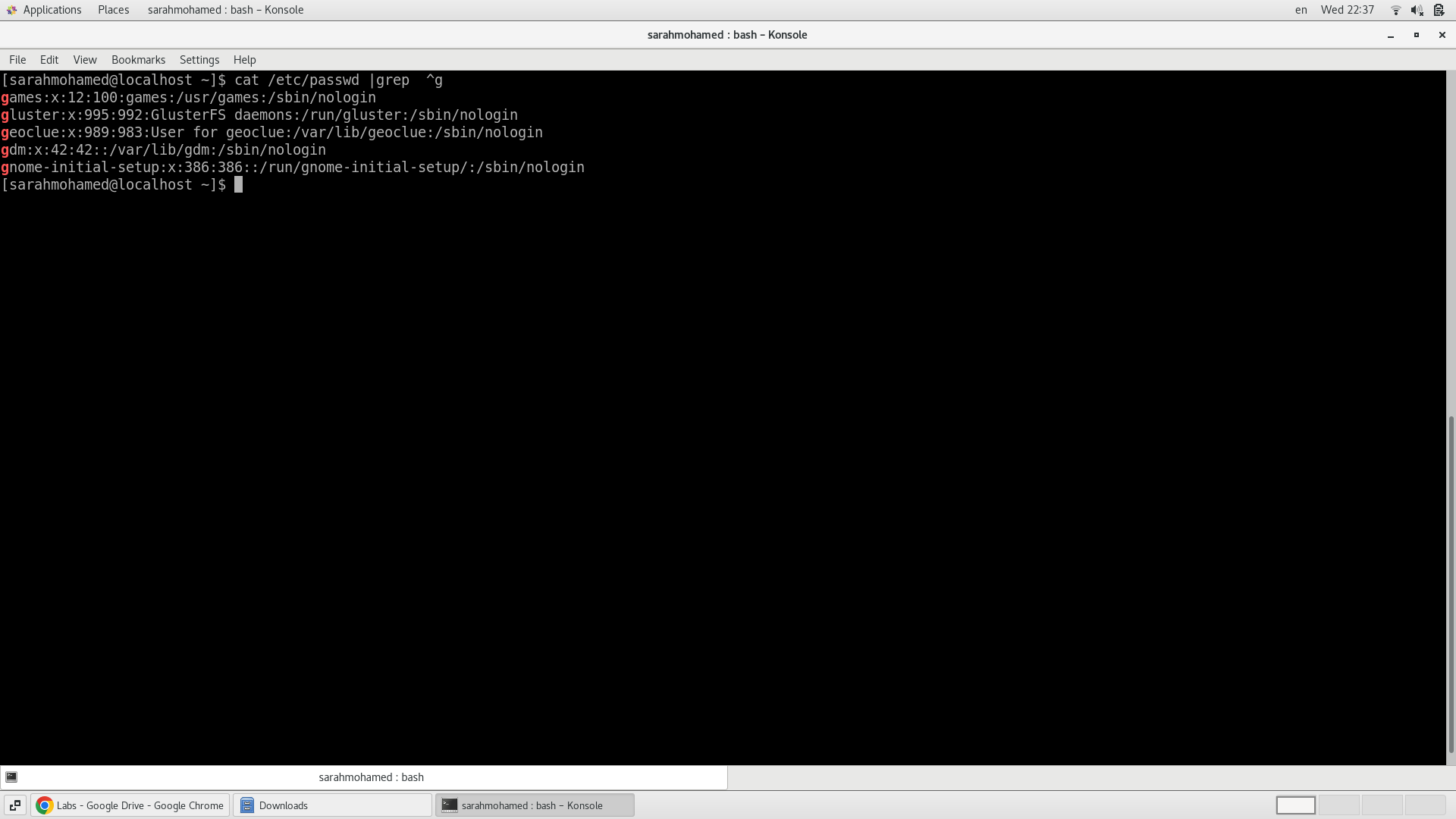
1. List the user commands and redirect the output to /tmp/commands.list



