



**Module Code & Module Title**  
**Level 5 – Network Operating Systems**

**Assessment Type**  
**Logbook 8**  
**Semester**  
**2023/24 Autumn**

**Student Name: Aayush Wanem Limbu**

**London Met ID: 22072043**

**College ID: np05cp4a220010**

**Assignment Due Date: Wednesday, December 20, 2023**

**Assignment Submission Date: Wednesday, December 20, 2023**

**Submitted To: Enjina Ghimire**

**Word Count (Where Required):2417**

*I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.*

## Table of Contents

1. UNIX Workshop Log: Directories, Files, and Alias .....	1
1.1 Aims and Objectives .....	1
1.1.1 Aims.....	1
1.1.2 Objectives.....	1
1.2 Required Tools and Concepts .....	2
2. Questions with their answers .....	3
3. Conclusion: .....	19

## Table of Figures

Figure 1 Creating Directory Structure .....	3
Figure 2 Changing Directory Using Relative Path.....	3
Figure 3 Using cat command to create and write into a filea. ....	4
Figure 4 Using cat command to create and write into a fileb.....	4
Figure 5 grep command without options. ....	5
Figure 6 Using grep with -v option. ....	5
Figure 7 Using grep with -n option. ....	6
Figure 8 Using grep with -l option with * to check for all files. ....	6
Figure 9 Using grep with -i option to search for all files.....	7
Figure 10 Again using grep with -i option to search for uppercase text patterns. ..	7
Figure 11 Using grep with -c option.....	8
Figure 12 Using grep to search for capital K .....	8
Figure 13 Using grep with -n option. ....	8
Figure 14 Defining alias. ....	9
Figure 15 Verifying for created alias.....	9
Figure 16 Using our alias in our home directory. ....	10
Figure 17 Removing alias named lsal.....	11
Figure 18 Verifying aliases. ....	11
Figure 19 Adding alias in .bashrc .....	12
Figure 20 Reading again from .bashrc .....	13
Figure 21 Verifying again.....	13
Figure 22 Adding nwho alias and verifying.....	14
Figure 23 Running nwho alias command.....	14
Figure 24 Executing history command .....	15
Figure 25 function redo command.....	16
Figure 26 executing the third last command .....	17
Figure 27 Executing last command that starts with l.....	18

## **1. UNIX Workshop Log: Directories, Files, and Alias**

### **1.1 Aims and Objectives**

#### **1.1.1 Aims**

The aim of this workshop is to improve the core UNIX skills and to streamline the command line tasks and to further improve the knowledge of directories and files.

#### **1.1.2 Objectives**

##### **Directory Skills:**

Create a clear directory structure using UNIX commands.

##### **File Handling with cat and grep:**

Practice making and managing files with cat.

Explore effective text searching using grep.

##### **Alias Proficiency:**

Learn to use, create, and remove aliases for common commands.

##### **Search Expertise:**

Master text searches with grep, trying out different options.

##### **System Understanding via Aliases:**

Connect aliases to system information with the nwho alias.

##### **Command History Tricks:**

Explore command history and redo actions using history and fc.

##### **Alias Persistence:**

Understand how alias can be made permanent by editing the .bashrc

## **1.2 Required Tools and Concepts**

Linux virtual machine or WSL machine

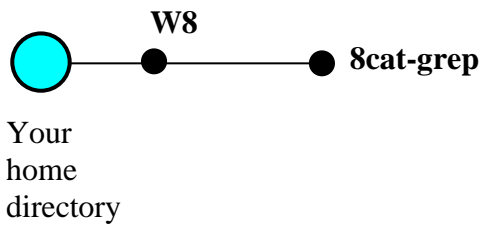
Linux basics

Text editors like nano, geany, vim etc

Linux file structures knowledge

## 2. Questions with their answers

1. Create the directory structure presented in the figure below.



Ans,

```
aayush@aayush: ~  
aayush@aayush:~$ mkdir -p w8/8cat-grep  
aayush@aayush:~$
```

Figure 1 Creating Directory Structure

2. Change to the 8cat-grep directory by one step using a relative pathname.

Ans,

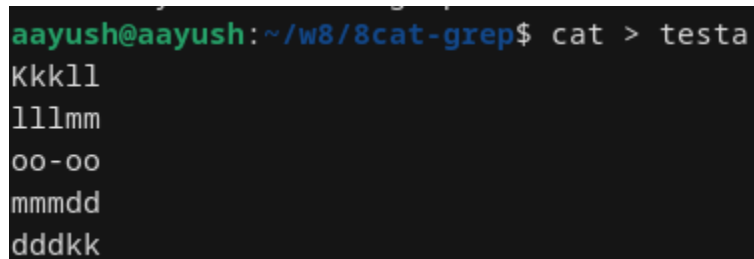
```
aayush@aayush:~$  
aayush@aayush:~$ cd w8/8cat-grep/  
aayush@aayush:~/w8/8cat-grep$ pwd  
/home/aayush/w8/8cat-grep  
aayush@aayush:~/w8/8cat-grep$
```

Figure 2 Changing Directory Using Relative Path

### 3. Using the cat utility, create two files

File testa	File testb
Kkkll	KKKKK
lllmm	LLLLL
oo--oo	MMMMM
mmmdd	DDDDD
dddkk	

Ans, We are using the write functionality of the cat utility to create and write into a file.



```
aayush@aayush:~/w8/8cat-grep$ cat > testa
Kkkll
lllmm
oo--oo
mmmdd
dddkk
```

Figure 3 Using cat command to create and write into a file.



```
aayush@aayush:~/w8/8cat-grep$ cat > testb
KKKKK
LLLLL
MMMMM
DDDDD
aayush@aayush:~/w8/8cat-grep$
```

Figure 4 Using cat command to create and write into a file.

**4. Give the following commands and explain the results for yourself**

- `grep ll testa`
- `grep -v ll testa`
- `grep -n ll testa`
- `grep -l ll *`
- `grep -i ll *`
- `grep -i LL *`
- `grep -c ll *`
- `grep '^K' testa testb`
- `grep -n '^' testa`

Ans,

The `grep` is one of the most used command line tool for UNIX based operating systems used mainly for filtering certain patterns of texts or characters within certain files.

