

Ananya Sahay

20BCE2790

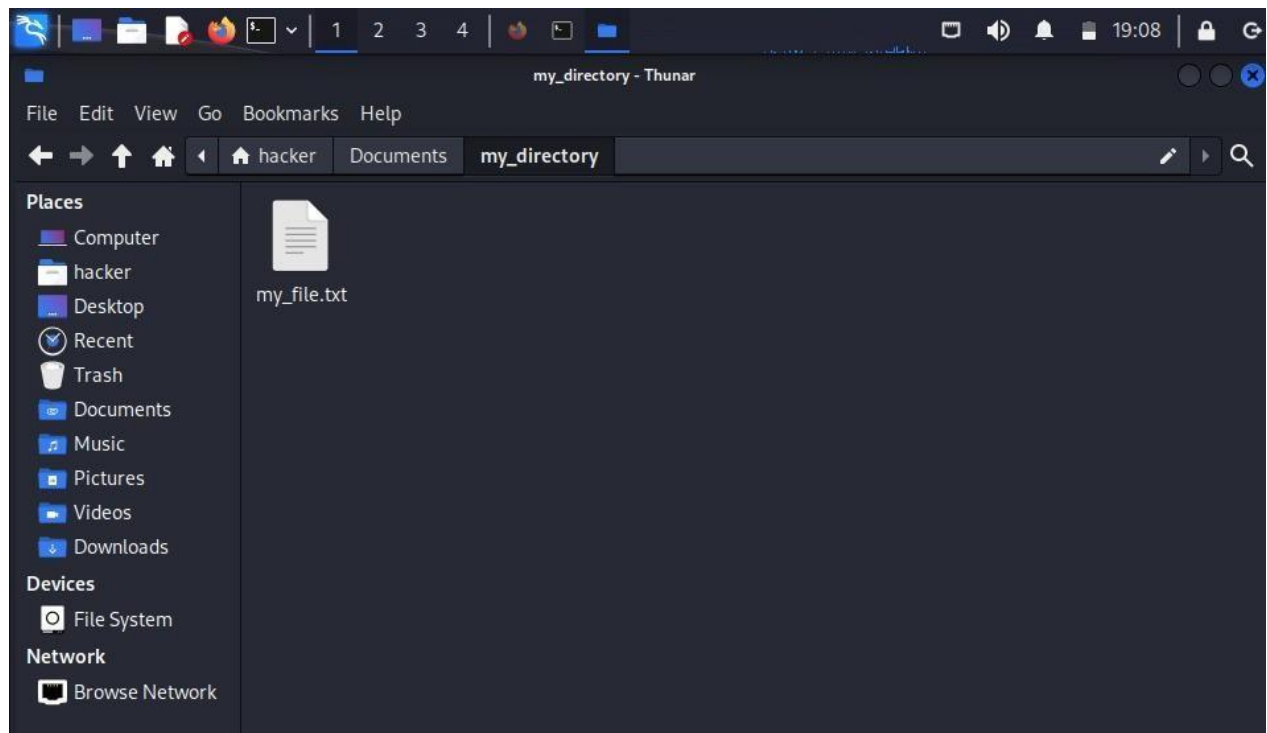
Assignment: Bash Shell BasicsTask

1: File and Directory Manipulation

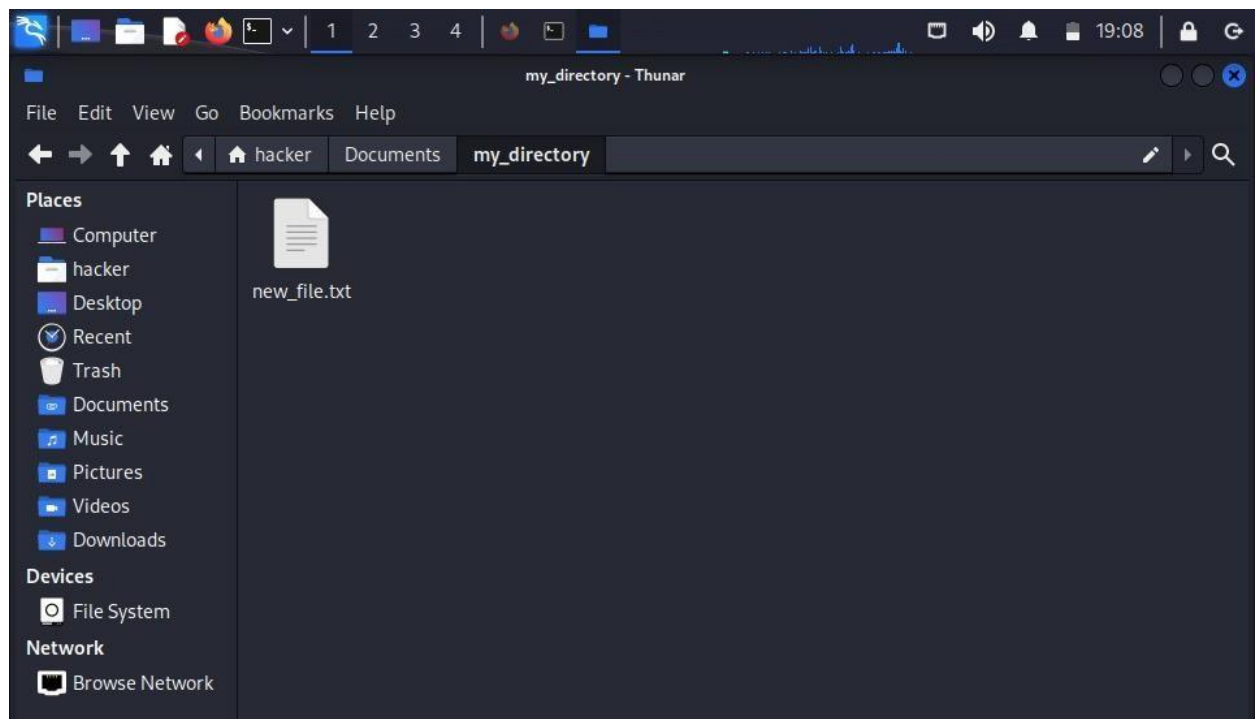
1. Create a directory called "my_directory".
2. Navigate into the "my_directory".
3. Create an empty file called "my_file.txt".
4. List all the files and directories in the current directory.
5. Rename "my_file.txt" to "new_file.txt".
6. Display the content of "new_file.txt" using a pager tool of your choice.
7. Append the text "Hello, World!" to "new_file.txt".
8. Create a new directory called "backup" within "my_directory".
9. Move "new_file.txt" to the "backup" directory.
10. Verify that "new_file.txt" is now located in the "backup" directory.
11. Delete the "backup" directory and all its contents.

```
hacker@kali: ~/Documents/my_directory
File Actions Edit View Help
(hacker@kali)-[~]
$ cd Documents/
(hacker@kali)-[~/Documents]
$ mkdir my_directory
(hacker@kali)-[~/Documents]
$ cd my_directory
(hacker@kali)-[~/Documents/my_directory]
$ touch my_file.txt
(hacker@kali)-[~/Documents/my_directory]
$ ls
my_file.txt
(hacker@kali)-[~/Documents/my_directory]
$ mv my_file.txt new_file.txt
(hacker@kali)-[~/Documents/my_directory]
$ cat new_file.txt
Hello, World!
(hacker@kali)-[~/Documents/my_directory]
$ echo "new_file.txt" > new_file.txt
(hacker@kali)-[~/Documents/my_directory]
$ cat new_file.txt
new_file.txt
(hacker@kali)-[~/Documents/my_directory]
$ mkdir backup
(hacker@kali)-[~/Documents/my_directory]
$ mv new_file.txt backup
(hacker@kali)-[~/Documents/my_directory]
$ rm -r backup
(hacker@kali)-[~/Documents/my_directory]
$
```

