Assessment-2

Assignment: Bash Shell Basics

Submitted by:

RAYUDU GAYATHRI

20BCB7058

Computer Science Engineering Spl. Business Systems

VIT-AP UNIVERSITY

Task 1: File and Directory Manipulation

1.Create a directory called “my_directory”.

```
mkdir my_dir
```

2.Navigate into the “my_directory”.

```
cd my_dir
```

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.

```
cat new_file.txt
```

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

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

Create a new directory called "backup" within “my_directory”.

.

```
mkdir backup
```

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

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

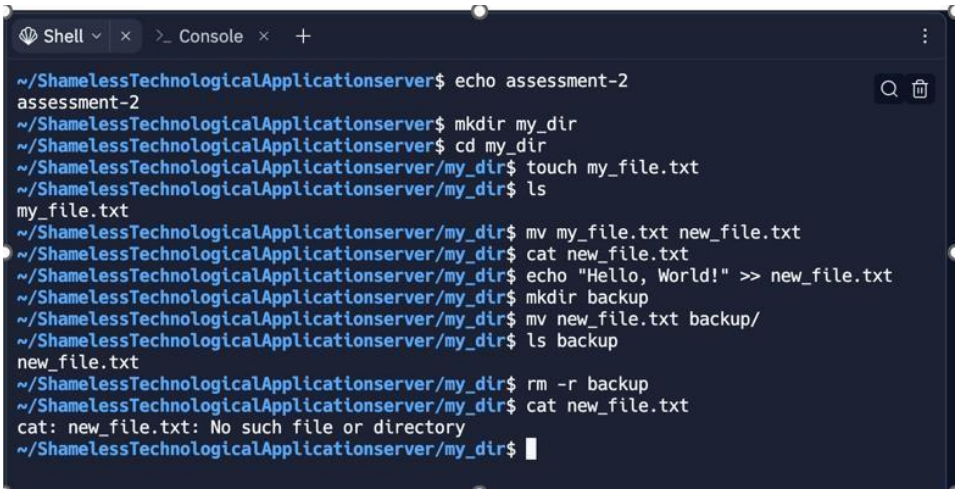
.Verify that "new_file.txt" is now located in the "backup" directory.

```
ls backup
```

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

```
rm -r backup
```

OUTPUT:



Task 2: Permissions and Scripting

Create a new file called "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.

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

Code:

```
touch my_script.sh nano
```

- sult.

Code:

```
touch my_script.sh nano
```

```
my s cript.h
```

Save and exit the file in nano. In nano, you can press “Ctrl + o” to save the file, then Enter to confirm, and “Ctrl + x” to exit the editor.

```
#!/bin/bash        echo "Welcome to
my script!" echo "Today's date is
$(date)." chmod +x my_script.sh  # This
command gives           the file
executable permissions.
```

