

EXTERNSHIP PROGRAM – CYBER SECURITY AND ETHICAL HACKING

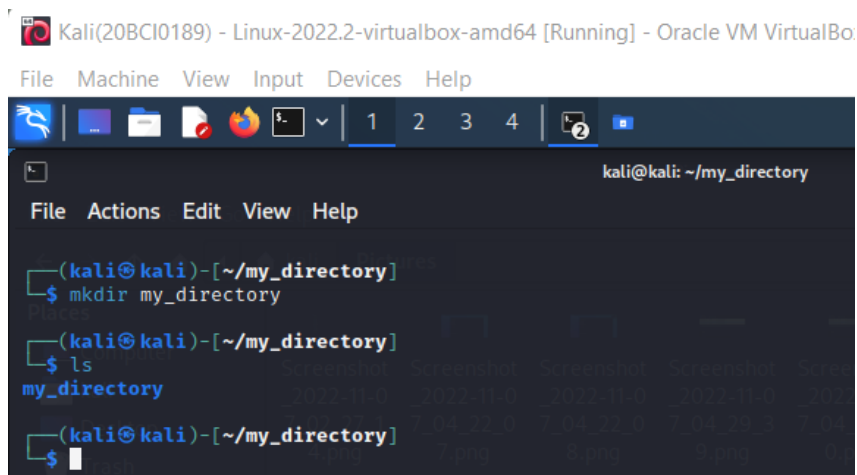
ASSIGNMENT II : BASH SHELL BASICS

TASK1 . File and directory manipulationsss

1. Create a directory called "my_directory".

```
mkdir my_directory
```

```
ls
```

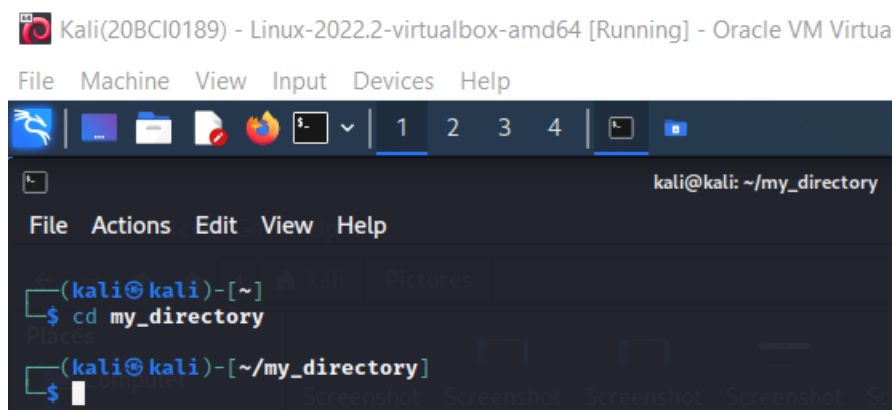


The screenshot shows a terminal window titled "Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The terminal prompt is "kali@kali: ~/my_directory". The user has entered the command "mkdir my_directory" and then "ls". The output of "ls" shows a directory named "my_directory".

```
Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBo
File Machine View Input Devices Help
kali@kali: ~/my_directory
(kali@kali)~[~/my_directory]
$ mkdir my_directory
(kali@kali)~[~/my_directory]
$ ls
my_directory
(kali@kali)~[~/my_directory]
$
```

2 . Navigate into the "my_directory".

```
cd my_directory
```



The screenshot shows a terminal window titled "Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The terminal prompt is "kali@kali: ~/my_directory". The user has entered the command "cd my_directory". The output of "cd" shows the user is now in the "my_directory" directory.

```
Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM Virtua
File Machine View Input Devices Help
kali@kali: ~/my_directory
(kali@kali)~[~]
$ cd my_directory
(kali@kali)~[~/my_directory]
$
```

3 . Create an empty file called "my_file.txt".

touch my_file.txt

ls

```
(kali㉿kali)-[~/my_directory]
$ touch my_file.txt

(kali㉿kali)-[~/my_directory]
$ ls
my_directory  my_file.txt  7.png  8.png  9.png
```

4 . List all the files and directories in the current directory.

ls -la

```
(kali㉿kali)-[~/my_directory]
$ ls -la
total 12
drwxr-xr-x  3 kali kali 4096 May 28 10:12 .
drwxr-xr-x 20 kali kali 4096 May 28 10:09 ..
drwxr-xr-x  2 kali kali 4096 May 28 10:07 my_directory
-rw-r--r--  1 kali kali    0 May 28 10:12 my_file.txt

(kali㉿kali)-[~/my_directory]
$
```

5 . Rename "my_file.txt" to "new_file.txt".