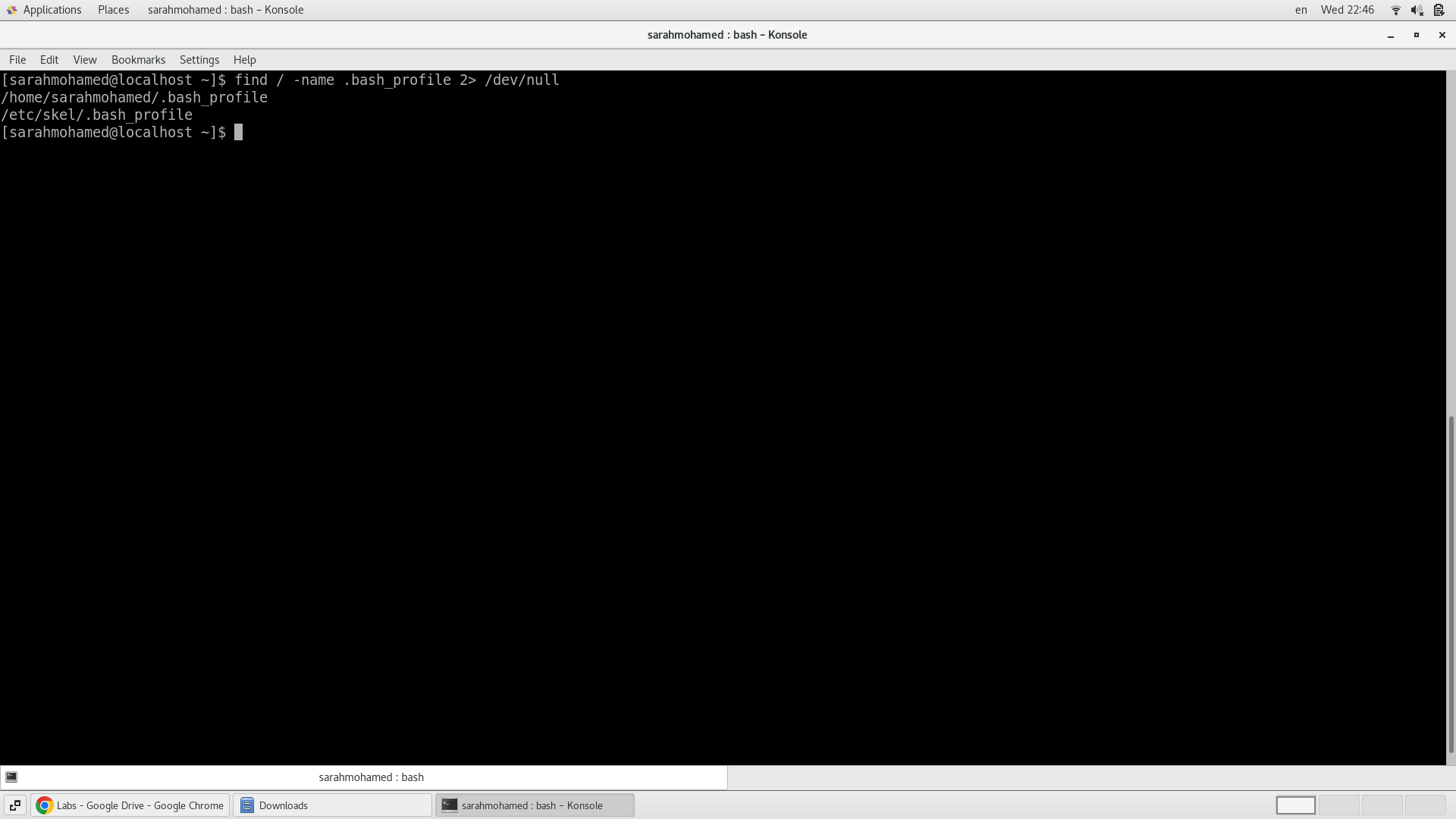
2. Count the number of user commands



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



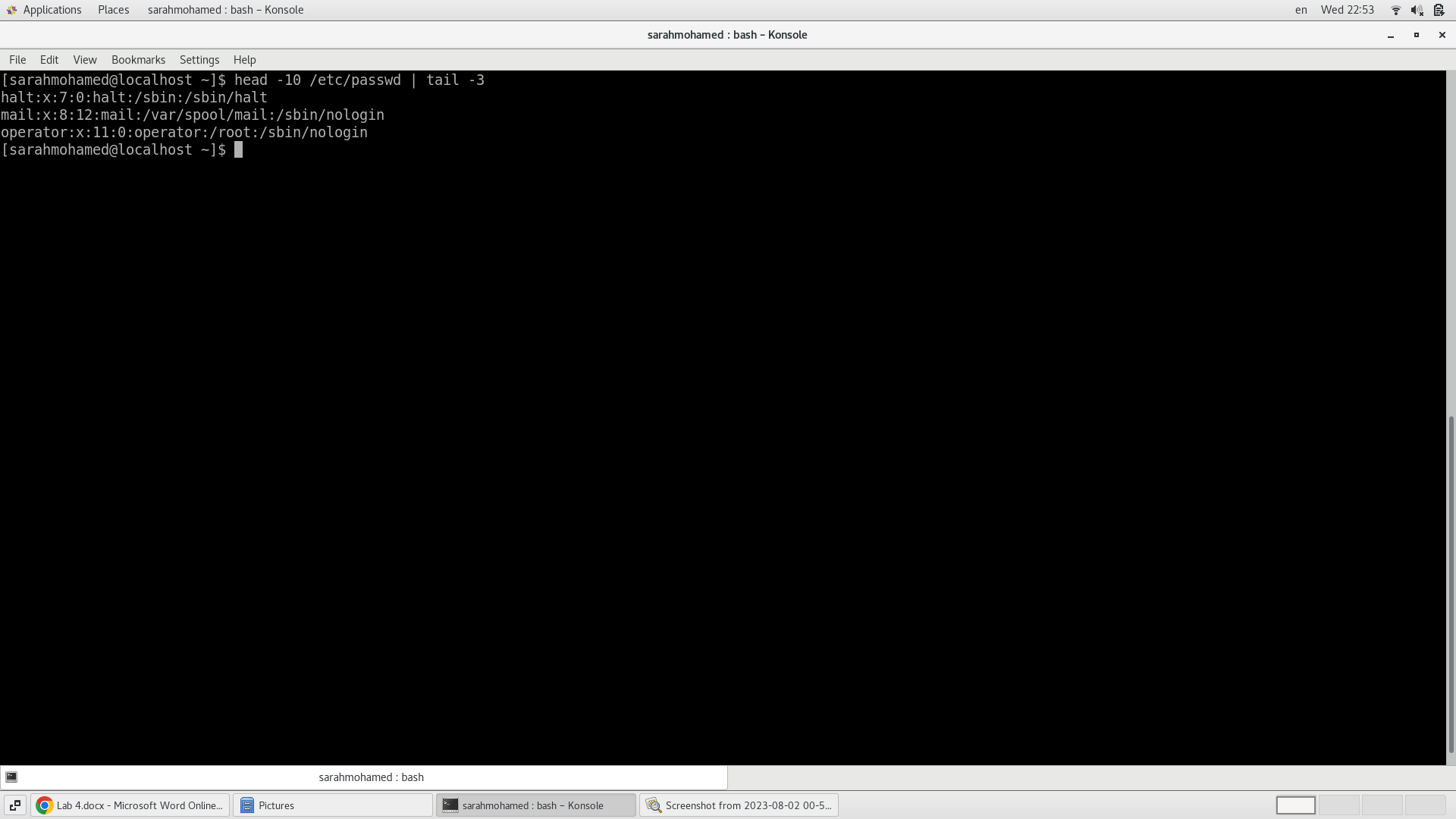
6. Write two commands: first: to search for all files on the system that named .bash\_profile.



