

**Q1 : What does CLI stand for, and how does it differ from GUI?**

Ans :

- CLI stands for **Command-Line Interface**. It allows users to interact with the operating system by typing commands in a terminal.
- GUI (Graphical User Interface) uses visual elements like icons, buttons, and windows for interaction.

**Difference:**

- CLI is faster and uses fewer resources.
  - GUI is easier for beginners as it provides a visual representation of tasks.
- 

**Q2 : Open a terminal and execute a simple command such as echo "Hello". Can you perform a similar action in a GUI? What are the advantages of using a CLI compared to a GUI?**

Ans :

Theory:

- CLI provides precision, faster automation, and access to advanced options not always available in GUI.
- GUIs are slower but more user-friendly.

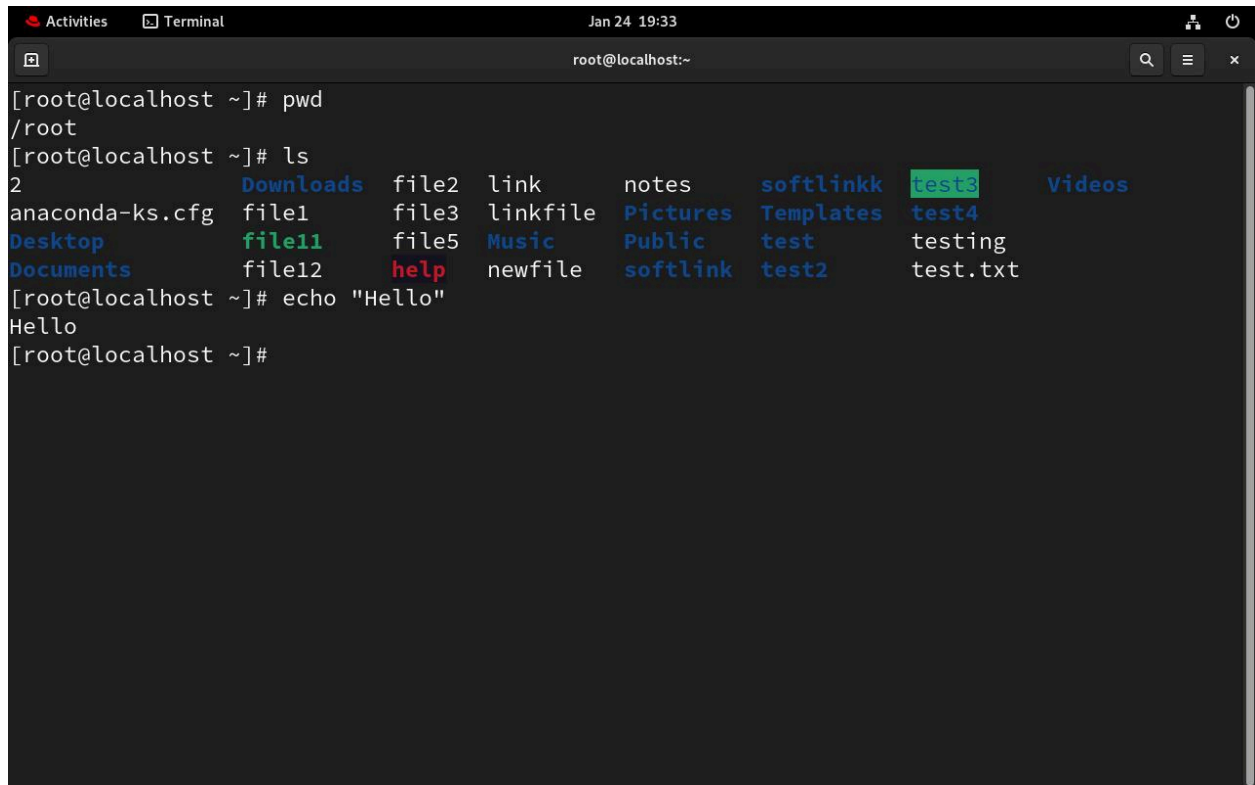
**Command :** `echo "Hello"`

GUI equivalent: Create a text file and type "Hello" manually.

**Advantages of CLI:**

- Speed and automation.
- Can be run remotely and in scripts.

## Q2 Image

A terminal window titled 'Terminal' with a date and time of 'Jan 24 19:33'. The user is logged in as 'root' at 'localhost'. The terminal shows the following commands and output:

```
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
2          Downloads  file2  link    notes    softlinkk  test3    Videos
anaconda-ks.cfg  file1  file3  linkfile Pictures Templates test4
Desktop        file11 file5   Music   Public   test       testing
Documents      file12 help    newfile softlink test2     test.txt
[root@localhost ~]# echo "Hello"
Hello
[root@localhost ~]#
```

**Q3 : To convert a minimal CLI-based Linux interface into a GUI-based one, which packages are typically required? Find them**

Ans :

**Theory:**

- For GUI, you need packages like a **Display Server (Xorg)**, **Window Manager**, and a **Desktop Environment** like GNOME or KDE.

**Commands to install GNOME Desktop Environment on RHEL:**

```
sudo yum groupinstall "Server with GUI" -y
```

```
sudo systemctl set-default graphical.target
```

```
sudo systemctl reboot
```

---

**Q4 : What are terminals in Linux? How many virtual terminals are available on your system, and which key combination is used to access them?**

ANS :

**Theory:**

- Terminals are interfaces to interact with the Linux system.
- Linux provides **6 virtual consoles** (tty1–tty6) by default.

**Command to switch between terminals:**

Press **Ctrl + Alt + F1** to **F6** to access virtual consoles.

Use **Ctrl + Alt + F7** to return to the GUI.

---

**Q5 : Write the commands to check a file and a directory in a long listing format. How can you determine whether it is a file or a directory?**

ANS :

**Theory:**

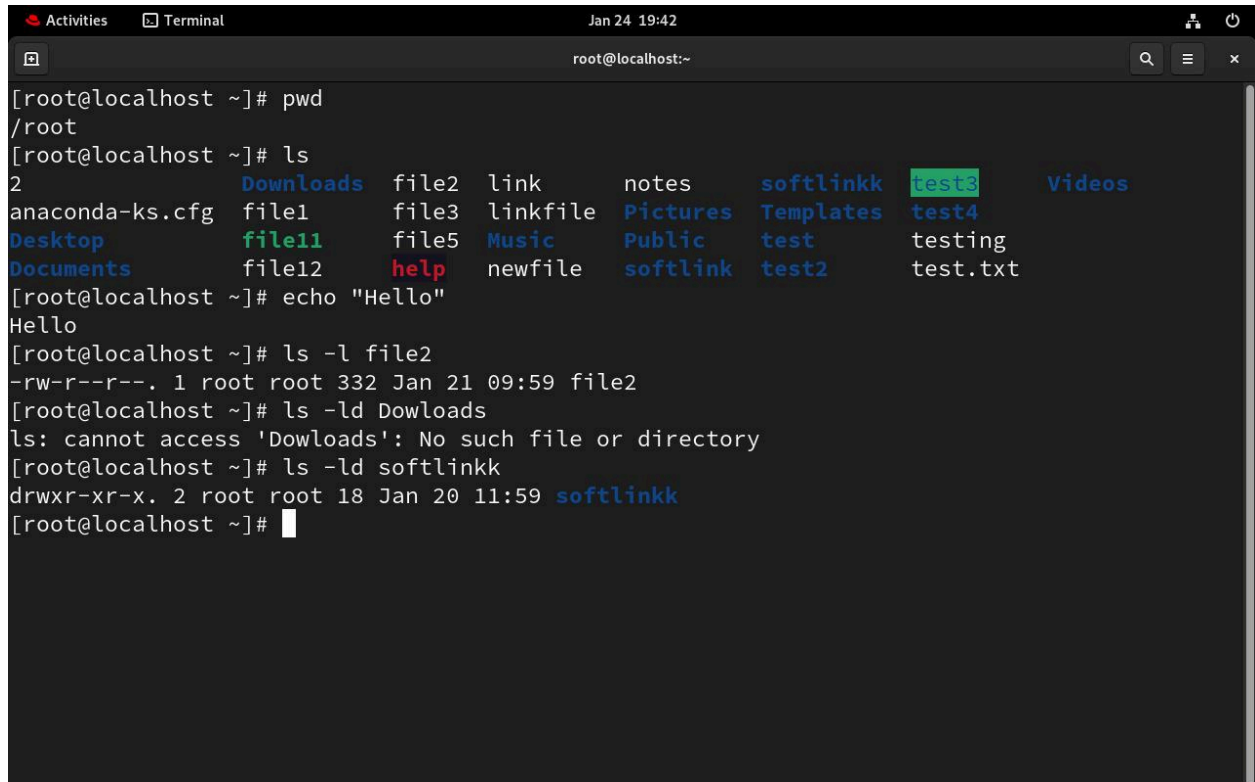
- The **ls -l** command displays file types and permissions.
- Directories start with **d** in the output.