mv my_file.txt new_file.txt

ls

```
(kali㉿kali)-[~/my_directory]
$ mv my_file.txt new_file.txt

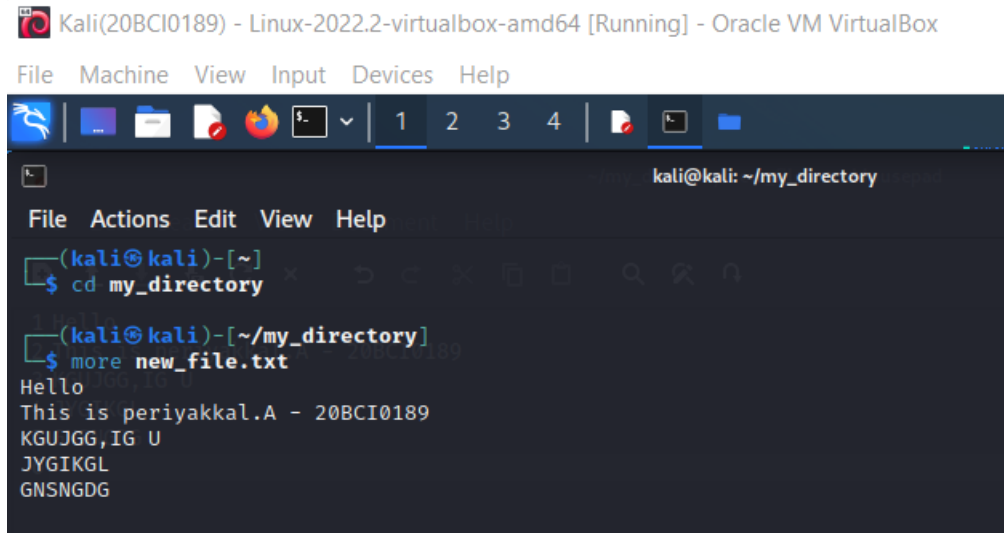
(kali㉿kali)-[~/my_directory]
$ ls
my_directory  new_file.txt

(kali㉿kali)-[~/my_directory]
$
```

6 . Display the content of "new_file.txt" using a pager tool of your choice.

more new_file.txt

(to display content of new_file.txt , I have added random words to it)



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

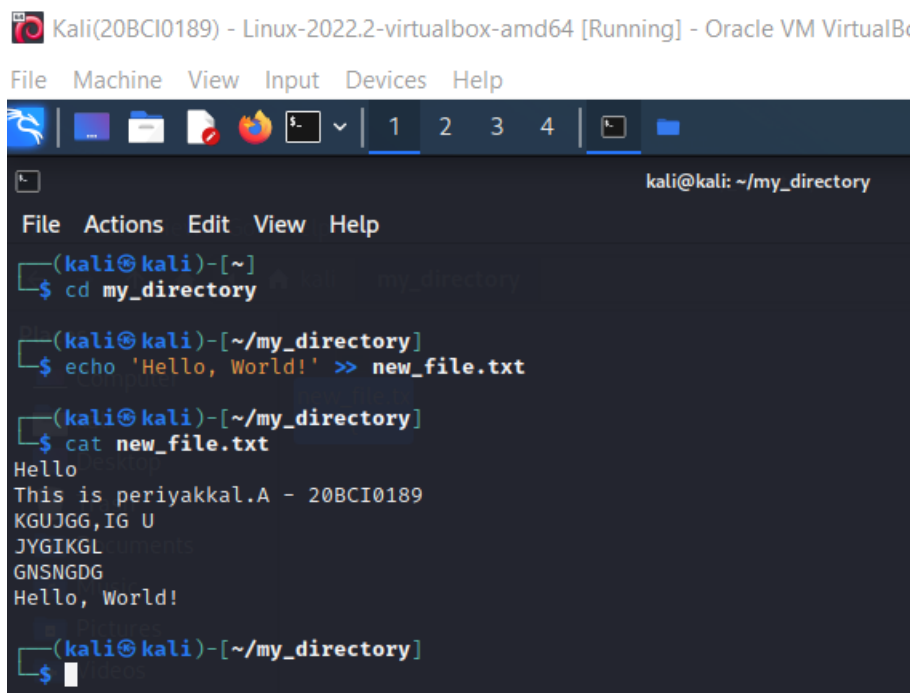
kali@kali: ~/my_directory

```
(kali@kali)-[~]
$ cd my_directory

(kali@kali)-[~/my_directory]
$ more new_file.txt
Hello
This is periyakkal.A - 20BCI0189
KGUJGG,IG U
JYGIKGL
GNSNGDG
```

7 . Append the text "Hello, World!" to "new_file.txt".