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

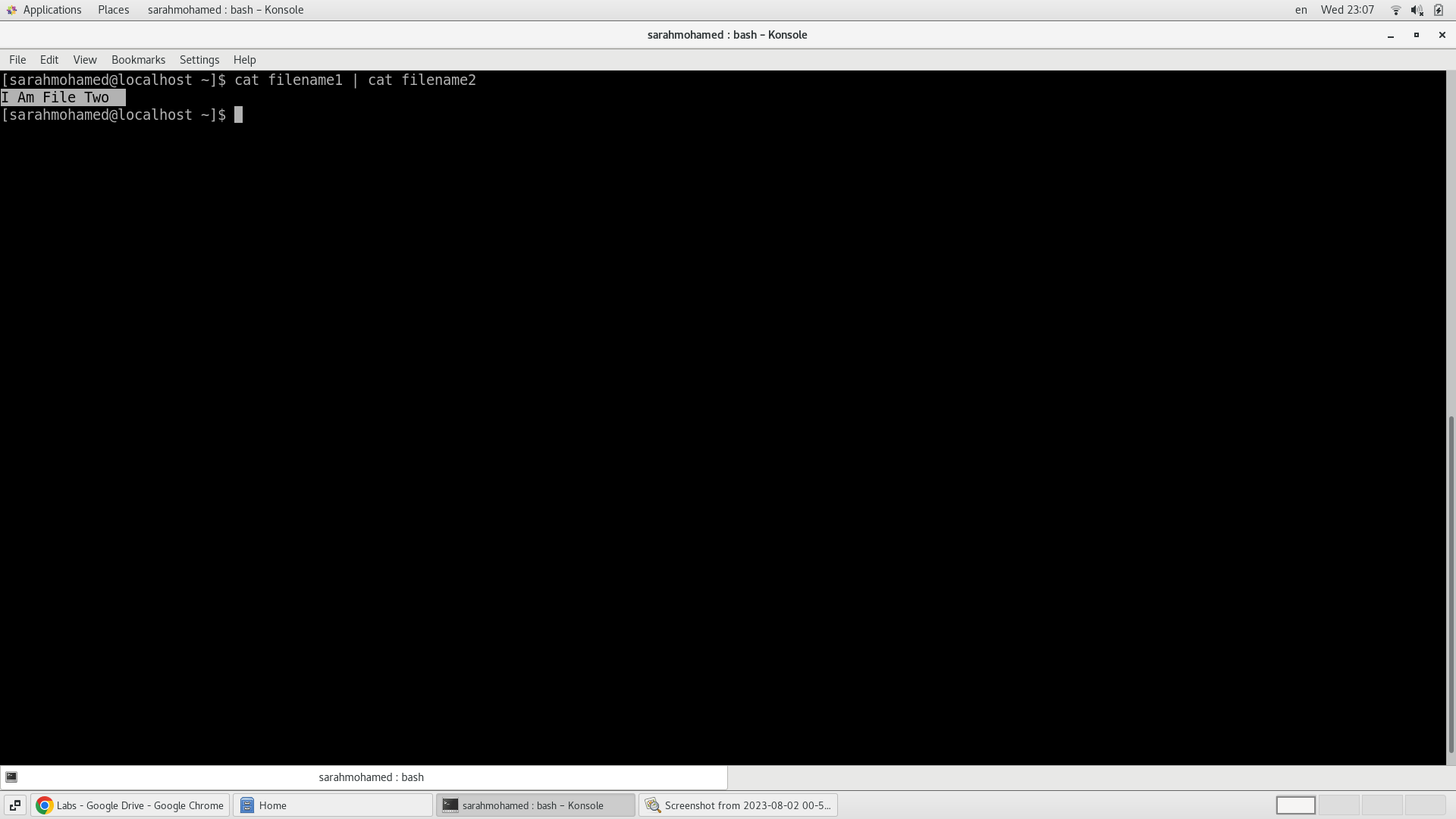


8. Display lines 7 to line 10 of /etc/passwd file



9. What happens if you execute:

* cat filename1 | cat filename2



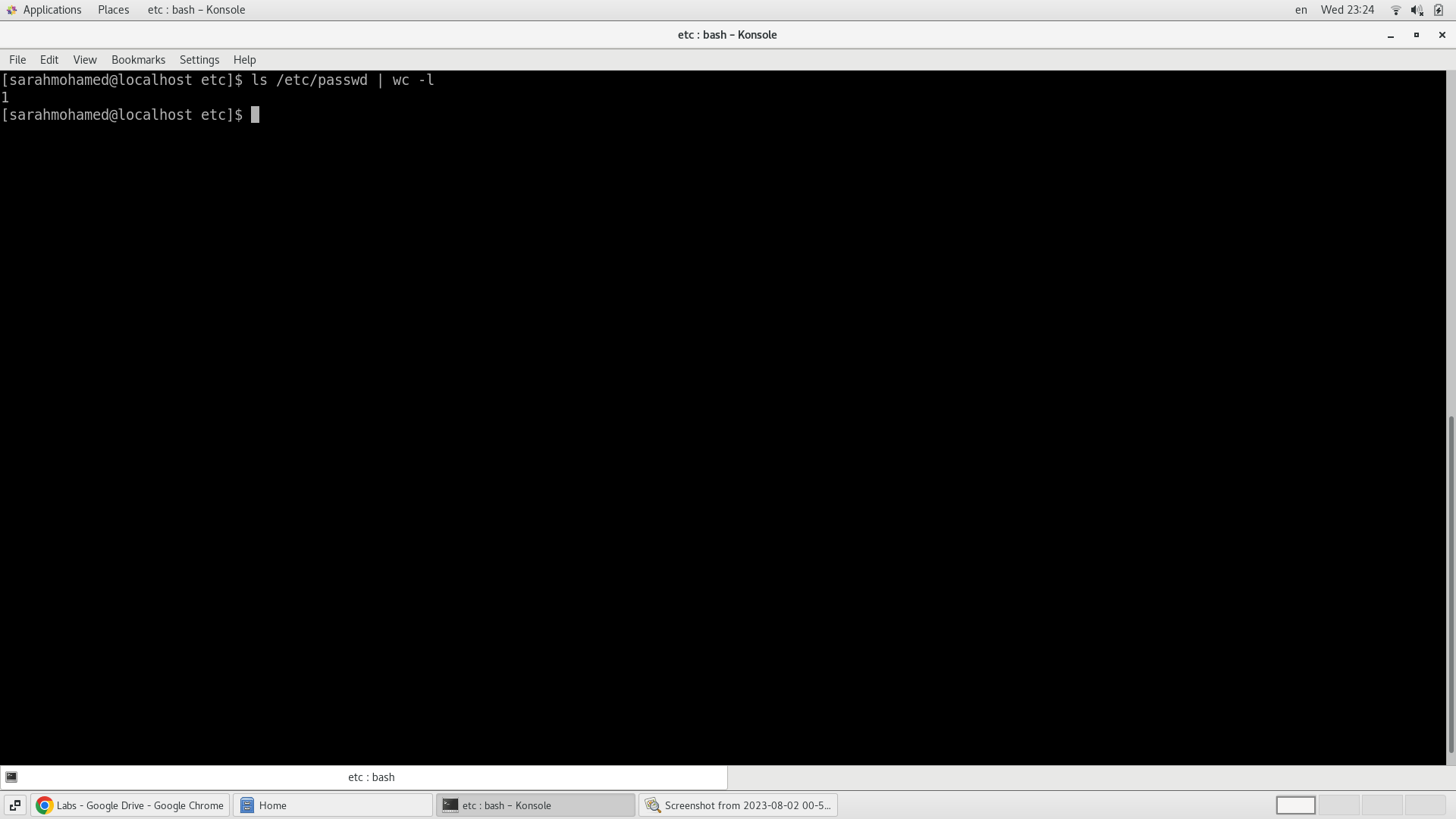
when you execute cat filename1 | cat filename2, the output of cat filename1 is passed through the pipe and becomes the input of cat filename2. However, since cat displays the contents of a file, the effect of this command is to display the contents of filename2 on the terminal, regardless of the contents of filename1.

* ls | rm



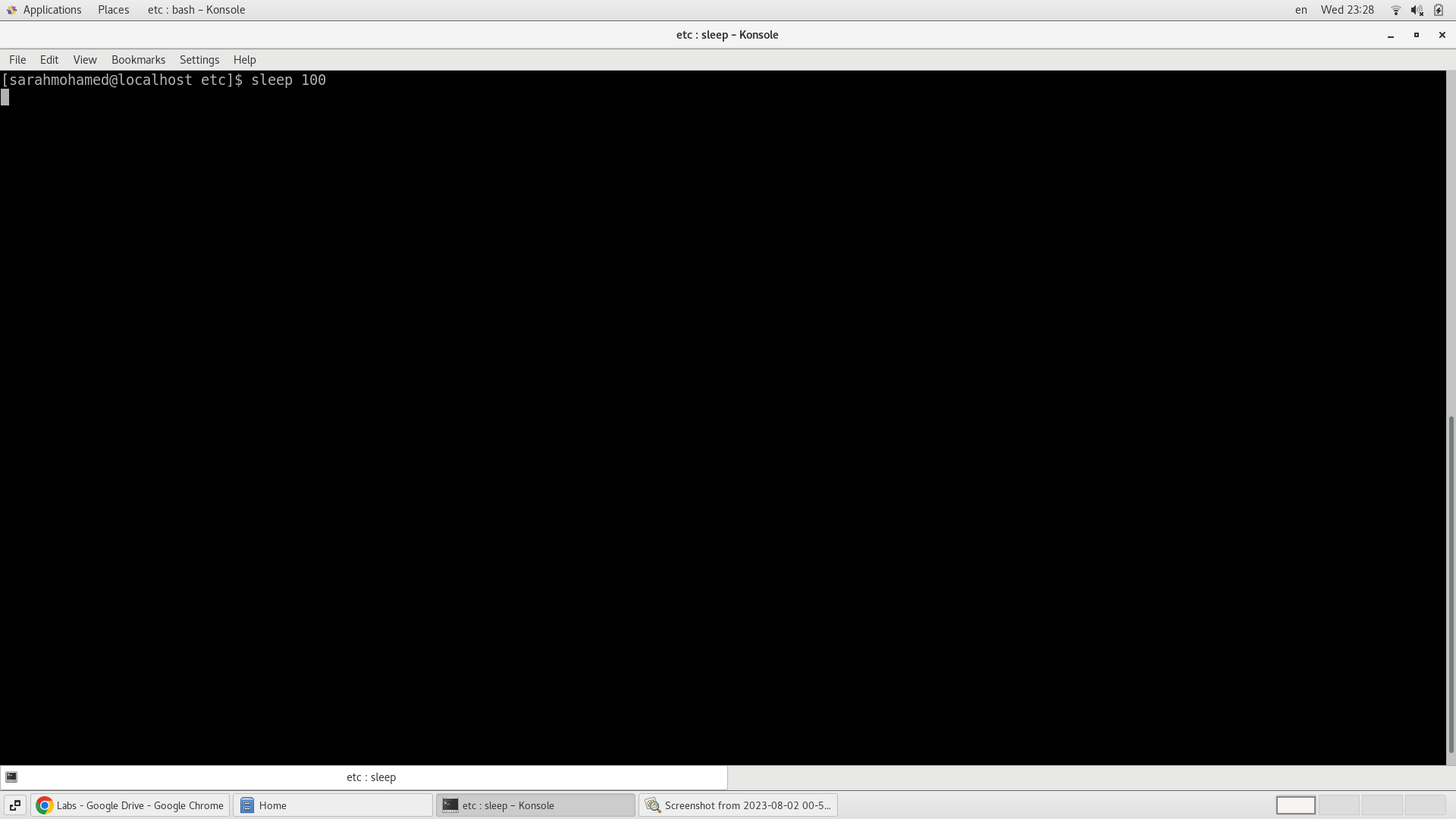
This command tries to pass the output of the ls command to the rm command. However, this will result in an error because the rm command expects file or directory names as arguments, not input from a pipe.

