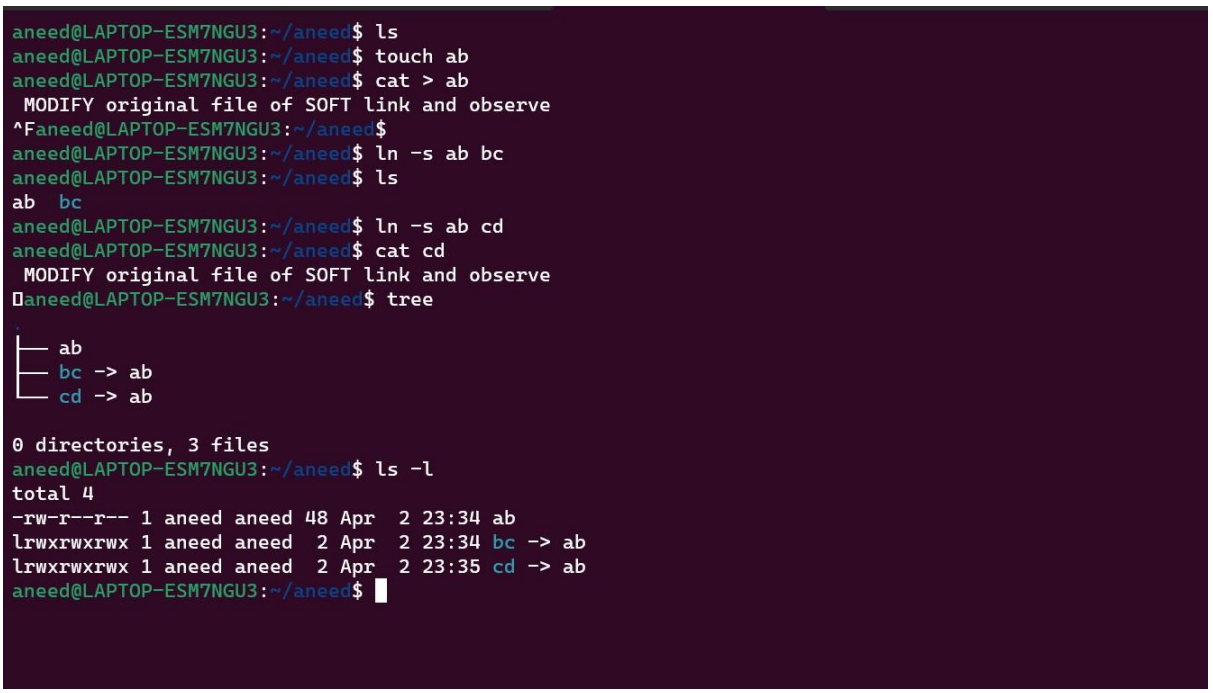
Assignment

links-

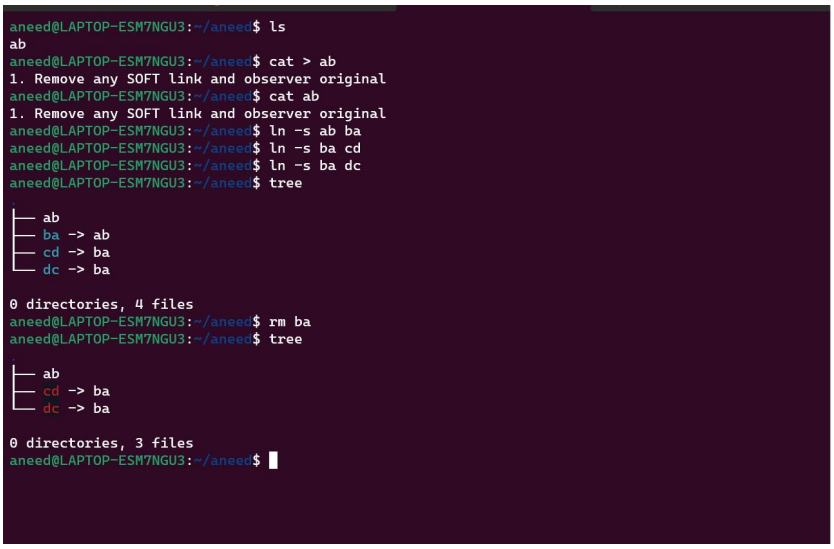
(Soft-link)