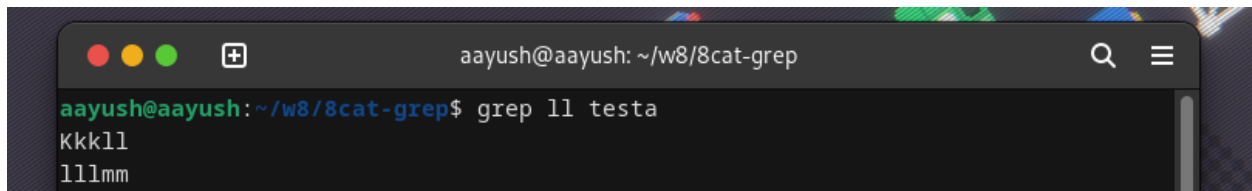
A terminal window titled 'aayush@aayush: ~/w8/8cat-grep' showing the command 'aayush@aayush:~/w8/8cat-grep\$ grep ll testa' and its output: 'Kkkll' and 'lllmm'.

Figure 5 `grep` command without options.

Using just the `grep` command without any option will just search for the pattern 'll' in the supplied file i.e., filea, In the picture above we can see that it displays the lines of text containing the pattern ll in them, in some terminals or other Linux distributions the specified pattern of text is highlighted.

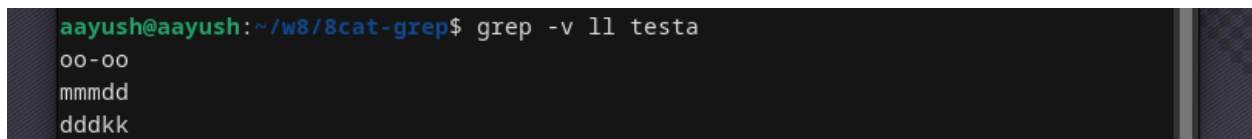
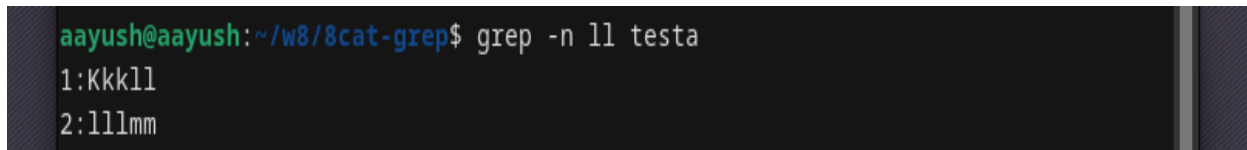
A terminal window titled 'aayush@aayush: ~/w8/8cat-grep' showing the command 'aayush@aayush:~/w8/8cat-grep\$ grep -v ll testa' and its output: 'oo-oo', 'mmdd', and 'dddkk'.

Figure 6 Using `grep` with `-v` option.

Using the `grep` with `-v` option or `--invert-match` option will select the non-matching lines from the file specified. In the figure above we have used the option `-v` with `grep` with the



'll' pattern and specified the file named testa and as the output we can see that it returned the lines which do not have the pattern 'll' in the specified file.

A terminal window with a dark background. The prompt is 'aayush@aayush: ~/w8/8cat-grep\$'. The command 'grep -n ll testa' has been entered. The output shows two lines: '1:Kkkll' and '2:lllmm'.

```
aayush@aayush: ~/w8/8cat-grep$ grep -n ll testa
1:Kkkll
2:lllmm
```

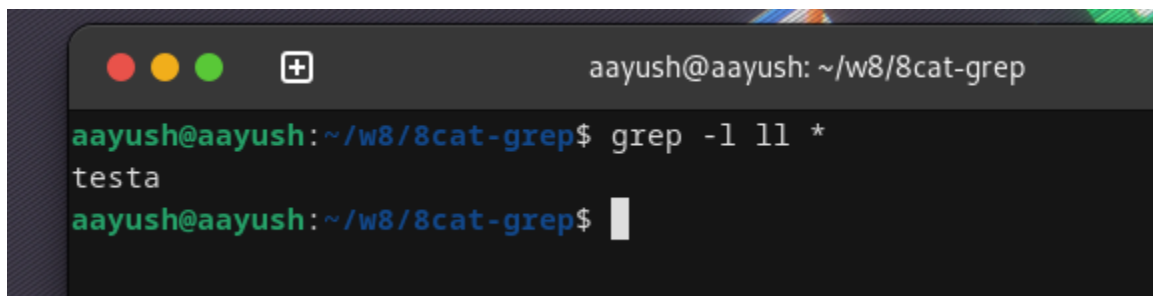
Figure 7 Using grep with -n option.

Using grep command with the -n option or --line-number or --line-buffered will do either one of two things:

**--line-number:** this option will print the line number with output lines.

**--line-buffered:** this option will flush the output on every line.

In the figure above we can see that when we are using the -n option of grep to search for 'll' pattern in test we can see it shows the line from the file which was supplied in the command i.e., testa that contains the pattern 'll' with its line number within that file.

A terminal window with a dark background. The prompt is 'aayush@aayush: ~/w8/8cat-grep\$'. The command 'grep -l ll \*' has been entered. The output shows the filename 'testa'.

```
aayush@aayush: ~/w8/8cat-grep$ grep -l ll *
testa
aayush@aayush: ~/w8/8cat-grep$
```

Figure 8 Using grep with -l option with \* to check for all files.

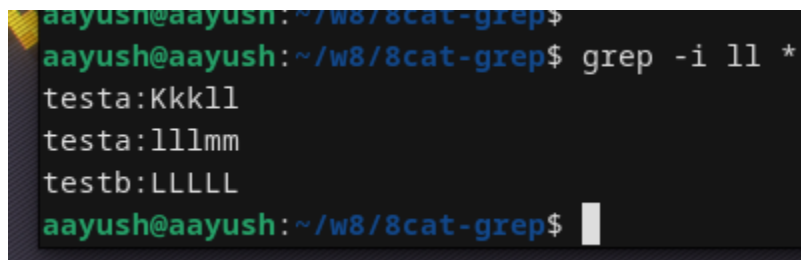
Using the grep command with lower caps -l or --files-with-matches option will print only the names of the file with selected lines, in the above figure we used asterisk symbol "\*" in place of any file names, using "\*" will tell our system to look for all the multiple files in our current working directory or any directory that we specify explicitly. This is helpful for us when we have a large number of files, and we have to check it all at once.



```
aayush@aayush:~/w8/8cat-grep$ grep -i ll *
testa:Kkkll
testa:lllmm
testb:LLLLL
aayush@aayush:~/w8/8cat-grep$
```

Figure 9 Using grep with -i option to search for all files.

