Data Analysis with Machine Learning (DA A23)

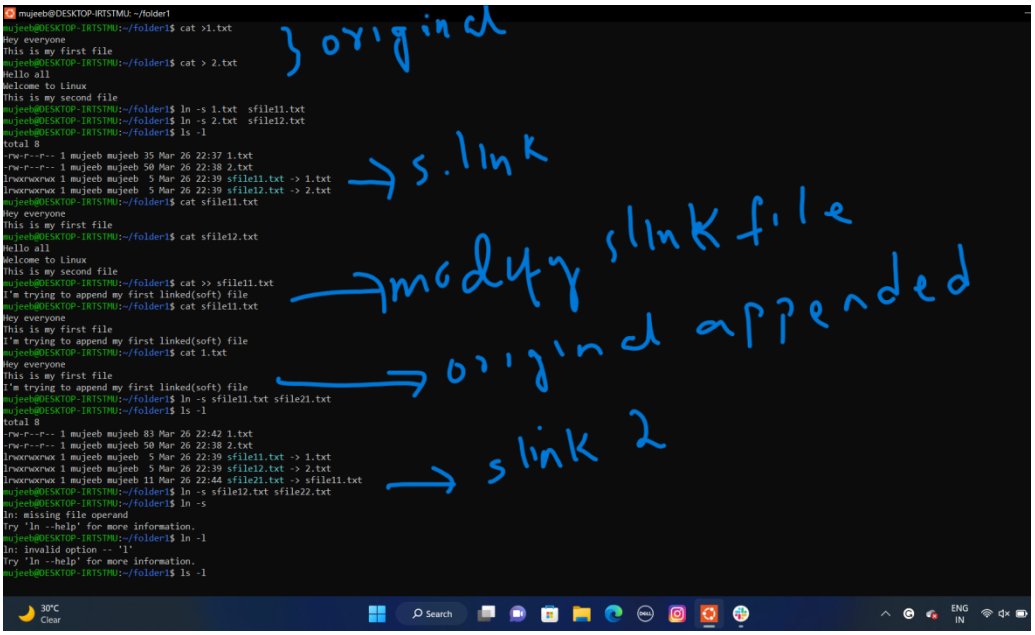
Assignment-2(Linux)

Links-

1. MODIFY any SOFT link and observer original?

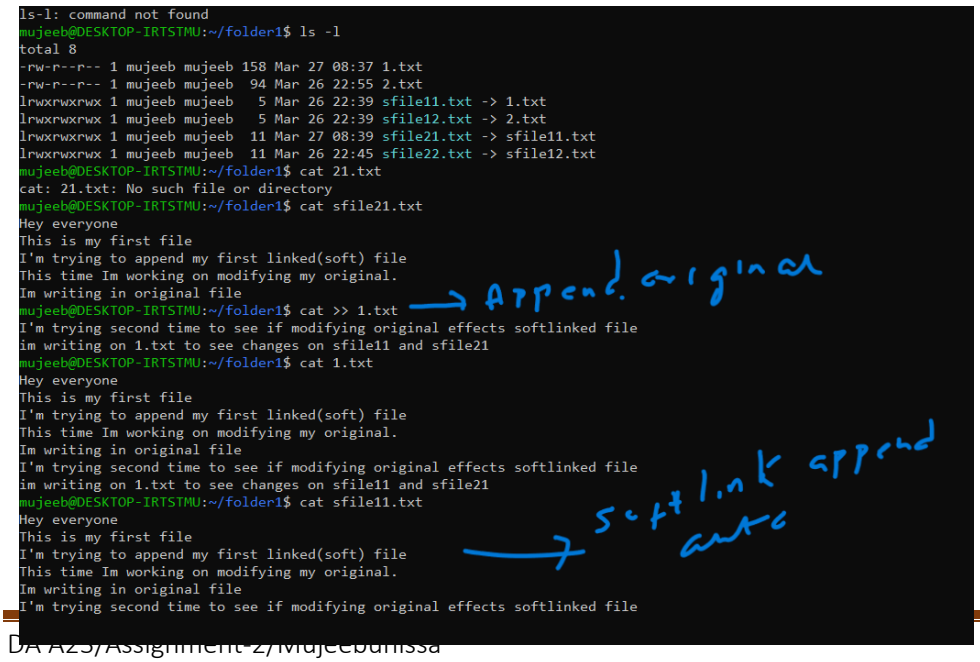
Soft links are like pointers to the files. Creating links is like making shortcuts to access a file.

If soft link file is modified/ appended, the change is reflected in original as well.