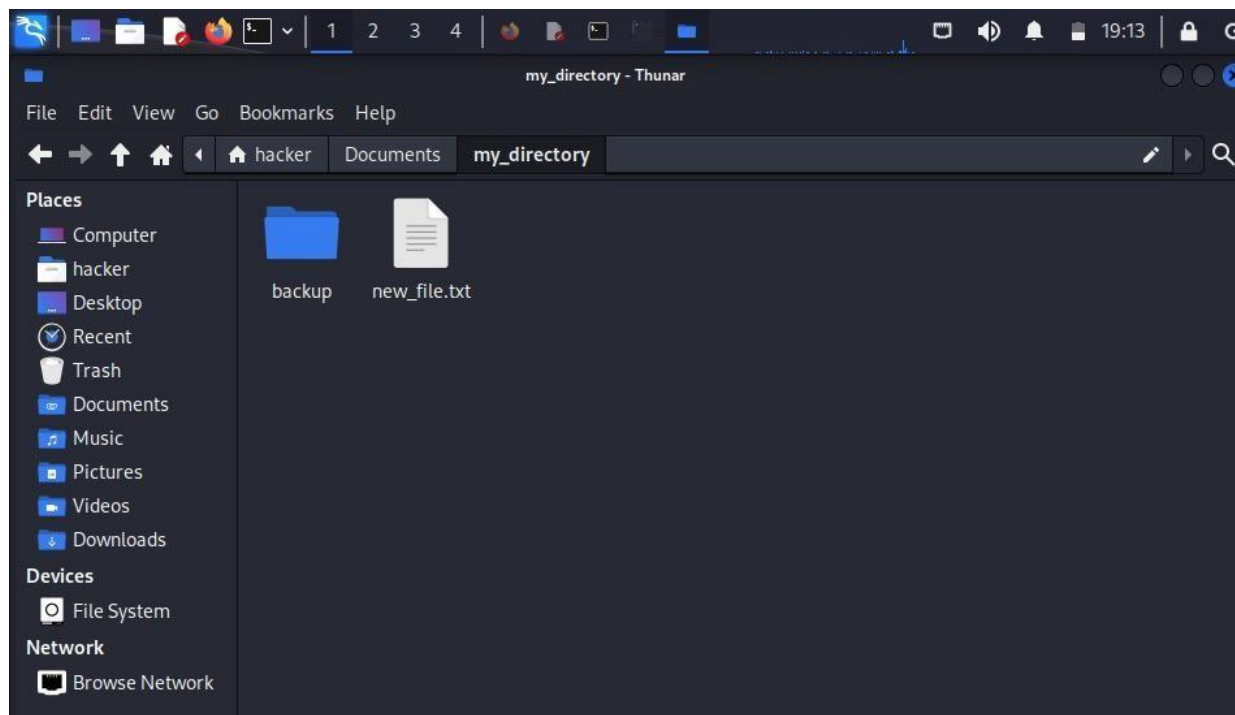
Creating a file name my_file.txt



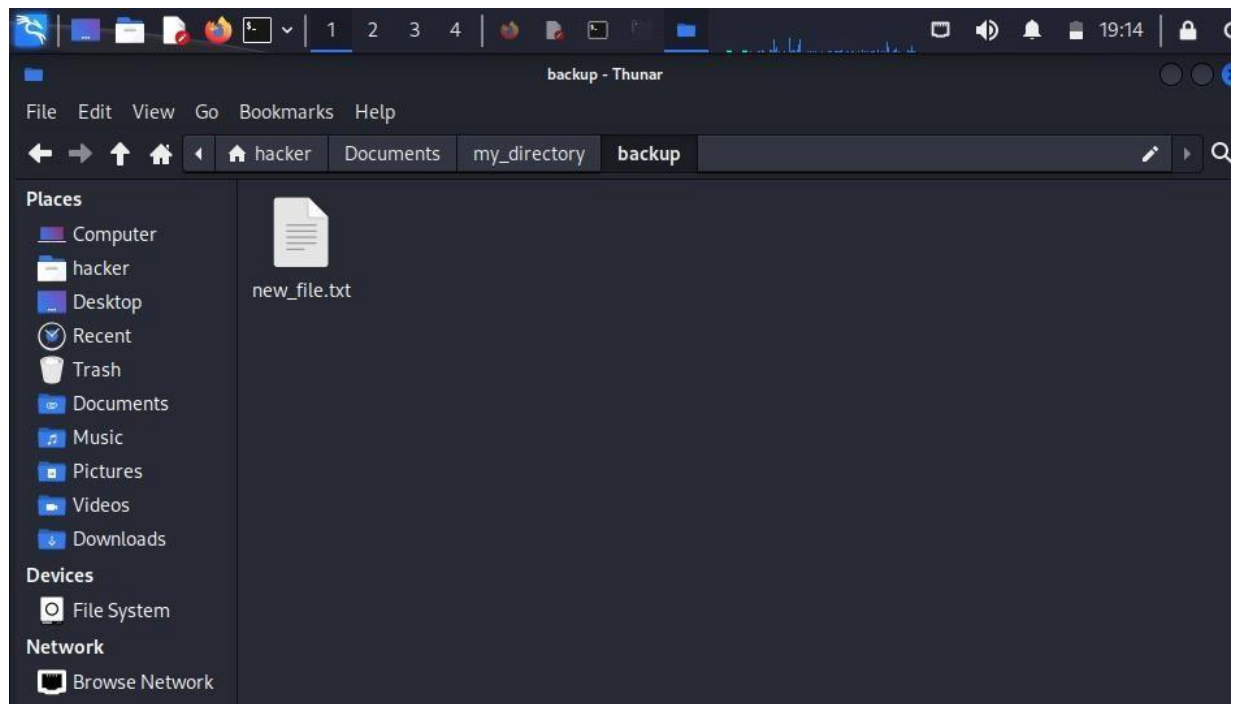
mv my_file.txt to new_file.txt