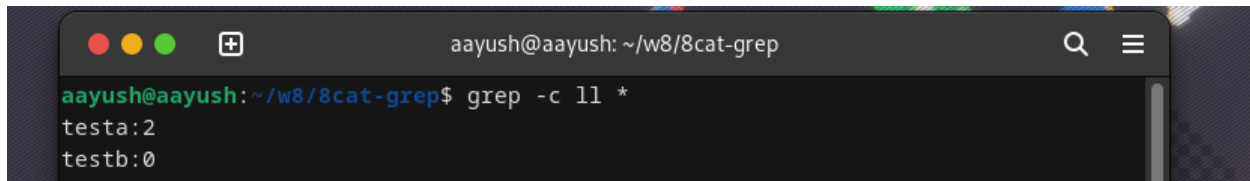
Using the grep command with -i or --no-ignore-lower-case option to search will perform a case-insensitive search for the pattern "ll" in all files in the current directory. In the figure above we can see that we have used the option -i of grep to search for 'll' lowercase 'll' on all of the files of our current working directory and we were expected to see the names of files having "ll" in them, but we can see that it showed the name of the file and the lines containing both lowercase and uppercase "ll" as the output.



```
aayush@aayush:~/w8/8cat-grep$ 
aayush@aayush:~/w8/8cat-grep$ grep -i ll *
testa:Kkkll
testa:lllmm
testb:LLLLL
aayush@aayush:~/w8/8cat-grep$
```

Figure 10 Again using grep with -i option to search for uppercase text patterns.

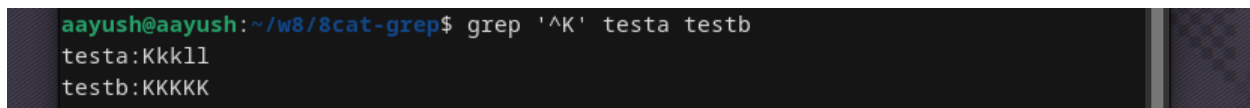
Again, using the -i option of grep to search for uppercase text patterns will display the same output as the previous step as it will only search for text patterns and ignore all the uppercase and lowercase texts.



```
aayush@aayush: ~/w8/8cat-grep
aayush@aayush:~/w8/8cat-grep$ grep -c ll *
testa:2
testb:0
```

Figure 11 Using grep with -c option.

Using grep with -c option or --count will print only the count of selected lines per file. In the above figure we are trying to count how many lines we have "ll" pattern in each file of our current working directory. We can see that at our output it displays the filenames present in our current directory and the number of times "ll" pattern has occurred within that files.



```
aayush@aayush:~/w8/8cat-grep$ grep '^K' testa testb
testa:Kkkll
testb:KKKKK
```

Figure 12 Using grep to search for capital K

This command searches for lines in files **testa** and **testb** that start with the character 'K'. It will display lines from both files where the line begins with 'K'.



```
aayush@aayush:~/w8/8cat-grep$ grep -n '^' testa
1:Kkkll
2:lllmm
3:oo-oo
4:mmmdd
5:dddkk
aayush@aayush:~/w8/8cat-grep$
```

Figure 13 Using grep with -n option.

Using the -n option of grep we can display line numbers along with lines that match the specified pattern. In this case, it searches for lines that start with the beginning of the line in the file **testa**. It will display lines from **testa** where each line is preceded by its line number.

### 5. (6) Define the lsal alias for ls -al command

Show that your system stores it giving the alias command (without arguments).

Use it in your home directory.

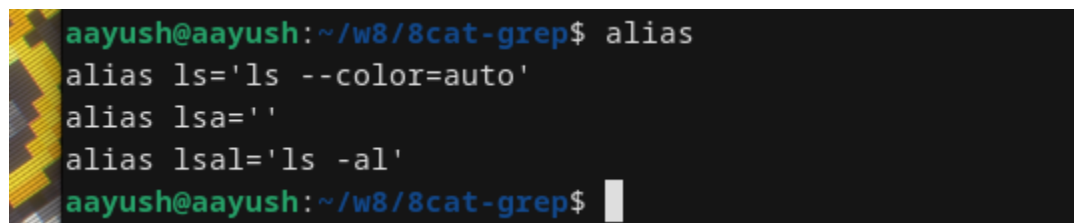
Ans,

An alias is a user-defined shorthand or substitute for a command or series of commands. It allows users to create custom shortcuts for complex or frequently used commands, making it easier and quicker to execute them. Aliases are commonly used in command-line interfaces (CLIs) in operating systems like UNIX, Linux, and macOS.



```
aayush@aayush:~/w8/8cat-grep$ alias lsal='ls -al'
```

Figure 14 Defining alias.



```
aayush@aayush:~/w8/8cat-grep$ alias
alias ls='ls --color=auto'
alias lsa=''
alias lsal='ls -al'
aayush@aayush:~/w8/8cat-grep$
```

Figure 15 Verifying for created alias.

Using the alias command without any arguments will show us all the available aliases which are present in our system. We can also see that there is our newly created alias named “lsal”

```

aayush@aayush: ~/w8/8cat-grep$ lsal ~
total 160
drwx----- 23 aayush aayush  4096 Dec 18 03:55 .
drwxr-xr-x  3 root   root    4096 Dec  4 04:23 ..
-rw-r--r--  1 aayush aayush 32768 Dec  7 21:46 a2script
-rw-----  1 aayush aayush  2058 Dec 15 05:13 .bash_history
-rw-r--r--  1 aayush aayush   220 Dec  4 04:23 .bash_logout
-rw-r--r--  1 aayush aayush  3568 Dec 15 00:12 .bashrc
-rw-r--r--  1 root   root     20 Dec  4 04:28 .bashshrc
drwxr-xr-x 11 aayush aayush  4096 Apr 23 2023 BigSur-Originals-Prime
drwxr-xr-x 10 aayush aayush  4096 Apr 23 2023 BigSur-Originals-Prime-dark
drwx----- 14 aayush aayush  4096 Dec 15 05:11 .cache
drwx----- 16 aayush aayush  4096 Dec 15 05:13 .config
drwxr-xr-x  2 aayush aayush  4096 Dec  4 04:25 Desktop
drwxr-xr-x  2 aayush aayush  4096 Dec  4 04:25 Documents
drwxr-xr-x  2 aayush aayush  4096 Dec 15 00:42 Downloads
-rw-r--r--  1 aayush aayush  5290 Dec  4 04:23 .face
lrwxrwxrwx  1 aayush aayush     5 Dec  4 04:23 .face.icon -> .face
-rw-r--r--  1 aayush aayush     0 Dec  8 04:14 file
drwx-----  2 aayush aayush  4096 Dec 18 03:55 .gnupg
drwxr-xr-x  3 aayush aayush  4096 Dec 15 05:12 .icons
-rw-----  1 aayush aayush    20 Dec 11 05:00 .lessht
drwx-----  4 aayush aayush  4096 Dec  4 04:25 .local
-rw-r--r--  1 aayush aayush   74 Dec  4 10:45 ~/.lock.a1script#
drwx-----  4 aayush aayush  4096 Dec  4 04:45 .mozilla
drwxr-xr-x  2 aayush aayush  4096 Dec  4 04:25 Music
drwxr-xr-x  2 aayush aayush  4096 Dec  4 04:25 Pictures
drwx-----  3 aayush aayush  4096 Dec  8 21:46 .pki
-rw-r--r--  1 aayush aayush   807 Dec  4 04:23 .profile
drwxr-xr-x  2 aayush aayush  4096 Dec  4 04:25 Public
drwx-----  2 aayush aayush  4096 Dec  4 04:26 .ssh
drwxr-xr-x  2 aayush aayush  4096 Dec  4 04:25 Templates
drwxr-xr-x  3 aayush aayush  4096 Dec 15 00:23 .themes

