

Linux Basic Commands Assignment

Assignment Part-3

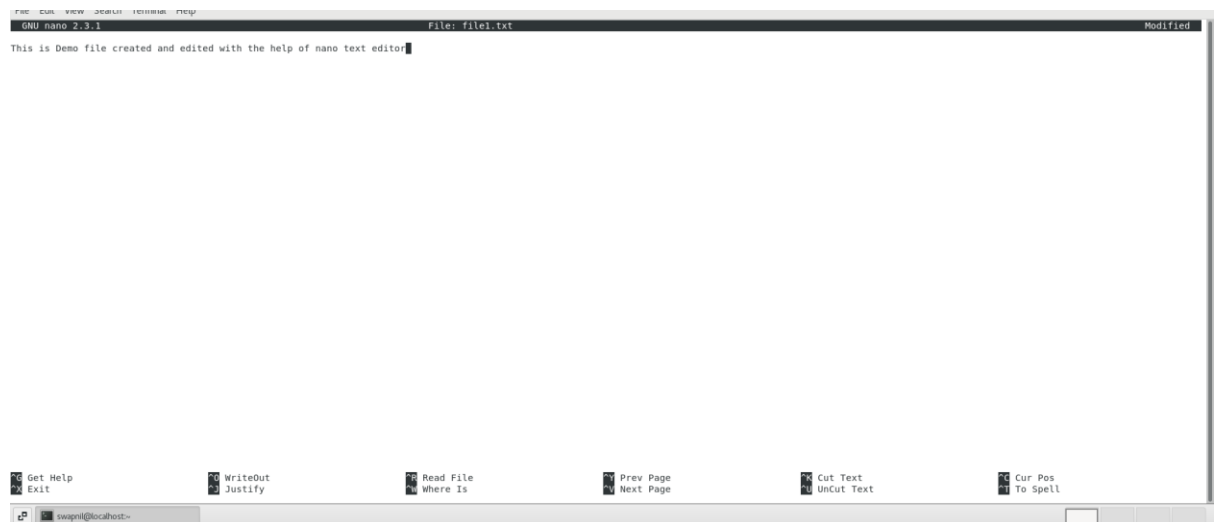
Playing with files

1. Create a file like nano file1.txt

o Edit some data and then save the file

→Ans: With “nano” command created a file1.txt, added text into the file, and finally saved the file with the help of “ctrl+x” to exit out of nano, “Y” to save the data in file and enter. PFB screenshots.

```
[swapnil@localhost ~]$ ls;pwd;date
Desktop Documents Downloads Music Pictures Public Templates Videos
/home/swapnil
Mon Oct 17 20:15:02 IST 2022
[swapnil@localhost ~]$ nano file1.txt
[swapnil@localhost ~]$ cat file1.txt
This is Demo file created and edited with the help of nano text editor
[swapnil@localhost ~]$ ls;pwd;date
Desktop Documents Downloads file1.txt Music Pictures Public Templates Videos
/home/swapnil
Mon Oct 17 20:19:25 IST 2022
[swapnil@localhost ~]$
```



2. Now we will copy data from file1 to new file2

o **cp file1.txt file2.txt**

o Then see the output of file2.txt, **cat file2.txt**

o Give screenshot

→Ans: “cp” command copies content from one file to another. In this case it copied data from file1.txt to file2.txx with file creation as well. PFB screenshot.

```

[swapnil@localhost ~]$ ls;pwd;date
Desktop Documents Downloads file1.txt Music Pictures Public Templates Videos
/home/swapnil
Mon Oct 17 20:22:59 IST 2022
[swapnil@localhost ~]$ cp file1.txt file2.txt
[swapnil@localhost ~]$ ls;pwd;date
Desktop Documents Downloads file1.txt file2.txt Music Pictures Public Templates Videos
/home/swapnil
Mon Oct 17 20:23:24 IST 2022
[swapnil@localhost ~]$ cat file2.txt
This is Demo file created and edited with the help of nano text editor
[swapnil@localhost ~]$ █

```

3. Now we will move the file2.txt to new folder **/home**

- o **mv file2.txt /home**

- o Then go to **home** directory and check **ls**, file exists or not?