echo 'Hello, World!' >> new_file.txt



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/my_directory

```
(kali@kali)-[~]
$ cd my_directory

(kali@kali)-[~/my_directory]
$ echo 'Hello, World!' >> new_file.txt

(kali@kali)-[~/my_directory]
$ cat new_file.txt
Hello
This is periyakkal.A - 20BCI0189
KGUJGG,IG U
JYGIKGL
GNSNGDG
Hello, World!

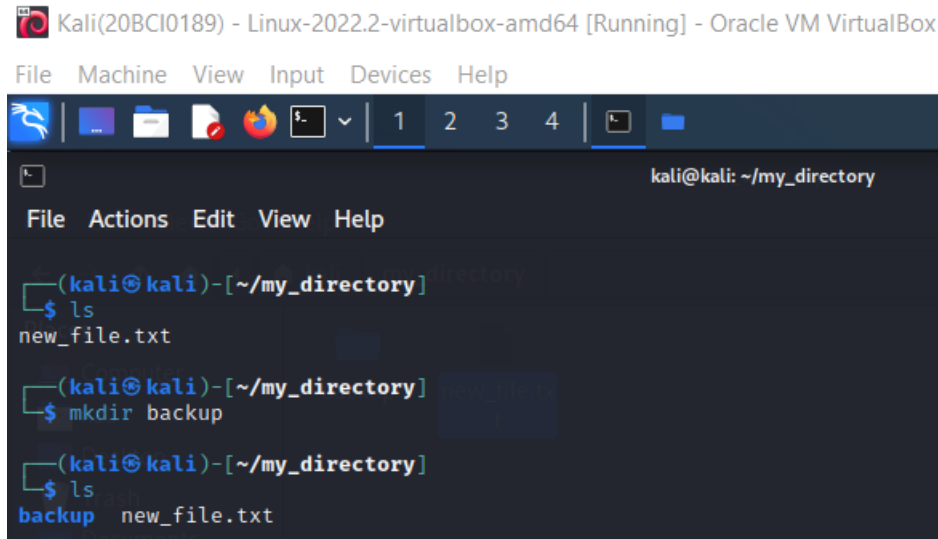
(kali@kali)-[~/my_directory]
$
```

8 . Create a new directory called "backup" within "my_directory".

ls

mkdir backup

ls

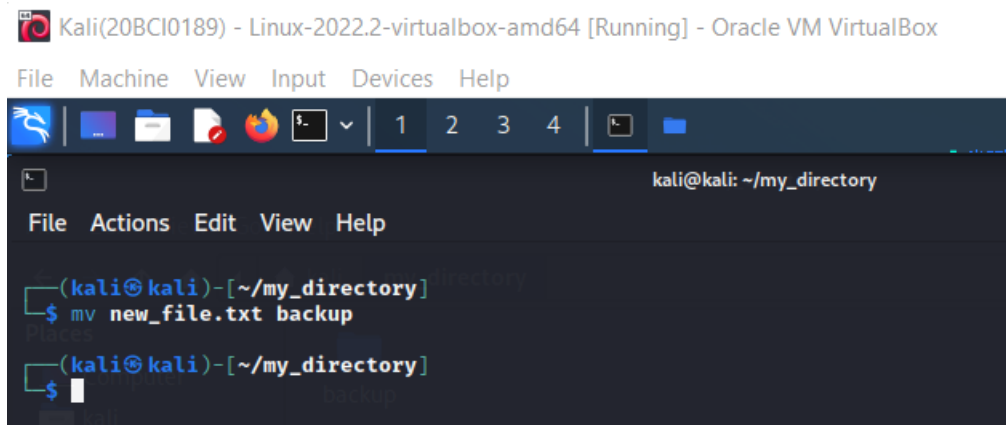


The screenshot shows a Kali Linux terminal window titled "Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The terminal is in the directory ~/my_directory. The user runs 'ls', showing 'new_file.txt'. Then they run 'mkdir backup'. Finally, they run 'ls' again, showing both 'backup' and 'new_file.txt'.

```
kali@kali: ~/my_directory
File Actions Edit View Help
(kali@kali)-[~/my_directory]
$ ls
new_file.txt
(kali@kali)-[~/my_directory]
$ mkdir backup
(kali@kali)-[~/my_directory]
$ ls
backup new_file.txt
```

9 . Move "new_file.txt" to the "backup" directory.

mv new_file.txt backup



The screenshot shows the same Kali Linux terminal window. The user runs 'mv new_file.txt backup'. The prompt returns, and the user has entered a new command, indicated by a cursor.

