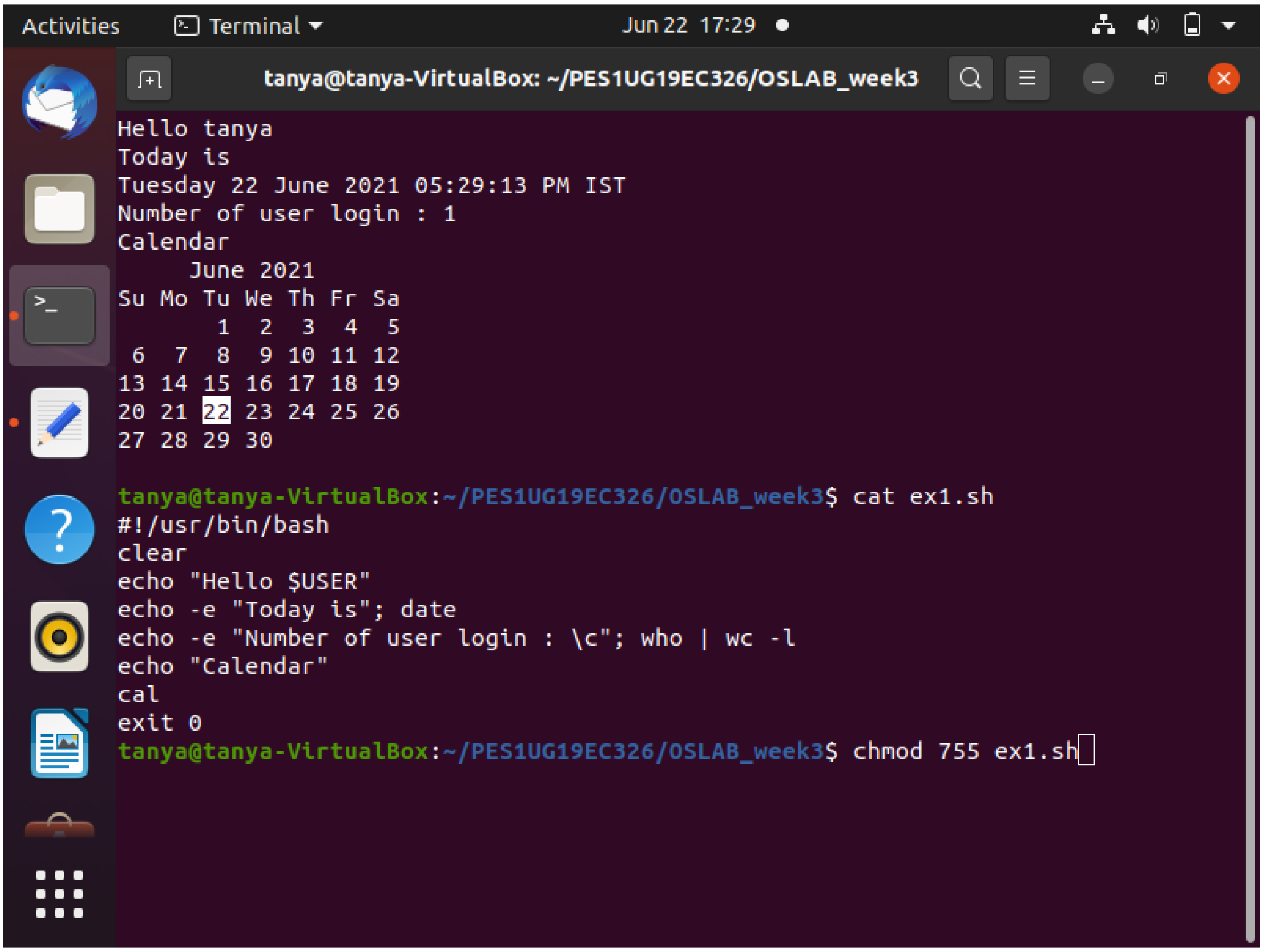
NAME: Tanya Chanchalani

SRN: PES1UG19EC326

OS_LAB_WEEK: 3

Exercise 1

a)



```
Hello tanya
Today is
Tuesday 22 June 2021 05:29:13 PM IST
Number of user login : 1
Calendar
     June 2021
Su Mo Tu We Th Fr Sa
       1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30

tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ cat ex1.sh
#!/usr/bin/bash
clear
echo "Hello $USER"
echo -e "Today is"; date
echo -e "Number of user login : \c"; who | wc -l
echo "Calendar"
cal
exit 0
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 ex1.sh
```

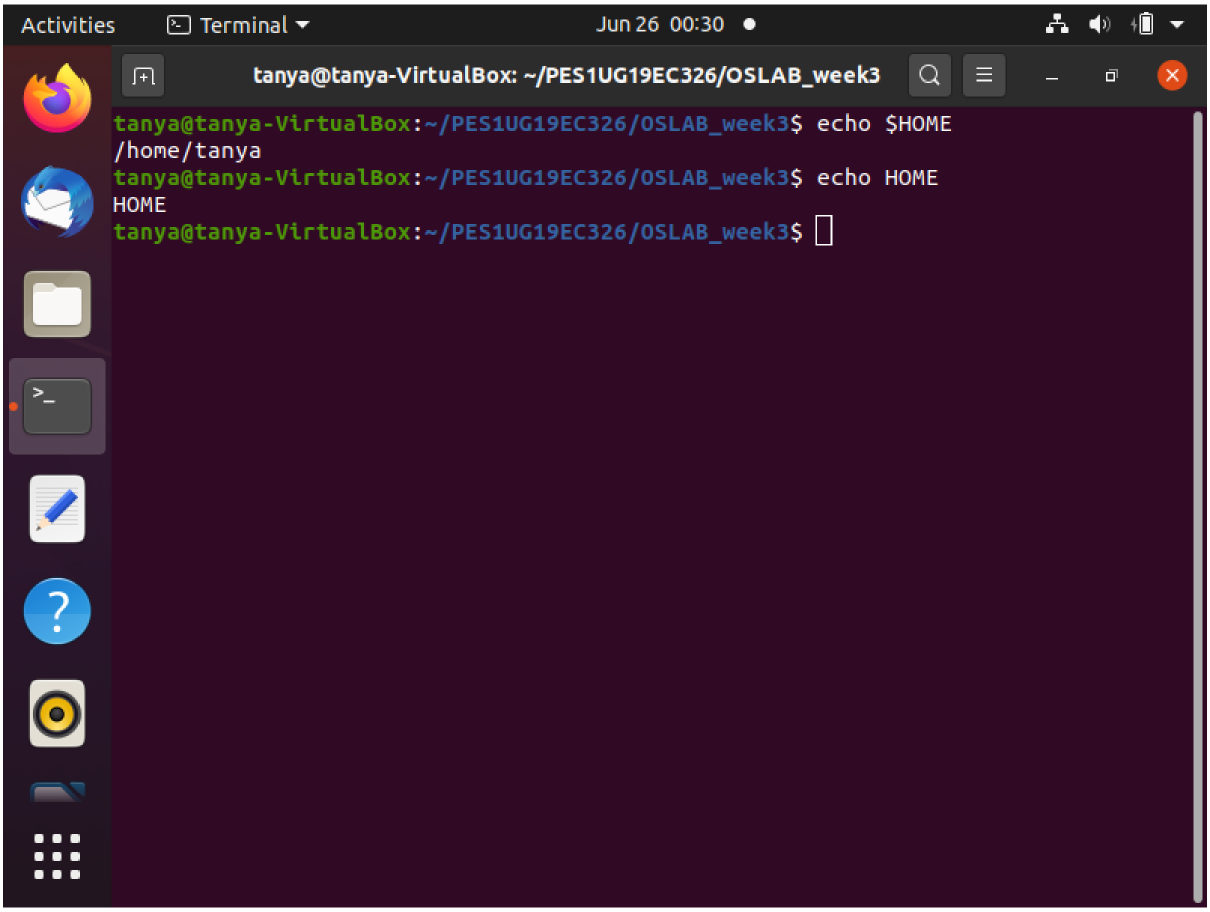
b)