## Command:

`ls -l filename`

`ls -ld directory name`

## Q5 Image



```
Activities Terminal Jan 24 19:42 root@localhost:~
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
2 Downloads file2 link notes softlinkk test3 Videos
anaconda-ks.cfg file1 file3 linkfile Pictures Templates test4
Desktop file11 file5 Music Public test testing
Documents file12 help newfile softlink test2 test.txt
[root@localhost ~]# echo "Hello"
Hello
[root@localhost ~]# ls -l file2
-rw-r--r--. 1 root root 332 Jan 21 09:59 file2
[root@localhost ~]# ls -ld Downloads
ls: cannot access 'Downloads': No such file or directory
[root@localhost ~]# ls -ld softlinkk
drwxr-xr-x. 2 root root 18 Jan 20 11:59 softlinkk
[root@localhost ~]#
```

---

**Q6 : . Which Linux commands are used to view the content of files and directories? Write the commands.**

**ANS :**

**Theory:**

- Use commands like `cat`, `less`, and `more` to view file content.

**Commands:**

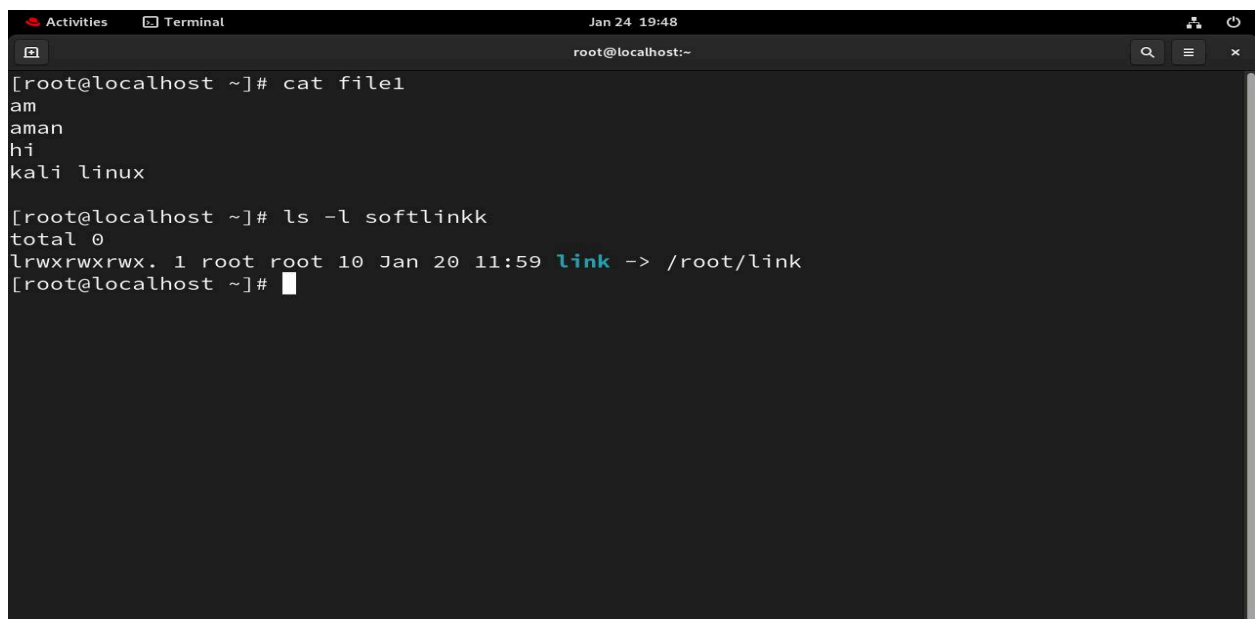
`cat filename`      # Show all content at once.

`less filename`      # Scroll through content.

`more filename`      # Similar to less but less flexible.

`ls -l directory`    # View directory details.

**Q6 IMAGE :**

A terminal window titled 'Terminal' with a date and time of 'Jan 24 19:48'. The prompt is 'root@localhost:~'. The user enters 'cat file1' and the output is 'am', 'aman', 'hi', and 'kali linux'. Then the user enters 'ls -l softlinkk' and the output is 'total 0', 'lrwxrwxrwx. 1 root root 10 Jan 20 11:59 link -> /root/link'. The prompt returns to 'root@localhost ~]#'.

```
[root@localhost ~]# cat file1
am
aman
hi
kali linux

[root@localhost ~]# ls -l softlinkk
total 0
lrwxrwxrwx. 1 root root 10 Jan 20 11:59 link -> /root/link
[root@localhost ~]#
```

---

**Q7. Change your current location to the /etc/yum.repos.d directory.**

**a. Using the relative path method, create a file named xyz.repo under the /etc/yum.repos.d/ directory.**

**b. Using the absolute path method, create a file named xyz.conf under the /etc/rsyslog.d/ directory.**

**c. What differences did you observe between using a relative path and an absolute path?**

a. Using Relative Path

```
cd /etc/yum.repos.d
```

```
touch xyz.repo
```

b : Using Absolute Path:

```
touch /etc/rsyslog.d/xyz.conf
```

c : Relative paths depend on the current directory, while absolute paths are independent

Q7 Image :



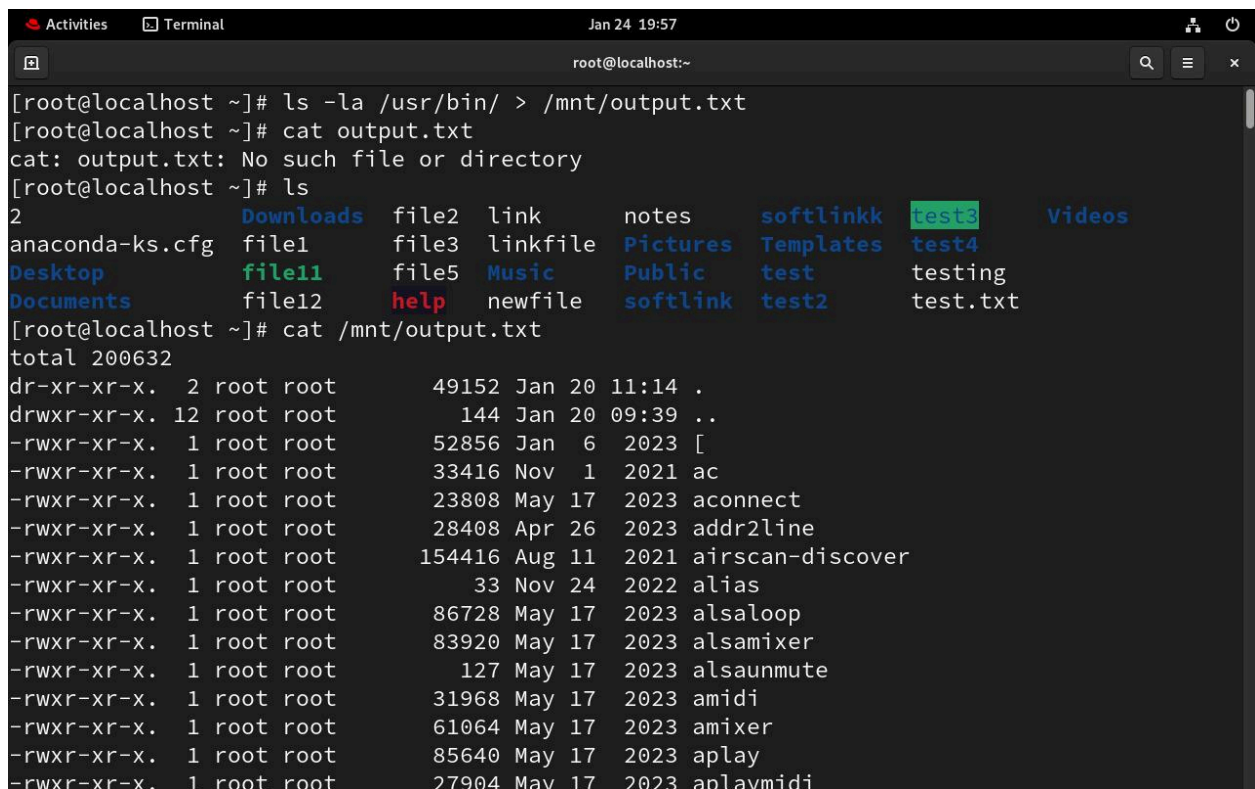
```
Activities Terminal Jan 24 19:52
root@localhost:/etc/yum.repos.d
[root@localhost ~]# cd /etc/yum.repos.d
[root@localhost yum.repos.d]# touch xyz.repo
[root@localhost yum.repos.d]# ls
redhat.repo  xyz.repo
[root@localhost yum.repos.d]#
```