1. MODIFY any SOFT link and observer original?

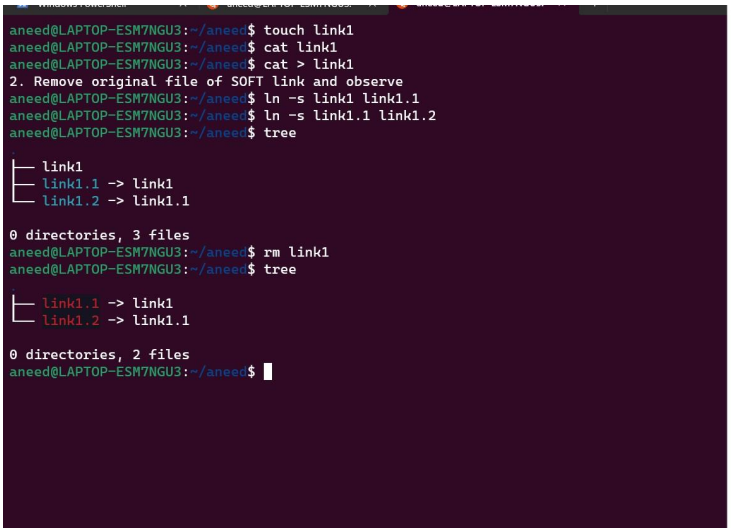
2. MODIFY original file of SOFT link and observe?



3. Remove any SOFT link and observer original?



4. Remove original file of SOFT link and observe?



(Hard-link)

1. MODIFY any HARD link and observer original?

```
aneed@LAPTOP-ESM7NGU3:~$ mkdir khan
aneed@LAPTOP-ESM7NGU3:~$ cd khan
aneed@LAPTOP-ESM7NGU3:~/khan$ touch ab
aneed@LAPTOP-ESM7NGU3:~/khan$ cat > ab
1. MODIFY any HARD link and observer original?
aneed@LAPTOP-ESM7NGU3:~/khan$
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab bc
aneed@LAPTOP-ESM7NGU3:~/khan$ ln bc cd
aneed@LAPTOP-ESM7NGU3:~/khan$ ln cd ef
aneed@LAPTOP-ESM7NGU3:~/khan$ cat ef
1. MODIFY any HARD link and observer original?
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── ab
├── bc
├── cd
└── ef

0 directories, 4 files
aneed@LAPTOP-ESM7NGU3:~/khan$
```

2. MODIFY Original file of HARD link and observe?

```
aneed@LAPTOP-ESM7NGU3:~/khan$ cd
aneed@LAPTOP-ESM7NGU3:~$ rm -rf khan
aneed@LAPTOP-ESM7NGU3:~$ mkdir khan
aneed@LAPTOP-ESM7NGU3:~$ cd khan
aneed@LAPTOP-ESM7NGU3:~/khan$ touch ab
aneed@LAPTOP-ESM7NGU3:~/khan$ cat > ab
2. MODIFY Original file of HARD link and observe?
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab bc
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab cd
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab ef
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── ab
├── bc
├── cd
└── ef

0 directories, 4 files
aneed@LAPTOP-ESM7NGU3:~/khan$
```

3. Remove any HARD link and observer original?

```
aneed@LAPTOP-ESM7NGU3:~/khan$ cd
aneed@LAPTOP-ESM7NGU3:~$ rm -rf khan
aneed@LAPTOP-ESM7NGU3:~$ mkdir khan
aneed@LAPTOP-ESM7NGU3:~$ cd khan
aneed@LAPTOP-ESM7NGU3:~/khan$ touch ab
aneed@LAPTOP-ESM7NGU3:~/khan$ cat > ab
2. MODIFY Original file of HARD link and observe?
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab bc
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab cd
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab ef
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── ab
├── bc
├── cd
└── ef

0 directories, 4 files
aneed@LAPTOP-ESM7NGU3:~/khan$ rm ef
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── ab
├── bc
└── cd

0 directories, 3 files
aneed@LAPTOP-ESM7NGU3:~/khan$
```