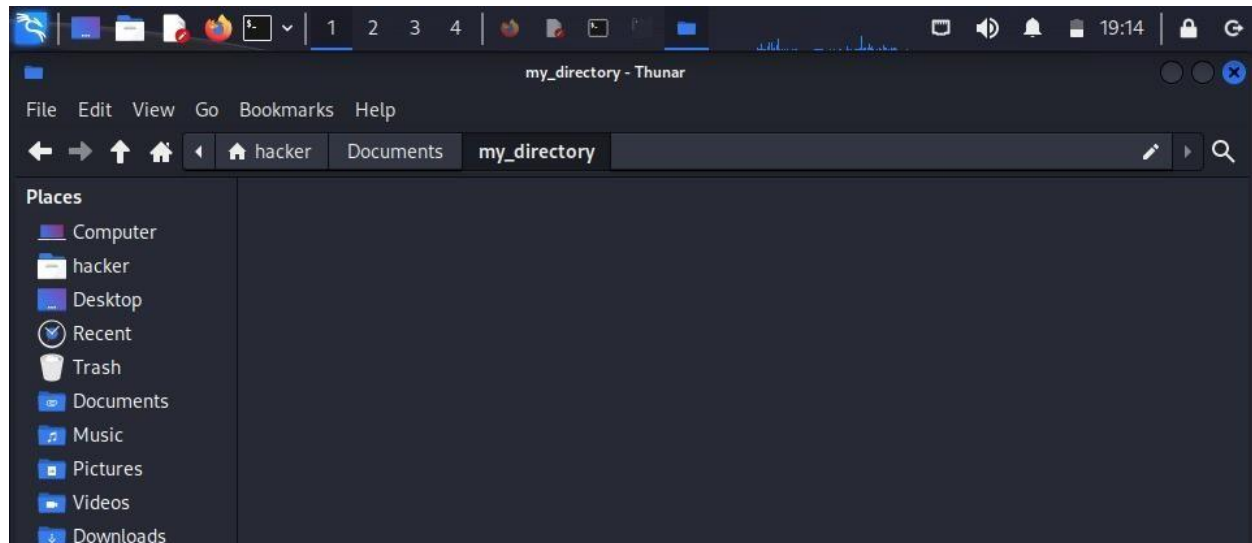
Creating a new directory with name backup in the same directory name
my_diractory