```

Figure 16 Using our alias in our home directory.

To use the alias that we recently created we need to use the name of that alias in our case it is the “**lsal**” to use it on our home directory without changing directories to the home directory we can use “**lsal ~**” and it will be equivalent of using this “**ls -al ~**” command.

## 6. Remove the alias.

**Show that your system does not store it.**

Ans, To remove any alias that we made we just need to use the `unalias` command with the name of the alias. In our case it will be “**unalias lsal**” it will just remove the alias having name **lsal** from our system.

```
aayush@aayush:~/w8/8cat-grep$  
aayush@aayush:~/w8/8cat-grep$ unalias lsal
```

Figure 17 Removing alias named **lsal**.

```
aayush@aayush:~/w8/8cat-grep$ alias  
alias ls='ls --color=auto'  
alias lsa=''  
aayush@aayush:~/w8/8cat-grep$
```

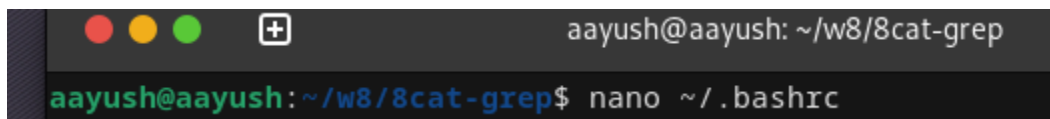
Figure 18 Verifying aliases.

After using the `alias` command, we can see that it will show us the alias command and we can see that the alias that we created named **lsal** is removed.

## 7. Define this alias again preserving it for the next session

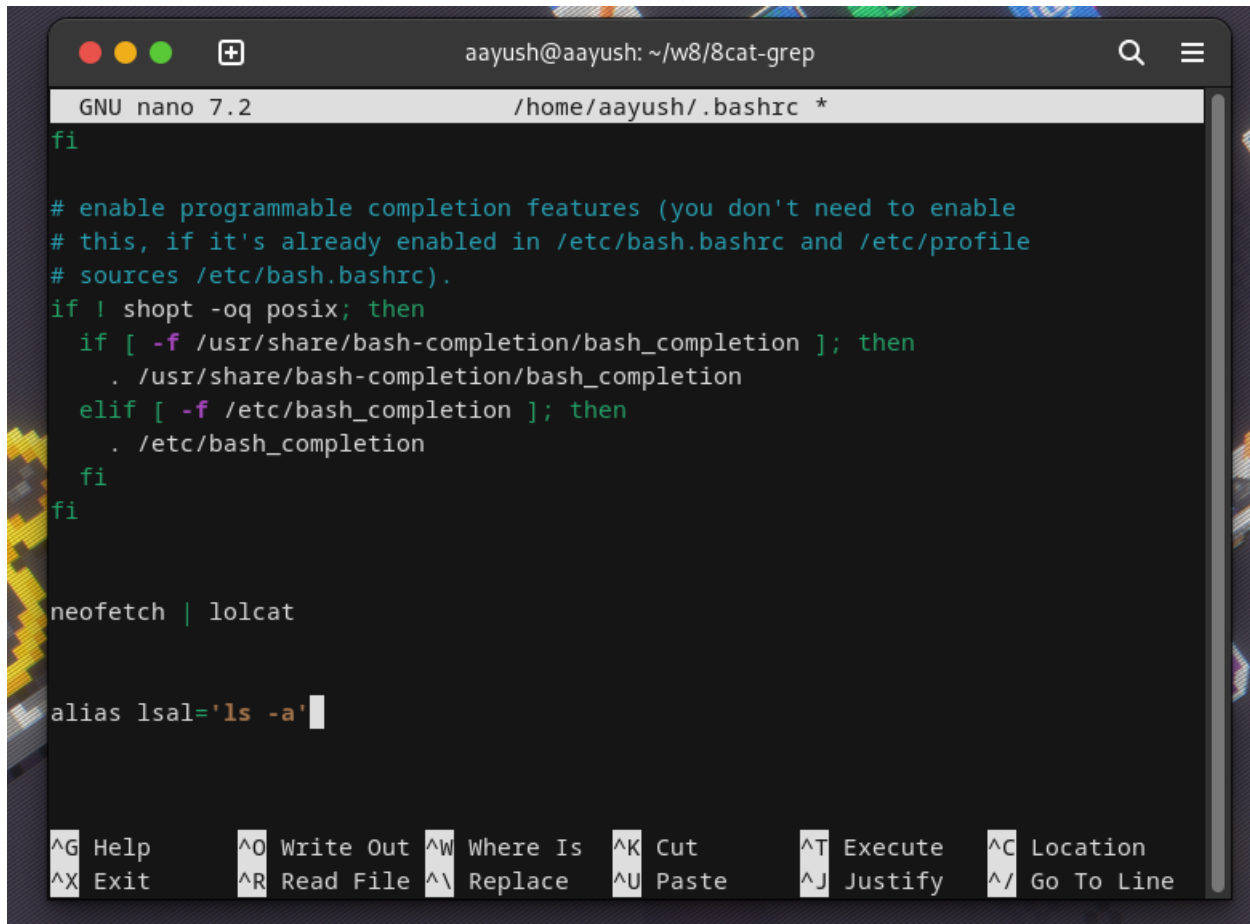
**Show that the system still keeps this your alias.**

Ans, To make an alias permanent we need to edit the `.bashrc` or `.zshrc` located in the home directory via a text editor such as nano, vim, geany etc. The `.bashrc` or `.zshrc` stores the configuration files for the shell that the device is using for example if a device is using bash shell as the terminal emulator it will have the `.bashrc` file in the user's home directory.



```
aayush@aayush: ~/w8/8cat-grep
aayush@aayush:~/w8/8cat-grep$ nano ~/.bashrc
```

