20BKT0075 GAGAN PARASHAR

Assignment - 2 : Bash Shell Basics

Task 1: File and Directory Manipulation

1. Create a directory called "my_directory".

   mkdir my_directory

2. Navigate into the "my_directory".

   cd my_directory

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

   touch my_file.txt

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

   ls

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

   mv my_file.txt new_file.txt

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

   less new_file.txt

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

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

   dquote>
   dquote> "Hello world
   Hello, World >> new_file.txt

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

   mkdir backup

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

```
mv new_file.txt backup/
```

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

```
mv new_file.txt backup/

new_file.txt
```

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

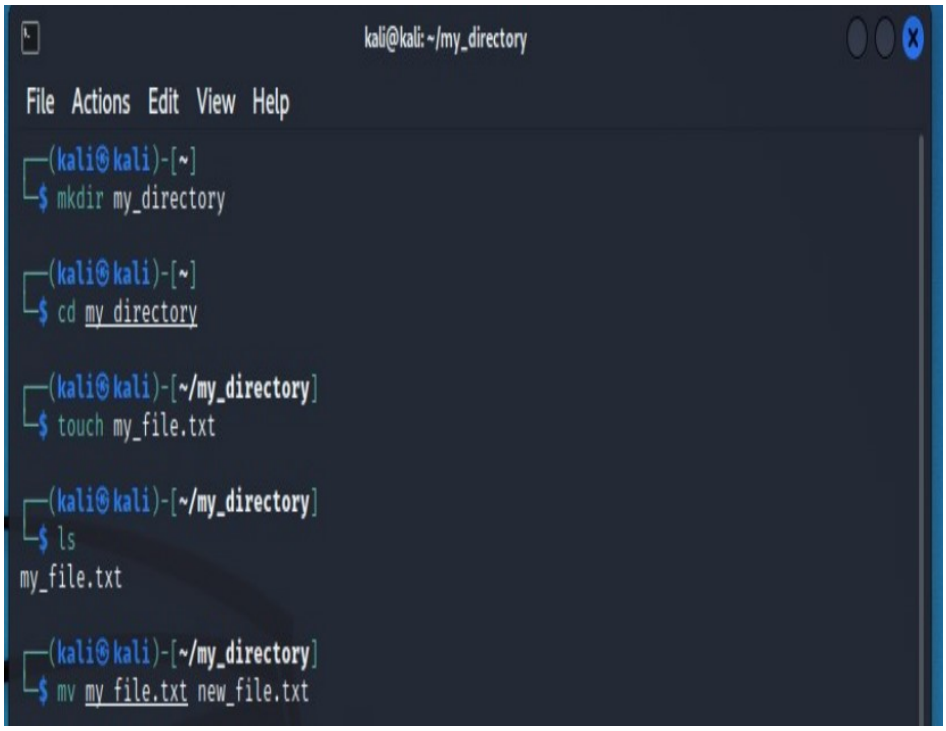
```
rm -r backup

ls backup
```

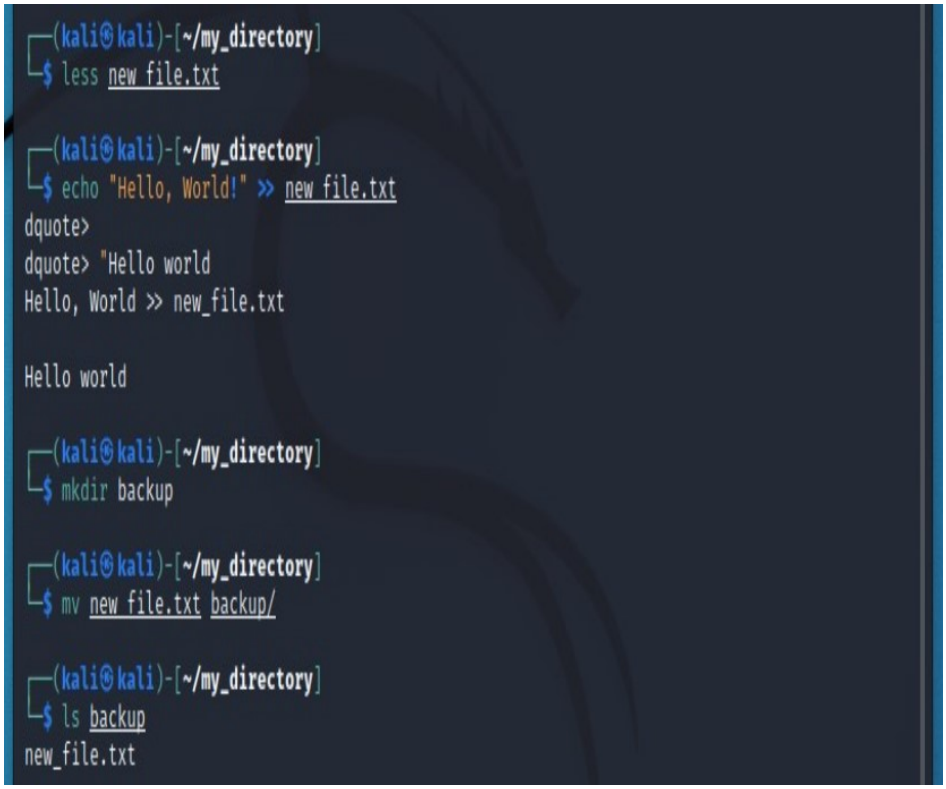
```
ls: cannot access 'backup': No such file or directory
```

```
ls
```