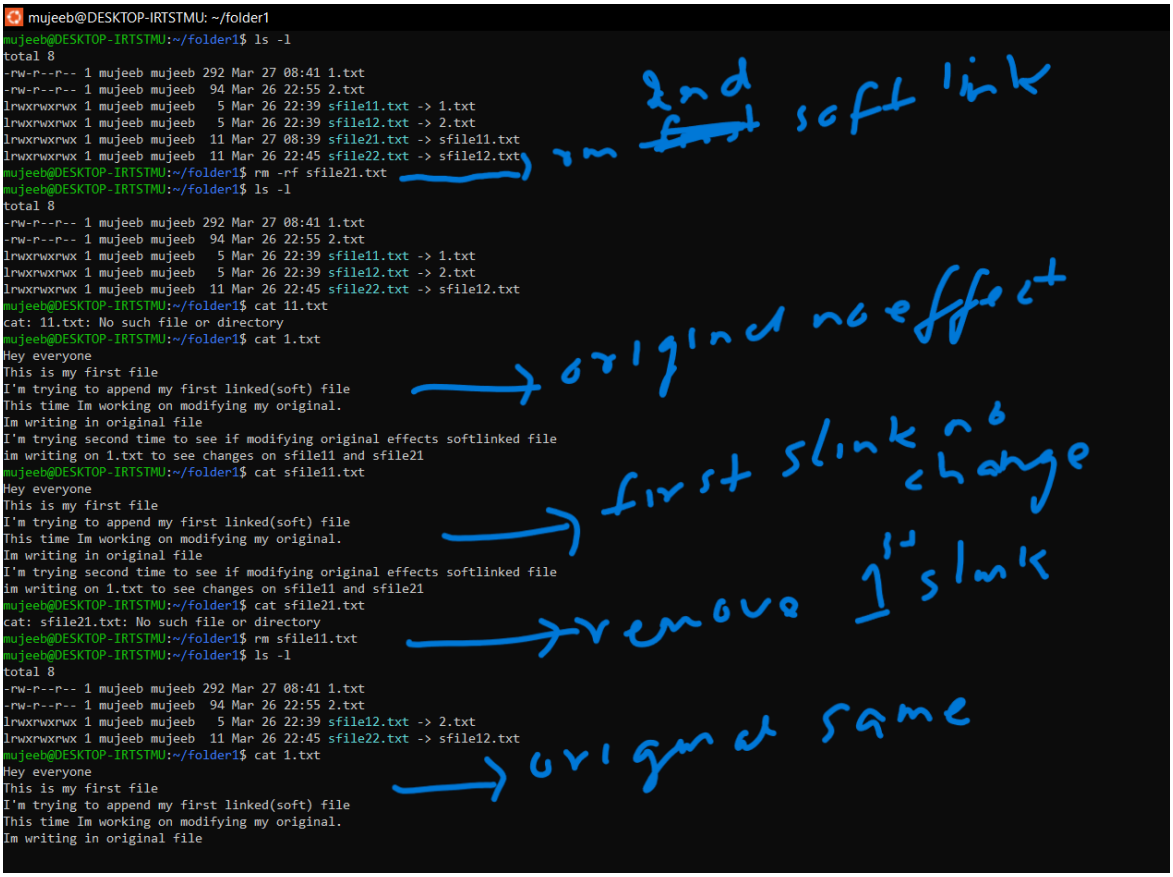
2. MODIFY original file of SOFT link and observe?

If modification is done on original, the linked file gets appended as well since the soft link is like backup of original. Any changes made in content will reflect on file as well