Figure 19 using nano to open `.bashrc`



```
GNU nano 7.2 /home/aayush/.bashrc *
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

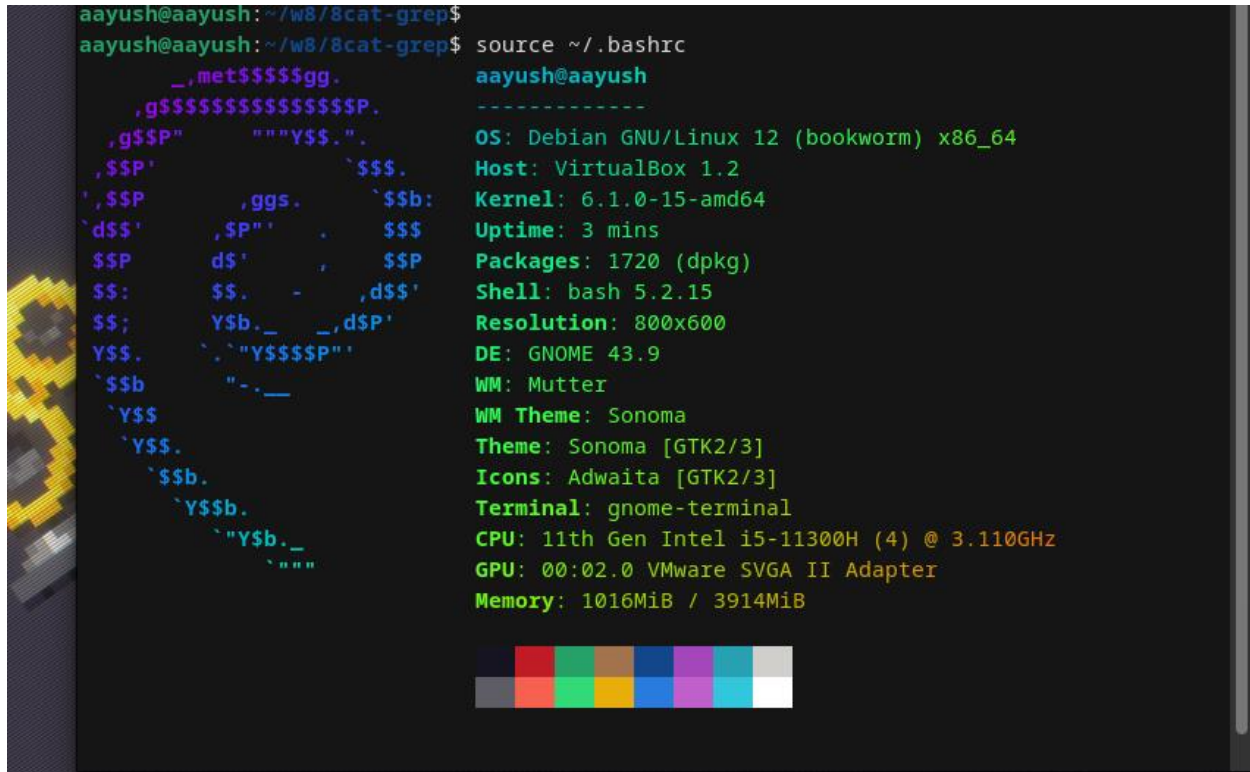
neofetch | lolcat

alias lsal='ls -a'
```

Figure 20 Adding alias in `.bashrc`

To open nano as a text editor, we need to type nano and the name of the file so it would be nano [file name]. since my Linux distribution has bash as its default terminal emulator

i have the .bashrc file so the command will be **nano ~/.bashrc**, it will open a new window which shows the .bashrc configurations we need to scroll down using the down arrow key and type **alias lsal='ls -al'** and exit nano text editor.



```

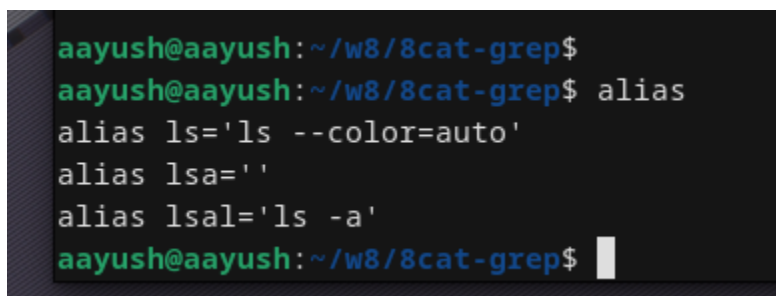
aayush@aayush:~/w8/8cat-grep$
aayush@aayush:~/w8/8cat-grep$ source ~/.bashrc
aayush@aayush
-----
OS: Debian GNU/Linux 12 (bookworm) x86_64
Host: VirtualBox 1.2
Kernel: 6.1.0-15-amd64
Uptime: 3 mins
Packages: 1720 (dpkg)
Shell: bash 5.2.15
Resolution: 800x600
DE: GNOME 43.9
WM: Mutter
WM Theme: Sonoma
Theme: Sonoma [GTK2/3]
Icons: Adwaita [GTK2/3]
Terminal: gnome-terminal
CPU: 11th Gen Intel i5-11300H (4) @ 3.110GHz
GPU: 00:02.0 VMware SVGA II Adapter
Memory: 1016MiB / 3914MiB

  _met$$$$$gg.
  ,g$$$$$$$$$$$$$P.
  ,g$$$P"      ""Y$$$.
  ,$$P'        `$$$_.
  ',$$P'      ,ggs.   `$$$
  `d$$'      ,P"     $$$
  $$$       d$'      $$$
  $$$       $$_.    ,d$$'
  $$;       Y$b._   ,dP'
  Y$$       `."Y$$$$P"
  `$$b      "-._
  `Y$$
  `Y$$_.
  `$$b.
  `Y$$b.
  `."Y$b._
  `""

```

Figure 21 Reading again from .bashrc

After exiting we need to read from the .bashrc file again for that we can restart our device, or we can type `source ~/.bashrc` to read from the file again. If we type `source ~/.bashrc` it will restart our terminal and the new alias which we set can be used.



```

aayush@aayush:~/w8/8cat-grep$
aayush@aayush:~/w8/8cat-grep$ alias
alias ls='ls --color=auto'
alias lsa=''
alias lsal='ls -a'
aayush@aayush:~/w8/8cat-grep$