Move the new_file.txt to backup directory



Remove the backup directory and its contents



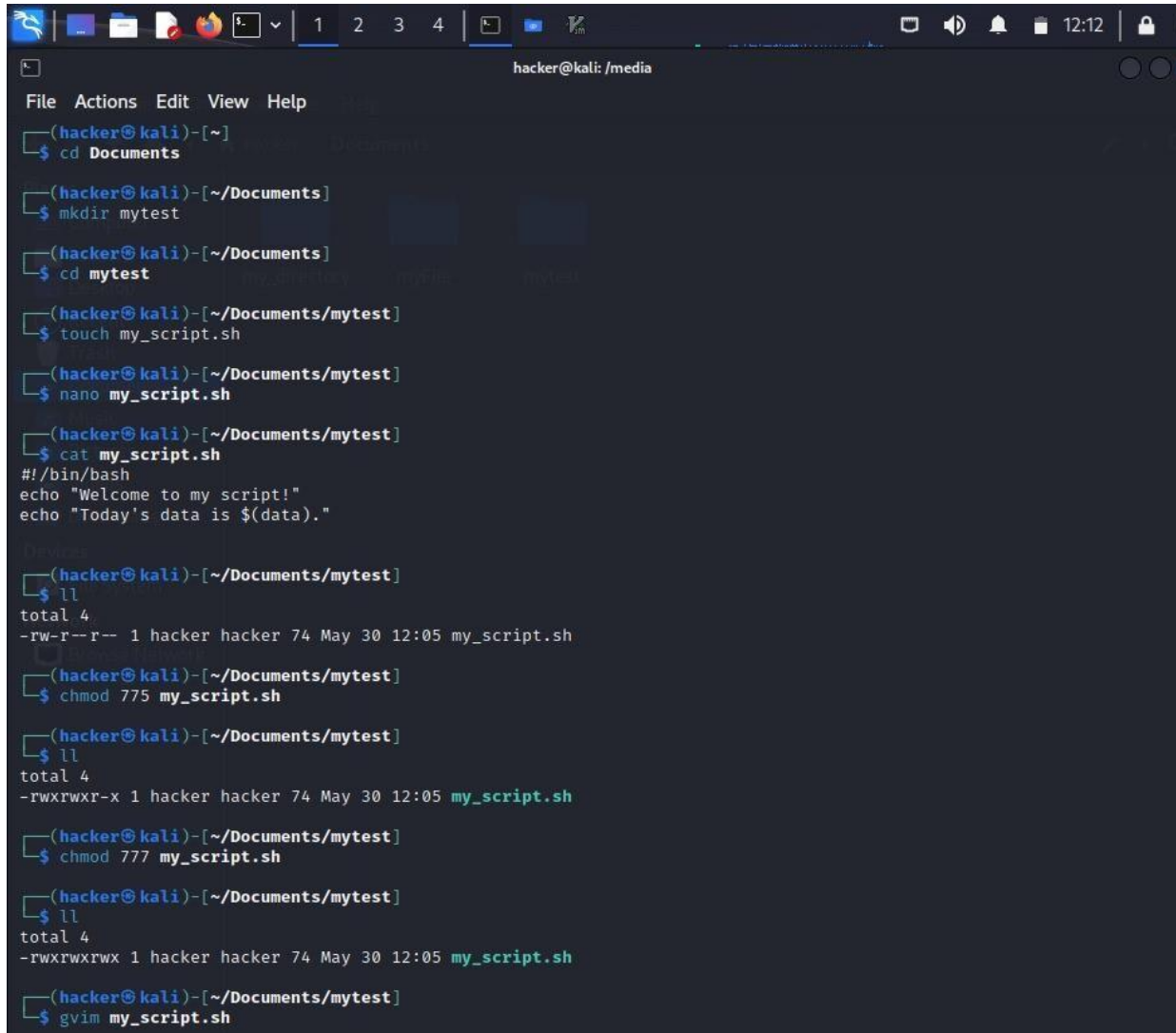
Task 2: Permissions and Scripting

- Create a new file called "my_script.sh".
 - Edit "my_script.sh" u

to

```
#!/bin/bash
echo "Welcome to my script!" echo
"Today's date is $(date)."Save and
exit the file.
```

- Make "my_script.sh" executable.
- Run "my_script.sh" and verify that the output matches the expected result.



```
hacker@kali: /media
File Actions Edit View Help
(hacker@kali)-[~]
$ cd Documents
(hacker@kali)-[~/Documents]
$ mkdir mytest
(hacker@kali)-[~/Documents]
$ cd mytest
(hacker@kali)-[~/Documents/mytest]
$ touch my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ nano my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ cat my_script.sh
#!/bin/bash
echo "Welcome to my script!"
echo "Today's data is ${data}."
(hacker@kali)-[~/Documents/mytest]
$ ll
total 4
-rw-r--r-- 1 hacker hacker 74 May 30 12:05 my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ chmod 775 my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ ll
total 4
-rwxrwxr-x 1 hacker hacker 74 May 30 12:05 my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ chmod 777 my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ ll
total 4
-rwxrwxrwx 1 hacker hacker 74 May 30 12:05 my_script.sh
(hacker@kali)-[~/Documents/mytest]
$ gvim my_script.sh
```

```
hacker@kali: /media

File Actions Edit View Help

(hacker@kali)~[~/Documents/mytest]
$ cat my_script.sh
#!/bin/bash
echo "Welcome to my script!"
echo "Today's data is $(data)."
```

```
(hacker@kali)~[~/Documents/mytest]
$ ll
total 4
-rwxrwxrwx 1 hacker hacker 74 May 30 12:05 my_script.sh
```

```
(hacker@kali)~[~/Documents/mytest]
$ my_script.sh
my_script.sh: command not found
```