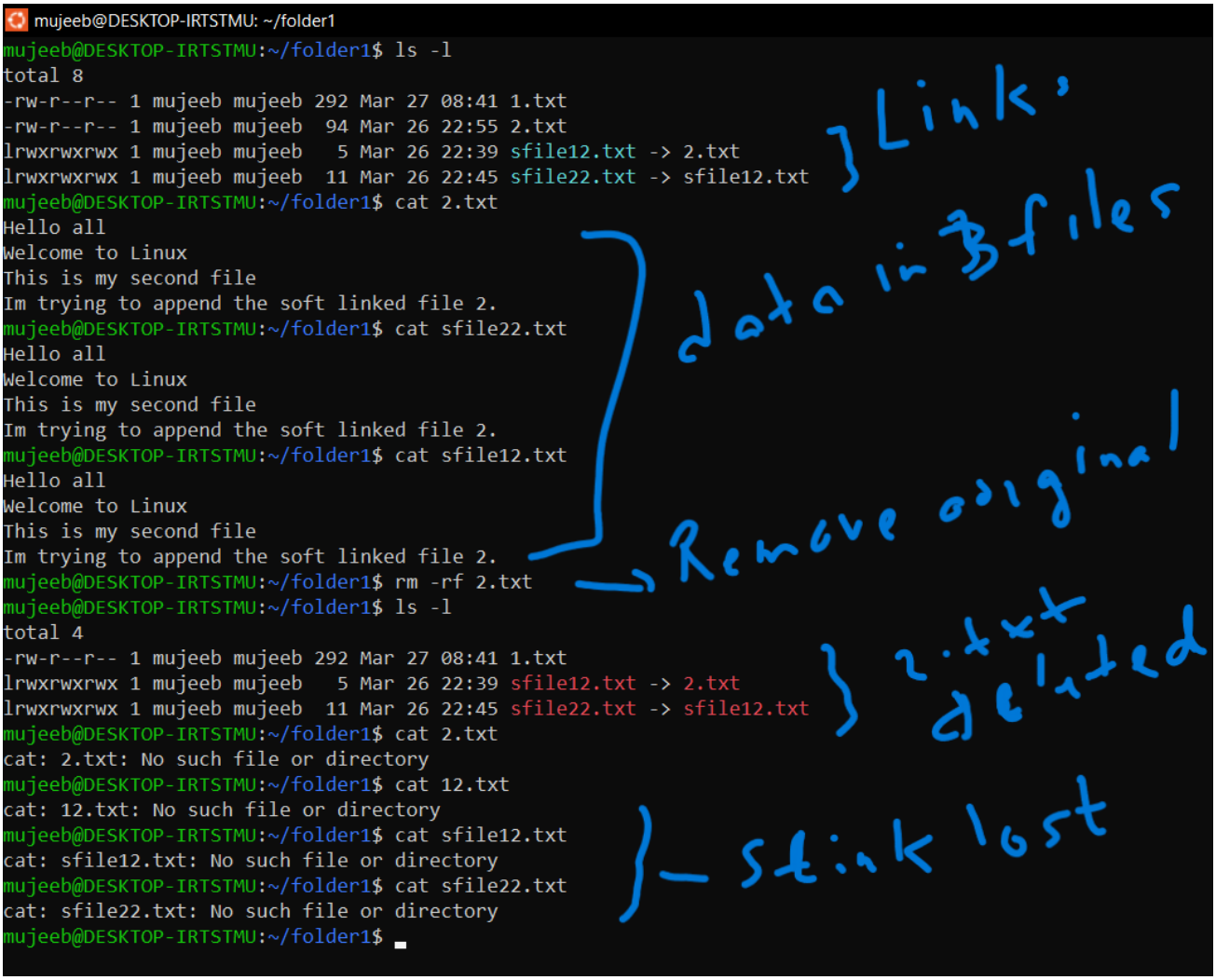
3. Remove any SOFT link and observer original?

If soft link is removed no effect on original file was observed.



4. Remove original file of SOFT link and observe?

Soft links are invalid when original file is deleted. Soft link is like shortcut, so if you delete the source file the shortcut is useless.



5. MODIFY any HARD link and observer original?

When a hardlink file is appended(11.txt),the change is reflected in original file.

```
mujeeb@DESKTOP-IRTSTMU:~$ mkdir Hlinkfolder
mujeeb@DESKTOP-IRTSTMU:~$ cd Hlinkfolder
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ touch 1h.txt 2h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ ln 1h.txt 11h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ ln 2h.txt 22h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ ls -l
total 0
-rw-r--r-- 2 mujeeb mujeeb 0 Mar 30 10:22 11h.txt
-rw-r--r-- 2 mujeeb mujeeb 0 Mar 30 10:22 1h.txt
-rw-r--r-- 2 mujeeb mujeeb 0 Mar 30 10:22 22h.txt
-rw-r--r-- 2 mujeeb mujeeb 0 Mar 30 10:22 2h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ cat > 1.txt
hey my hlink original
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ cat > 2.txt
Hello my hlink second original file
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ cat > 11h.txt
I'm appending my hlink file to see the change
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ cat 1h.txt
I'm appending my hlink file to see the change
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$
```