**Q8 : . List all files, including hidden ones, in the /usr/bin/ directory with details like file permissions. Save the output to a file named output.txt in the /mnt directory. Write the command**

**ANS :**

**Command:** `ls -la /usr/bin > /mnt/output.txt`

**Q8 IMAGE :**



```
[root@localhost ~]# ls -la /usr/bin/ > /mnt/output.txt
[root@localhost ~]# cat output.txt
cat: output.txt: No such file or directory
[root@localhost ~]# ls
2          Downloads  file2  link    notes    softlinkk  test3      Videos
anaconda-ks.cfg  file1  file3  linkfile Pictures  Templates  test4
Desktop         file11 file5   Music   Public    test       testing
Documents       file12 help    newfile softlink  test2      test.txt
[root@localhost ~]# cat /mnt/output.txt
total 200632
dr-xr-xr-x.  2 root root      49152 Jan 20 11:14 .
drwxr-xr-x. 12 root root       144 Jan 20 09:39 ..
-rwxr-xr-x.  1 root root    52856 Jan  6 2023 [
-rwxr-xr-x.  1 root root    33416 Nov  1 2021 ac
-rwxr-xr-x.  1 root root   23808 May 17 2023 aconnect
-rwxr-xr-x.  1 root root   28408 Apr 26 2023 addr2line
-rwxr-xr-x.  1 root root  154416 Aug 11 2021 aircan-discover
-rwxr-xr-x.  1 root root      33 Nov 24 2022 alias
-rwxr-xr-x.  1 root root   86728 May 17 2023 alsaloop
-rwxr-xr-x.  1 root root   83920 May 17 2023 alsamixer
-rwxr-xr-x.  1 root root    127 May 17 2023 alsaunmute
-rwxr-xr-x.  1 root root   31968 May 17 2023 amidi
-rwxr-xr-x.  1 root root   61064 May 17 2023 amixer
-rwxr-xr-x.  1 root root   85640 May 17 2023 aplay
-rwxr-xr-x.  1 root root   27904 May 17 2023 aplaymidi
```

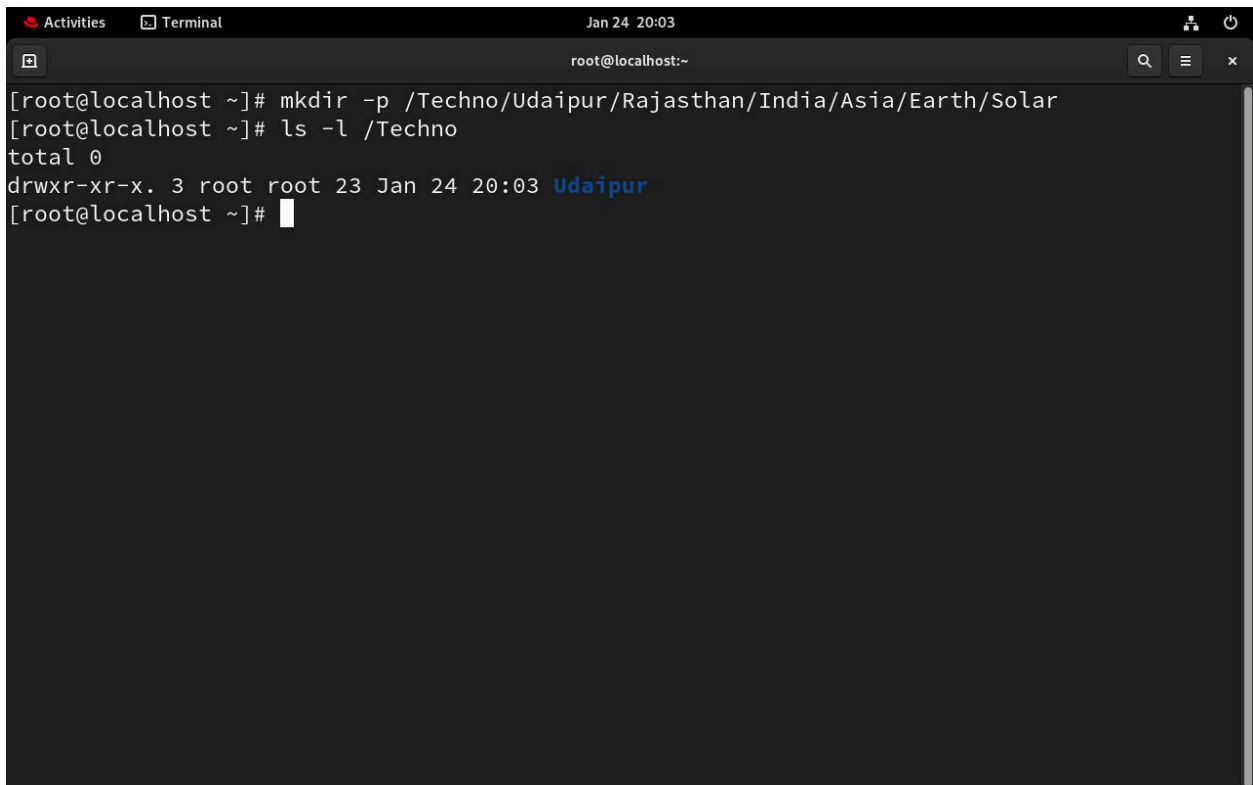
---

**Q9. Create nested directories and check the structure in a long listing format.**

**Command:**

```
mkdir -p  
/Techno/Udaipur/Rajasthan/India/Asia/Earth/Solar  
  
ls -lR /Techno
```

**Q9 IMAGE :**

A terminal window titled 'Terminal' with a timestamp of 'Jan 24 20:03'. The prompt is 'root@localhost:~'. The user enters the command 'mkdir -p /Techno/Udaipur/Rajasthan/India/Asia/Earth/Solar'. The next prompt is '[root@localhost ~]# ls -l /Techno'. The output shows 'total 0' followed by a line 'drwxr-xr-x. 3 root root 23 Jan 24 20:03 Udaipur'. The prompt returns to '[root@localhost ~]#'.

```
Activities Terminal Jan 24 20:03 root@localhost:~  
[root@localhost ~]# mkdir -p /Techno/Udaipur/Rajasthan/India/Asia/Earth/Solar  
[root@localhost ~]# ls -l /Techno  
total 0  
drwxr-xr-x. 3 root root 23 Jan 24 20:03 Udaipur  
[root@localhost ~]#
```