```

Figure 22 Verifying again

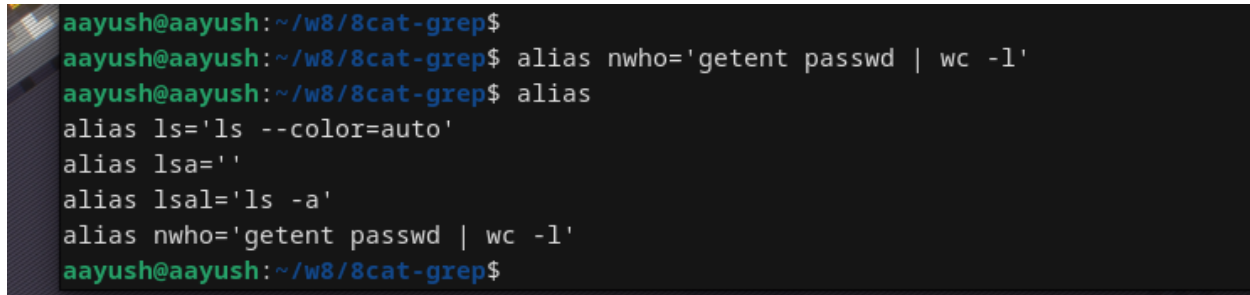
Verifying again to see all the aliases using the alias command.



8. Define the **nwho** alias for the number of system file at UNIX computers.

***alias nwho='getent passwd|wc -l'***

Ans,



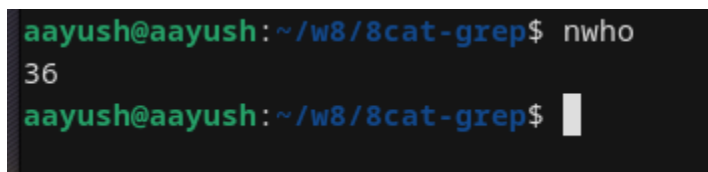
```
aayush@aayush:~/w8/8cat-grep$  
aayush@aayush:~/w8/8cat-grep$ alias nwho='getent passwd | wc -l'  
aayush@aayush:~/w8/8cat-grep$ alias  
alias ls='ls --color=auto'  
alias lsa=''  
alias lsal='ls -a'  
alias nwho='getent passwd | wc -l'  
aayush@aayush:~/w8/8cat-grep$
```

Figure 23 Adding nwho alias and verifying.

9. Give the command **nwho**. Compare the figure displayed with ones got by your UNIX-mates.

Ans,

This **nwho** alias will count the number of user accounts on the system using the **getent passwd|wc -l** command.



```
aayush@aayush:~/w8/8cat-grep$ nwho  
36  
aayush@aayush:~/w8/8cat-grep$
```

Figure 24 Running nwho alias command.

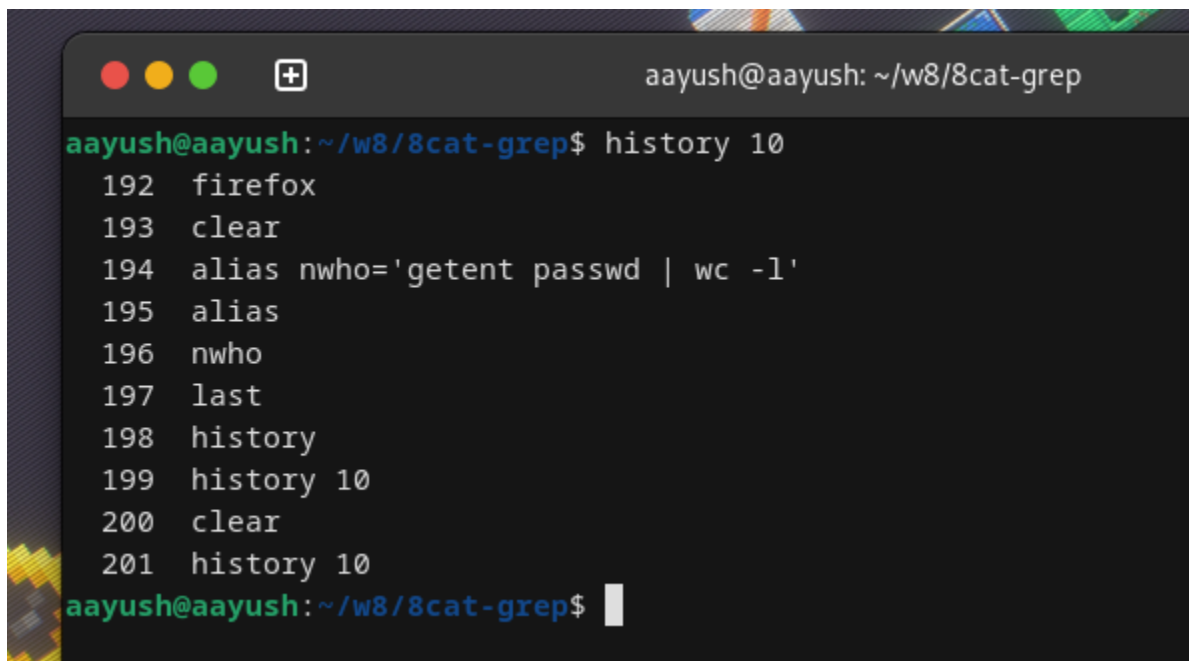
The output of the **nwho** alias, counting user accounts with **getent passwd | wc -l**, may vary across different Linux or UNIX operating systems. Various factors contribute to these differences, including the distinct user management approaches employed by different OS distributions. System configurations, default user accounts, and the presence of networked users can also influence the count. Therefore, when executing **nwho** on different UNIX systems, one can expect divergent values due to the unique characteristics and user environments of each operating system.

**10. List your last commands executed giving the history command.**

Ans,

The **history** command in UNIX-like operating systems is used to display a list of previously executed commands. When you append a number (e.g., **history 10**), it shows the specified number of most recent commands from your command history.

In the case of **history 10**, the command will display the last 10 commands that you have executed in the current terminal session. The output typically includes the command numbers and the corresponding commands.