- o Given screenshot

→Ans: “mv” command helps to move the files from one directory to another. It also helps in renaming of the file. In this case, due to permission issues on /home directory, we switched to root user first and then moved the file2.txt to /home. PFB screenshot.

```

[swapnil@localhost ~]$ ls;pwd;date
Desktop Documents Downloads file1.txt file2.txt Music Pictures Public Templates Videos
/home/swapnil
Mon Oct 17 20:27:23 IST 2022
[swapnil@localhost ~]$ mv file2.txt /home/
mv: cannot move 'file2.txt' to '/home/file2.txt': Permission denied
[swapnil@localhost ~]$ su -
Password:
Last login: Mon Oct 17 19:47:57 IST 2022 on pts/0
[root@localhost ~]# mv /home/swapnil/file2.txt /home/
[root@localhost ~]# cd /home/
[root@localhost home]# ls
file2.txt swapnil
[root@localhost home]# date;pwd
Mon Oct 17 20:29:57 IST 2022
/home
[root@localhost home]#

```

4. Then we create a new **file3.txt** and **file4.txt** in **home** directory and add content in it.

- o Now do **echo “Hello I am newline” > file3.txt** and provide the output of file3.txt

- o Now do **echo “Hello I am newline” >> file4.txt** and provide the output of file4.txt

- o Tell the different between both step you follow and the reason behind it

→Ans: Created two files, file3.txt and file4.txt with the help of “touch” command. With the help of “echo” command we inserted the data in both files separately. Below is the **difference** between both commands: “>” and “>>” both are **output direction operators** in Linux.

- A.** echo "Hello I am newline" > file3.txt
Here we used ">" after echo command which refer to **Overwriting of existing file** and creates the file in case it is not present.
- B.** echo "Hello I am newline" >> file4.txt
Here we used ">>" after echo command which refers to **Appending the existing file** at the end and creates the file in case it is not present.

```
[root@localhost home]# touch file3.txt file4.txt
[root@localhost home]# ls
file2.txt file3.txt file4.txt swapnil
[root@localhost home]# echo "Hello I am newline" > file3.txt
[root@localhost home]# cat file3.txt
"Hello I am newline"
[root@localhost home]# echo "Hello I am newline" >> file4.txt
[root@localhost home]# cat file4.txt
"Hello I am newline"
[root@localhost home]# ls;date;pwd
file2.txt file3.txt file4.txt swapnil
Mon Oct 17 20:37:18 IST 2022
/home
[root@localhost home]#
```

- 5.** For remove a file or directory you can use the below two commands
- o To delete a file – **rm <any_filename>**
 - o To delete a directory - **rmdir <any_directoryname>**

→Ans: "rm" command is useful for removing the files and "rmdir" is useful for removing the directories in Linux. To demonstrate this, we will create the file "demo.txt" and the directory "demo", then with use of rm and rmdir commands, we will remove them. PFB screenshot.

```
[swapnil@localhost ~]$ touch demo.txt
[swapnil@localhost ~]$ mkdir demo
[swapnil@localhost ~]$ ls; pwd; date
demo demo.txt Desktop Documents Downloads file1.txt Music Pictures Public Templates Videos
/home/swapnil
Mon Oct 17 20:49:42 IST 2022
[swapnil@localhost ~]$ rm demo.txt
[swapnil@localhost ~]$ ls
demo Desktop Documents Downloads file1.txt Music Pictures Public Templates Videos
[swapnil@localhost ~]$ rmdir demo
[swapnil@localhost ~]$ ls
Desktop Documents Downloads file1.txt Music Pictures Public Templates Videos
[swapnil@localhost ~]$
```