Linux if particular command/shell script is executed, it return two type of values which is used to see whether command or shell script executed is successful or not.

(1) If return value is zero (0), command is successful.

(2) If return value is nonzero, command is not successful or some sort of error executing command/shell script. This value is know as Exit Status. $? is a special variable in shell that reads the exit status of the last command executed. After a function returns, $? gives the exit status of the last command executed in the function.

Exercise 2



```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ echo $HOME
/home/tanya
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ echo HOME
HOME
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```

echo $HOME - the shell evaluates $HOME and passes its value as an argument to echo.

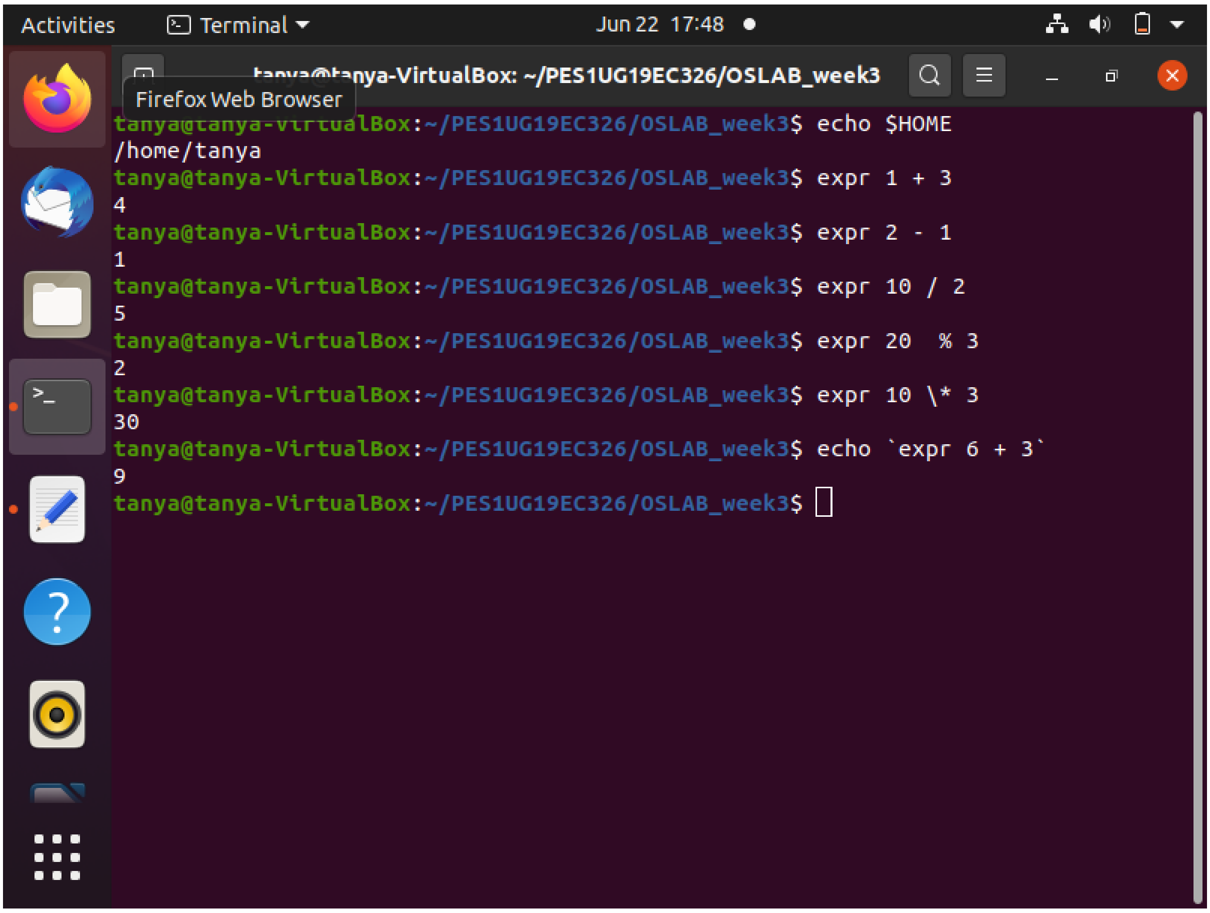
While echo HOME just prints HOME in the terminal.

Echo is particularly useful for showing the values of environmental variables, which tell the shell how to behave as a user works at the command line or in scripts (short programs).

It is to see the value of $HOME, the environmental value that shows the current user's home directory.

Exercise 3

a)



```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ echo $HOME
/home/tanya
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ expr 1 + 3
4
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ expr 2 - 1
1
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ expr 10 / 2
5
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ expr 20  % 3
2
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ expr 10 \* 3
30
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ echo `expr 6 + 3`
9
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```

Exercise 4

a)

SINGLE QUOTES :-

Single quotes can be used around text to prevent the shell from interpreting any special characters. Dollar signs, spaces, ampersands, asterisks and other special characters are all ignored when enclosed within single quotes.

$ echo 'All sorts of things are ignored in single quotes, like $ & * ; |.'

RESULT = All sorts of things are ignored in single quotes, like $ & * ; |.

DOUBLE QUOTES :-

Enclosing characters in double quotes (") preserves the literal value of all characters within the quotes.Within double quotes, backslashes that are followed by one of these characters are removed.