* ls /etc/passwd | wc –l

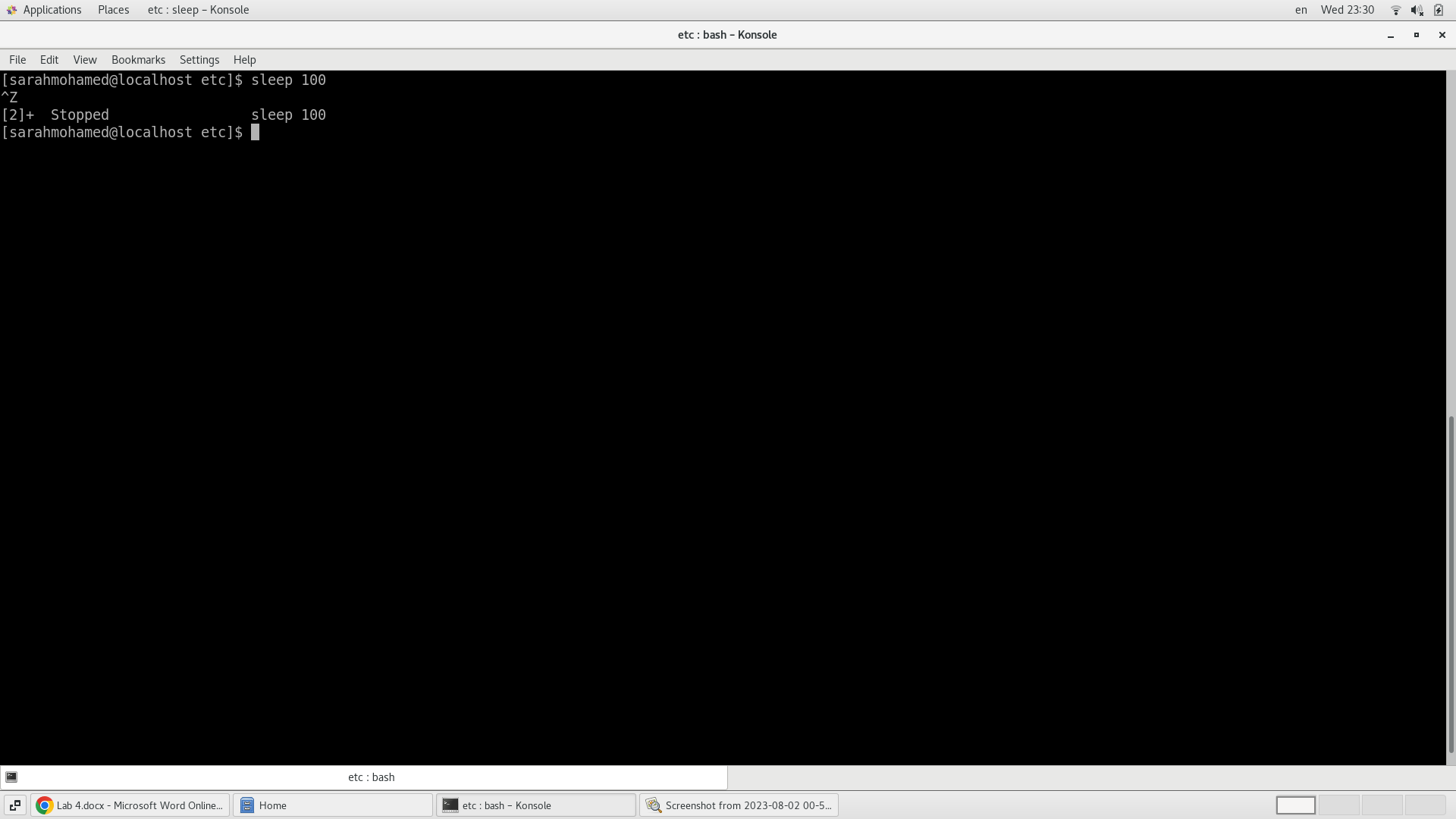


The Output of ls => /etc/passwd , wc –l => 1

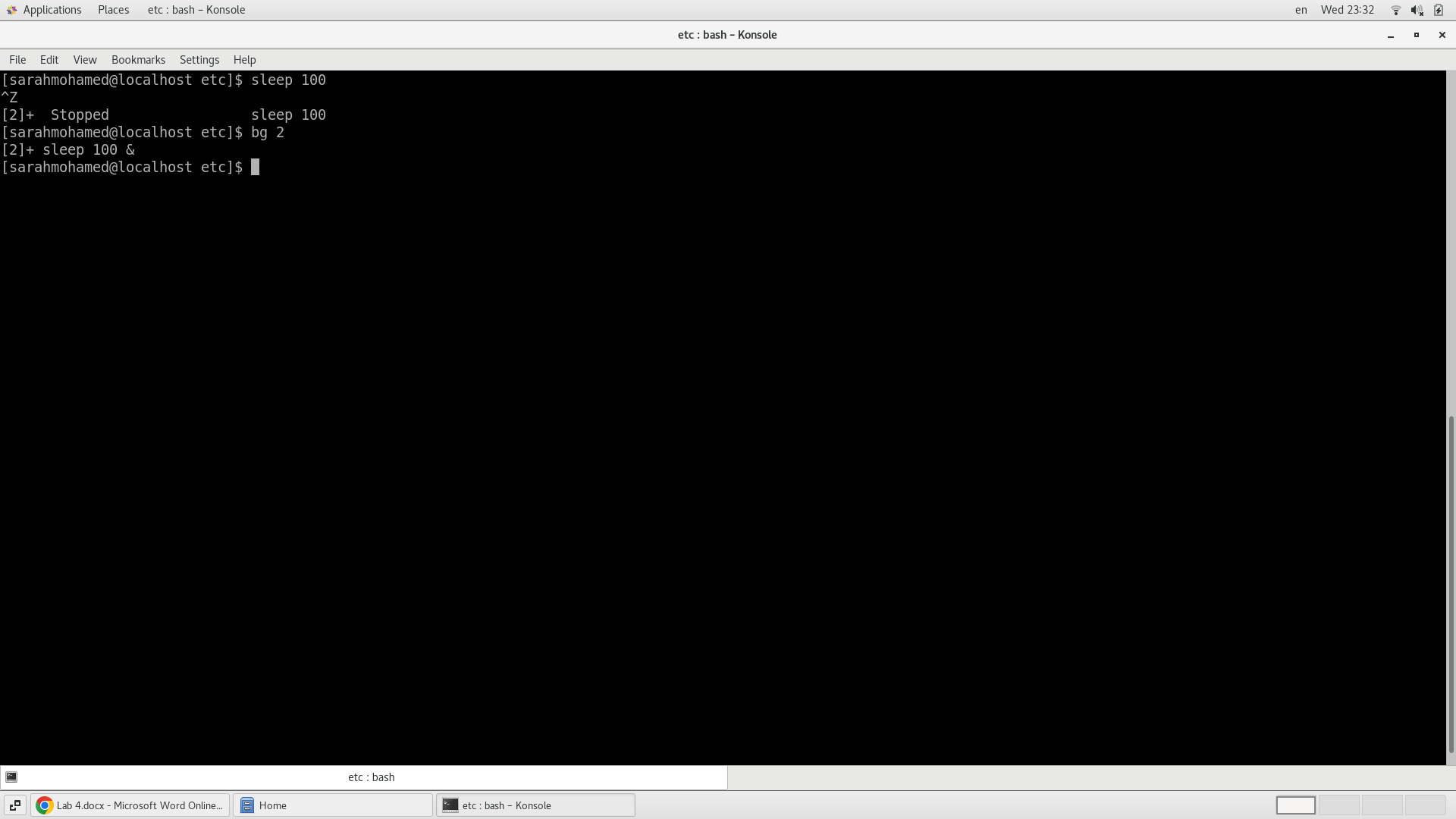
10.Issue the command sleep 100.