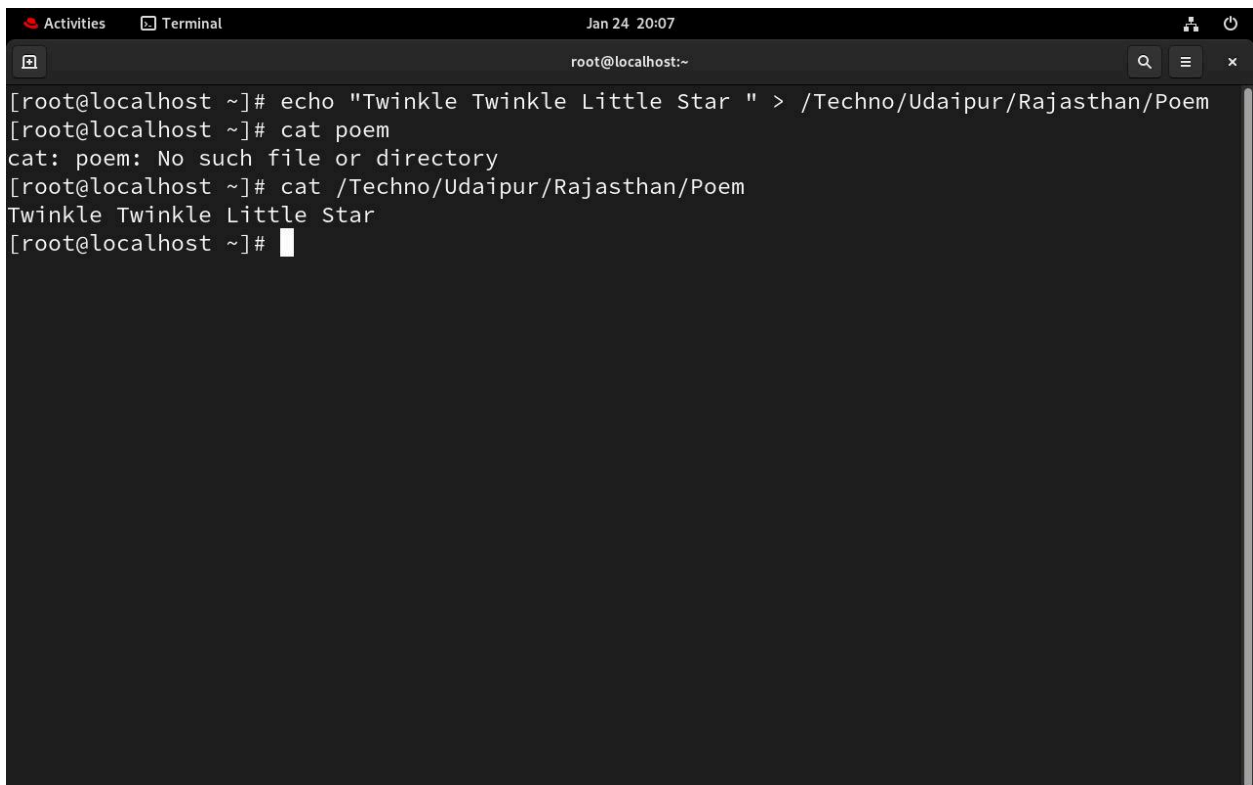
---

**10. Create a file named “Poem” under the /Techno/Udaipur/Rajasthan/ directory. Write the text "Twinkle Twinkle Little Star" into the file and save it. Perform all actions using the absolute path method.**

**Commands:**

```
echo "Twinkle Twinkle Little Star" >  
/Techno/Udaipur/Rajasthan/Poem
```

Q10 IMAGE :

A screenshot of a Linux terminal window. The window title is "Activities Terminal" and the date/time is "Jan 24 20:07". The terminal shows the following commands and output:

```
[root@localhost ~]# echo "Twinkle Twinkle Little Star " > /Techno/Udaipur/Rajasthan/Poem  
[root@localhost ~]# cat poem  
cat: poem: No such file or directory  
[root@localhost ~]# cat /Techno/Udaipur/Rajasthan/Poem  
Twinkle Twinkle Little Star  
[root@localhost ~]#
```

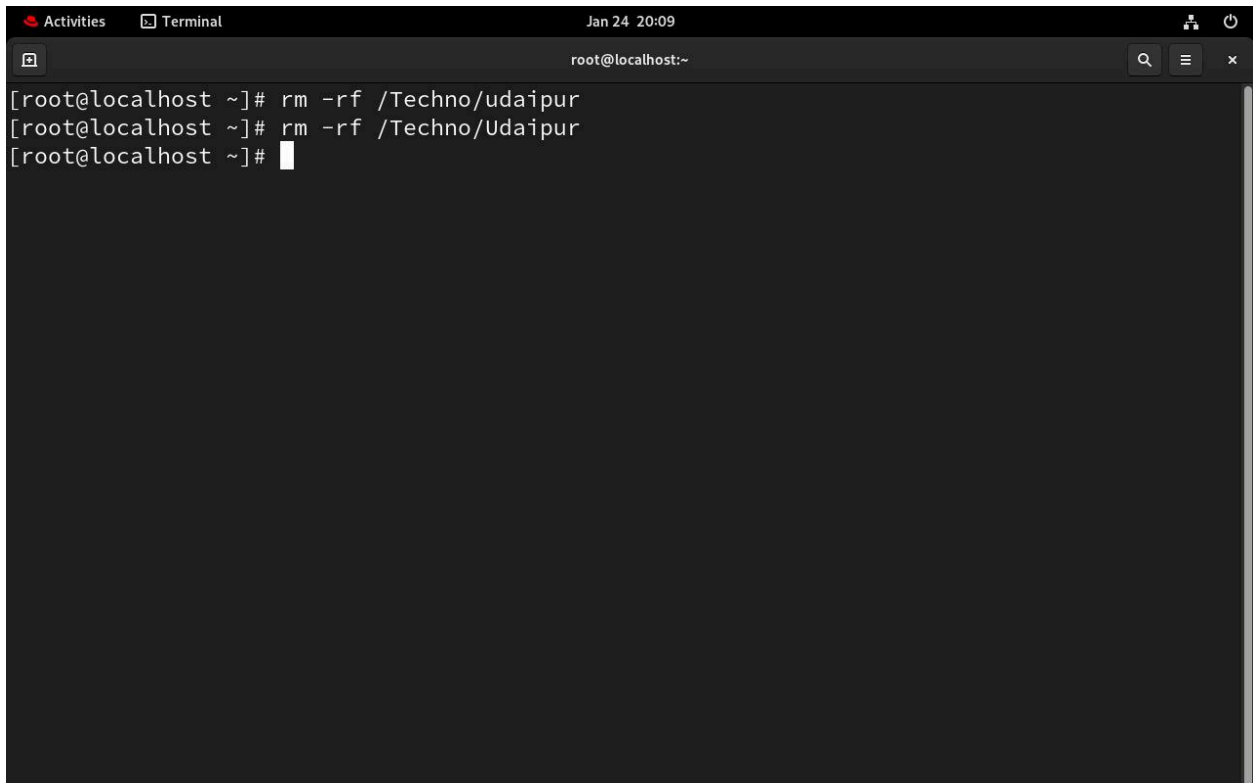
---

11. Delete the **/Techno/Udaipur** directory and its contents.

**Command:**

```
rm -rf /Techno/Udaipur
```

Q11 IMAGE :

A screenshot of a Linux terminal window. The window has a title bar with 'Activities', 'Terminal', and the date/time 'Jan 24 20:09'. The terminal shows the prompt '[root@localhost ~]#' followed by the command 'rm -rf /Techno/udaipur' on the first line, and the same command on the second line. The prompt is followed by a cursor on the third line.

```
[root@localhost ~]# rm -rf /Techno/udaipur
[root@localhost ~]# rm -rf /Techno/Udaipur
[root@localhost ~]#
```

---

**12. How can you view the manual page for the useradd command?  
From the manual page, identify which files are important for user administration.**

**Command:**

```
man useradd
```

---

**13. You have two files: Fruits.txt and Vegetables.txt, each containing related content.**

**a. Write content in fruits.txt using vi. (EX: mango, apple, kiwi, grapes, cherry).**

```
vi Fruits.txt    # Add: mango, apple, kiwi, grapes,  
cherry.
```

