**1. List all the files and subdirectories of the directory /bin.**

ls /bin

**2. List all the files including hidden files in your current directory.**

ls -a

**3. List all the files starting with the letter ‘r’ in your current directory.**

ls r\*

**4. List all the files having three characters in their names, from your current directory.**

ls ???

**5. List all the files with the extension .doc in your current directory.**

ls \*.doc

**6. List all the files having the first letter of their name within the range ‘l’ to ‘s’, from your current directory.**

ls [l-s]\*

**7. Create a file text1 and read its input from the keyboard.**

touch text1

cat > text1

**8. Copy the contents of file text1 to another file text2.**

cp text1 text2

**9. Append the contents of file text2 to file text1.**

cat text2 >> text1

**10. Count the number of files in the current directory.**

ls | wc -l

**11. Display the output of the `ls -l` command to a file and on the output screen.**

ls -l | tee ls\_output.txt

**12. From file text1, print all lines starting from the 10th line.**

tail -n +10 text1

**13. Find the number of users currently logged on to the system.**

who | wc -l

**14. Delete all the files with names starting with "tmp".**

rm tmp\*

**Section 2:**

**1. Count the total number of words in file `text1`.**

wc -w text1

**Output Snip:**

**2. List the contents of the `ls` command page-wise.**

ls | less

**Output Snip:**

(Displays the contents of the current directory page by page using the 'less' command)

# You can navigate through the list using arrow keys or other navigation keys.

**3. Create a file `FILE2` with some text in it. Increase the number of hard links to the file `FILE2` to 3 and check the inode number and link count for those names.**

echo "This is some text." > FILE2

ln FILE2 FILE2\_link1

ln FILE2 FILE2\_link2

ln FILE2 FILE2\_link3

ls -i FILE2\*

**Output Snip:**

Note: In the output, `123456` represents the inode number of the file.

**4. Using one single command, display the output of "who" and "pwd" commands.**

who && pwd

**Output Snip:**

username tty1 2023-09-28 09:00 (:0)

/home/username

**5. Display the system date in the following format: "Today is Friday, 17 May 96".**

date "+Today is %A, %d %B %y"

**Output Snip:**

**6. Display the following text message on the monitor screen.**

echo "Deposited \$100 to your account"

**Output Snip:**

Deposited $100 to your account

**7. Display the following message on the monitor.**

echo "The long listing of my home dir ... is ..."

ls -l $(pwd)

**Output Snip:**

The long listing of my home dir ... is ...

total 4

drwxr-xr-x 2 username users 4096 Sep 28 09:00 Desktop

(other files and directories in the home directory)

Note: Replace `username` with your actual username in the above commands.

**Section 3:**

To perform the tasks using the vi editor, follow these steps:

a) Create a file "Data1.txt" and edit it:

vi Data1.txt

This opens the "Data1.txt" file in vi editor.

b) Save the file and exit from the vi editor:

- Press `Esc` to ensure you are in normal mode.

- Type `:w` and press `Enter` to save the file.

- Type `:q` and press `Enter` to exit the vi editor.

c) Open the vi editor without specifying a file name:

vi

This opens vi editor without a specific file.

d) Write some text and save it to a file "MyData2.txt":

- Type your desired text.

- Press `Esc` to ensure you are in normal mode.

- Type `:w MyData2.txt` and press `Enter` to save the text to "MyData2.txt".

- Type `:q` and press `Enter` to exit the vi editor.

e) Repeat point (c) but after writing some text, don't save and just exit "vi":

- Type your desired text.

- Press `Esc` to ensure you are in normal mode.

- Type `:q!` and press `Enter`. The `!` is used to forcefully exit without saving changes.

Now, you've created and edited files "Data1.txt" and "MyData2.txt" using the vi editor as specified.