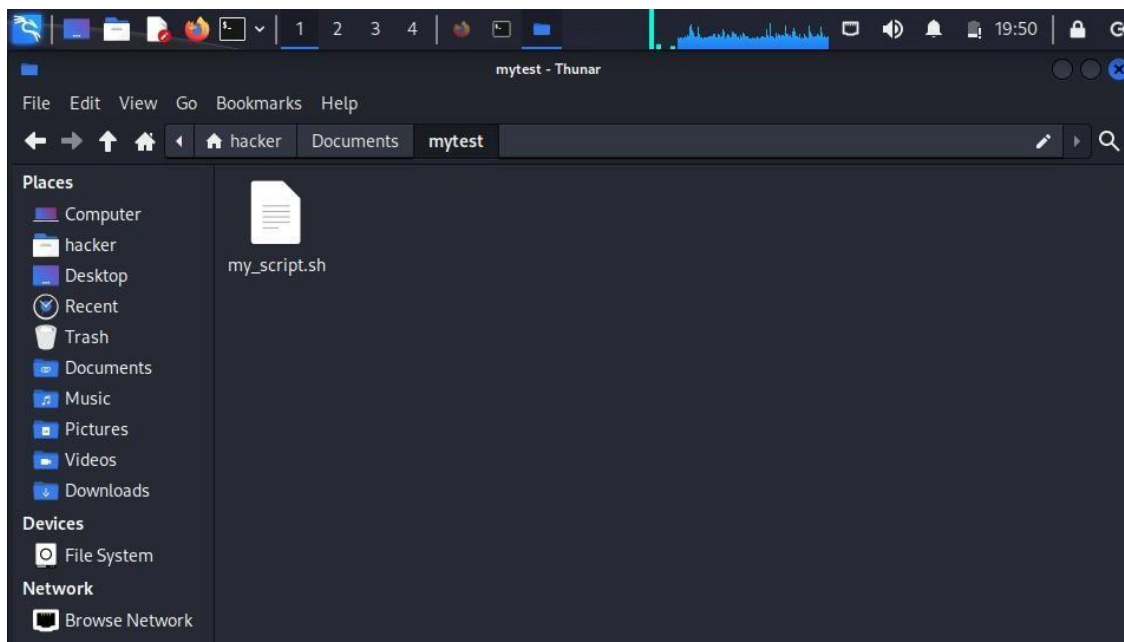
```
(hacker@kali)~[~/Documents/mytest]
$ pwd
/home/hacker/Documents/mytest
```

```
(hacker@kali)~[~/Documents/mytest]
$ which mv
/usr/bin/mv
```

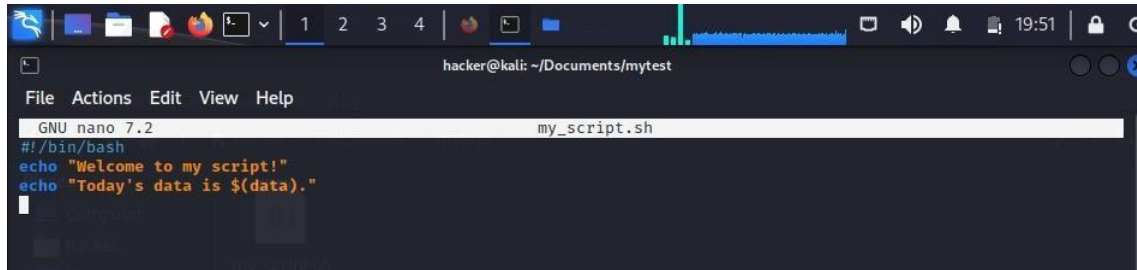
```
(hacker@kali)~[~/Documents/mytest]
$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/usr/games
```

```
(hacker@kali)~[~/Documents/mytest]
$ ./my_script.sh
Welcome to my script!
./my_script.sh: line 3: data: command not found
Today's data is .
```