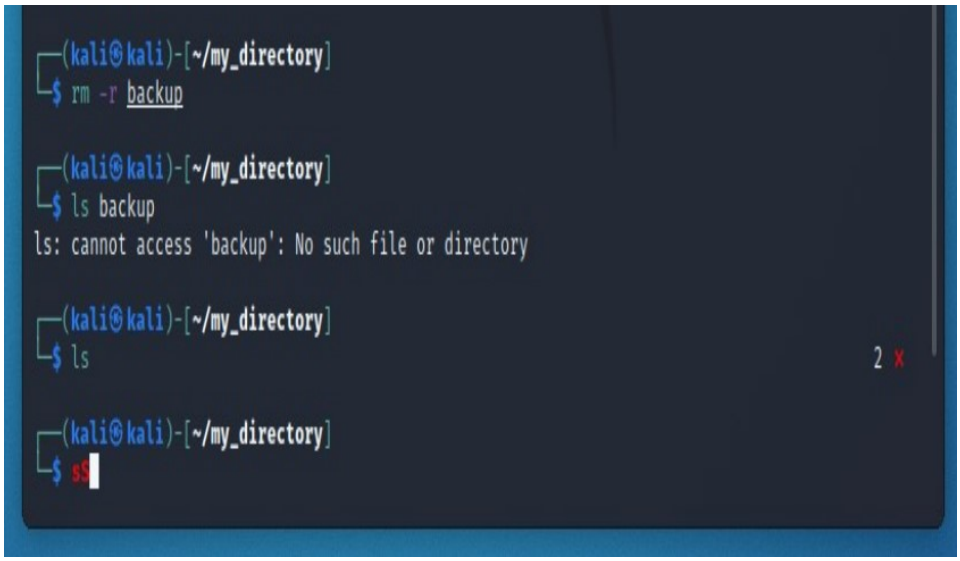
20BKT0075 GAGAN PARASHAR

Output





20BKT0075 GAGAN PARASHAR



Task 2: Permissions and Scripting

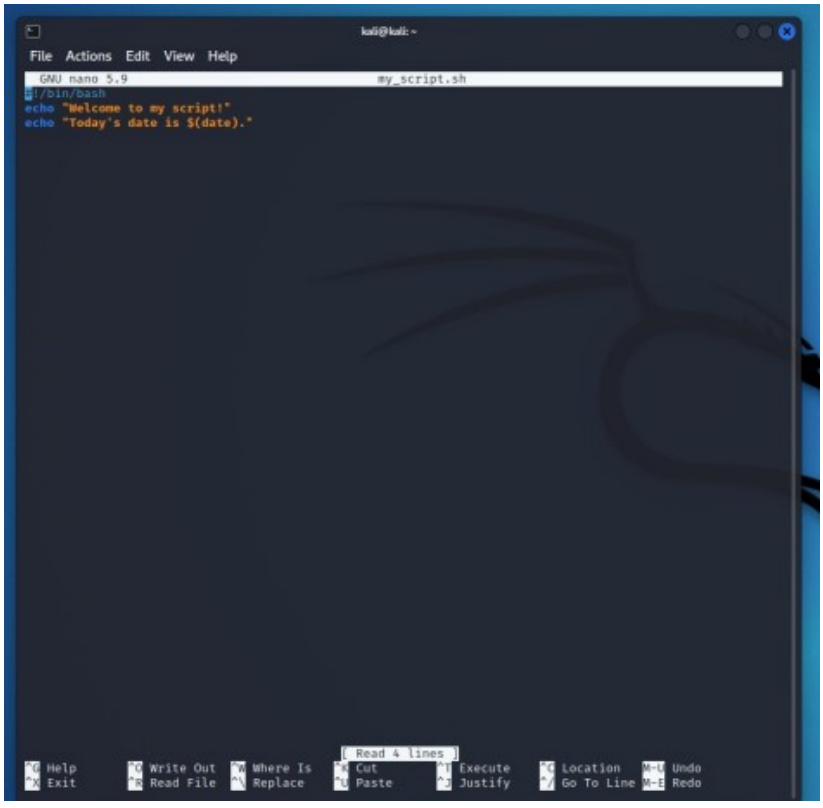
1,Create a new file called "my_script.sh".