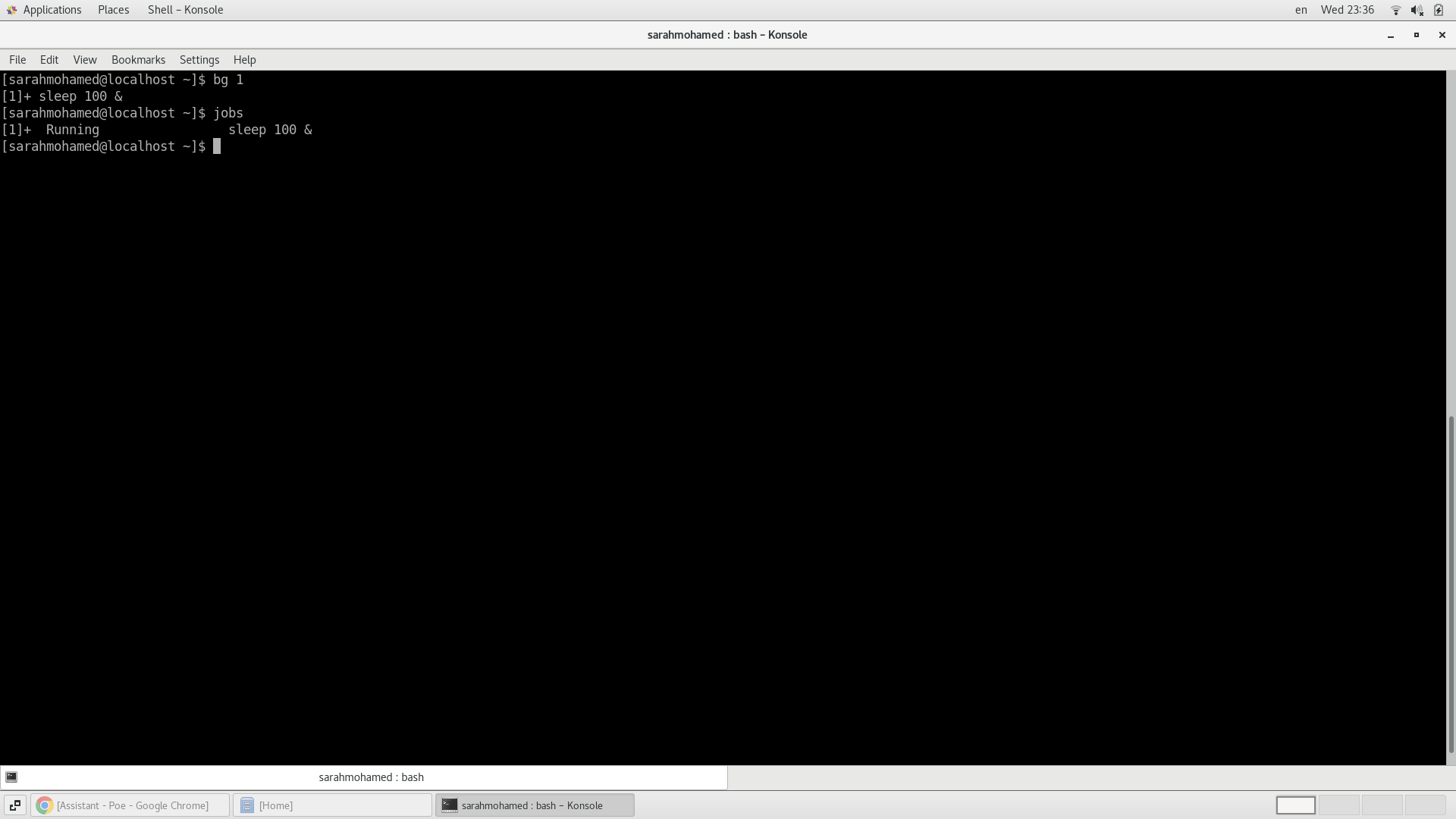
11.Stop the last command.



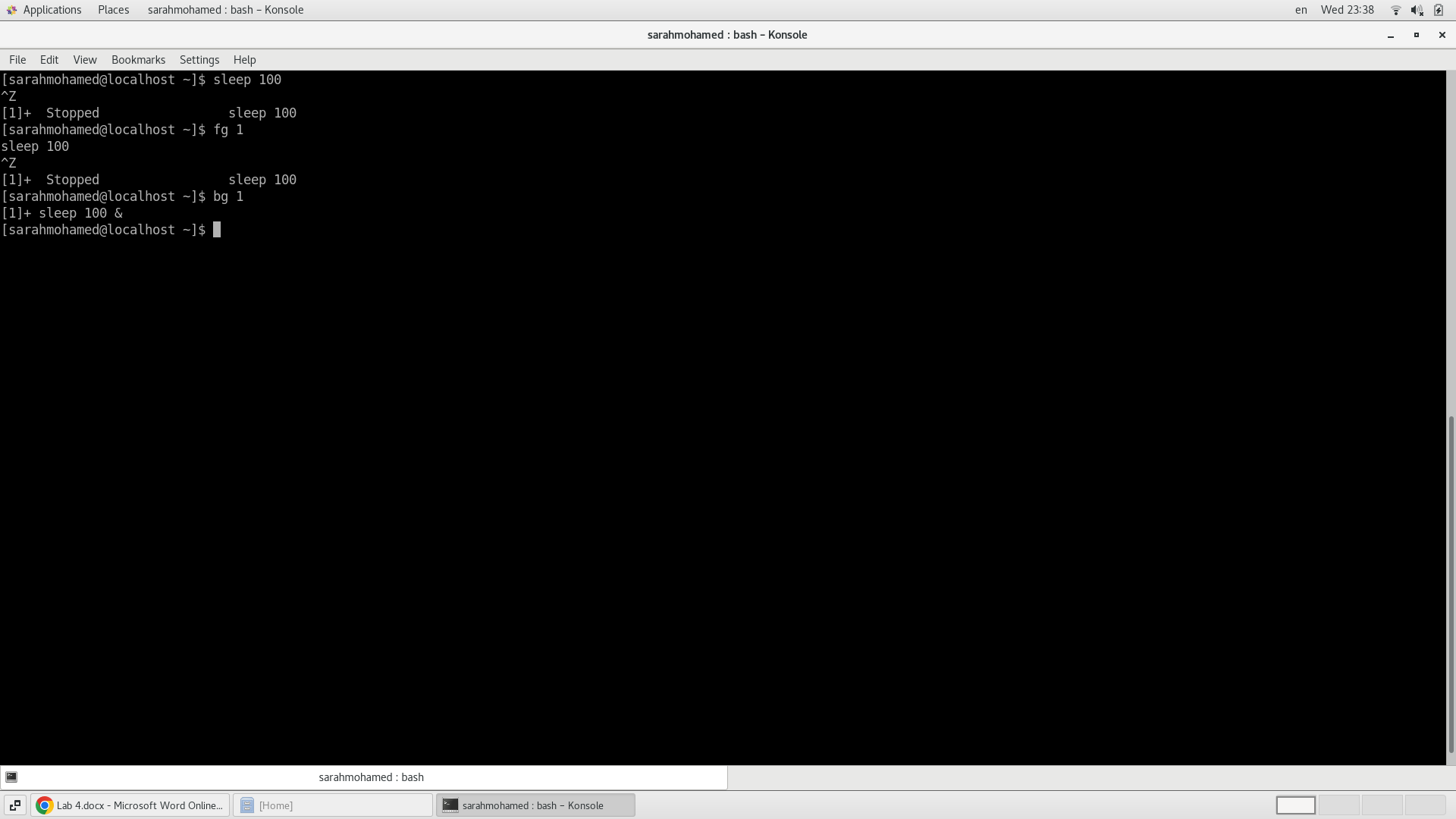
12.Resume the last command in the background