**b. Write content in vegetables.txt using nano. (Ex: potato, tomato, onion, chilli, garlic).**

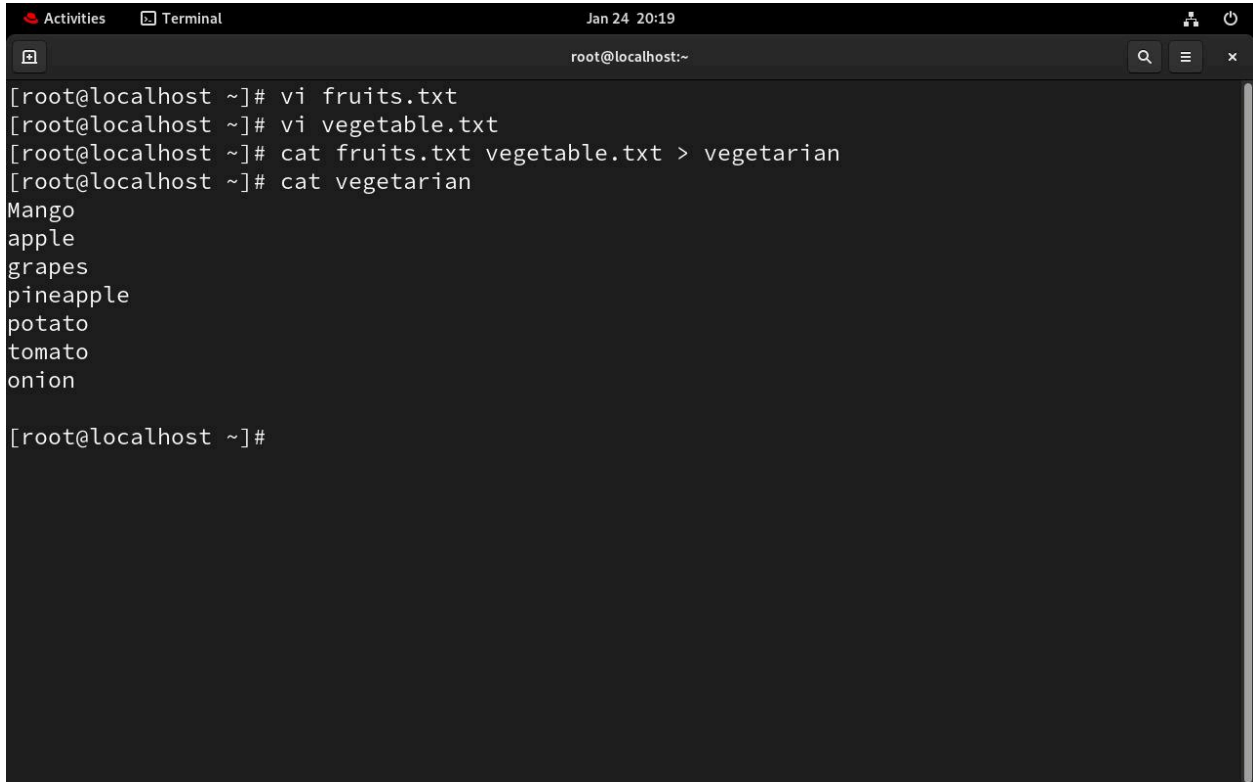
```
nano Vegetables.txt    # Add: potato, tomato, onion,  
chili, garlic.
```

**c. Write the single command to Combine the contents of both files into a single file named vegetarian and display its content.**

```
cat Fruits.txt Vegetables.txt > vegetarian
```

```
cat vegetarian
```

## Q13 IMAGE :

A terminal window titled 'Terminal' with a dark background. The window shows a series of commands and their outputs. The commands are: 'vi fruits.txt', 'vi vegetable.txt', 'cat fruits.txt vegetable.txt > vegetarian', and 'cat vegetarian'. The output of the last command is a list of fruits and vegetables: 'Mango', 'apple', 'grapes', 'pineapple', 'potato', 'tomato', and 'onion'. The terminal prompt is '[root@localhost ~]#'.

```
[root@localhost ~]# vi fruits.txt
[root@localhost ~]# vi vegetable.txt
[root@localhost ~]# cat fruits.txt vegetable.txt > vegetarian
[root@localhost ~]# cat vegetarian
Mango
apple
grapes
pineapple
potato
tomato
onion

[root@localhost ~]#
```

---

**14. . Write the command to copy all files, including related sub-files, from /var to a new location /tmp/data/. The output should be displayed during the copying process..**

**Command:**

```
cp -r /var /tmp/data
```

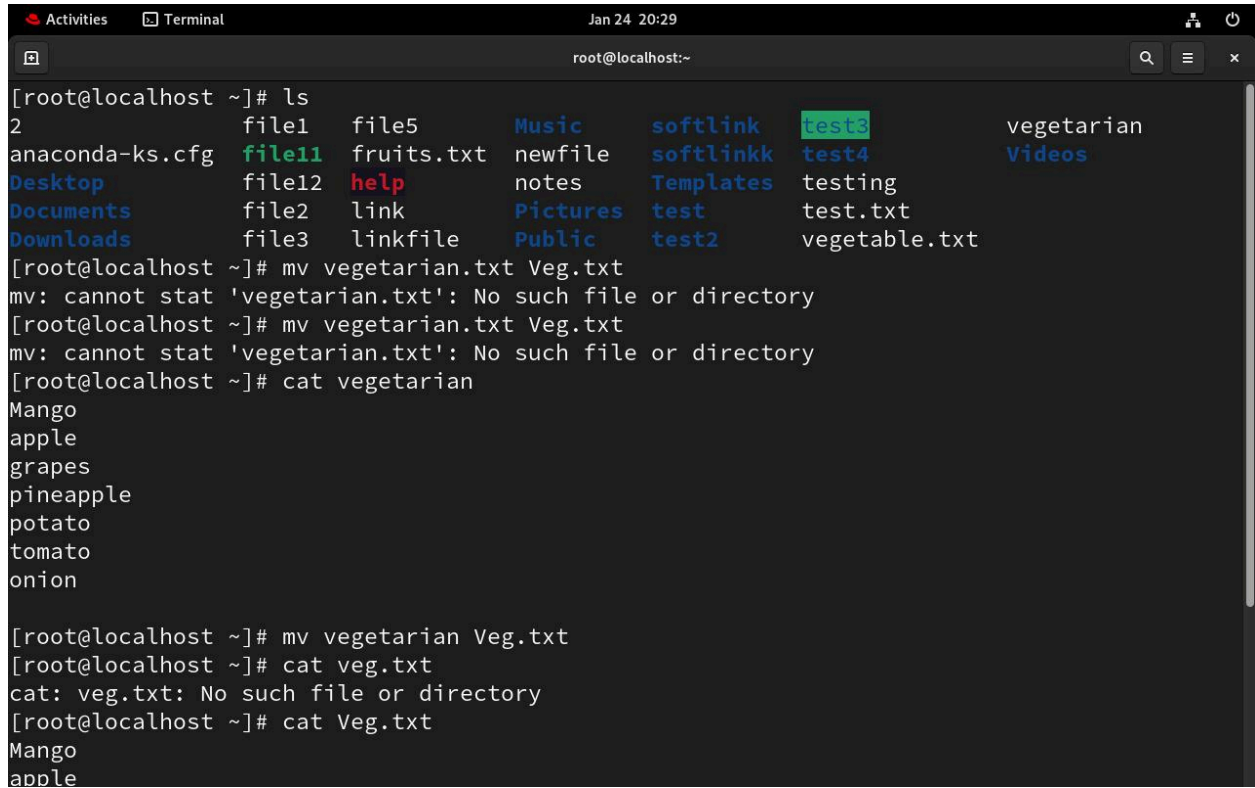
---

**15. Rename the file “Vegetrian.txt” to “Veg.txt”. Write the command.**

**Command:**

mv Vegetarian Veg.txt

Q15 IMAGE :

A terminal window titled 'Terminal' with a timestamp of 'Jan 24 20:29' and a user prompt 'root@localhost:~'. The terminal shows a series of commands and their outputs. First, 'ls' is run, displaying a directory listing with files like 'file1', 'file5', 'fruits.txt', 'newfile', 'softlink', 'test3', 'vegetarian', 'anaconda-ks.cfg', 'file11', 'help', 'notes', 'softlinkk', 'test4', 'Videos', 'Desktop', 'file12', 'link', 'Pictures', 'Templates', 'testing', 'Documents', 'file2', 'linkfile', 'Public', 'test', 'test.txt', 'Downloads', 'file3', 'test2', and 'vegetable.txt'. Then, the command 'mv vegetarian.txt Veg.txt' is entered twice, both resulting in the error 'mv: cannot stat 'vegetarian.txt': No such file or directory'. Next, 'cat vegetarian' is run, outputting a list of fruits: 'Mango', 'apple', 'grapes', 'pineapple', 'potato', 'tomato', and 'onion'. Finally, 'mv vegetarian Veg.txt' is run again successfully, followed by 'cat veg.txt' which gives the error 'cat: veg.txt: No such file or directory', and 'cat Veg.txt' which outputs 'Mango' and 'apple'.

```
[root@localhost ~]# ls
2          file1   file5   Music   softlink test3     vegetarian
anaconda-ks.cfg file11  fruits.txt newfile softlinkk test4     Videos
Desktop    file12 help    notes   Templates testing
Documents  file2  link    Pictures test     test.txt
Downloads  file3  linkfile Public  test2    vegetable.txt

[root@localhost ~]# mv vegetarian.txt Veg.txt
mv: cannot stat 'vegetarian.txt': No such file or directory
[root@localhost ~]# mv vegetarian.txt Veg.txt
mv: cannot stat 'vegetarian.txt': No such file or directory
[root@localhost ~]# cat vegetarian
Mango
apple
grapes
pineapple
potato
tomato
onion

[root@localhost ~]# mv vegetarian Veg.txt
[root@localhost ~]# cat veg.txt
cat: veg.txt: No such file or directory
[root@localhost ~]# cat Veg.txt
Mango
apple
```