4. Remove original file of HARD link and observe?

```
aneed@LAPTOP-ESM7NGU3:~/khan$ cd
aneed@LAPTOP-ESM7NGU3:~$ rm -rf khan
aneed@LAPTOP-ESM7NGU3:~$ mkdir khan
aneed@LAPTOP-ESM7NGU3:~$ cd khan
aneed@LAPTOP-ESM7NGU3:~/khan$ touch ab
aneed@LAPTOP-ESM7NGU3:~/khan$ cat > ab
2. MODIFY Original file of HARD link and observe?
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab bc
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab cd
aneed@LAPTOP-ESM7NGU3:~/khan$ ln ab ef
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── ab
├── bc
├── cd
└── ef

0 directories, 4 files
aneed@LAPTOP-ESM7NGU3:~/khan$ rm ef
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── ab
├── bc
└── cd

0 directories, 3 files
aneed@LAPTOP-ESM7NGU3:~/khan$ rm ab
aneed@LAPTOP-ESM7NGU3:~/khan$ tree
.
├── bc
└── cd

0 directories, 2 files
```

9. Comparison among soft,hard,copy?

Hard Link :

A hard link acts as a copy (mirrored) of the selected file. It accesses the data available in the original file. If the earlier selected file is deleted, the hard link to the file will still contain the data of that file.

Soft Link :

A soft link (also known as Symbolic link) acts as a pointer or a reference to the file name. It does not access the data available in the original file. If the earlier file is deleted, the soft link will be pointing to a file that does not exist anymore.

Copy :

cp command for copying files from one location to another. This command can also copy directories (folders). The syntax of this command is: cp [... file/directory-sources] [destination] [file/directory-sources] specifies the sources of the files or directories you want to copy.

10. what are wget , curl commands differences?

Curl and wget both have unique features that make them suitable for different tasks.

Curl, for example, is often used to interact with APIs and can handle a variety of data formats. It also supports authentication, cookies, and more. Here are some of key features of curl –

Supports a wide range of protocols, including HTTP, HTTPS, FTP, FTPS, SCP, SFTP, and more.

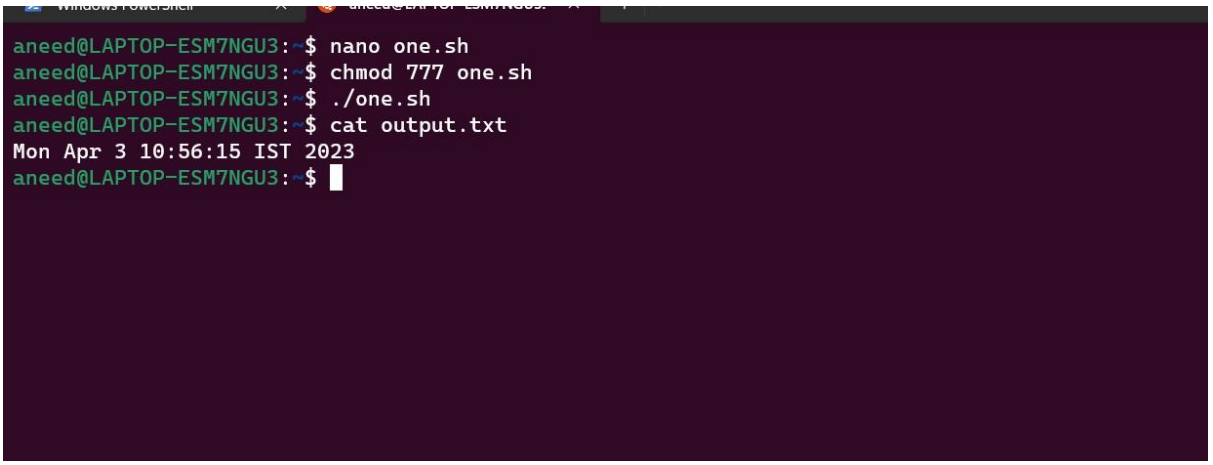
Wget, on other hand, is primarily designed for downloading files and has a number of features that make it a reliable tool for this task. Here are some of key features of wget –