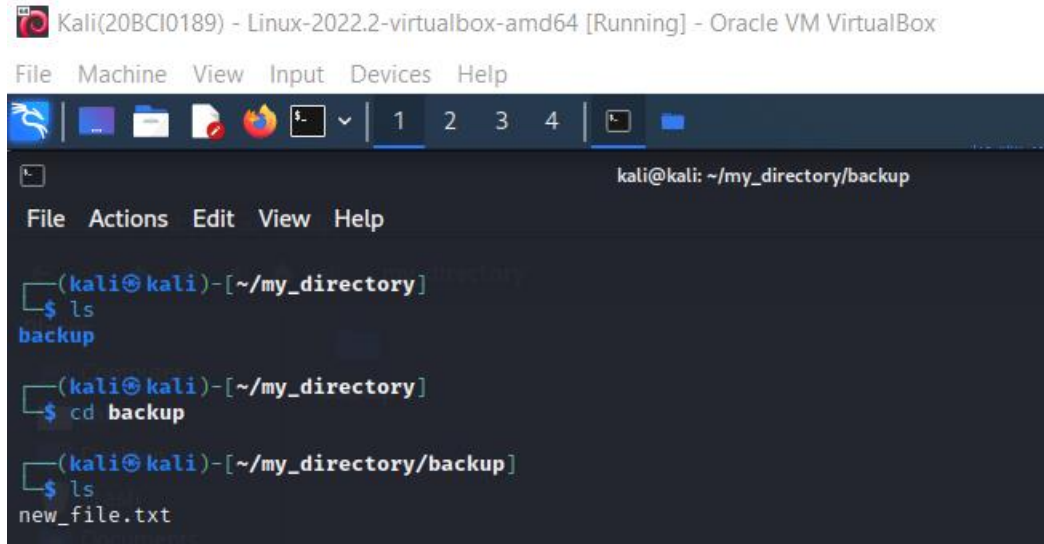
```
kali@kali: ~/my_directory
File Actions Edit View Help
(kali@kali)-[~/my_directory]
$ mv new_file.txt backup
(kali@kali)-[~/my_directory]
$
```

10 . Verify that "new_file.txt" is now located in the "backup" directory.

ls

cd backup

ls



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/my_directory/backup

```
(kali@kali)~[~/my_directory]
$ ls
backup

(kali@kali)~[~/my_directory]
$ cd backup

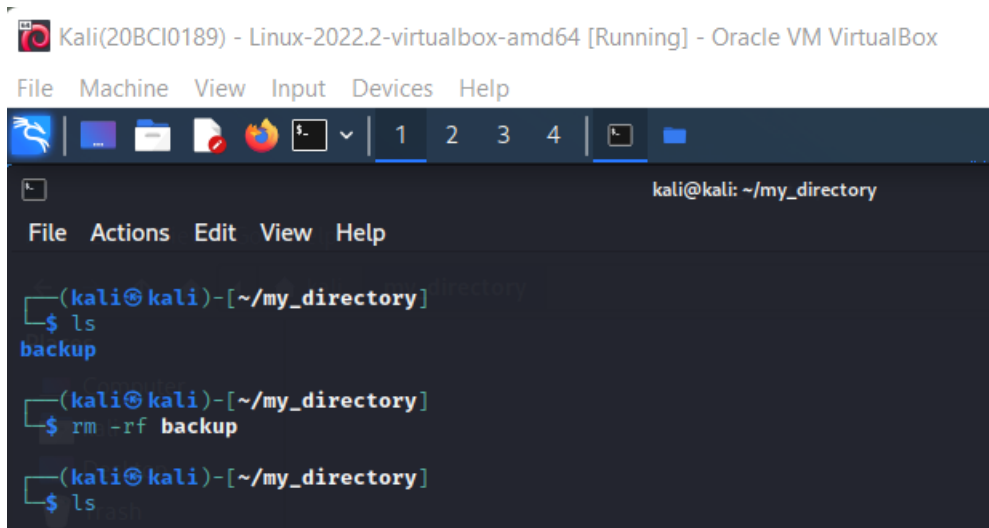
(kali@kali)~[~/my_directory/backup]
$ ls
new_file.txt
```

11 . Delete the "backup" directory and all its contents.

ls

rm -rf backup

ls



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~/my_directory

```
(kali@kali)~[~/my_directory]
$ ls
backup

(kali@kali)~[~/my_directory]
$ rm -rf backup

(kali@kali)~[~/my_directory]
$ ls
```

r : recursive (remove directories and their contents recursively)

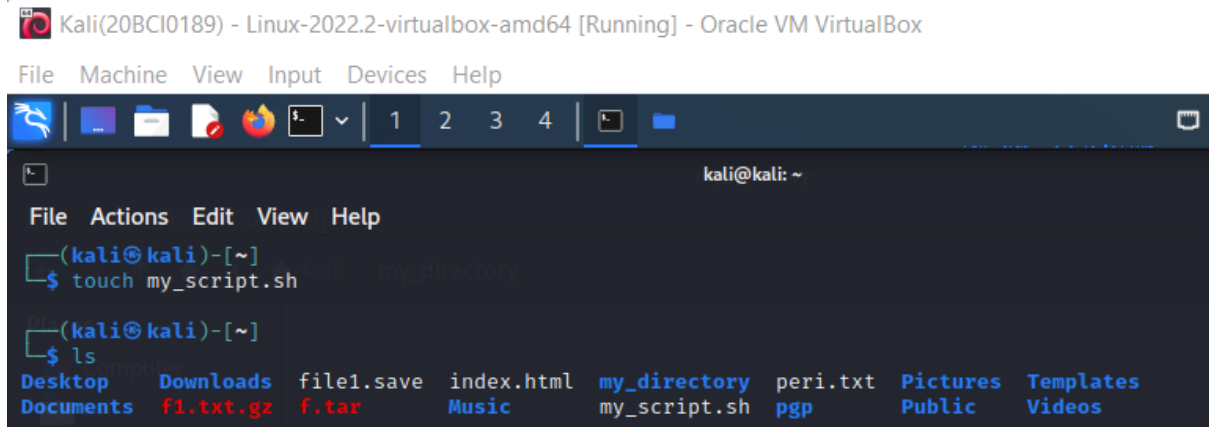
f: force (ignore non-existent file , never prompt)