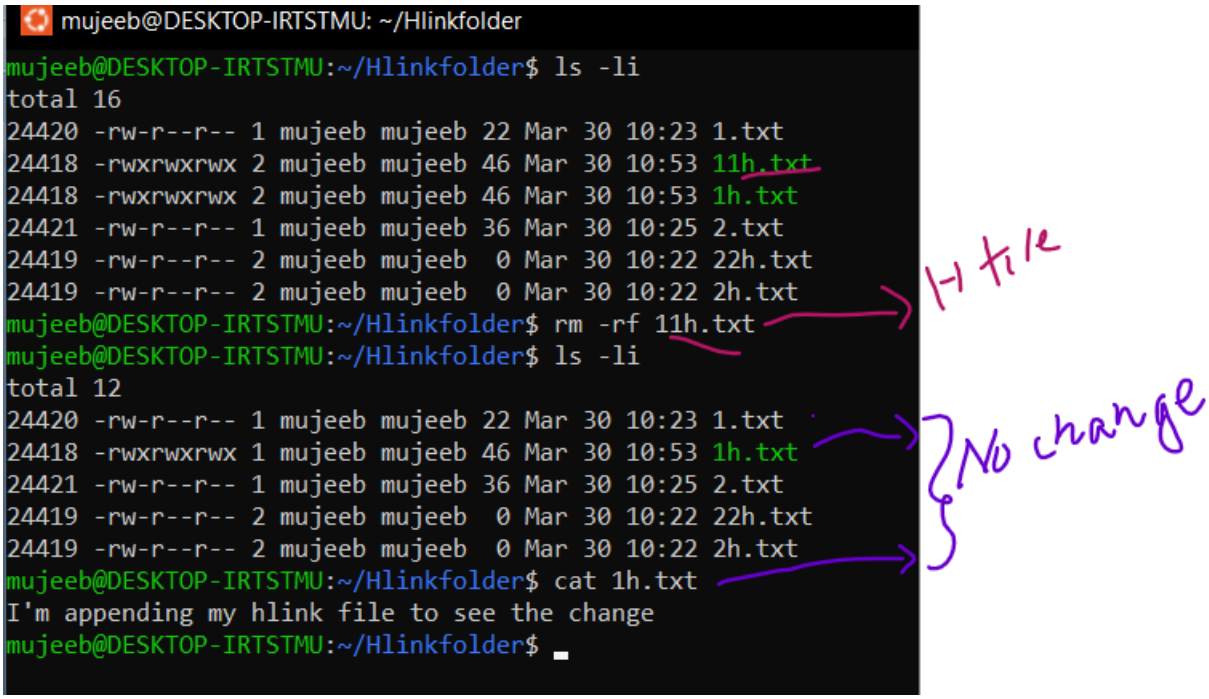
Append hlink

6. MODIFY Original file of HARD link and observe?

Unlike softlink, in hardlink the source file and the linked file share the same Inode and permissions.If the source file is modified such as file permissions are changed we observe the same has been reflected in hard linked file

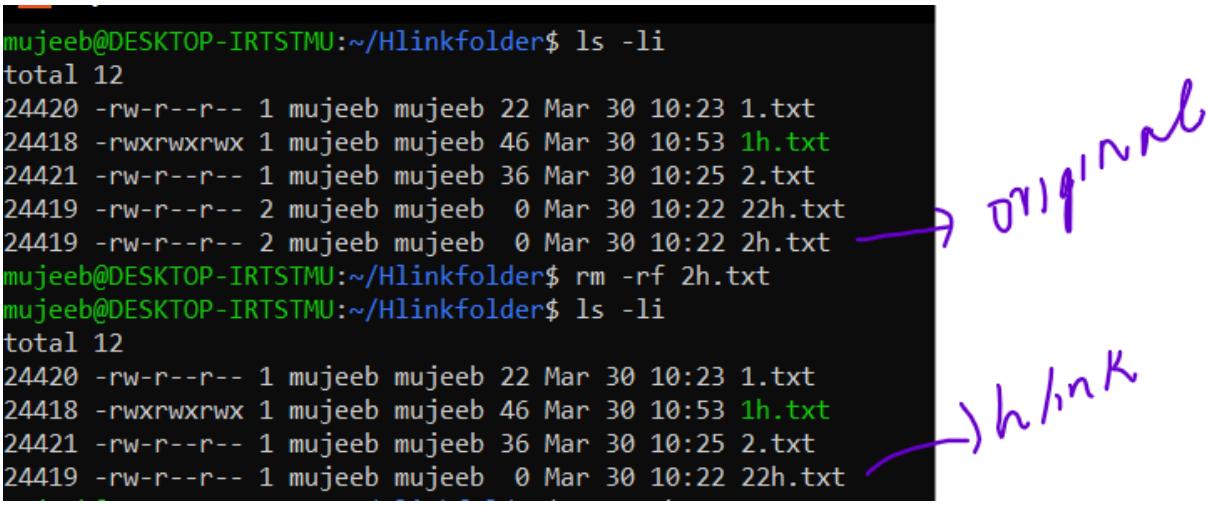
```
mujeeb@DESKTOP-IRTSTMU: ~/Hlinkfolder
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ ls -li
total 16
24420 -rw-r--r-- 1 mujeeb mujeeb 22 Mar 30 10:23 1.txt
24418 -rw-r--r-- 2 mujeeb mujeeb 46 Mar 30 10:53 11h.txt
24418 -rw-r--r-- 2 mujeeb mujeeb 46 Mar 30 10:53 1h.txt
24421 -rw-r--r-- 1 mujeeb mujeeb 36 Mar 30 10:25 2.txt
24419 -rw-r--r-- 2 mujeeb mujeeb  0 Mar 30 10:22 22h.txt
24419 -rw-r--r-- 2 mujeeb mujeeb  0 Mar 30 10:22 2h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ chmod 777 1h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$ ls -li
total 16
24420 -rw-r--r-- 1 mujeeb mujeeb 22 Mar 30 10:23 1.txt
24418 -rwxrwxrwx 2 mujeeb mujeeb 46 Mar 30 10:53 11h.txt
24418 -rwxrwxrwx 2 mujeeb mujeeb 46 Mar 30 10:53 1h.txt
24421 -rw-r--r-- 1 mujeeb mujeeb 36 Mar 30 10:25 2.txt
24419 -rw-r--r-- 2 mujeeb mujeeb  0 Mar 30 10:22 22h.txt
24419 -rw-r--r-- 2 mujeeb mujeeb  0 Mar 30 10:22 2h.txt
mujeeb@DESKTOP-IRTSTMU:~/Hlinkfolder$
```

