

Assignment 2: Bash Shell Basics

NAME: JAYASREE N

REGISTER NUMBER: 20MIS0370

Task 1: File and Directory Manipulation

1. Create a directory called "my_directory".

```
(jayasree@kali)-[~]  
$ mkdir my_directory
```

2. Navigate into the "my_directory".

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

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

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

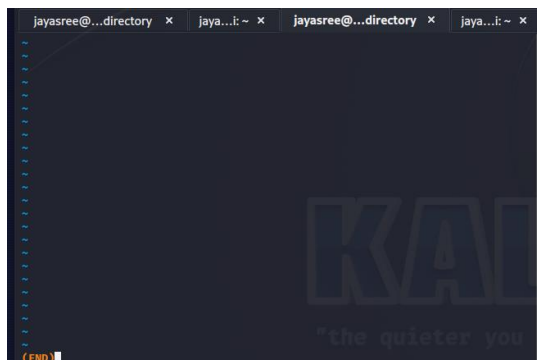
4. List all the files and directories in the current directory.

```
(jayasree@kali)-[~/my_directory]  
$ ls  
my_file.txt
```

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

```
(jayasree@kali)-[~/my_directory]  
$ mv my_file.txt 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".

```
(jayasree@kali)-[~]  
$ nano new_file.txt
```

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

```
(jayasree@kali)-[~]  
$ mkdir backup
```

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

```
(jayasree@kali)-[~]  
$ mv new_file.txt backup/
```

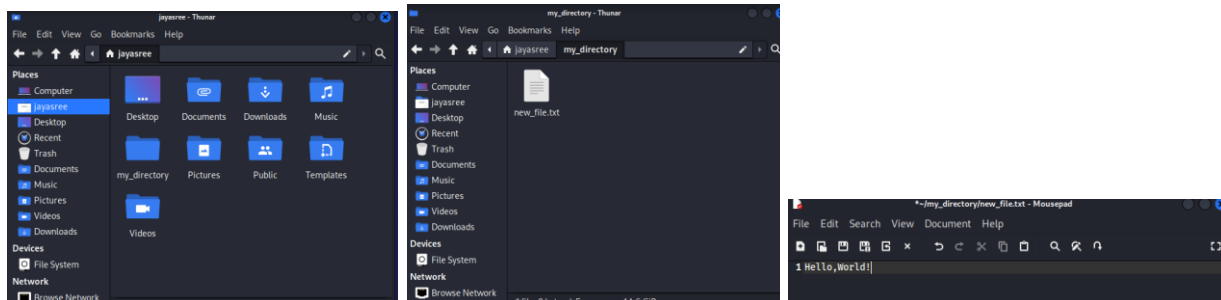
10. Verify that "new_file.txt" is now located in the "backup" directory.

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

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

```
(jayasree@kali)-[~]  
$ rm -r backup  
  
(jayasree@kali)-[~]  
$ ls backup/  
ls: cannot access 'backup/': No such file or directory
```

Output:



Task 2: Permissions and Scripting

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

```
(jayasree@kali)-[~]  
$ touch my_script.sh
```

- Edit "my_script.sh" using a text editor of your choice and add the following lines:

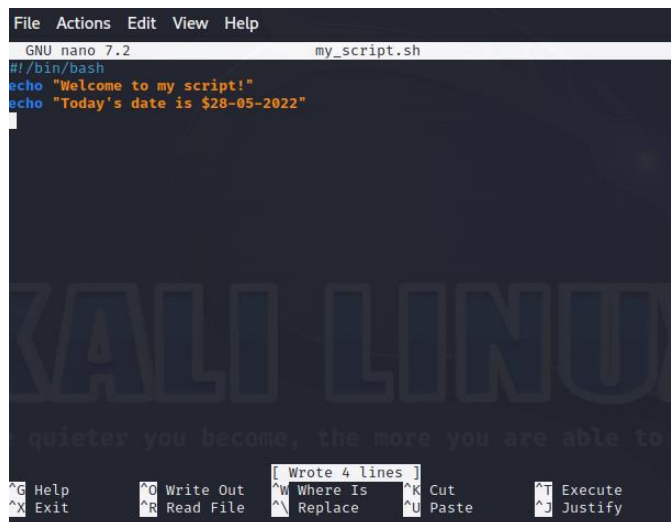
bash

#!/bin/bash

echo "Welcome to my script!"

echo "Today's date is \$(date)."

Save and exit the file.

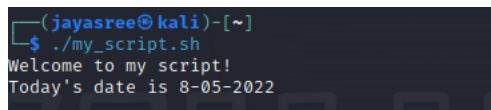