TASK 2 : PERMISSIONS AND SCRIPTING

1.Create a new file called “my_script.sh”

```
touch my_script.sh
```

```
ls
```



The screenshot shows a Kali Linux terminal window titled "Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox". The terminal displays the following commands and output:

```
kali@kali: ~  
File Actions Edit View Help  
$ touch my_script.sh  
$ ls  
Desktop Downloads file1.save index.html my_directory peri.txt Pictures Templates  
Documents fl.txt.gz f.tar Music my_script.sh pgp Public Videos
```

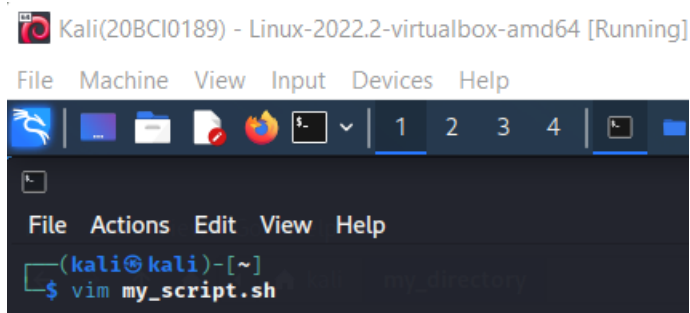
2. Edit my_script.sh using any text editor , add the given lines, make it executable , and run.

```
vim my_script.sh
```

```
#!/bin/bash
```

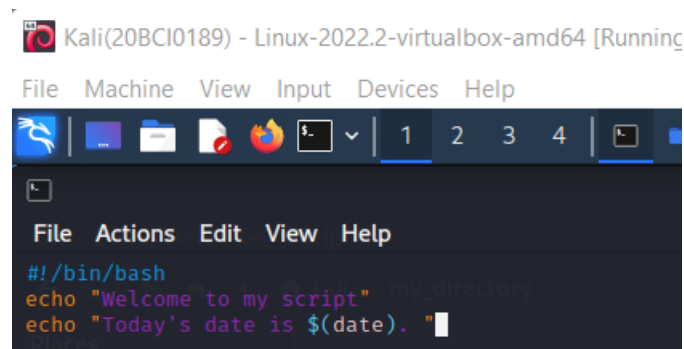
```
echo “Welcome to my script!”
```

```
echo “Today’s date is $(date).”
```



The screenshot shows a Kali Linux terminal window titled "Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running]". The terminal displays the following command:

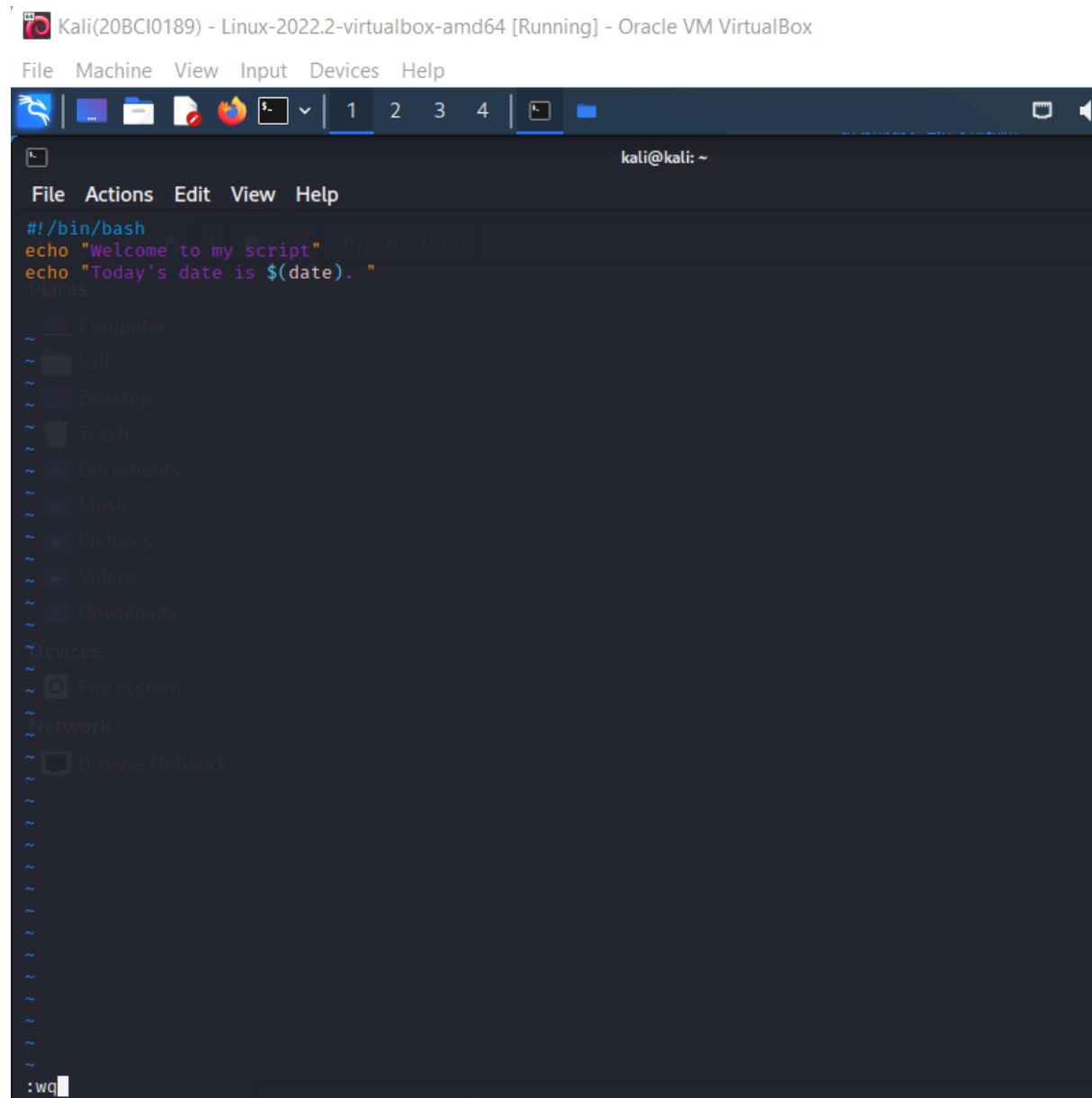
```
kali@kali: ~  
File Machine View Input Devices Help  
$ vim my_script.sh
```



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running]

File Machine View Input Devices Help

```
#!/bin/bash
echo "Welcome to my script" my_directory
echo "Today's date is $(date). "
```



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
#!/bin/bash
echo "Welcome to my script" my_directory
echo "Today's date is $(date). "
```

Places

- Computer
- kali
- Desktop
- Trash
- Documents
- Music
- Pictures
- Videos
- Downloads
- Devices
- File System
- Network
- Browse Network

kali@kali: ~

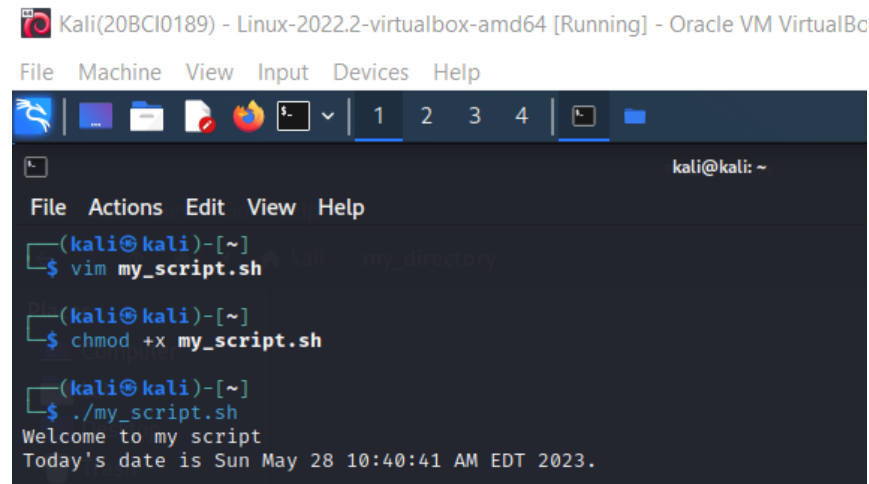
:wq

w : save changes made to the file

q : exit Vim