Create a new file with name my_script.sh



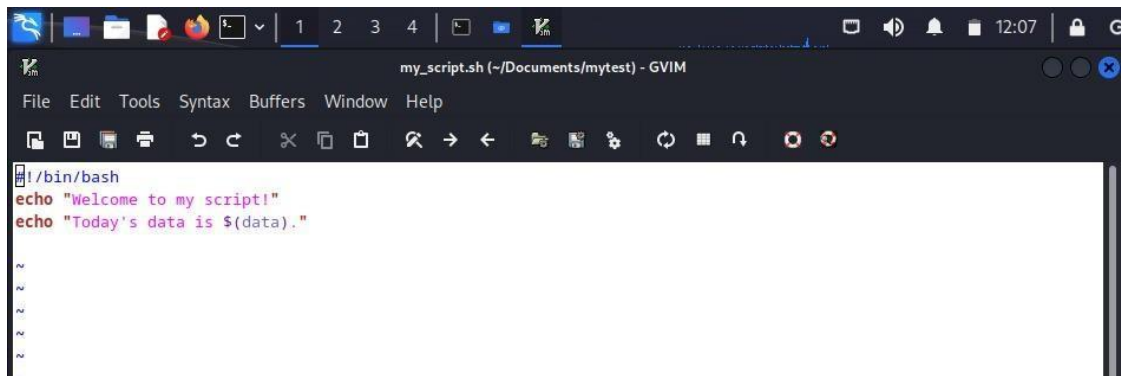
nano command window



A screenshot of a terminal window running the nano text editor. The window title is "hacker@kali: ~/Documents/mytest". The menu bar includes "File", "Actions", "Edit", "View", and "Help". The status bar at the bottom shows "GNU nano 7.2" and "my_script.sh". The editor contains the following text:

```
#!/bin/bash
echo "Welcome to my script!"
echo "Today's data is ${data}."
```

Running the file "my_script.sh" and verifying that the output matches with the expected output.



A screenshot of a terminal window running the GVIM text editor. The window title is "my_script.sh (~/.Documents/mytest) - GVIM". The menu bar includes "File", "Edit", "Tools", "Syntax", "Buffers", "Window", and "Help". The status bar at the bottom shows "my_script.sh (~/.Documents/mytest) - GVIM". The editor contains the following text:

```
#!/bin/bash
echo "Welcome to my script!"
echo "Today's data is ${data}."
```


Task 3: Command Execution and Pipelines

- List all the processes running on your system using the "ps" command.
- Use the "grep" command to filter the processes list and display only the processes with "bash" in their name.
- Use the "wc" command to count the number of lines in the filtered output.

cat – to display the content of the file

cat file.txt | sort – to sort the names in alphabetical order

cat file.txt | sort | uniq – to sort the names in alphabetical order and the names should not repeat

cat file.txt | sort > copy.txt – copy the file content from one file to otherhead
– top of the content in the file