Anything enclose in double quotes removedmeaning of that characters except symbols like \ and $.

Although, double quotes still allow the shell to interpret dollar signs, back quotes and backslashes.That is, variables are expanded inside "".

```
$ echo "The current Oracle SID is $ORACLEID"
```

RESULT = The current Oracle SID is demo.

BACK QUOTES :-

Back quotes force the execution of the commands they enclose. After the enclosed commands are executed, their output is substituted in place of the back quotes in the original line.

```
$ today=`date '+%A, %B %d, %Y'`

$ echo $today

Tuesday, June 22, 2021
```

Exercise 5

a)

```
$ sort < myfile > sorted_file
```

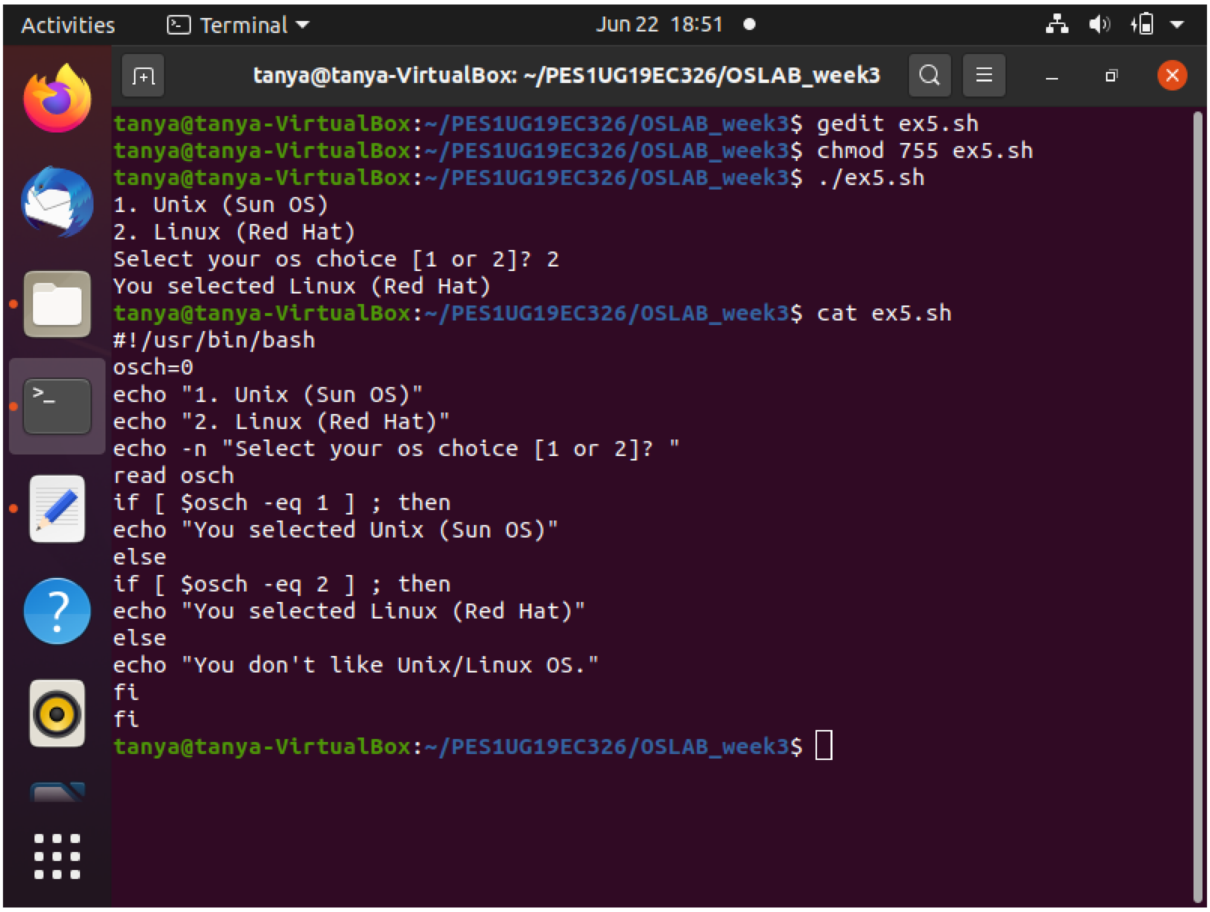
This command will use the input redirection to obtain the contents of myfile, which is sent to sort command, the contents get sorted line by line once finished it gets redirected to sorted_file and sorted.

sort command is used to sort a file, arranging the records in a particular order. It is a command which sorts the contents of a text file, line by line. sort is a standard command line program that prints the lines of its input or concatenation of all files listed in its argument list in sorted order.

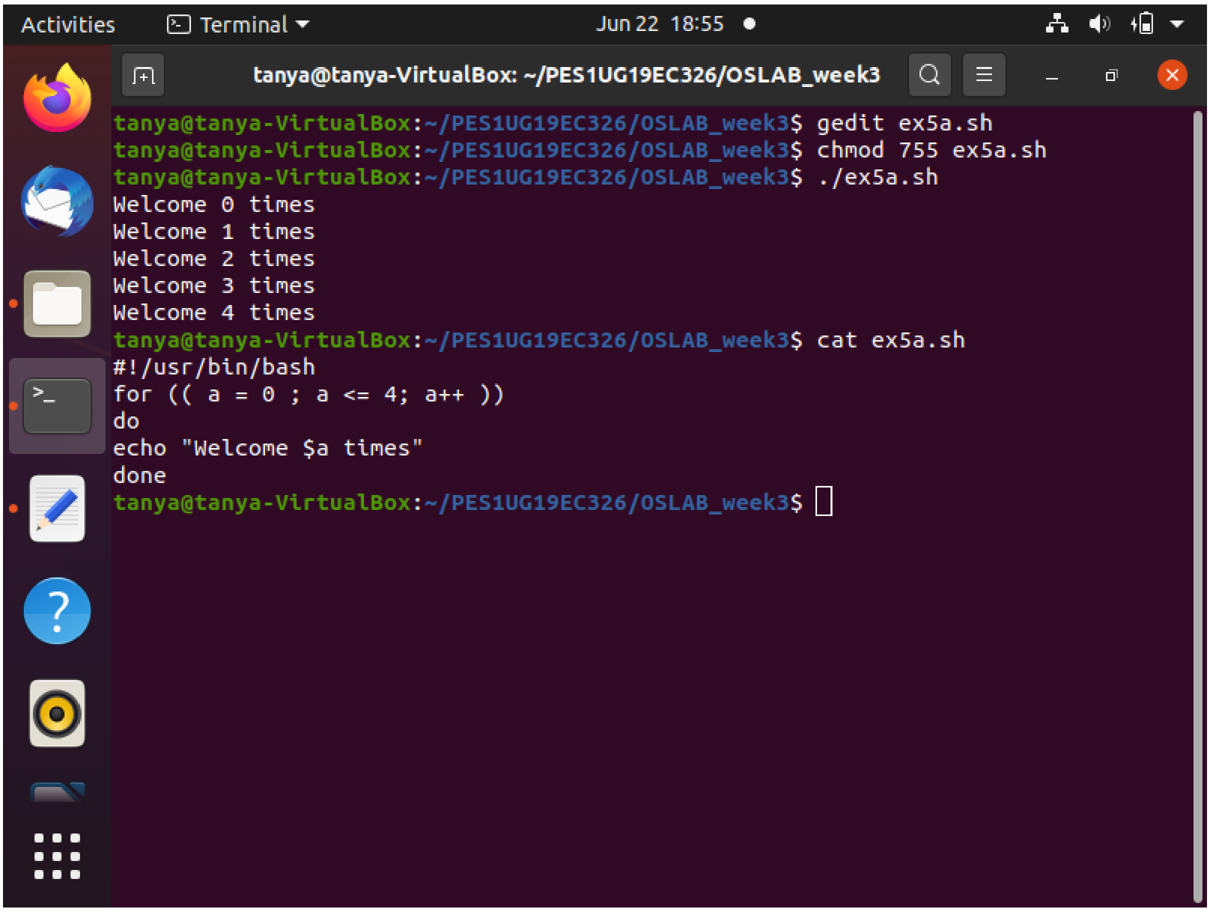
$ sort myfile > sorted_file – would take input from my_file and sort the content line by line, and redirect it to sorted_file.