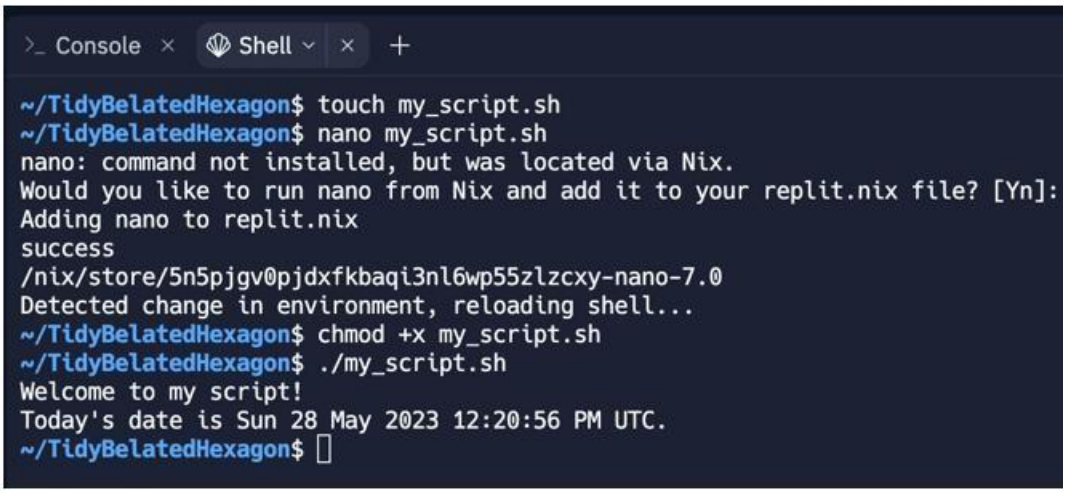
./my_script.sh **Output:**

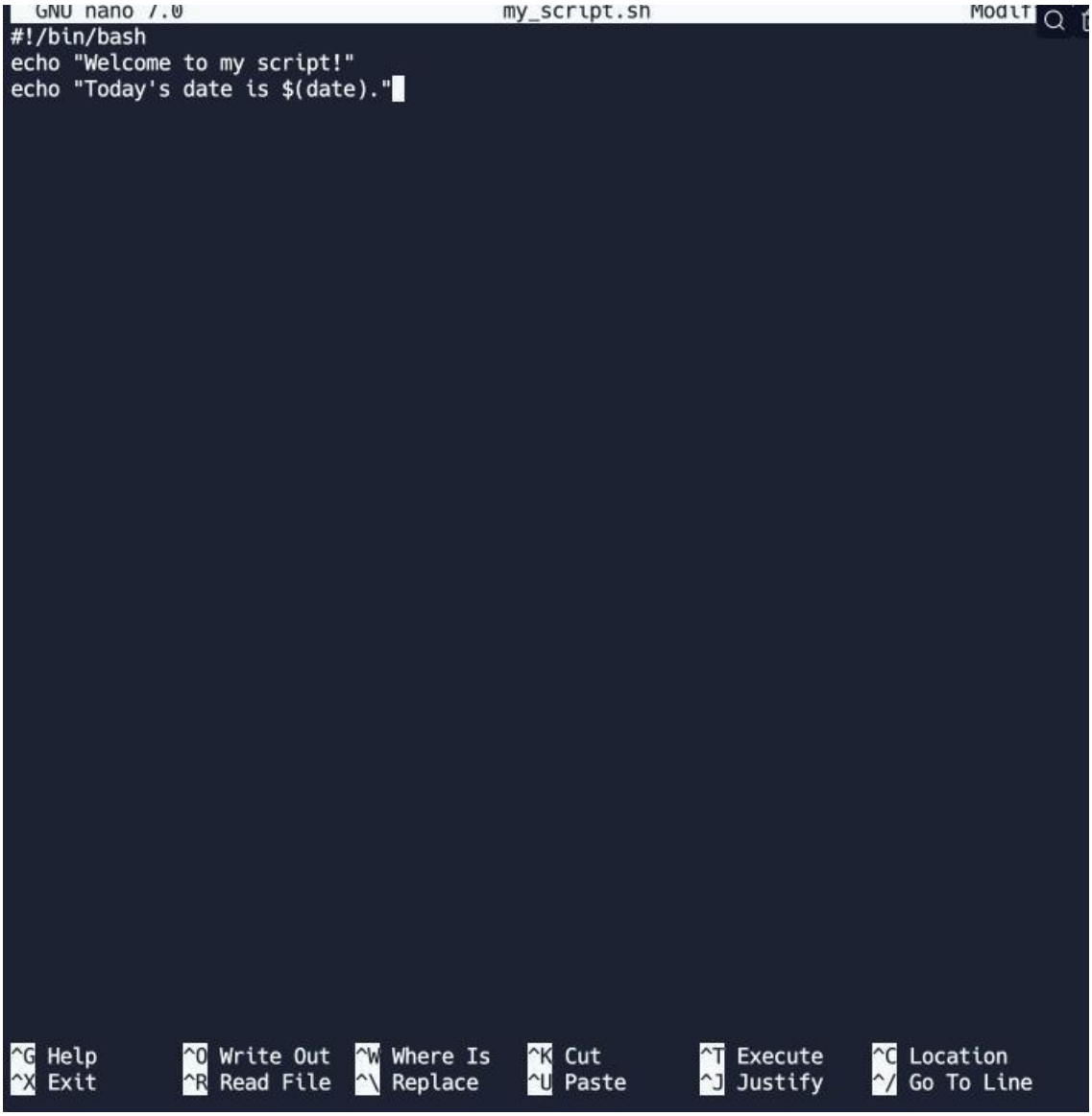
Save and exit the file in nano. In nano, you can press “Ctrl + o” to save the file, then Enter to confirm, and “Ctrl + x” to exit the editor.

```
#!/bin/bash      echo "Welcome to my script!" echo
"Today's date is n          $(date)." chmod +x my_script.sh
# This command gives the
 File executable permissions.
```

./my_script.sh

Output:





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.