tail – bottom of the content in the file

wc – it us a word count

wc -l – number of lines in the list or file

```
hacker@kali: ~/Documents/my_directory
File Actions Edit View Help
(hacker@kali)-[~]
$ cd Documents
(hacker@kali)-[~/Documents]
$ cd my_directory
(hacker@kali)-[~/Documents/my_directory]
$ touch file.txt
(hacker@kali)-[~/Documents/my_directory]
$ nano file.txt
(hacker@kali)-[~/Documents/my_directory]
$ cat file.txt
land rover
ford
tata rover
Frod
maruthi suzuki
nexon
tata safari
Tata
Nexon
Land rover
(hacker@kali)-[~/Documents/my_directory]
$ cat file.txt | sort
ford
Frod
land rover
Land rover
maruthi suzuki
nexon
Nexon
Tata
tata rover
tata safari
```

```
hacker@kali: ~/Documents/my_directory
File Actions Edit View Help
(hacker@kali)-[~/Documents/my_directory]
$ cat file.txt | sort | uniq
Frod
Frod
Land rover
Land rover
maruthi suzuki
nixon
Nexon
Tata
tata rover
tata safarile.txt

(hacker@kali)-[~/Documents/my_directory]
$ cat file.txt | sort > copy.txt

(hacker@kali)-[~/Documents/my_directory]
$ cat copy.txt
Frod
Frod
Land rover
Land rover
maruthi suzuki
nixon
Nexon
Tata
tata rover
tata safari
nixon

(hacker@kali)-[~/Documents/my_directory]
$ cat copy.txt | head -5
Frod
Frod
Land rover
Land rover
maruthi suzuki

(hacker@kali)-[~/Documents/my_directory]
$ cat copy.txt | tail -4
Nexon
Tata
tata rover
tata safari

(hacker@kali)-[~/Documents/my_directory]
$ cat copy.txt | wc
 10   15   87

(hacker@kali)-[~/Documents/my_directory]
$ cat copy.txt | wc -l
10
```

```
hacker@kali: ~/Documents/my_directory

File Actions Edit View Help

(hacker@kali)-[~/Documents/my_directory]
$ ls
copy.txt  file.txt  new_file.txt

(hacker@kali)-[~/Documents/my_directory]
$ grep "Nexon" file.txt
Nexon

(hacker@kali)-[~/Documents/my_directory]
$ grep "Nexon" file.txt -nt~/my_directory
9:Nexon file.txt

(hacker@kali)-[~/Documents/my_directory]
$ grep "Nexon" file.txt -v
land rover
frod
tata rover
Frod
maruthi suzuki
nexon
tata safari
Tata
Land rover

(hacker@kali)-[~/Documents/my_directory]
$ grep "Nexon" file.txt -v -n
1:land rover
2:frod
3:tata rover
4:Frod
5:maruthi suzuki
6:nexon
7:tata safari
8:Tata
10:Land rover

(hacker@kali)-[~/Documents/my_directory]
$ grep "Nexon" file.txt -i
nexon
Nexon

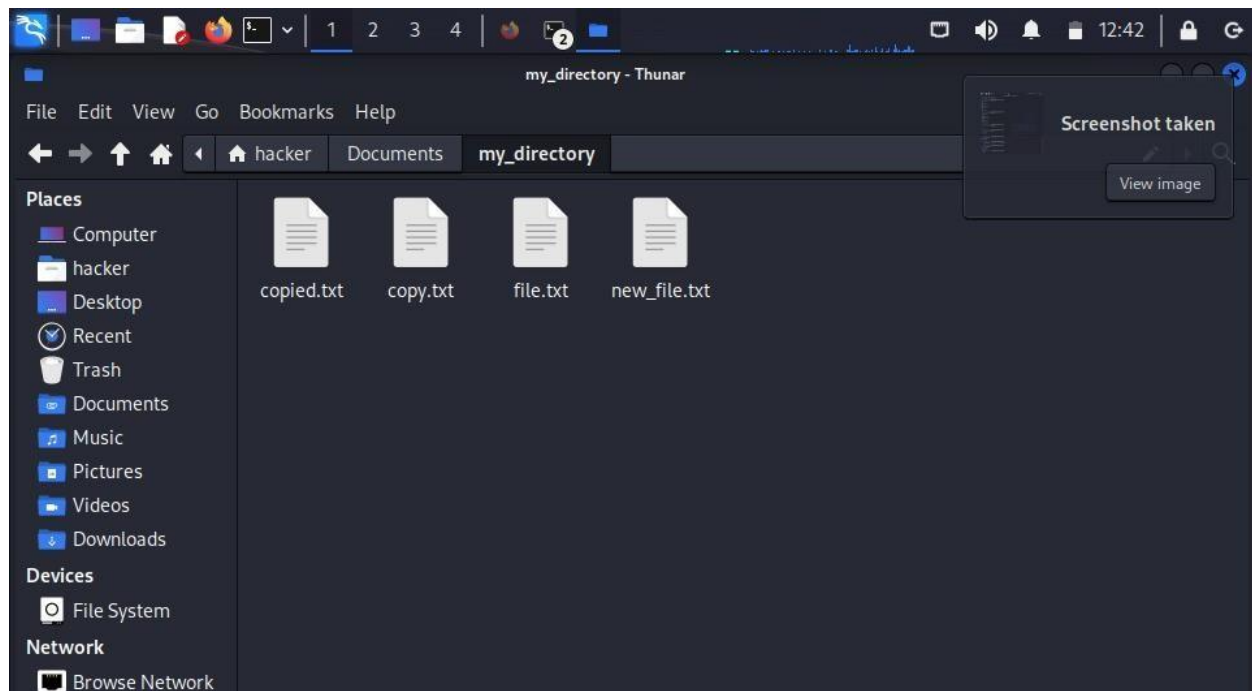
(hacker@kali)-[~/Documents/my_directory]
$ grep "tata" file.txt -i
tata rover
tata safari
Tata
```

```
(hacker@kali)-[~/Documents/my_directory]
$ cat file.txt | grep "tata" | tee copied.txt | wc
2      4      23

(hacker@kali)-[~/Documents/my_directory]
$ cat copied.txt
tata rover
tata safari

(hacker@kali)-[~/Documents/my_directory]
$ ps
  PID TTY          TIME CMD
 23690 pts/0    00:00:08 zsh
 34989 pts/0    00:00:00 ps

(hacker@kali)-[~/Documents/my_directory]
$
```



All tasks are performed in kali linux and verified the outputs.