<> sorted_file = the file is open in read+write mode, without truncation and creating the file if it didn't exist beforehand. The data is redirected into myfile as input, then sort command starts running, and finally the contents are transferred to sorted_file. Redirection of output causes the file to be opened for writing if the file does not exist it is created; if it does exist it is truncated to zero size.

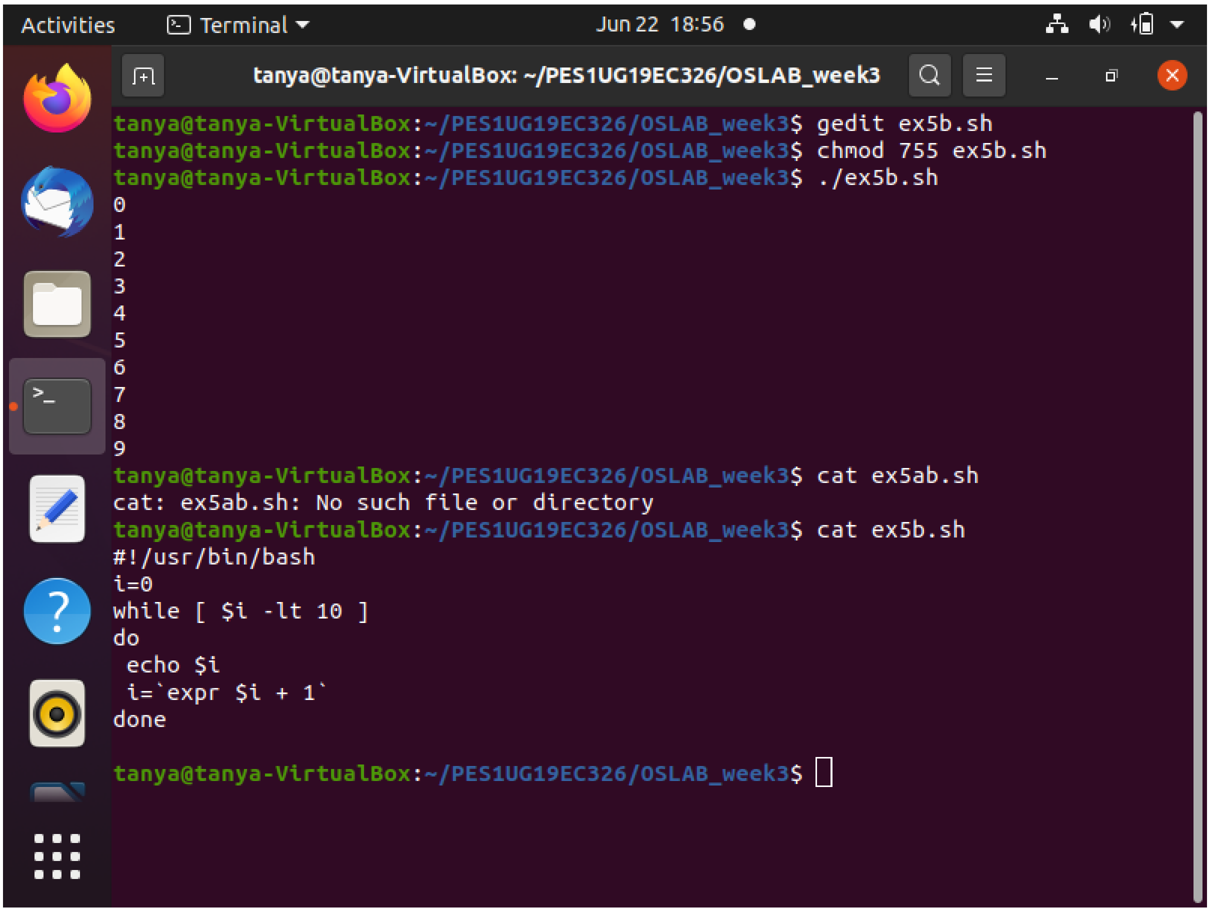
b)