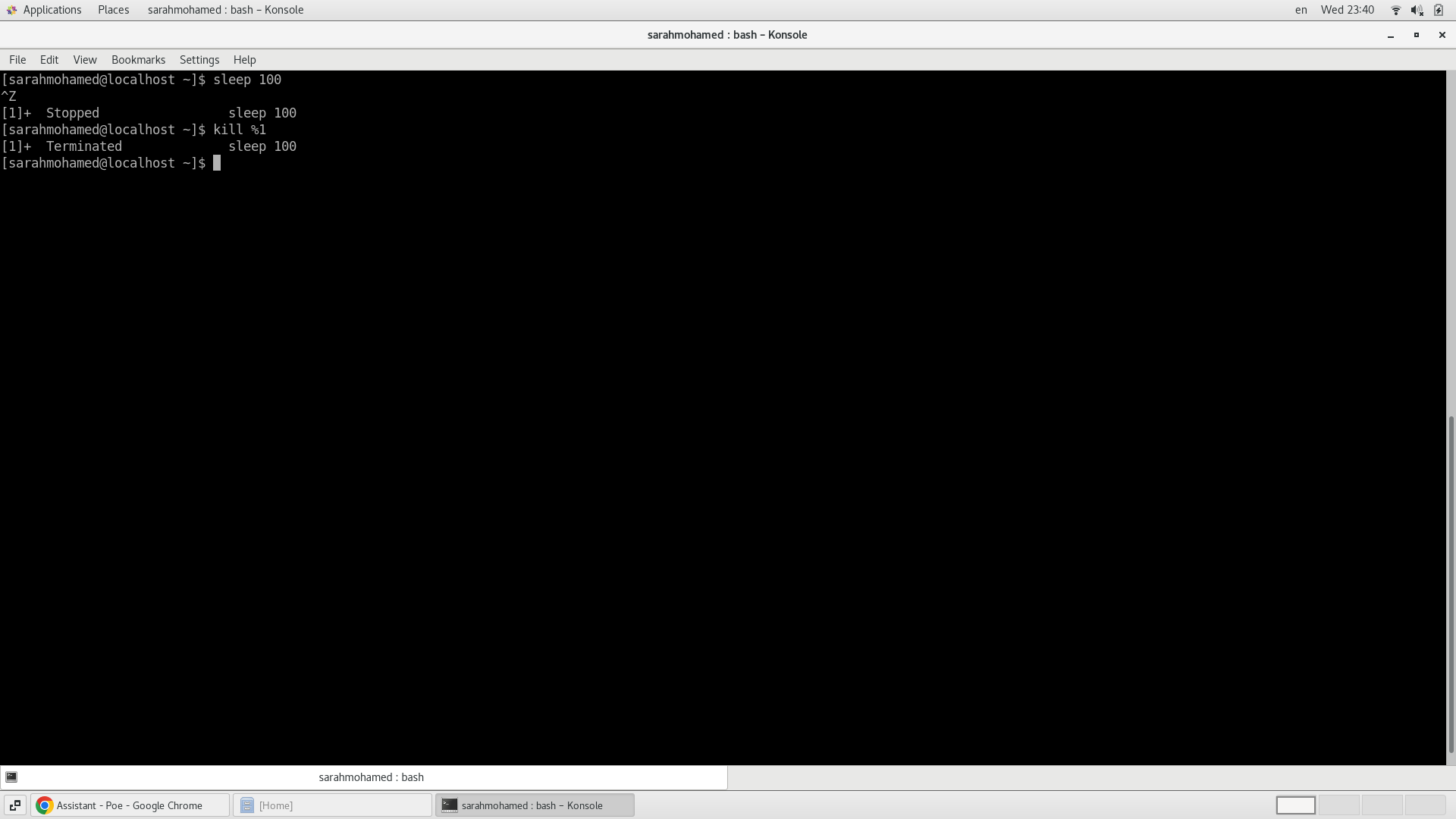
13.Issue the jobs command and see its output.



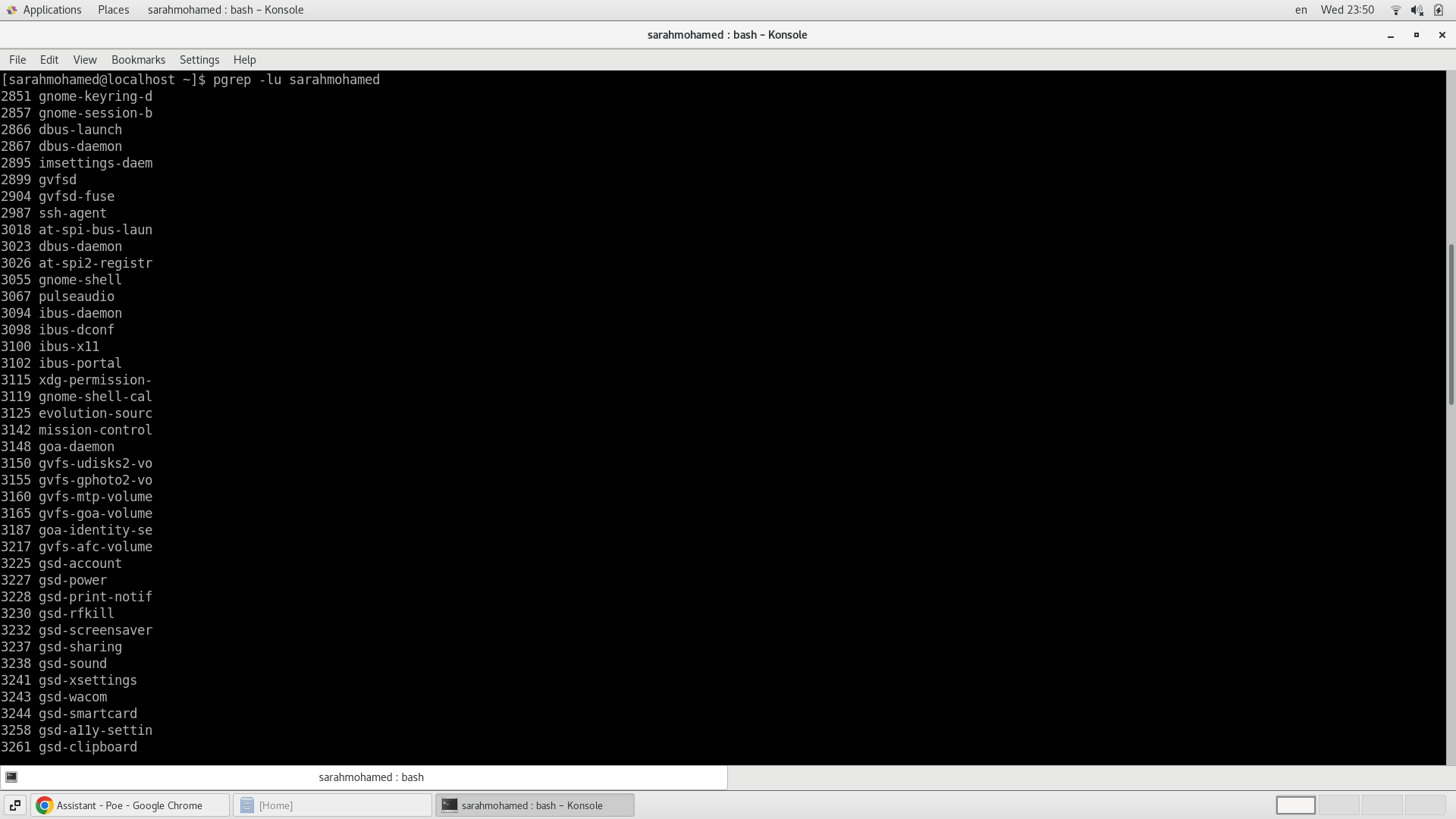
14.Send the sleep command to the foreground and send it again to the background.