```
chmod +x my_script.sh
```

```
./my_script.sh
```



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ vim my_script.sh
(kali@kali)-[~]
$ chmod +x my_script.sh
(kali@kali)-[~]
$ ./my_script.sh
Welcome to my script
Today's date is Sun May 28 10:40:41 AM EDT 2023.

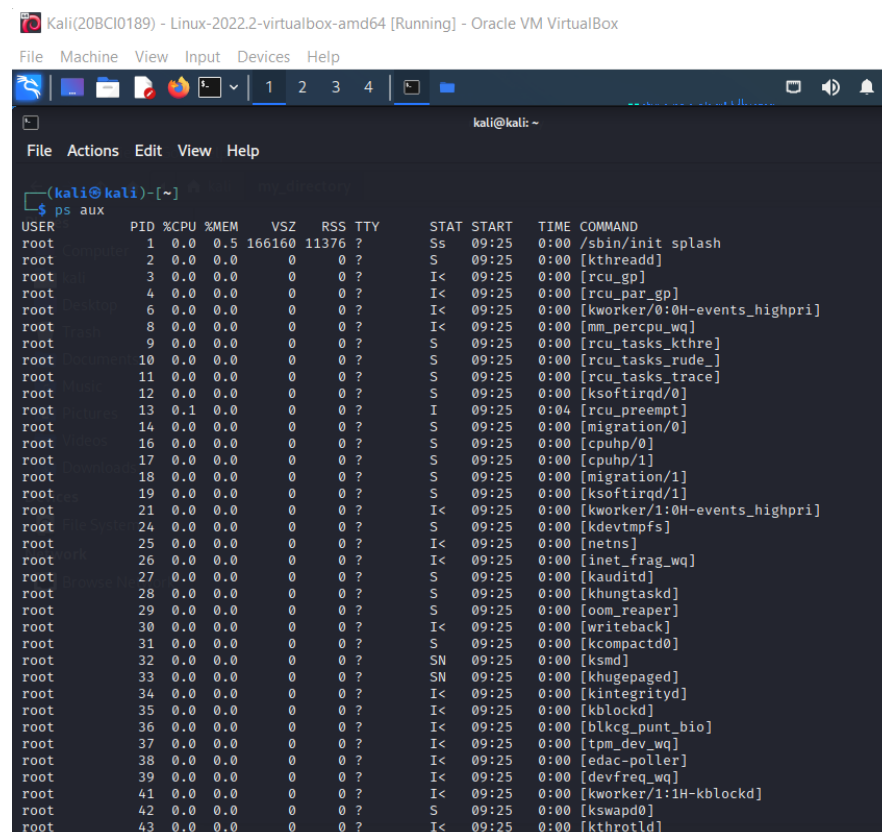
```

TASK 3 : COMMAND EXECUTION AND PIPELINES

1. List all the processes running on your system using the "ps" command.

ps aux

The ps aux command is used to display a detailed list of all running processes on a Linux or Unix system.