Supports HTTP and FTP protocols.

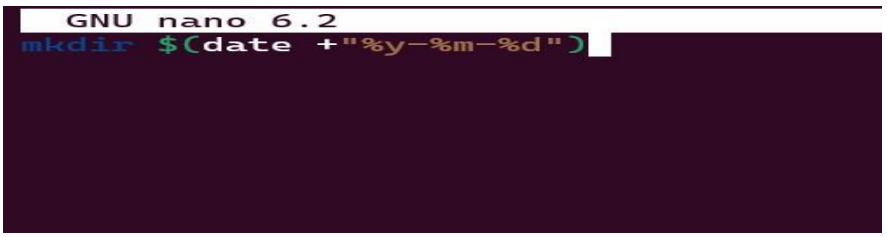
Can download recursively to download all linked files.

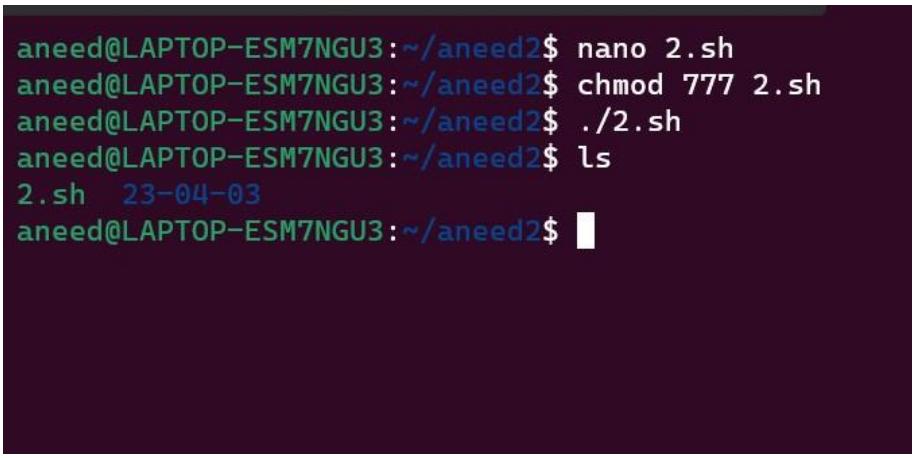
Scripts-

1. write a script to print the date and redirect it to output.txt?