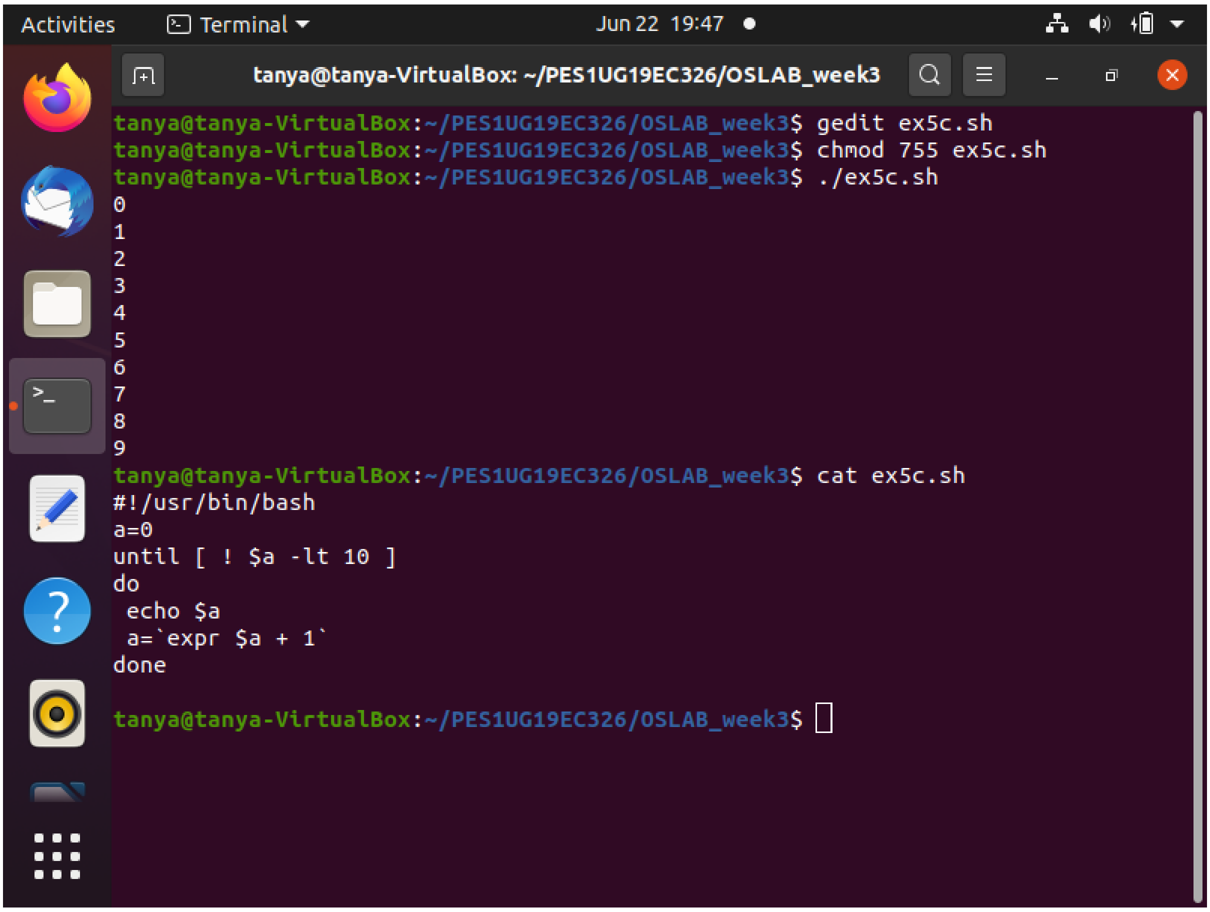
```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex5.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 ex5.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ ./ex5.sh
1. Unix (Sun OS)
2. Linux (Red Hat)
Select your os choice [1 or 2]? 2
You selected Linux (Red Hat)
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ cat ex5.sh
#!/usr/bin/bash
osch=0
echo "1. Unix (Sun OS)"
echo "2. Linux (Red Hat)"
echo -n "Select your os choice [1 or 2]? "
read osch
if [ $osch -eq 1 ] ; then
echo "You selected Unix (Sun OS)"
else
if [ $osch -eq 2 ] ; then
echo "You selected Linux (Red Hat)"
else
echo "You don't like Unix/Linux OS."
fi
fi
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```



```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex5a.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 ex5a.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ ./ex5a.sh
Welcome 0 times
Welcome 1 times
Welcome 2 times
Welcome 3 times
Welcome 4 times
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ cat ex5a.sh
#!/usr/bin/bash
for (( a = 0 ; a <= 4; a++ ))
do
echo "Welcome $a times"
done
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```