7. Remove any HARD link and observer original?



8. Remove original file of HARD link and observe?

If we remove the original file,the hard linked file still exists there's no change



9. Comparision among soft,hard,copy?

Links are like pointers to a file in Linux. In a way we can say there are like creating backups for your files. However the copy command is quite different compared to links because in copy the date is copied but if any modifications done on source will not be copied automatically. Every time we modify we need to copy the file. To ease this job if we want to take a backup we can use these linked files. Modification to content is done automatically irrespective of source or linked file. The only difference is in softlink if original gets deleted the link are useless, its viceversa in case of hardlink if source is deleted linked file still exists.

10. What are wget , curl commands differences?

Wget and curl commands allow you to transfer data from a network server, with curl being the more robust. They are used to copy/download data from internet sources such as http or https or so on. The difference between wget and curl is if we need to redirect the downloaded content in a file the we can only use curl command as wget fails to redirect url content in file.

Scripts-

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

```
mujeeb@DESKTOP-IRTSTMU:~$ echo $(date)
Mon Mar 27 10:23:03 IST 2023
mujeeb@DESKTOP-IRTSTMU:~$ echo $(date) > output.ttxt
mujeeb@DESKTOP-IRTSTMU:~$ cat output.txt
cat: output.txt: No such file or directory
mujeeb@DESKTOP-IRTSTMU:~$ cat output.ttxt
Mon Mar 27 10:23:29 IST 2023
mujeeb@DESKTOP-IRTSTMU:~$
```

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

```
mujeeb@DESKTOP-IRTSTMU:~$ touch ($date +%d-%m-%y)
-bash: syntax error near unexpected token `$date'
mujeeb@DESKTOP-IRTSTMU:~$ touch $(date +%d-%m-%y)
mujeeb@DESKTOP-IRTSTMU:~$ ls
1       10:31:19      12zAB.txt   2023       33.txt     Mar        bc1.txt           fila        fileab      'index.php?utm_source=Google'   output.ttxt    testdir
1.c     11.txt        17.txt      27         ABC.txt    Mon        copiedtext1.txt   file.txt    fileb       log1.txt                        outt.txt       text1.txt
1.js    123.txt       18.txt      27-03-23   Abc.txt    a.out      dir1              file1.txt   filec       names.txt                       p.txt          typing.txt
1.py    12a.txt       1a.txt      3.txt      BC12.txt   a123.txt   dir2              file2.txt   folder1     newdir                          pytonlog.txt   typing2.txt
1.txt   12cccAB.txt   2.txt       31.txt     IST        abc.txt    error.txt         file4.txt   index.html  out.txt                         tesdir         wget-log
mujeeb@DESKTOP-IRTSTMU:~$ touch $(date +%d-%m-%y-%T)
mujeeb@DESKTOP-IRTSTMU:~$
mujeeb@DESKTOP-IRTSTMU:~$ ls
1       10:31:19      12zAB.txt   2023               31.txt     IST        abc.txt           error.txt   file4.txt   index.html                      out.txt        tesdir         wget-log
1.c     11.txt        17.txt      27                 33.txt     Mar        bc1.txt           fila        fileab      'index.php?utm_source=Google'   output.ttxt    testdir
1.js    123.txt       18.txt      27-03-23           ABC.txt    Mon        copiedtext1.txt   file.txt    fileb       log1.txt                        outt.txt       text1.txt
1.py    12a.txt       1a.txt      27-03-23-10:34:14  Abc.txt    a.out      dir1              file1.txt   filec       names.txt                       p.txt          typing.txt
1.txt   12cccAB.txt   2.txt       3.txt              BC12.txt   a123.txt   dir2              file2.txt   folder1     newdir                          pytonlog.txt   typing2.txt
mujeeb@DESKTOP-IRTSTMU:~$
```