16. Open the file “/etc/passwd and locate the following lines using less and more.

**Commands:**

a. Search "Root" using **less**:

**less /etc/passwd**

**/SearchText**

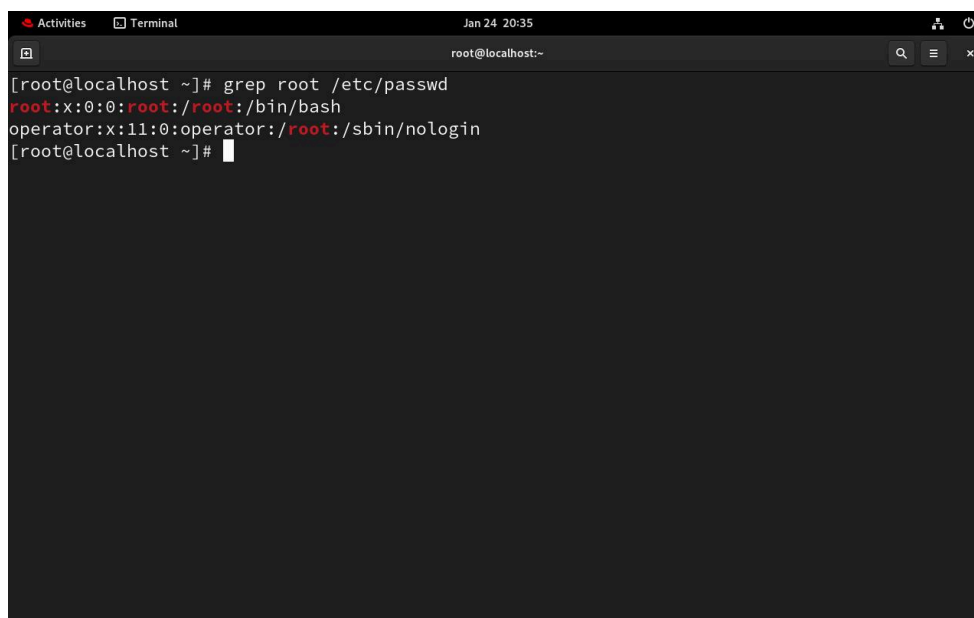
b. Search "root" using **grep**:

```
grep root /etc/passwd
```

### Difference:

**less** scrolls interactively; **grep** filters results directly.

Q16 IMAGE :

A terminal window titled 'Terminal' with a timestamp of 'Jan 24 20:35'. The prompt is 'root@localhost:~'. The command '[root@localhost ~]# grep root /etc/passwd' has been executed. The output shows two lines: 'root:x:0:0:root:/root:/bin/bash' and 'operator:x:11:0:operator:/root:/sbin/nologin'. The word 'root' is highlighted in red in the prompt and the first line of output. The prompt returns to '[root@localhost ~]#'.

---

## 17. Display specific lines from files:

### Commands:

a. Display the top 7th line of the /etc/passwd file.

```
sed -n '7p' /etc/passwd
```

b. Display the last 3 lines of the /etc/group file.

```
tail -n 3 /etc/group
```

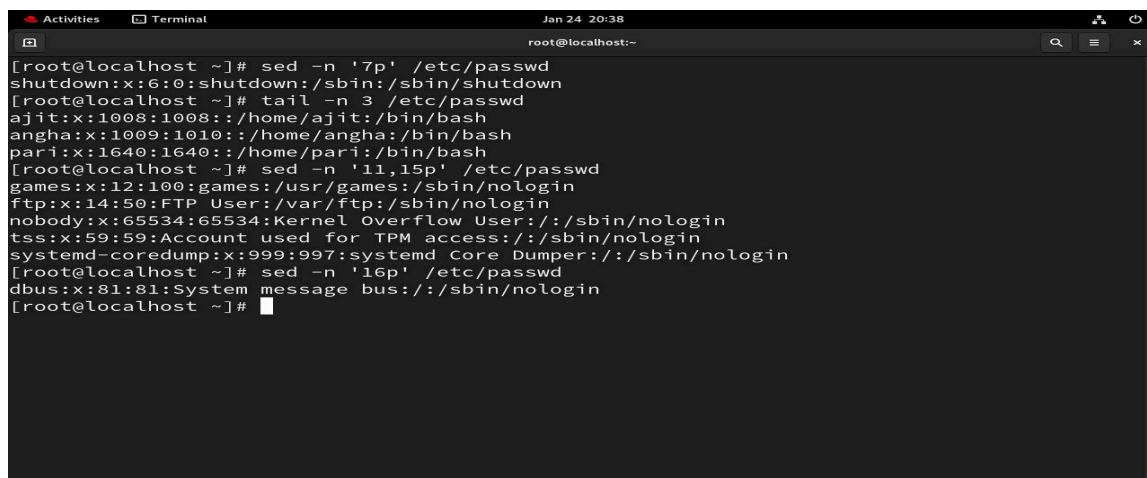
c. Display the lines 11th to 15th from the /etc/shadow file using a pipeline

```
sed -n '11,15p' /etc/shadow
```

d. Display only the 16th line of the /etc/passwd file

```
sed -n '16p' /etc/passwd
```

Q17 IMAGE :

A terminal window titled 'Terminal' with a date and time of 'Jan 24 20:38'. The prompt is 'root@localhost:~'. The user enters the command '[root@localhost ~]# sed -n '7p' /etc/passwd', which outputs 'shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown'. The user then enters '[root@localhost ~]# tail -n 3 /etc/passwd', which outputs three lines of user entries. Finally, the user enters '[root@localhost ~]# sed -n '11,15p' /etc/passwd', which outputs five lines of user entries. The prompt returns to '[root@localhost ~]#'.

```
[root@localhost ~]# sed -n '7p' /etc/passwd
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
[root@localhost ~]# tail -n 3 /etc/passwd
ajit:x:1008:1008::/home/ajit:/bin/bash
angha:x:1009:1010::/home/angha:/bin/bash
pari:x:1640:1640::/home/pari:/bin/bash
[root@localhost ~]# sed -n '11,15p' /etc/passwd
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
tss:x:59:59:Account used for TPM access:/:/sbin/nologin
systemd-coredump:x:999:997:systemd Core Dumper:/:/sbin/nologin
[root@localhost ~]# sed -n '16p' /etc/passwd
dbus:x:81:81:System message bus:/:/sbin/nologin
[root@localhost ~]#
```

## 18. Search with **grep** and save matches:

### Commands:

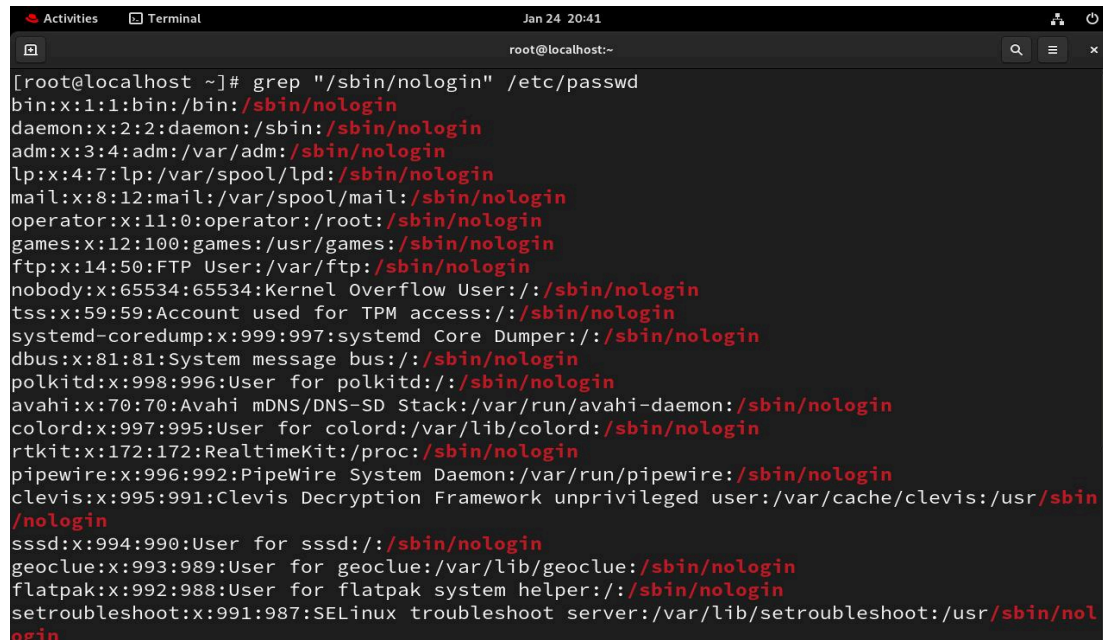
a. a. Write a command to match and display lines containing the word /sbin/nologin

```
grep "/sbin/nologin" /etc/passwd
```

b. . Write a command to match and display lines containing the multiple words (root, sbin, and /home) simultaneously, ignoring typographical case errors. Save the output for all three matches into the file /root/test. Without losing data. :

```
grep -iE "root|sbin|/home" /etc/passwd > /root/test
```