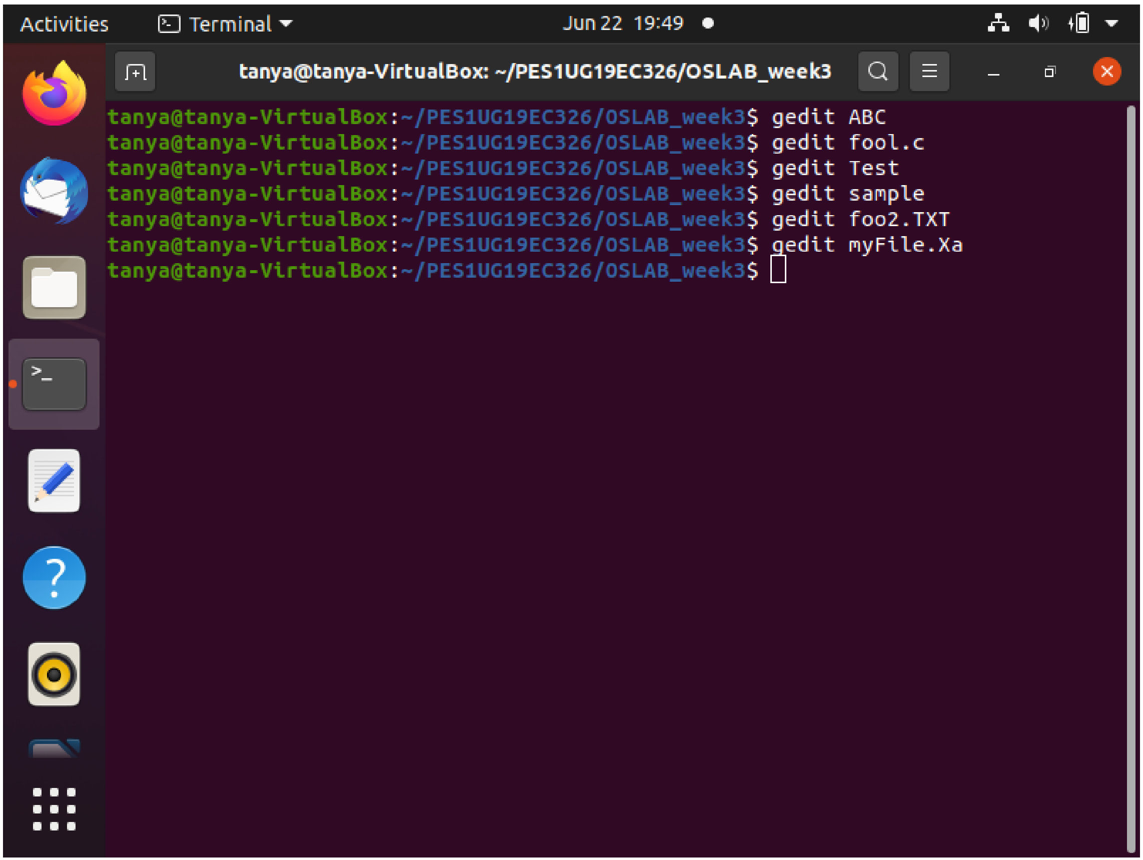


```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex5b.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 ex5b.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ ./ex5b.sh
0
1
2
3
4
5
6
7
8
9
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ cat ex5ab.sh
cat: ex5ab.sh: No such file or directory
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ cat ex5b.sh
#!/usr/bin/bash
i=0
while [ $i -lt 10 ]
do
 echo $i
 i=`expr $i + 1`
done

tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```

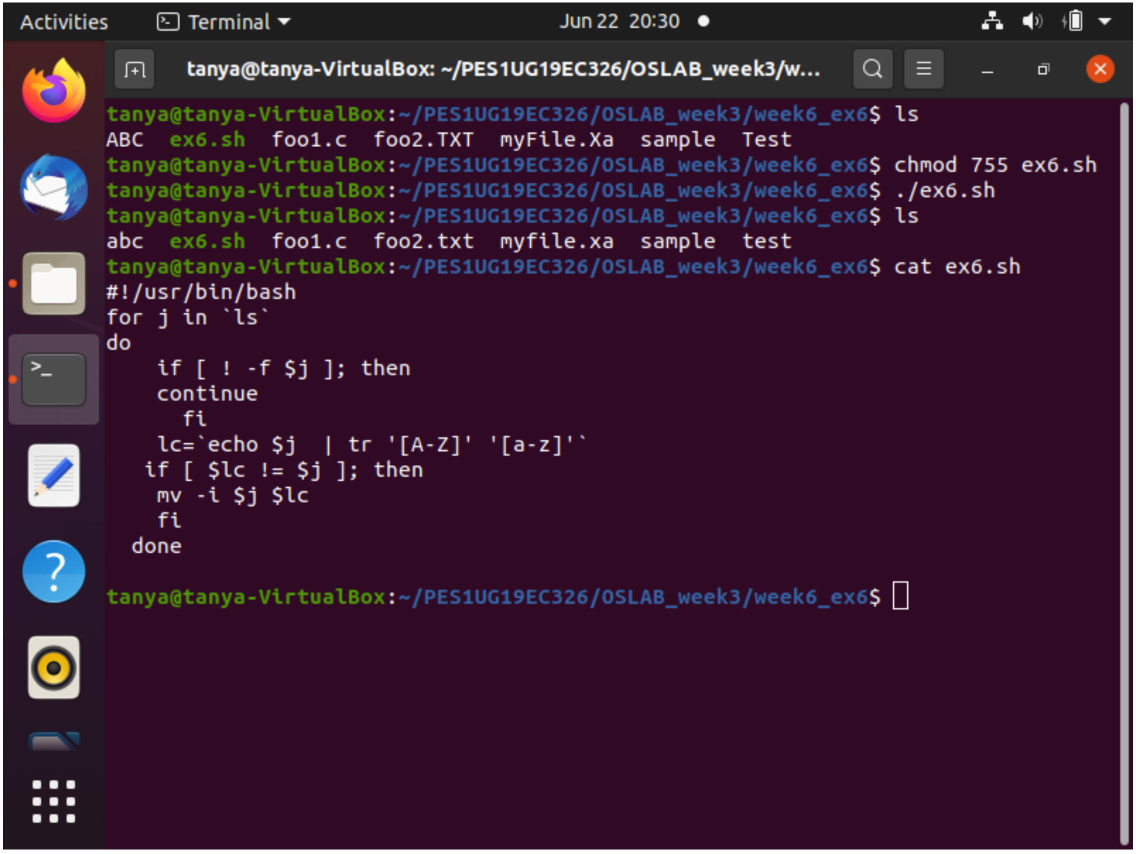


```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex5c.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 ex5c.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ ./ex5c.sh
0
1
2
3
4
5
6
7
8
9
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ cat ex5c.sh
#!/usr/bin/bash
a=0
until [ ! $a -lt 10 ]
do
 echo $a
 a=`expr $a + 1`
done

tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```

Exercise 6



```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ABC
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit fool.c
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit Test
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit sample
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit foo2.TXT
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit myFile.Xa
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ 
```



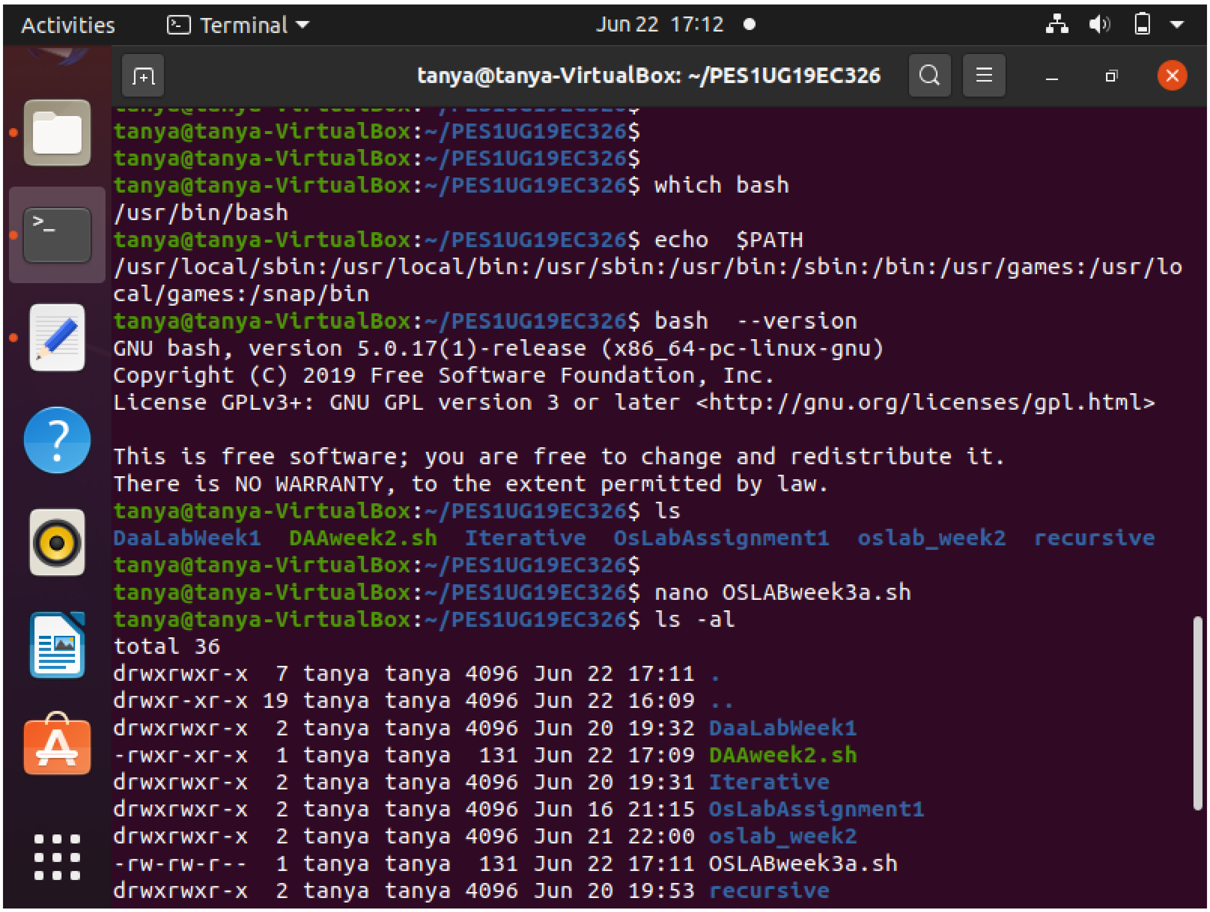
```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3/week6_ex6$ ls
ABC  ex6.sh  foo1.c  foo2.TXT  myFile.Xa  sample  Test
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3/week6_ex6$ chmod 755 ex6.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3/week6_ex6$ ./ex6.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3/week6_ex6$ ls
abc  ex6.sh  foo1.c  foo2.txt  myfile.xa  sample  test
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3/week6_ex6$ cat ex6.sh
#!/usr/bin/bash
for j in `ls`
do
    if [ ! -f $j ]; then
    continue
      fi
    lc=`echo $j  | tr '[A-Z]' '[a-z]'`
   if [ $lc != $j ]; then
    mv -i $j $lc
    fi
  done

tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3/week6_ex6$ 
```