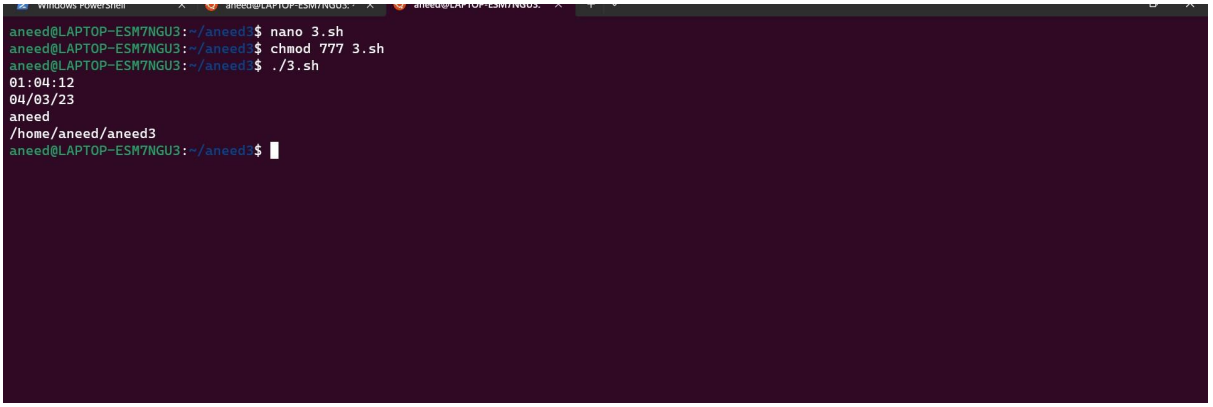


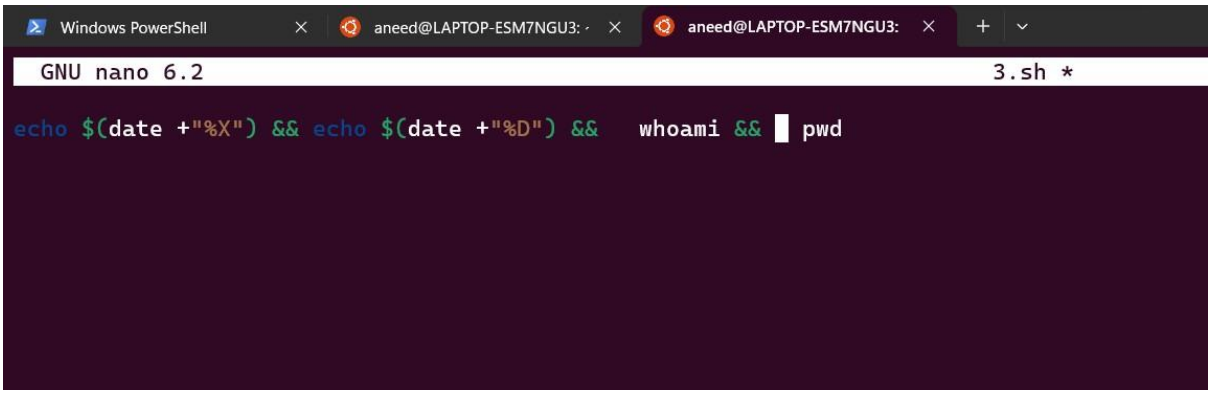
2. create a file or folder using the date as the name?



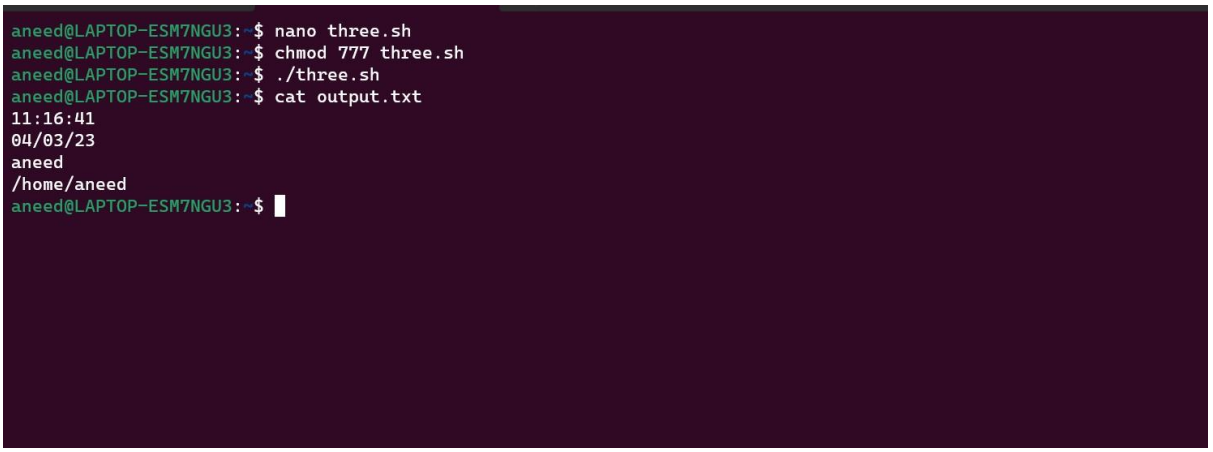


3. i) Create a bash script to print the local time, date, username of your system, and your current path.