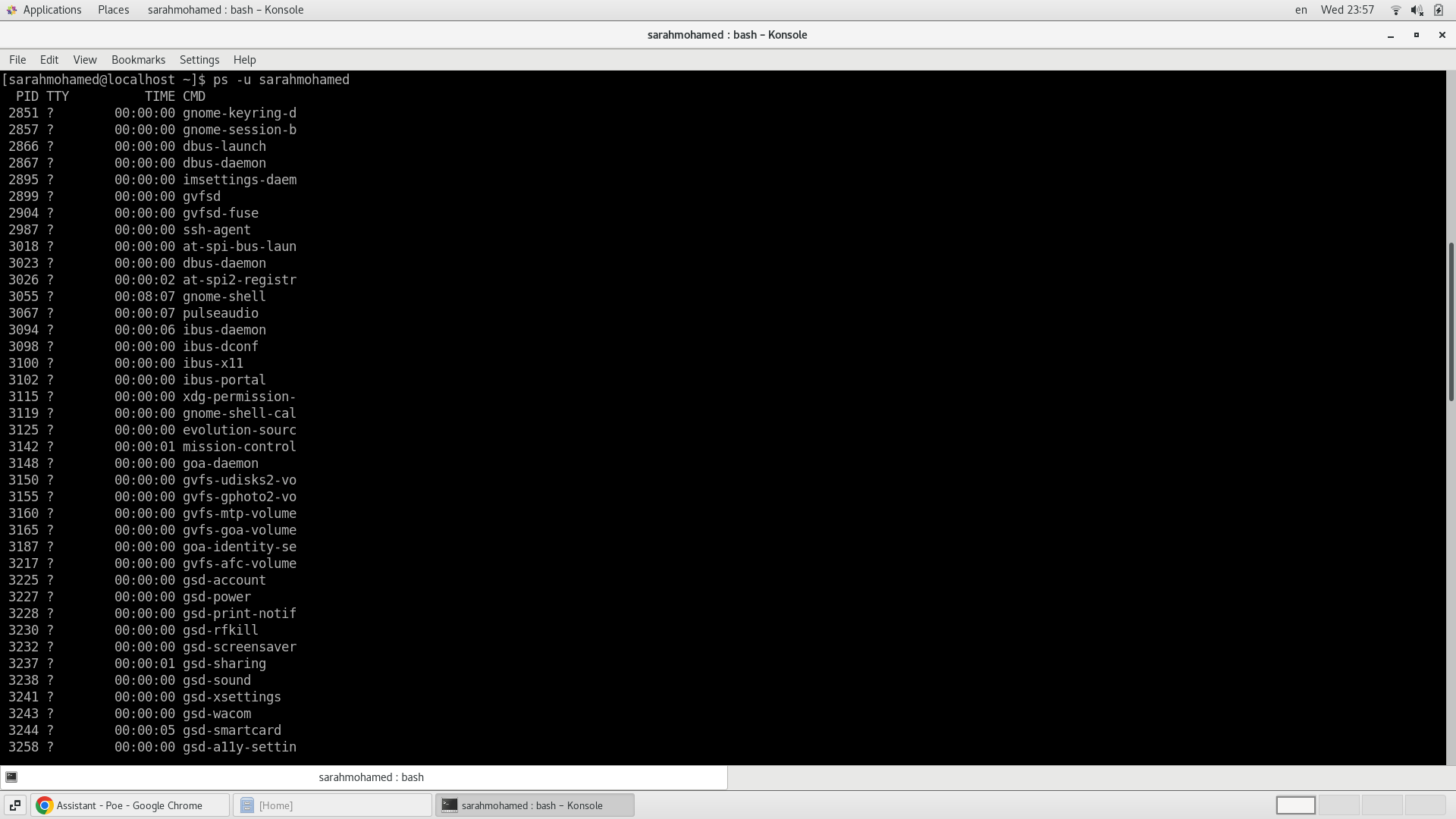


15.Kill the sleep command.