//The tr command reads a byte stream from standard input (stdin), translates or deletes characters, then writes the result to the standard output (stdout). If the translated result has to be written back to the input file, redirect stdout to a temporary file and then rename and overwrite the input file.

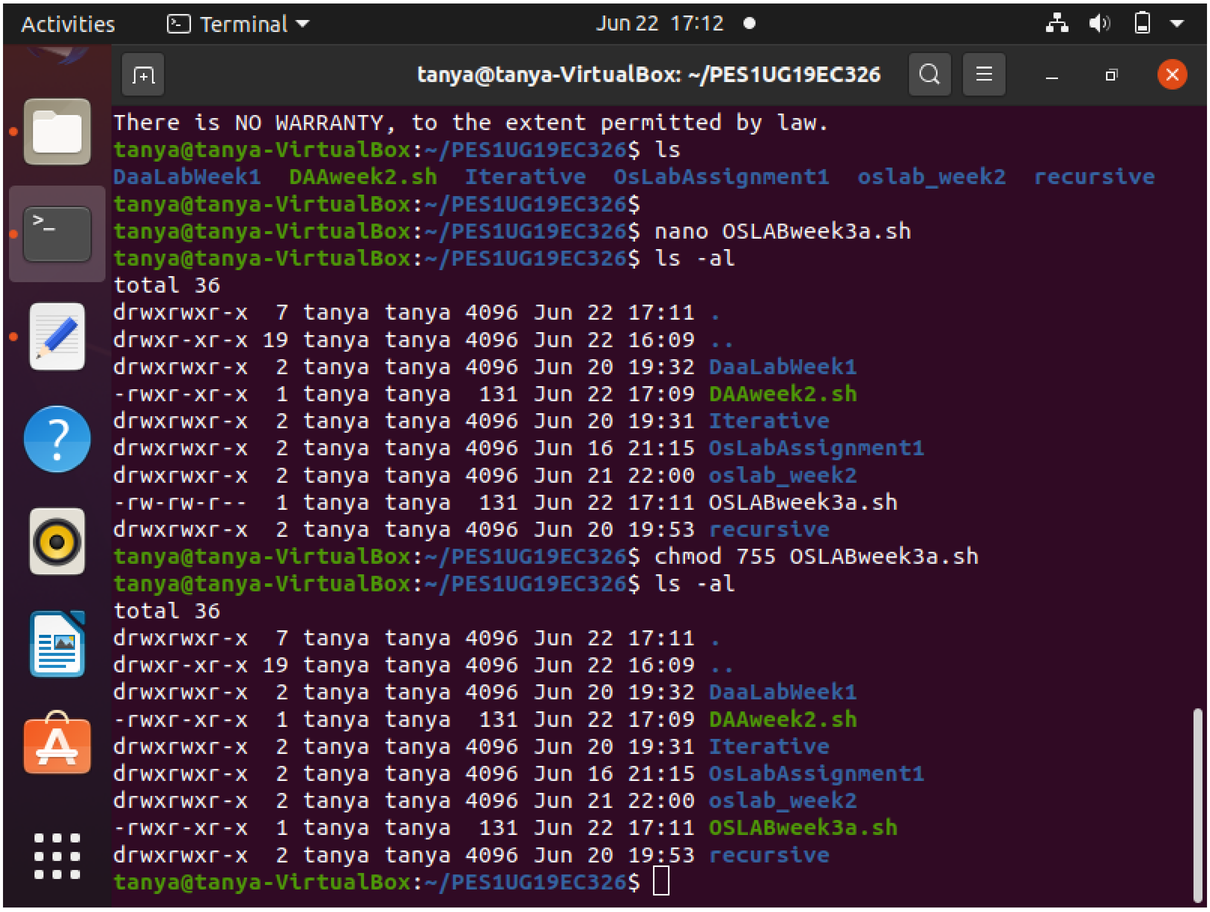
```
tr [OPTION] SET1 [SET2]
```

```
#!/usr/bin/bash
for j in `ls`
do
   if [ ! -f $j ]; then
   continue
     fi
   lc=`echo $j  | tr '[A-Z]' '[a-z]'`
  if [ $lc != $j ]; then
   mv -i $j $lc
   fi
 done
```