ii) After printing, redirect the output into a file called output.txt



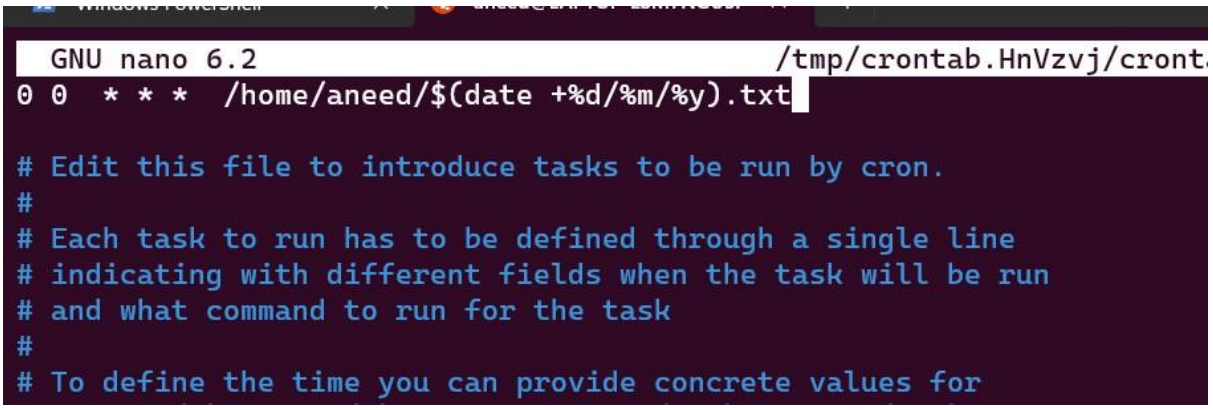
iii) Insert output.txt into a new directory, where the directory name is the current timestamp.

```
aneed@LAPTOP-ESM7NGU3:~/aneed4$ cd
aneed@LAPTOP-ESM7NGU3:~$ mkdir  $(date +"%X") && mv output.txt $(date +"%X")
aneed@LAPTOP-ESM7NGU3:~$ ls
 1           Apr                             ahmed    aneed5   file1          one.sh
 10:36:53    IST                             aneed    b        first.sh       ouput.txt
 11:56:22    Sat                             aneed2   c        first.sh.save  out.txt
 3.sh        'Sat Apr  1 10:33:40 IST 2023'   aneed3   dat.sh   khan           test1
 9.txt.save  a                               aneed4   file     my.txt         three.sh
aneed@LAPTOP-ESM7NGU3:~$ ls
 1           Apr                             ahmed    aneed5   file1          one.sh
 10:36:53    IST                             aneed    b        first.sh       ouput.txt
 11:56:22    Sat                             aneed2   c        first.sh.save  out.txt
 3.sh        'Sat Apr  1 10:33:40 IST 2023'   aneed3   dat.sh   khan           test1
 9.txt.save  a                               aneed4   file     my.txt         three.sh
aneed@LAPTOP-ESM7NGU3:~$
```