Q18 IMAGE :

A terminal window titled 'Activities Terminal' with a timestamp of 'Jan 24 20:41'. The prompt is 'root@localhost:~'. The command entered is '[root@localhost ~]# grep "/sbin/nologin" /etc/passwd'. The output lists system users and their shell paths, all of which are '/sbin/nologin'. The users listed are bin, daemon, adm, lp, mail, operator, games, ftp, nobody, tss, systemd-coredump, dbus, polkitd, avahi, colord, rtkit, pipewire, clevis, sssd, geoclue, flatpak, and setroubleshoot. The output is color-coded with red for the shell path and green for the username.

```
[root@localhost ~]# grep "/sbin/nologin" /etc/passwd
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
tss:x:59:59:Account used for TPM access:/:/sbin/nologin
systemd-coredump:x:999:997:systemd Core Dumper:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:998:996:User for polkitd:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
colord:x:997:995:User for colord:/var/lib/colord:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
pipewire:x:996:992:PipeWire System Daemon:/var/run/pipewire:/sbin/nologin
clevis:x:995:991:Clevis Decryption Framework unprivileged user:/var/cache/clevis:/usr/sbin/nologin
sssd:x:994:990:User for sssd:/:/sbin/nologin
geoclue:x:993:989:User for geoclue:/var/lib/geoclue:/sbin/nologin
flatpak:x:992:988:User for flatpak system helper:/:/sbin/nologin
setroubleshoot:x:991:987:SELinux troubleshoot server:/var/lib/setroubleshoot:/usr/sbin/nologin
```

---

19. Replace text using **sed** and count lines with "success":

**Commands:**

a. . Write the command to replace the word localhost with localhost.localhost in the file /etc/hosts without opening the file in an editor

```
sed -i 's/localhost/localhost.localhost/g' /etc/hosts
```

b. The /var/log/audit/audit.log file contains audit log messages, some of which include the word "success." Write the command to count how many lines contain the word success.



```
grep -c "success" /var/log/audit/audit.log
```

---

**20. . Create a directory named “demo” on “/root”:**

**Commands:**

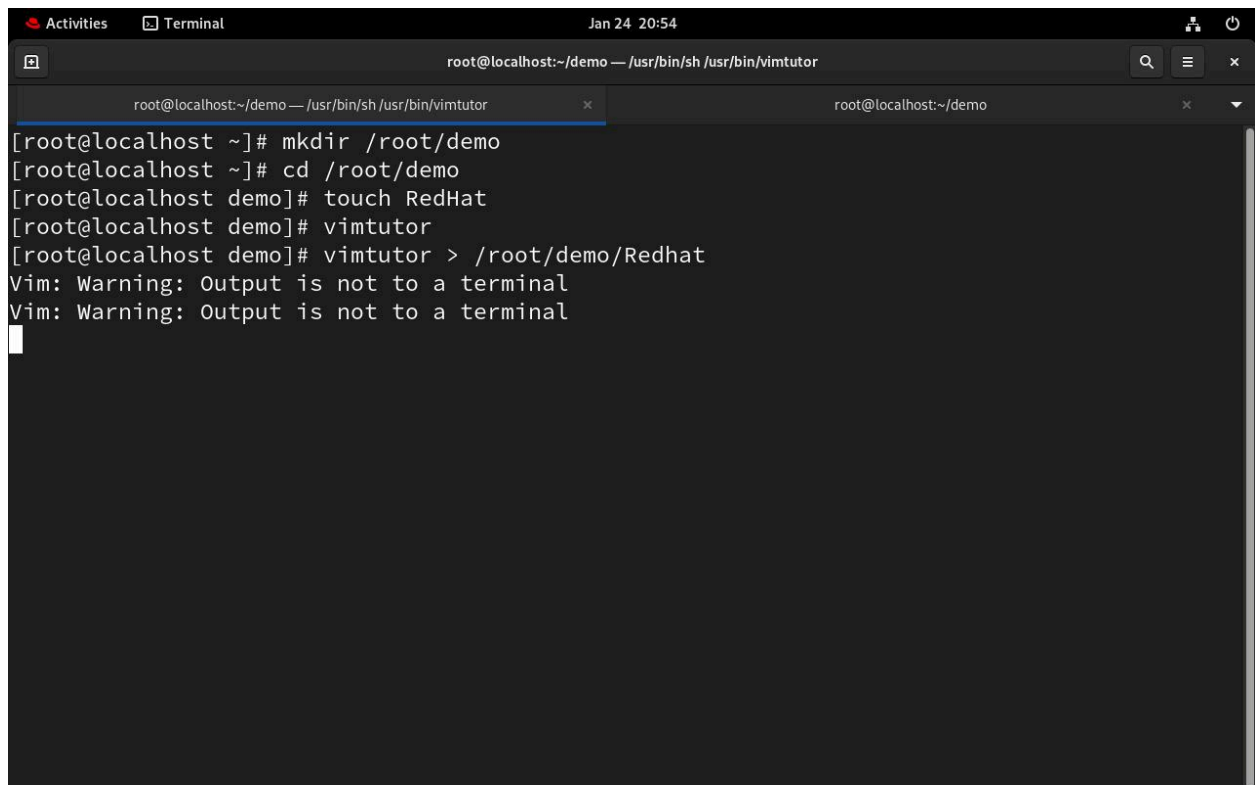
**a. Create a file “RedHat” under the “demo” directory:**

```
mkdir /root/demo
```

```
touch /root/demo/RedHat
```

```
vimtutor > /root/demo/RedHat
```

Q20 (a) Image :

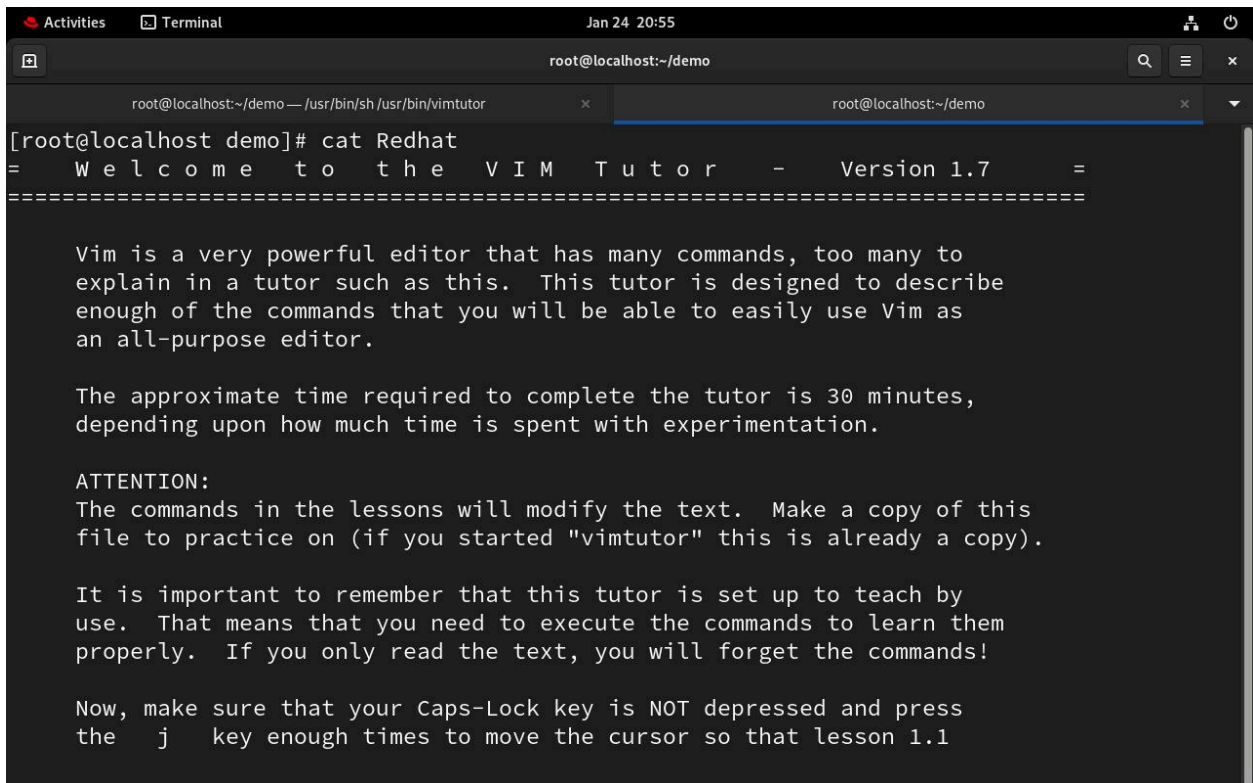


```
Activities Terminal Jan 24 20:54
root@localhost:~/demo — /usr/bin/sh /usr/bin/vimtutor
root@localhost:~/demo — /usr/bin/sh /usr/bin/vimtutor x root@localhost:~/demo x
[root@localhost ~]# mkdir /root/demo
[root@localhost ~]# cd /root/demo
[root@localhost demo]# touch RedHat
[root@localhost demo]# vimtutor
[root@localhost demo]# vimtutor > /root/demo/Redhat
Vim: Warning: Output is not to a terminal
Vim: Warning: Output is not to a terminal
```