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ ps aux

```

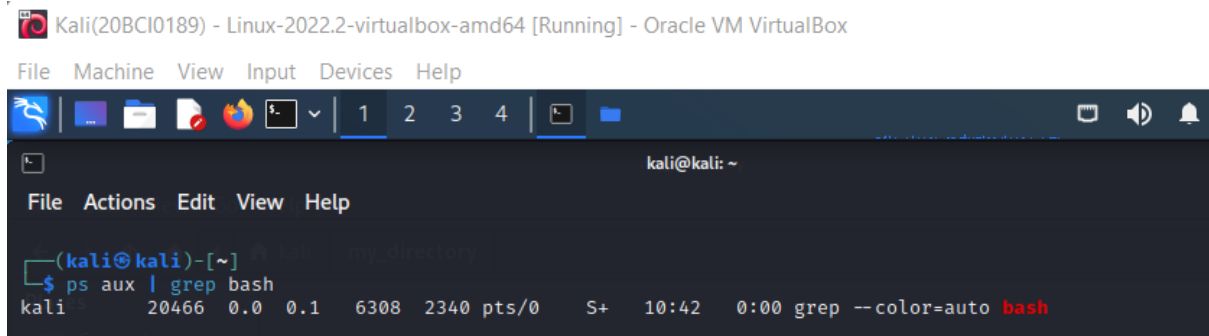
USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.5	166160	11376	?	Ss	09:25	0:00	/sbin/init splash
root	2	0.0	0.0	0	0	?	S	09:25	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	I<	09:25	0:00	[rcu_gp]
root	4	0.0	0.0	0	0	?	I<	09:25	0:00	[rcu_par_gp]
root	6	0.0	0.0	0	0	?	I<	09:25	0:00	[kworker/0:0H-events_highpri]
root	8	0.0	0.0	0	0	?	I<	09:25	0:00	[mm_percpu_wq]
root	9	0.0	0.0	0	0	?	S	09:25	0:00	[rcu_tasks_kthre]
root	10	0.0	0.0	0	0	?	S	09:25	0:00	[rcu_tasks_rude_]
root	11	0.0	0.0	0	0	?	S	09:25	0:00	[rcu_tasks_trace]
root	12	0.0	0.0	0	0	?	S	09:25	0:00	[ksoftirqd/0]
root	13	0.1	0.0	0	0	?	I	09:25	0:04	[rcu_preempt]
root	14	0.0	0.0	0	0	?	S	09:25	0:00	[migration/0]
root	16	0.0	0.0	0	0	?	S	09:25	0:00	[cpuhp/0]
root	17	0.0	0.0	0	0	?	S	09:25	0:00	[cpuhp/1]
root	18	0.0	0.0	0	0	?	S	09:25	0:00	[migration/1]
root	19	0.0	0.0	0	0	?	S	09:25	0:00	[ksoftirqd/1]
root	21	0.0	0.0	0	0	?	I<	09:25	0:00	[kworker/1:0H-events_highpri]
root	24	0.0	0.0	0	0	?	S	09:25	0:00	[kdevtmpfs]
root	25	0.0	0.0	0	0	?	I<	09:25	0:00	[netns]
root	26	0.0	0.0	0	0	?	I<	09:25	0:00	[inet_frag_wq]
root	27	0.0	0.0	0	0	?	S	09:25	0:00	[kauditd]
root	28	0.0	0.0	0	0	?	S	09:25	0:00	[khungtaskd]
root	29	0.0	0.0	0	0	?	S	09:25	0:00	[oom_reaper]
root	30	0.0	0.0	0	0	?	I<	09:25	0:00	[writeback]
root	31	0.0	0.0	0	0	?	S	09:25	0:00	[kcompactd0]
root	32	0.0	0.0	0	0	?	SN	09:25	0:00	[ksmd]
root	33	0.0	0.0	0	0	?	SN	09:25	0:00	[khugepaged]
root	34	0.0	0.0	0	0	?	I<	09:25	0:00	[kintegrityd]
root	35	0.0	0.0	0	0	?	I<	09:25	0:00	[kblockd]
root	36	0.0	0.0	0	0	?	I<	09:25	0:00	[blkcg_punt_bio]
root	37	0.0	0.0	0	0	?	I<	09:25	0:00	[tpm_dev_wq]
root	38	0.0	0.0	0	0	?	I<	09:25	0:00	[edac-poller]
root	39	0.0	0.0	0	0	?	I<	09:25	0:00	[devfreq_wq]
root	41	0.0	0.0	0	0	?	I<	09:25	0:00	[kworker/1:1H-kblockd]
root	42	0.0	0.0	0	0	?	S	09:25	0:00	[kswapd0]
root	43	0.0	0.0	0	0	?	I<	09:25	0:00	[kthrotld]

2. Use the "grep" command to filter the processes list and display only the processes with

"bash" in their name.

`ps aux | grep bash`

(grep is used for matching a pattern or string)



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

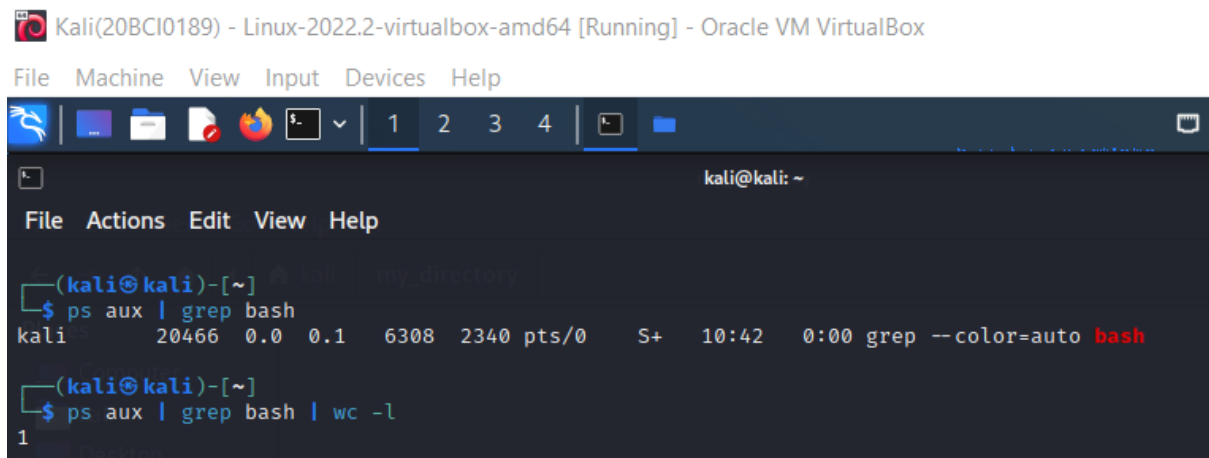
File Machine View Input Devices Help

kali@kali: ~

```
(kali@kali)-[~]  
$ ps aux | grep bash  
kali 20466 0.0 0.1 6308 2340 pts/0 S+ 10:42 0:00 grep --color=auto bash
```

3. Use the "wc" command to count the number of lines in the filtered output.

`ps aux | grep bash | wc -l`



Kali(20BCI0189) - Linux-2022.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

kali@kali: ~

```
(kali@kali)-[~]  
$ ps aux | grep bash  
kali 20466 0.0 0.1 6308 2340 pts/0 S+ 10:42 0:00 grep --color=auto bash  
  
(kali@kali)-[~]  
$ ps aux | grep bash | wc -l  
1
```