touch myscript.sh

2, 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.

nano script.sh



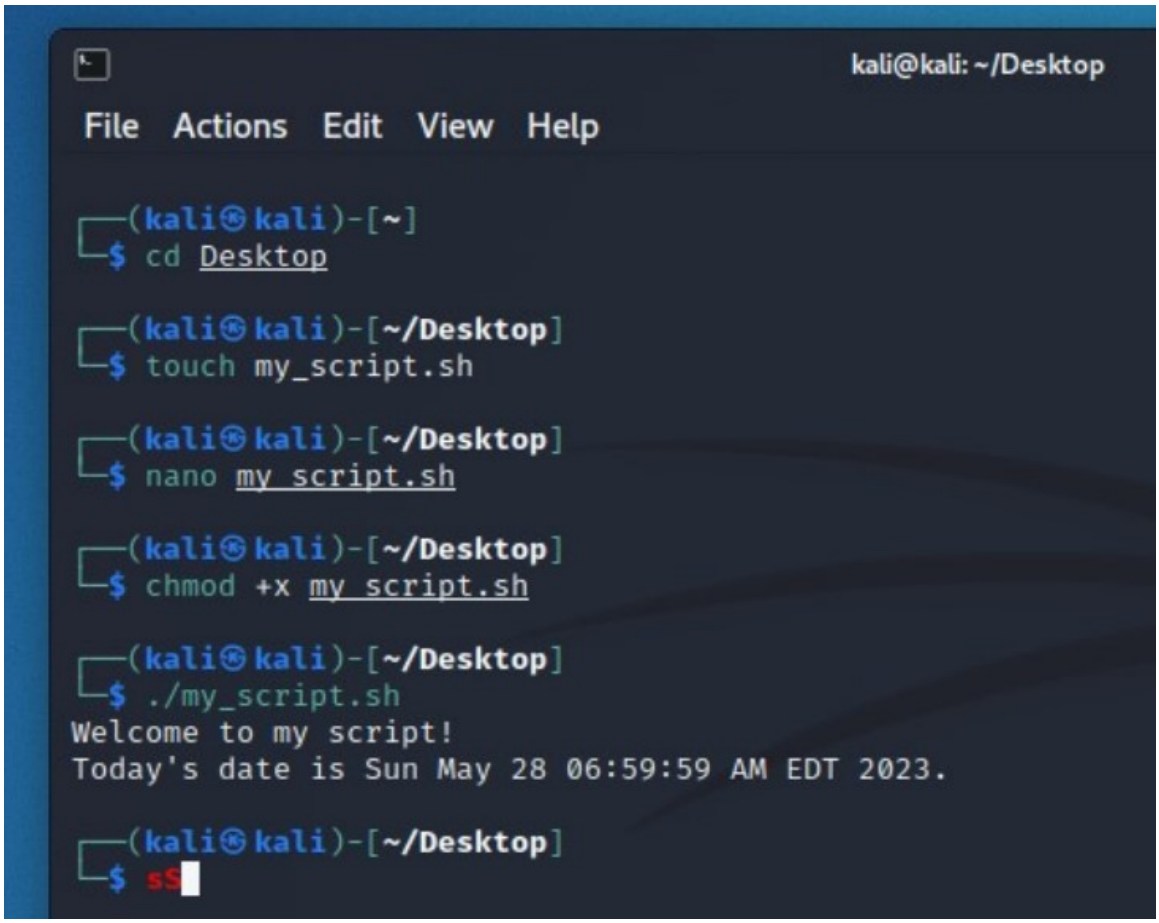
3, Make "my_script.sh" executable.