```
File Actions Edit View Help  
GNU nano 7.2 my_script.sh  
#!/bin/bash  
echo "Welcome to my script!"  
echo "Today's date is $(date)"  
KALI LINUX  
quieter you become, the more you are able to  
[ Wrote 4 lines ]  
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^J Execute  
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^_ Justify
```

- Make "my_script.sh" executable.

```
(jayasree@kali)-[~]  
$ chmod +x my_script.sh
```

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



```
(jayasree@kali)-[~]  
$ ./my_script.sh  
Welcome to my script!  
Today's date is 8-05-2022
```

Task 3: Command Execution and Pipelines

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

```
jayasree@kali: ~  
File Actions Edit View Help  
(jayasree@kali)-[~]  
$ ps aux  
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
root           1  0.0  0.4 167764 12200 ?        Ss   02:23   0:02 /sbin/init  
root           2  0.0  0.0      0     0 ?        S    02:23   0:00 [kthreadd]  
root           3  0.0  0.0      0     0 ?        I<   02:23   0:00 [rcu_gp]  
root           4  0.0  0.0      0     0 ?        I<   02:23   0:00 [rcu_par_g  
root           5  0.0  0.0      0     0 ?        I<   02:23   0:00 [slub_flus  
root           6  0.0  0.0      0     0 ?        I<   02:23   0:00 [netns]  
root          10  0.0  0.0      0     0 ?        I<   02:23   0:00 [mm_percpu  
root          11  0.0  0.0      0     0 ?        I    02:23   0:00 [rcu_tasks  
root          12  0.0  0.0      0     0 ?        I    02:23   0:00 [rcu_tasks  
root          13  0.0  0.0      0     0 ?        I    02:23   0:00 [rcu_tasks  
root          14  0.0  0.0      0     0 ?        S    02:23   0:00 [ksoftirqd  
root          15  0.1  0.0      0     0 ?        I    02:23   0:37 [rcu_preem  
root          16  0.0  0.0      0     0 ?        S    02:23   0:02 [migration  
root          18  0.0  0.0      0     0 ?        S    02:23   0:00 [cpuhp/0]  
root          19  0.0  0.0      0     0 ?        S    02:23   0:00 [cpuhp/1]  
root          20  0.0  0.0      0     0 ?        S    02:23   0:02 [migration  
root          21  0.0  0.0      0     0 ?        S    02:23   0:00 [ksoftirqd  
root          23  0.0  0.0      0     0 ?        I<   02:23   0:00 [kworker/1  
root          24  0.0  0.0      0     0 ?        S    02:23   0:00 [cpuhp/2]  
root          25  0.0  0.0      0     0 ?        S    02:23   0:01 [migration  
root          26  0.0  0.0      0     0 ?        S    02:23   0:01 [ksoftirqd  
root          28  0.0  0.0      0     0 ?        I<   02:23   0:00 [kworker/2  
root          32  0.0  0.0      0     0 ?        S    02:23   0:00 [kdevtmpfs  
root          33  0.0  0.0      0     0 ?        I<   02:23   0:00 [inet_frag
```

- Use the "grep" command to filter the processes list and display only the processes with "bash" in their name.

```
(jayasree@kali)-[~]  
$ ps aux | grep bash  
jayasree 266107 0.0 0.0 6332 2128 pts/0    S+   11:26   0:00 grep --col  
or-auto bash
```

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

```
(jayasree@kali)-[~]  
$ ps aux | grep bash | wc -l  
1
```