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

```
mujeeb@DESKTOP-IRTSTMU: ~
  GNU nano 6.2                                                scr5.sh
echo $(date +%D) && echo $(date +%T) && (whoami) && (pwd)
```

```
mujeeb@DESKTOP-IRTSTMU: ~
mujeeb@DESKTOP-IRTSTMU:~$ ./scr5.sh
mujeeb@DESKTOP-IRTSTMU:~$
./scr5.sh: line 1: `echo (date +%D) && echo (date +%T) && (whoami) && (pwd) > OUtput.txt'
mujeeb@DESKTOP-IRTSTMU:~$ nano scr5.sh
mujeeb@DESKTOP-IRTSTMU:~$ ./scr5.sh
./scr5.sh: line 1: syntax error near unexpected token `date'
./scr5.sh: line 1: `echo (date +%D) && echo (date +%T) && (whoami) && (pwd) '
mujeeb@DESKTOP-IRTSTMU:~$ nano scr5.sh
mujeeb@DESKTOP-IRTSTMU:~$ ./scr5.sh
03/28/23
05:57:20
mujeeb
/home/mujeeb
mujeeb@DESKTOP-IRTSTMU:~$
```

o/p q3i

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

```
mujeeb@DESKTOP-IRTSTMU: ~
  GNU nano 6.2
(echo $(date +%D) && echo $(date +%T) && (whoami) && (pwd)) > OUtput.txt
```

```
mujeeb@DESKTOP-IRTSTMU:~$ nano scr6.sh
mujeeb@DESKTOP-IRTSTMU:~$ chmod +x scr6.sh
mujeeb@DESKTOP-IRTSTMU:~$ ./scr6.sh
03/28/23
06:10:24
mujeeb
mujeeb@DESKTOP-IRTSTMU:~$ nano scr6.sh
mujeeb@DESKTOP-IRTSTMU:~$ ./scr6.sh
mujeeb@DESKTOP-IRTSTMU:~$ ls
1.c         14:14:45            OUtput.txt        dat.txt   dt.txt      fila      folder.sh   index.html                  out.txt       p.txt       scr2.sh   scr6.sh    text1.txt
1.js        14:15:18            Output.txt        dd.txt    e.txt       fileab    folder1     index.html.1                output.ttxt   posts       scr3.sh   scrdat.sh  typing.txt
1.py        14:20:02            a.out             dir1      error.txt   fileb     hello.txt   'index.php?utm_source=Google'  output.txt    pytonlog.txt scr4.sh  tesdir     typing2.txt
10:31:19    27-03-23-10:34:14   copiedtext1.txt   dir2      f.txt       filec     hi.txt      names.txt                   outt.txt      scr1.sh     scr5.sh   testdir    wget-log
mujeeb@DESKTOP-IRTSTMU:~$ cat OUtput.txt
03/28/23
06:11:08
mujeeb
/home/mujeeb
mujeeb@DESKTOP-IRTSTMU:~$
```

o/p 3ii

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

```
mujeeb@DESKTOP-IRTSTMU:~$ mkdir $(date +%d-%m-%Y,%H:%M:%S) && mv OUTPUT.txt $(date +%d-%m-%Y,%H:%M:%S)
mujeeb@DESKTOP-IRTSTMU:~$ ls
1.c         23:27:57            BACKup.zip  HI.txt       dir2          hku          pytonlog.txt  scr5.sh
1.js        29-03-2023,10:51    Backup.zip  Hlinkfolder  dt.txt        log          samplegit     scr8.sh
1.py        29-03-2023,10:53    Bla         a.out        dummy         mygit        scr1.sh       scrdat.sh
21:00:54    31-03-2023          Count       backup       dynamic.sh    mygit3       scr10.sh      scrtim.sh
21:01:09    31-03-2023,00:14:48 Count.sh    backup.sh    filefolder.sh mysecondgit  scr2.sh       testdir
23-21-40    31-03-2023,00:16    D.txt       date.txt     folder.sh     output.txt   scr3.sh       timefolder.sh
23:27:35    31-03-2023,10:30:43 Dat.sh      dir1         folder1       posts        scr4.sh       wget-log
mujeeb@DESKTOP-IRTSTMU:~$ cd 31-03-2023,10:30:43
mujeeb@DESKTOP-IRTSTMU:~/31-03-2023,10:30:43$ ls
OUTPUT.txt
```