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

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

```
./my_script.sh
```

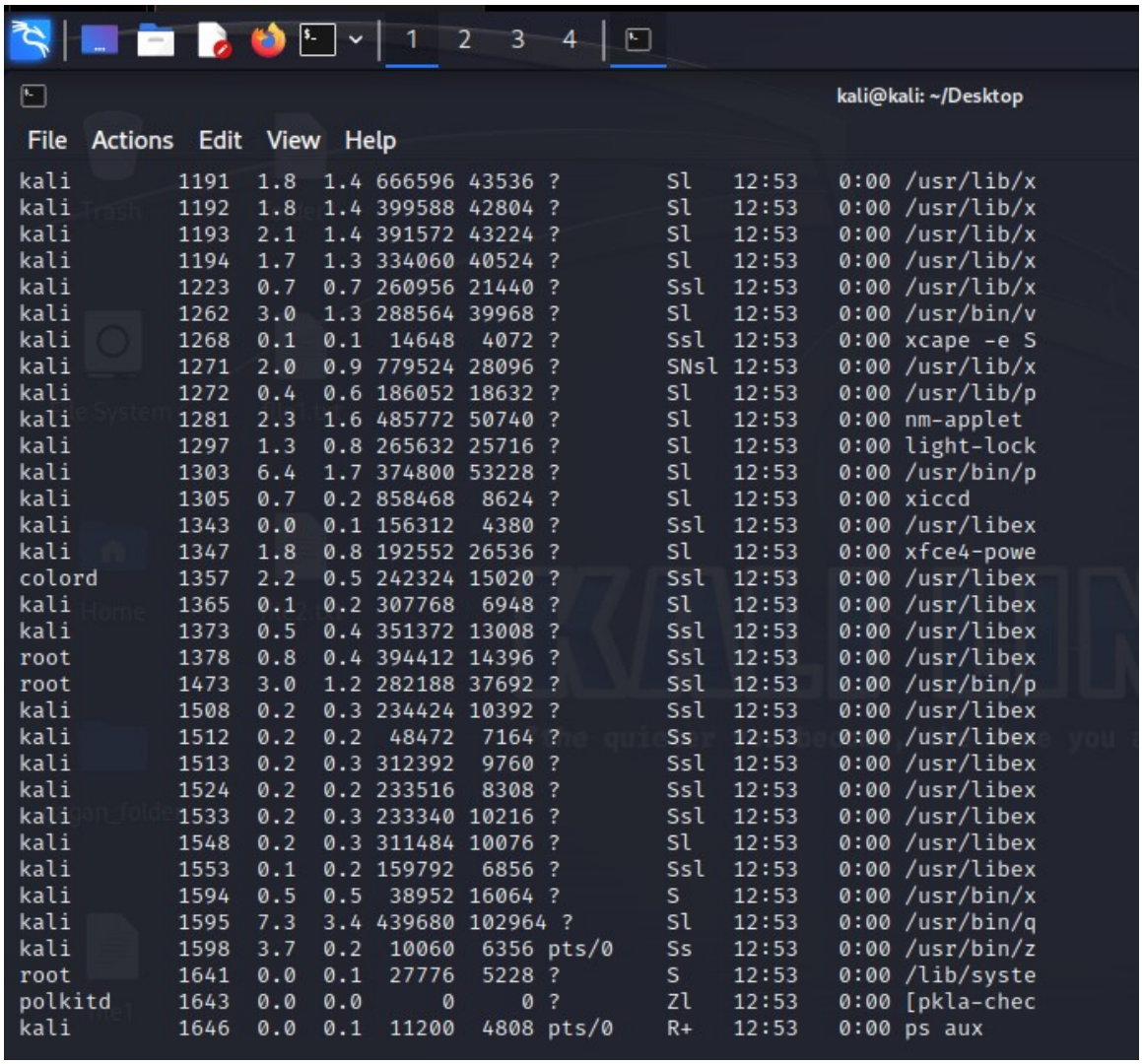
Welcome to my script!
Today's date is Sun May 28 06:59:59 AM EDT 2023.



Task 3: Command Execution and Pipelines

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

ps aux

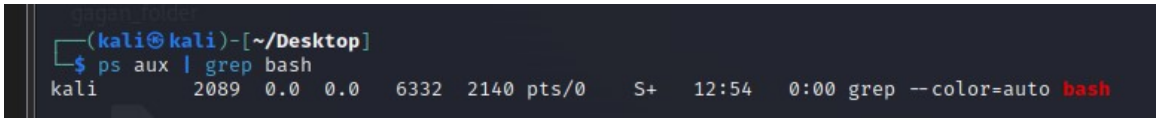


20BKT0075 GAGAN PARASHAR

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

ps aux | grep bash

kali 21398 0.0 0.1 6184 2228 pts/0 S+ 06:43 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 -

Submitted By Gagan parashar

20BKT0075