A screenshot of a terminal window titled 'aayush@aayush: ~/w8/8cat-grep'. The prompt is 'aayush@aayush:~/w8/8cat-grep\$'. The command 'history 10' has been entered, and the output shows a list of 10 commands with their corresponding line numbers: 192 firefox, 193 clear, 194 alias nwho='getent passwd | wc -l', 195 alias, 196 nwho, 197 last, 198 history, 199 history 10, 200 clear, and 201 history 10. The prompt is now 'aayush@aayush:~/w8/8cat-grep\$' with a cursor at the end.

```
aayush@aayush:~/w8/8cat-grep$ history 10
192  firefox
193  clear
194  alias nwho='getent passwd | wc -l'
195  alias
196  nwho
197  last
198  history
199  history 10
200  clear
201  history 10
aayush@aayush:~/w8/8cat-grep$
```

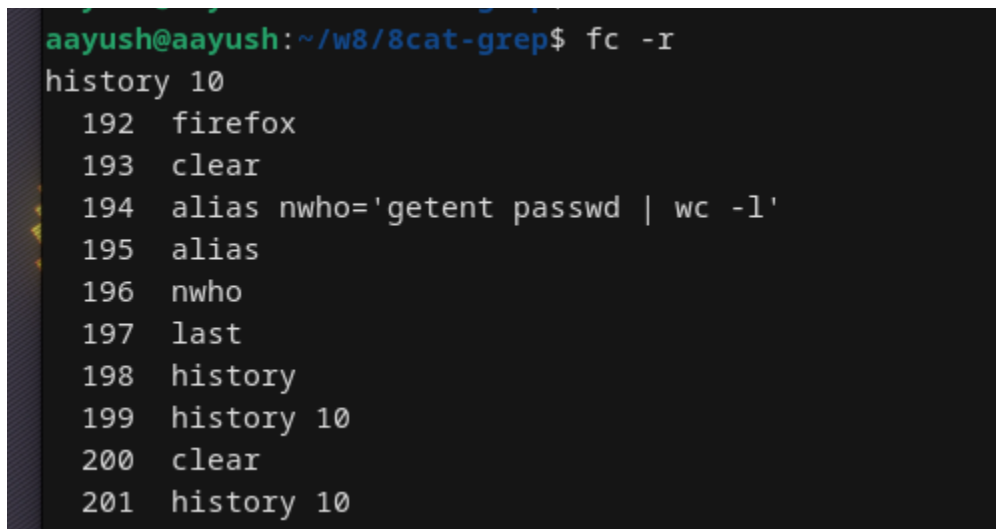
Figure 25 Executing history command

**11. Re-execute the *last but one* command using the redo (r) command and the number of the event.**

***fc -r***

Ans,

The **fc -r** command is a shell command used to redo (re-execute) the last command that was previously executed. It simplifies the process of rerunning the most recent command without retyping it.



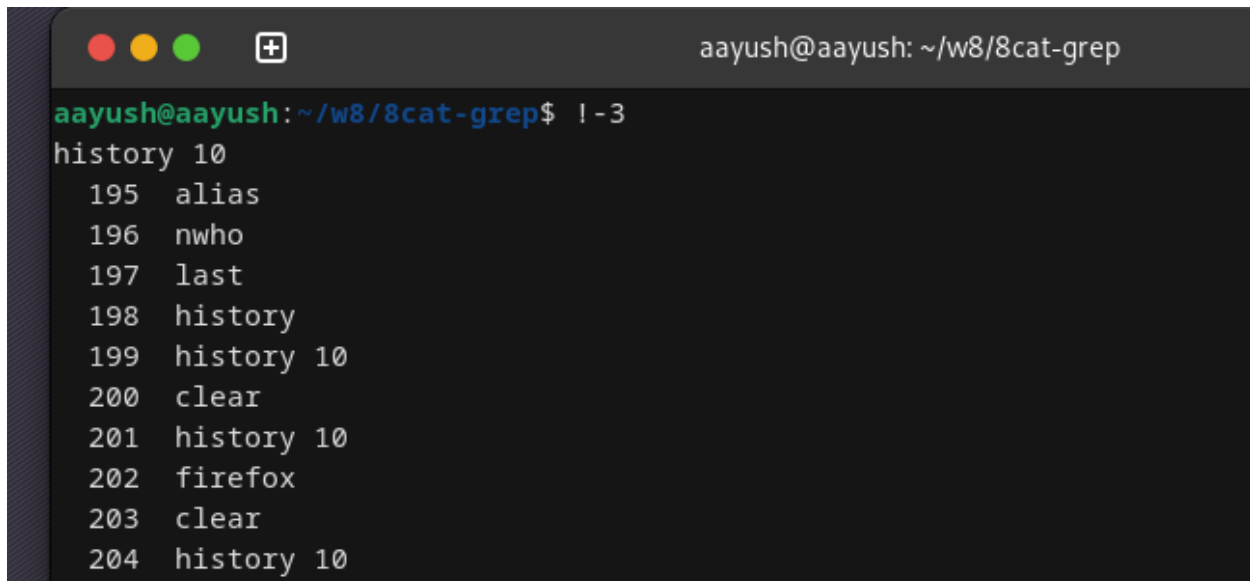
```
aayush@aayush:~/w8/8cat-grep$ fc -r
history 10
192  firefox
193  clear
194  alias nwho='getent passwd | wc -l'
195  alias
196  nwho
197  last
198  history
199  history 10
200  clear
201  history 10
```

Figure 26 function redo command

**12. Re-execute the command given *three commands ago* using the negative integer.**

**!-3**

Ans, The command **!-3** is a history expansion in UNIX-like shells. It retrieves and re-executes the command that was executed three commands ago from the current command history. This provides a quick way to repeat a specific command without typing it out again.

A terminal window with a dark background. The title bar shows three colored circles (red, yellow, green) and a plus icon, followed by the text 'aayush@aayush: ~/w8/8cat-grep'. The terminal content shows the prompt 'aayush@aayush:~/w8/8cat-grep\$' followed by the command '!-3'. Below this, the command 'history 10' is executed, displaying a list of commands with their line numbers: 195 alias, 196 nwho, 197 last, 198 history, 199 history 10, 200 clear, 201 history 10, 202 firefox, 203 clear, and 204 history 10.

```
aayush@aayush:~/w8/8cat-grep$ !-3
history 10
195 alias
196 nwho
197 last
198 history
199 history 10
200 clear
201 history 10
202 firefox
203 clear
204 history 10
```