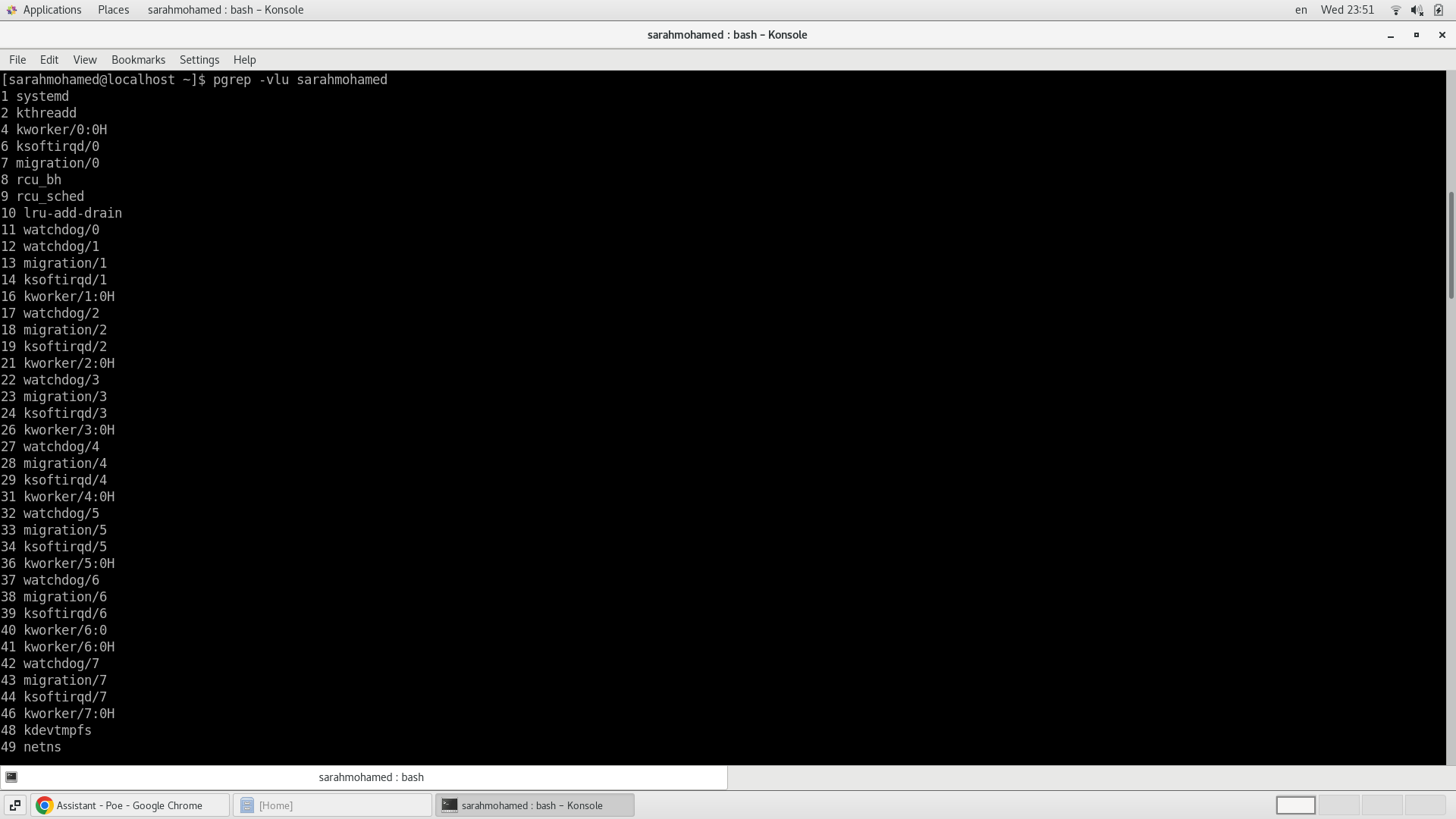


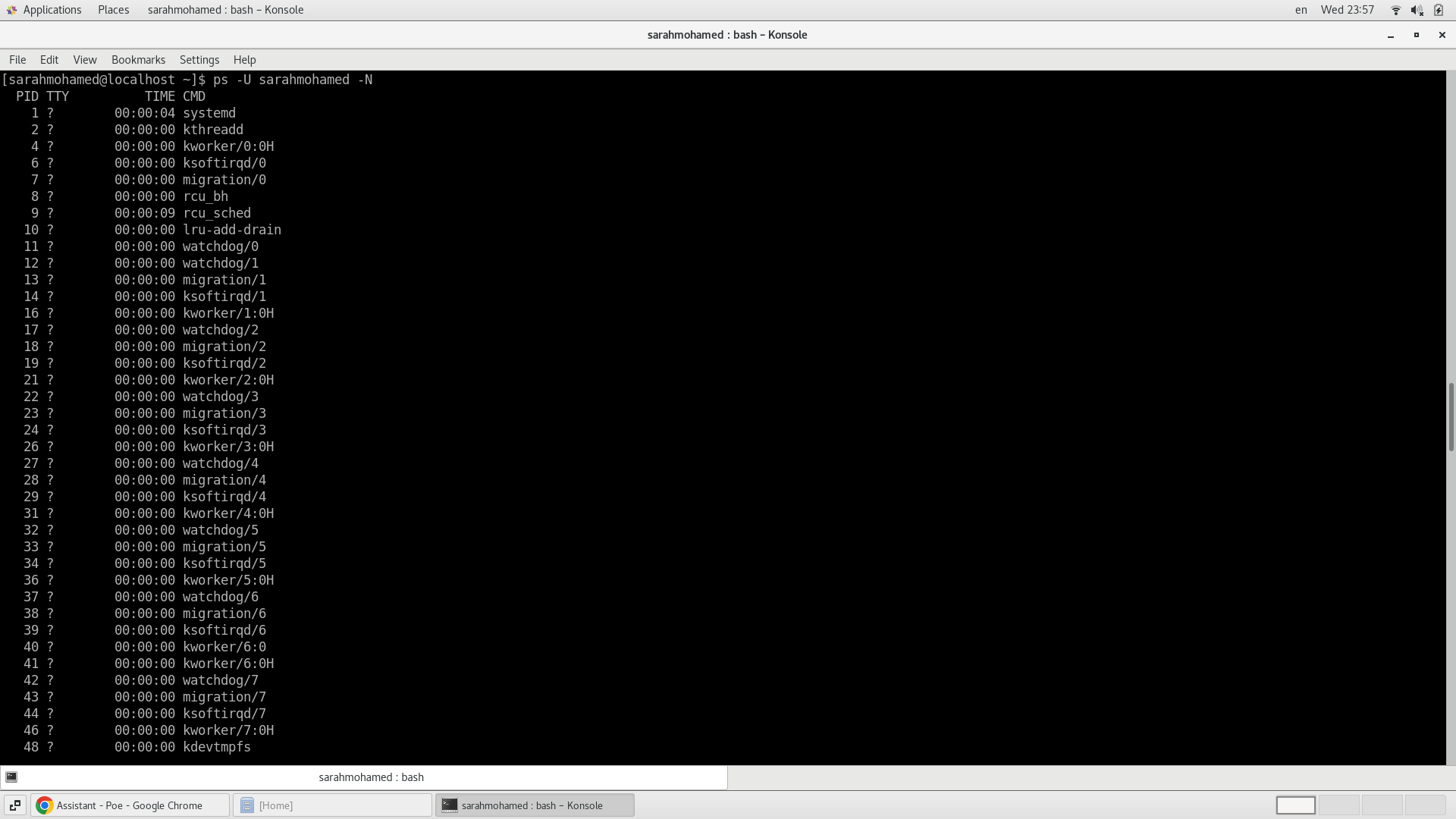
16.Display your processes only



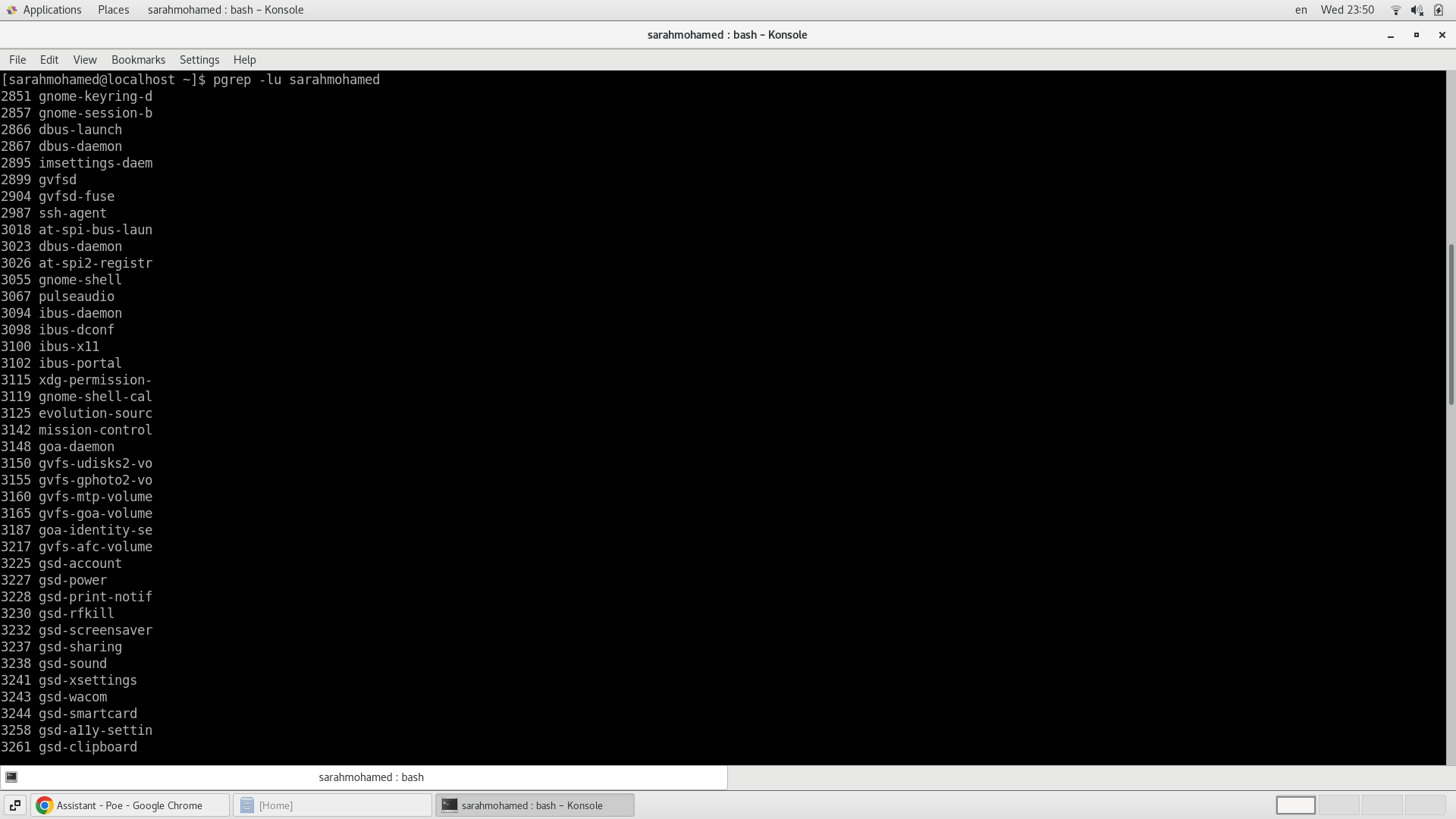


17.Display all processes except yours





18.Use the pgrep command to list your processes only



19.Kill your processes only.