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

```
mujeeb@DESKTOP-IRTSTMU: ~
  GNU nano 6.2                                                Dat.sh *
(echo $(date +"%d-%m-%Y")) >> D.txt
```

```
*/2 * * * sat /home/mujeeb/Dat.sh
```

```
mujeeb@DESKTOP-IRTSTMU:~$ nano Dat.sh
mujeeb@DESKTOP-IRTSTMU:~$ ./Dat.sh
mujeeb@DESKTOP-IRTSTMU:~$ ls
1.c       23:27:35            BACKup.zip  Dat.sh       date.txt    filefolder.sh  mygit3       scr1.sh  scrdat.sh
1.js      23:27:57            Backup.zip  HI.txt       dir1        folder.sh      mysecondgit  scr2.sh  scrtim.sh
1.py      29-03-2023,10:51    Bla         Hlinkfolder  dir2        folder1        output.txt   scr3.sh  testdir
21:00:54  29-03-2023,10:53    Count       a.out        dt.txt      hku            posts        scr4.sh  timefolder.sh
21:01:09  31-03-2023,00:14:48  Count.sh    backup       dummy       log            pytonlog.txt scr5.sh  wget-log
23-21-40  31-03-2023,00:16    D.txt       backup.sh    dynamic.sh  mygit          samplegit    scr8.sh
mujeeb@DESKTOP-IRTSTMU:~$ cat D.txt
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
28-03-2023
22:26:58
31-03-2023
mujeeb@DESKTOP-IRTSTMU:~$
```

o/p

5. Take a backup of a folder every month twice?

```
mujeeb@DESKTOP-IRTSTMU: ~
  GNU nano 6.2                                    backup.sh
zip -r ~/BACKup ~/log/*
```

```
* 8 1,15 * * /home/mujeeb/backup.sh
```

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

```
mujeeb@DESKTOP-IRTSTMU:~$ mkdir Count
mujeeb@DESKTOP-IRTSTMU:~$ cd Count
mujeeb@DESKTOP-IRTSTMU:~/Count$ touch f1.txt f2.txt f3 f4 f5
mujeeb@DESKTOP-IRTSTMU:~/Count$ ls
f1.txt  f2.txt  f3  f4  f5
mujeeb@DESKTOP-IRTSTMU:~/Count$ ls -1|wc -l
5
mujeeb@DESKTOP-IRTSTMU:~/Count$
```

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

```
mujeeb@DESKTOP-IRTSTMU: ~
  GNU nano 6.2                                    dynamic.sh
touch $(date +"%d-%m-%Y")
```

```
0 0 * * * /home/mujeeb/dynamic.sh
```

```
mujeeb@DESKTOP-IRTSTMU:~$ nano dynamic.sh
mujeeb@DESKTOP-IRTSTMU:~$ ./dynamic.sh
mujeeb@DESKTOP-IRTSTMU:~$ ls
1.c         23:27:57              Backup.zip   Hlinkfolder
1.js        29-03-2023,10:51      Bla          a.out
1.py        29-03-2023,10:53      Count        backup
21:00:54    31-03-2023            Count.sh     backup.sh
21:01:09    31-03-2023,00:14:48   D.txt        date.txt
23-21-40    31-03-2023,00:16      Dat.sh       dir1
23:27:35    BACKup.zip            HI.txt       dir2
```

CRONTAB SCRIPTS LABS:

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

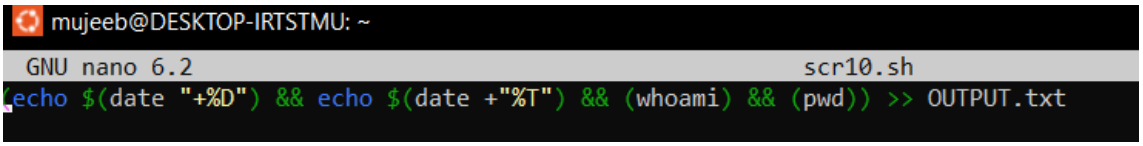
```
mujeeb@DESKTOP-IRTSTMU: ~
 GNU nano 6.2                                          scr2.sh
(whoami&& pwd)>output.txt
```