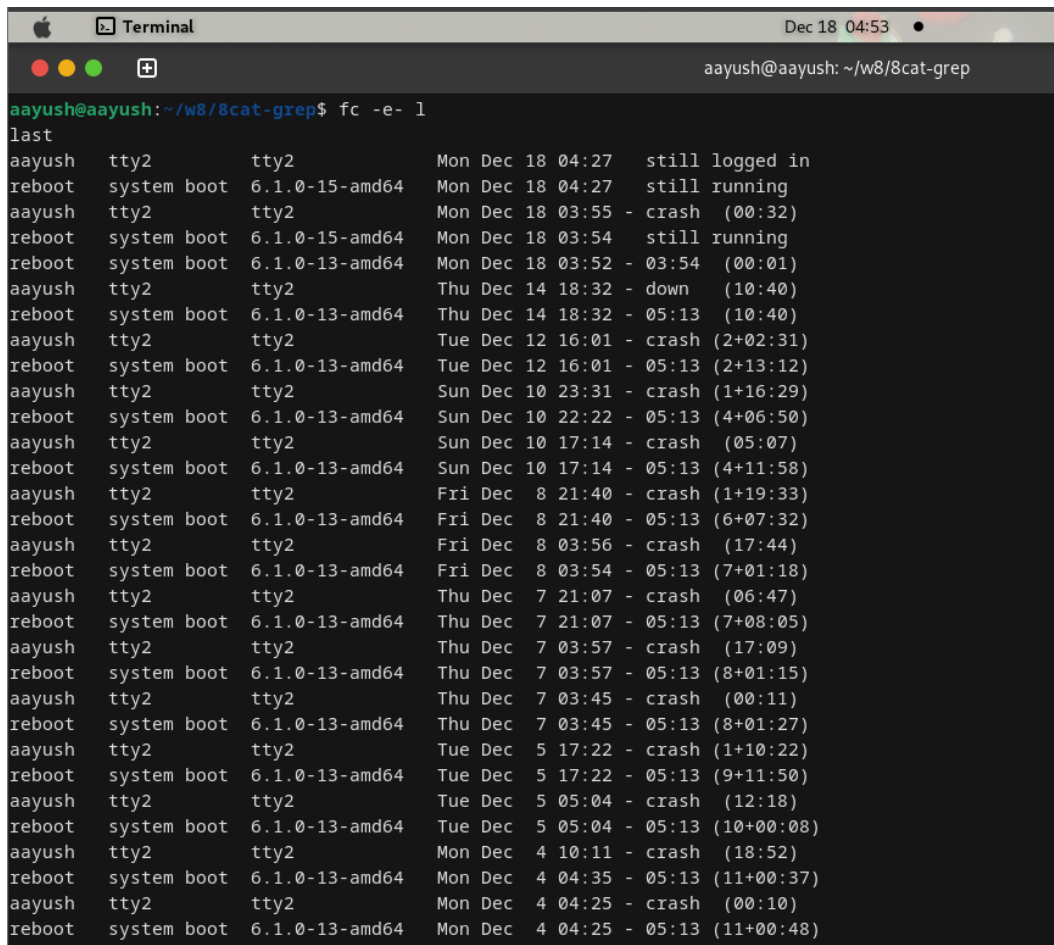
Figure 27 executing the third last command

### 13. Re-execute the last command which name begins with 'l'.

**fc -e- l**

Ans,

The **fc -e- l** command is a history expansion in UNIX-like shells. It re-executes the most recent command from the command history that starts with the letter 'l'. This provides a quick way to repeat a specific command with a particular prefix.



```

aayush@aayush: ~/w8/8cat-grep
aayush@aayush:~/w8/8cat-grep$ fc -e- l
last
aayush  tty2      tty2      Mon Dec 18 04:27  still logged in
reboot  system boot 6.1.0-15-amd64  Mon Dec 18 04:27  still running
aayush  tty2      tty2      Mon Dec 18 03:55 - crash (00:32)
reboot  system boot 6.1.0-15-amd64  Mon Dec 18 03:54  still running
reboot  system boot 6.1.0-13-amd64  Mon Dec 18 03:52 - 03:54 (00:01)
aayush  tty2      tty2      Thu Dec 14 18:32 - down (10:40)
reboot  system boot 6.1.0-13-amd64  Thu Dec 14 18:32 - 05:13 (10:40)
aayush  tty2      tty2      Tue Dec 12 16:01 - crash (2+02:31)
reboot  system boot 6.1.0-13-amd64  Tue Dec 12 16:01 - 05:13 (2+13:12)
aayush  tty2      tty2      Sun Dec 10 23:31 - crash (1+16:29)
reboot  system boot 6.1.0-13-amd64  Sun Dec 10 22:22 - 05:13 (4+06:50)
aayush  tty2      tty2      Sun Dec 10 17:14 - crash (05:07)
reboot  system boot 6.1.0-13-amd64  Sun Dec 10 17:14 - 05:13 (4+11:58)
aayush  tty2      tty2      Fri Dec 8 21:40 - crash (1+19:33)
reboot  system boot 6.1.0-13-amd64  Fri Dec 8 21:40 - 05:13 (6+07:32)
aayush  tty2      tty2      Fri Dec 8 03:56 - crash (17:44)
reboot  system boot 6.1.0-13-amd64  Fri Dec 8 03:54 - 05:13 (7+01:18)
aayush  tty2      tty2      Thu Dec 7 21:07 - crash (06:47)
reboot  system boot 6.1.0-13-amd64  Thu Dec 7 21:07 - 05:13 (7+08:05)
aayush  tty2      tty2      Thu Dec 7 03:57 - crash (17:09)
reboot  system boot 6.1.0-13-amd64  Thu Dec 7 03:57 - 05:13 (8+01:15)
aayush  tty2      tty2      Thu Dec 7 03:45 - crash (00:11)
reboot  system boot 6.1.0-13-amd64  Thu Dec 7 03:45 - 05:13 (8+01:27)
aayush  tty2      tty2      Tue Dec 5 17:22 - crash (1+10:22)
reboot  system boot 6.1.0-13-amd64  Tue Dec 5 17:22 - 05:13 (9+11:50)
aayush  tty2      tty2      Tue Dec 5 05:04 - crash (12:18)
reboot  system boot 6.1.0-13-amd64  Tue Dec 5 05:04 - 05:13 (10+00:08)
aayush  tty2      tty2      Mon Dec 4 10:11 - crash (18:52)
reboot  system boot 6.1.0-13-amd64  Mon Dec 4 04:35 - 05:13 (11+00:37)
aayush  tty2      tty2      Mon Dec 4 04:25 - crash (00:10)
reboot  system boot 6.1.0-13-amd64  Mon Dec 4 04:25 - 05:13 (11+00:48)

```

Figure 28 Executing last command that starts with l.

### 3. Conclusion:

During this week's workshop tasks, I practiced using UNIX utilities, focusing on directory manipulation and text processing commands. Successfully creating directories, generating files, and exploring various **grep** commands enhanced my proficiency. Alias management, including definition, verification, removal, and redefinition, demonstrated practical skills. The workshop also covered history navigation, command redo, and history expansion, contributing to a comprehensive understanding of essential UNIX concepts. Overall, the exercises provided valuable hands-on experience, reinforcing key skills in UNIX system navigation and manipulation.