4. Create a bash script to execute the date every 2 minutes once on Saturdays only.

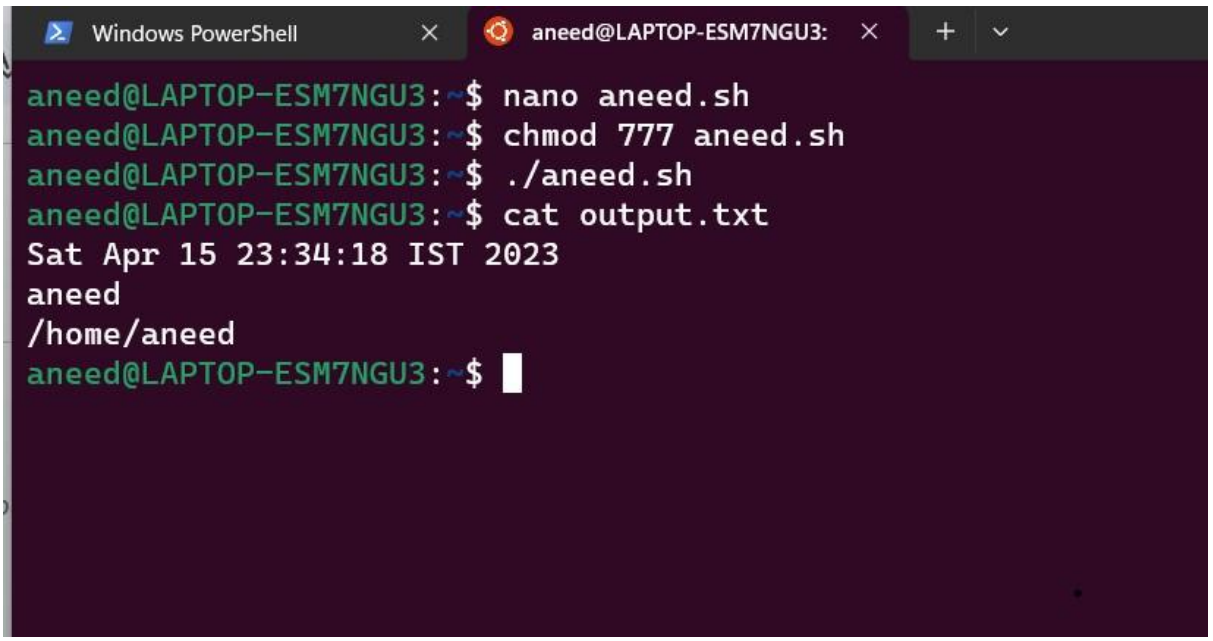
6. Print the count of the number of files in a file called count.txt?

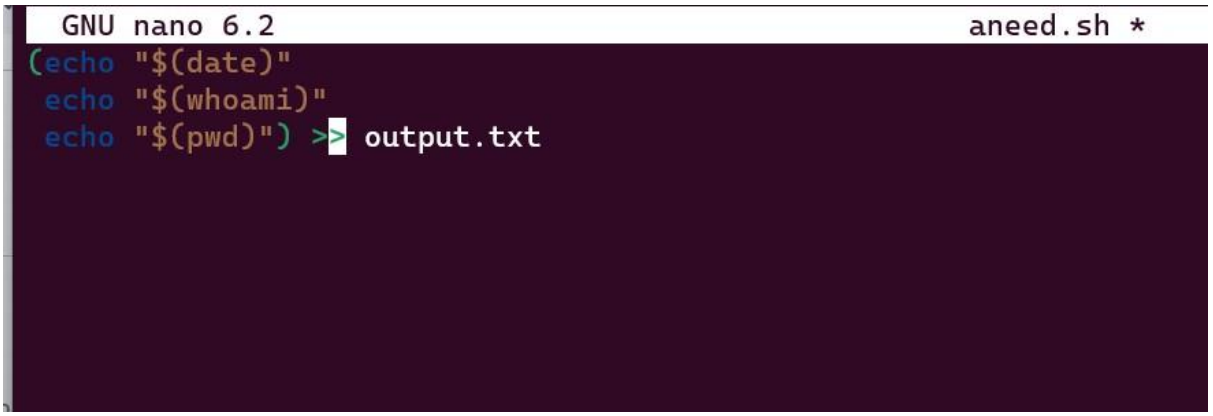
7. Create files dynamically every day at 12 AM where the file name is a date?