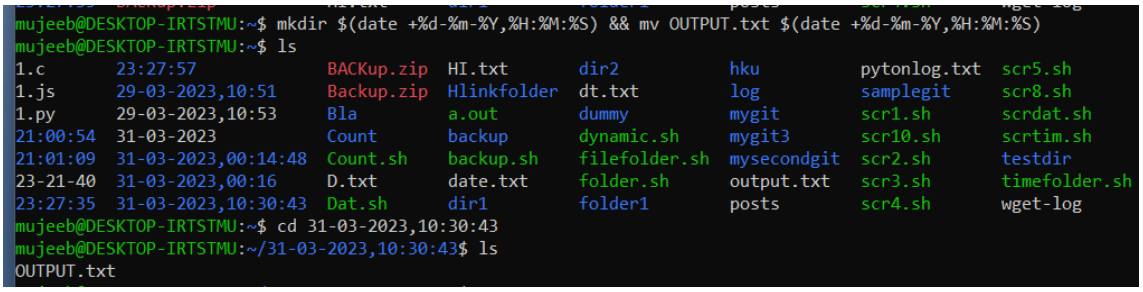
```
mujeeb@DESKTOP-IRTSTMU: ~
mujeeb@DESKTOP-IRTSTMU:~$ cat output.txt
mujeeb
/home/mujeeb
mujeeb@DESKTOP-IRTSTMU:~$
```

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

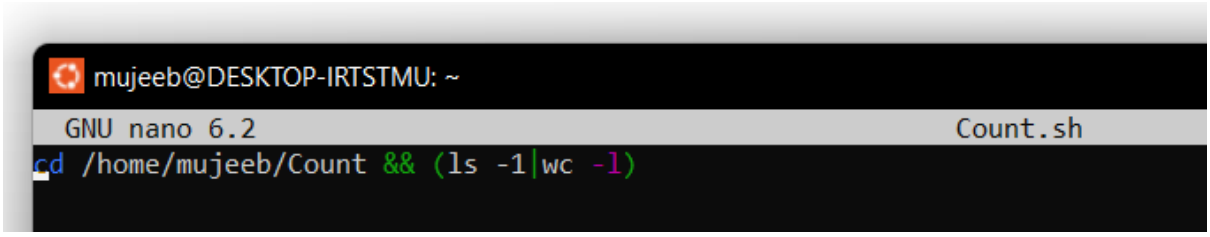
```
mujeeb@DESKTOP-IRTSTMU: ~/03-04-23
mujeeb@DESKTOP-IRTSTMU:~$ mkdir $(date +"%d-%m-%y") && cd $(date +"%d-%m-%y")
mujeeb@DESKTOP-IRTSTMU:~/03-04-23$ touch $(date +"%H:%M:%S")
mujeeb@DESKTOP-IRTSTMU:~/03-04-23$ LS
LS: command not found
mujeeb@DESKTOP-IRTSTMU:~/03-04-23$ ls
11:53:27
mujeeb@DESKTOP-IRTSTMU:~/03-04-23$
```

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 file called output.txt. Insert output.txt into a new directory, where the directory name is the current timestamp.

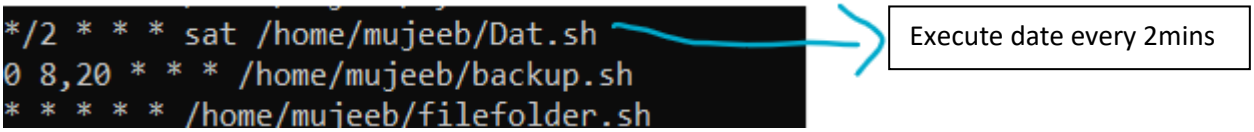


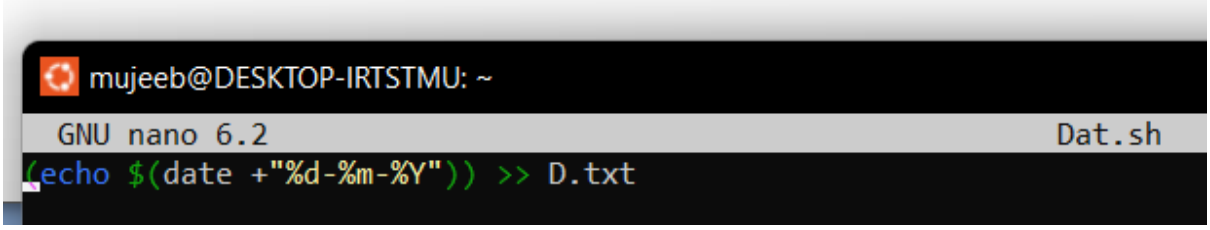


4. Write a script to print the count of the number of files in a folder and redirect the count to a file called count.txt.



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?