**b. Run the command “vimtutor” and save the output to the “RedHat” file:**

Command : `vimtutor > /root/demo/RedHat`

Q20 (b) image :



```
root@localhost:~/demo — /usr/bin/sh /usr/bin/vimtutor
[roo@localhost demo]# cat Redhat
=  Welcome to the VIM Tutor - Version 1.7  =
=====

Vim is a very powerful editor that has many commands, too many to
explain in a tutor such as this.  This tutor is designed to describe
enough of the commands that you will be able to easily use Vim as
an all-purpose editor.

The approximate time required to complete the tutor is 30 minutes,
depending upon how much time is spent with experimentation.

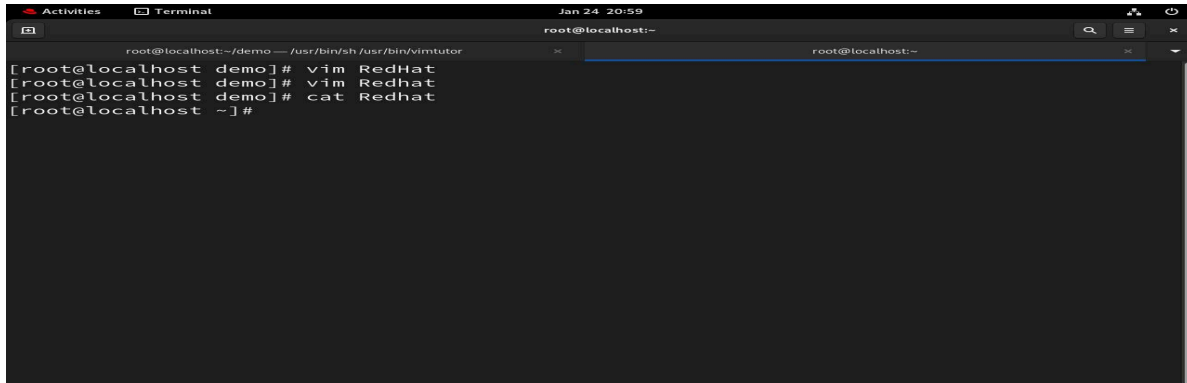
ATTENTION:
The commands in the lessons will modify the text.  Make a copy of this
file to practice on (if you started "vimtutor" this is already a copy).

It is important to remember that this tutor is set up to teach by
use.  That means that you need to execute the commands to learn them
properly.  If you only read the text, you will forget the commands!

Now, make sure that your Caps-Lock key is NOT depressed and press
the  j  key enough times to move the cursor so that lesson 1.1
```

**c. Now create a soft link of “RedHat” to “/etc/” location.**

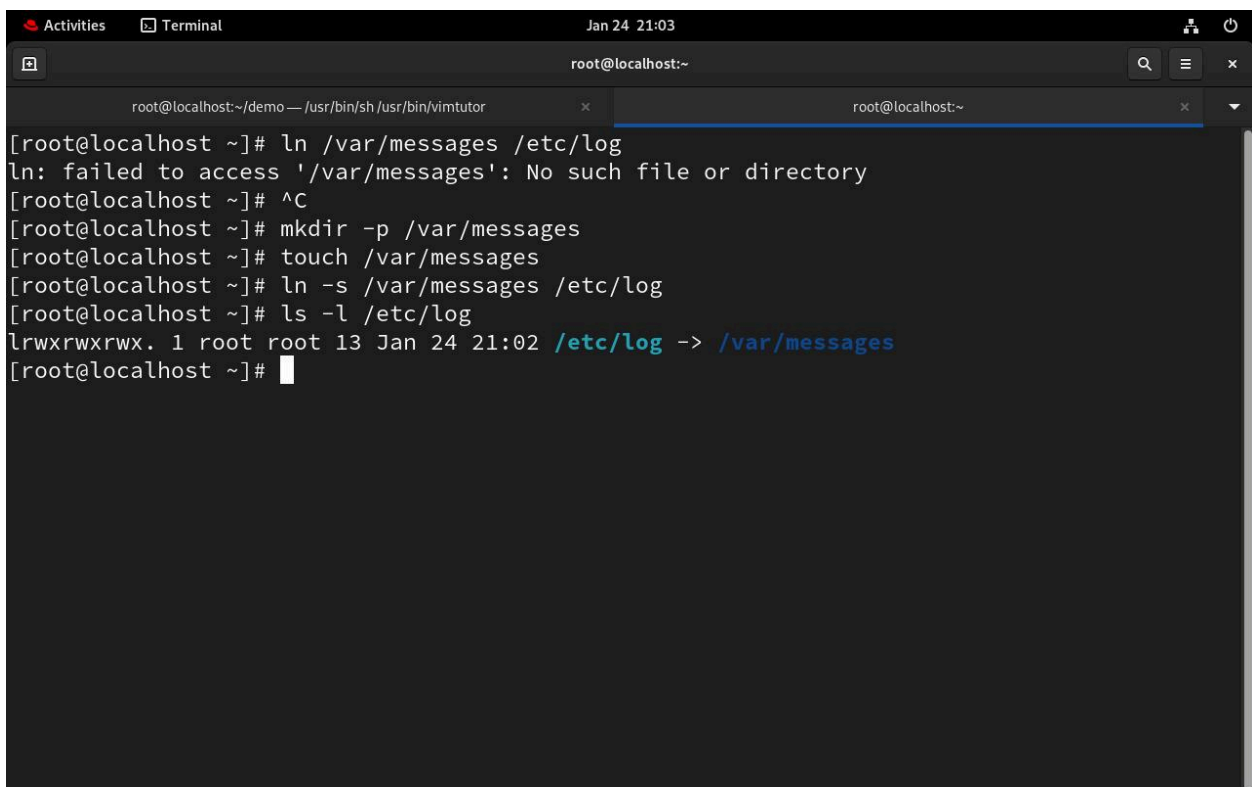
Command : `ln -s /root/demo/RedHat /etc/RedHat`



```
root@localhost:~# vim RedHat
root@localhost:demo]# vim RedHat
root@localhost:demo]# cat RedHat
root@localhost:~]#
```

**D . Create a hard link of /var/log/messages to /etc/log.**

Command : `ln /var/log/messages /etc/log`



```
root@localhost:~]# ln /var/messages /etc/log
ln: failed to access '/var/messages': No such file or directory
root@localhost:~]# ^C
root@localhost:~]# mkdir -p /var/messages
root@localhost:~]# touch /var/messages
root@localhost:~]# ln -s /var/messages /etc/log
root@localhost:~]# ls -l /etc/log
lrwxrwxrwx. 1 root root 13 Jan 24 21:02 /etc/log -> /var/messages
root@localhost:~]#
```