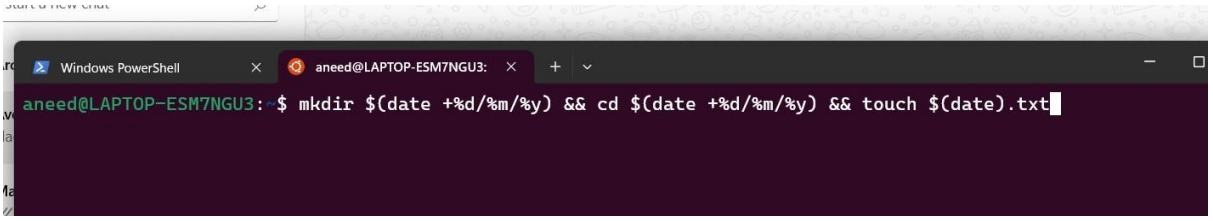
CRONTAB SCRIPTS LABS:

1. Write a script to print the current directory and username and redirect it to a file called output.txt?





2. Create a file with the current timestamp as its name inside a folder with the current date as its name?



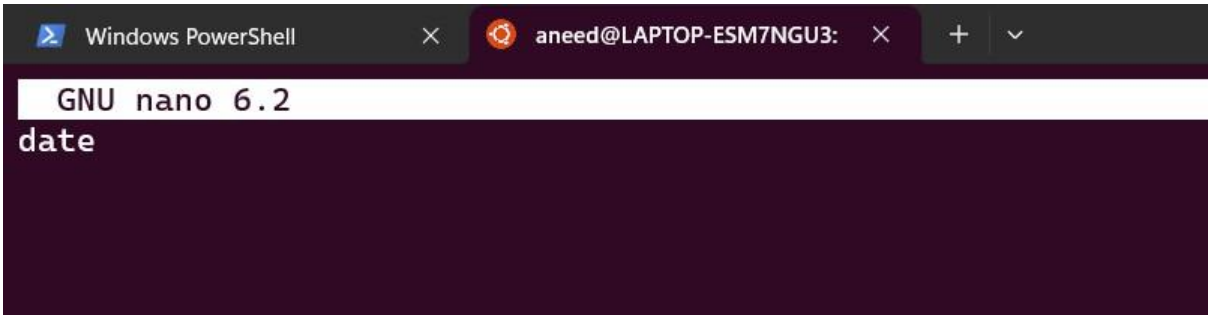
3. Create a bash script to print the local time, date, username of your system, and your current path and redirect the output into a your system, and your current path and redirect the output into a where the directory name is the current timestamp.

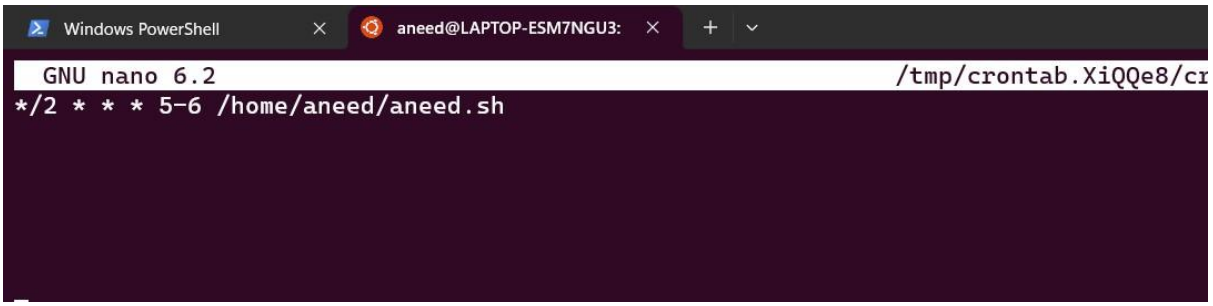
```
 GNU nano 6.2
(echo "$(date)"
 echo "$(whoami)"
 echo "$(pwd)") >> output.txt

mkdir $(date)
mv output.txt $(date)
```

4. Write a script to print the count of the number of files in a folder and redirect the count to a file called count.txt.

(Reference: Google about word count and pipe commands in Linux)

5. Create a bash script to execute the date every 2 minutes once on weekends only?





6. Take a backup of a folder daily twice?