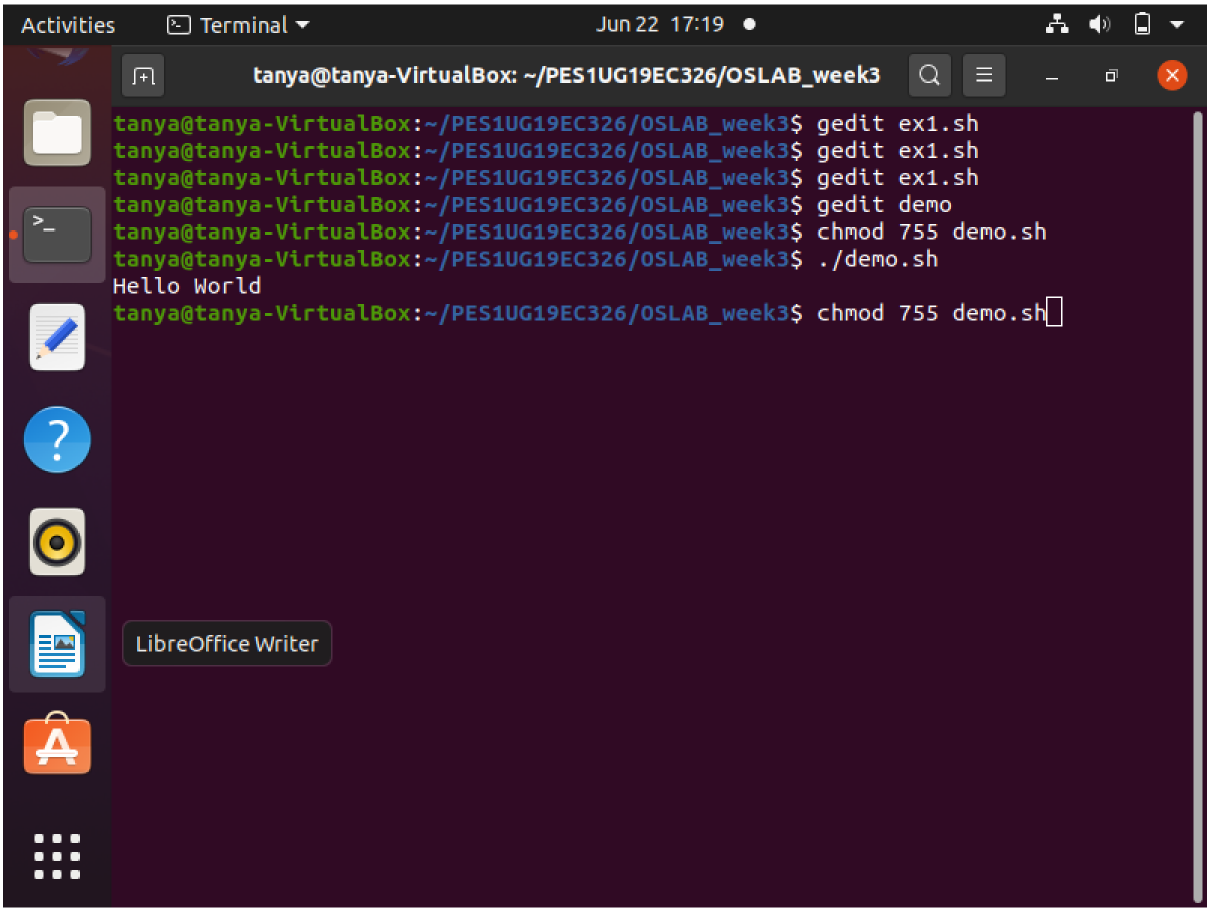
PRACTICE:



```
tanya@tanya-VirtualBox:~/PES1UG19EC326$
tanya@tanya-VirtualBox:~/PES1UG19EC326$
tanya@tanya-VirtualBox:~/PES1UG19EC326$
tanya@tanya-VirtualBox:~/PES1UG19EC326$ which bash
/usr/bin/bash
tanya@tanya-VirtualBox:~/PES1UG19EC326$ echo  $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/lo
cal/games:/snap/bin
tanya@tanya-VirtualBox:~/PES1UG19EC326$ bash  --version
GNU bash, version 5.0.17(1)-release (x86_64-pc-linux-gnu)
Copyright (C) 2019 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
tanya@tanya-VirtualBox:~/PES1UG19EC326$ ls
DaaLabWeek1  DAAweek2.sh  Iterative  OsLabAssignment1  oslab_week2  recursive
tanya@tanya-VirtualBox:~/PES1UG19EC326$
tanya@tanya-VirtualBox:~/PES1UG19EC326$ nano OSLABweek3a.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326$ ls -al
total 36
drwxrwxr-x  7 tanya tanya 4096 Jun 22 17:11 .
drwxr-xr-x 19 tanya tanya 4096 Jun 22 16:09 ..
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:32 DaaLabWeek1
-rwxr-xr-x  1 tanya tanya  131 Jun 22 17:09 DAAweek2.sh
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:31 Iterative
drwxrwxr-x  2 tanya tanya 4096 Jun 16 21:15 OsLabAssignment1
drwxrwxr-x  2 tanya tanya 4096 Jun 21 22:00 oslab_week2
-rw-rw-r--  1 tanya tanya  131 Jun 22 17:11 OSLABweek3a.sh
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:53 recursive
```



```
There is NO WARRANTY, to the extent permitted by law.
tanya@tanya-VirtualBox:~/PES1UG19EC326$ ls
DaaLabWeek1  DAAweek2.sh  Iterative  OsLabAssignment1  oslab_week2  recursive
tanya@tanya-VirtualBox:~/PES1UG19EC326$
tanya@tanya-VirtualBox:~/PES1UG19EC326$ nano OSLABweek3a.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326$ ls -al
total 36
drwxrwxr-x  7 tanya tanya 4096 Jun 22 17:11 .
drwxr-xr-x 19 tanya tanya 4096 Jun 22 16:09 ..
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:32 DaaLabWeek1
-rwxr-xr-x  1 tanya tanya  131 Jun 22 17:09 DAAweek2.sh
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:31 Iterative
drwxrwxr-x  2 tanya tanya 4096 Jun 16 21:15 OsLabAssignment1
drwxrwxr-x  2 tanya tanya 4096 Jun 21 22:00 oslab_week2
-rw-rw-r--  1 tanya tanya  131 Jun 22 17:11 OSLABweek3a.sh
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:53 recursive
tanya@tanya-VirtualBox:~/PES1UG19EC326$ chmod 755 OSLABweek3a.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326$ ls -al
total 36
drwxrwxr-x  7 tanya tanya 4096 Jun 22 17:11 .
drwxr-xr-x 19 tanya tanya 4096 Jun 22 16:09 ..
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:32 DaaLabWeek1
-rwxr-xr-x  1 tanya tanya  131 Jun 22 17:09 DAAweek2.sh
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:31 Iterative
drwxrwxr-x  2 tanya tanya 4096 Jun 16 21:15 OsLabAssignment1
drwxrwxr-x  2 tanya tanya 4096 Jun 21 22:00 oslab_week2
-rwxr-xr-x  1 tanya tanya  131 Jun 22 17:11 OSLABweek3a.sh
drwxrwxr-x  2 tanya tanya 4096 Jun 20 19:53 recursive
tanya@tanya-VirtualBox:~/PES1UG19EC326$
```



```
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex1.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex1.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit ex1.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ gedit demo
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 demo.sh
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ ./demo.sh
Hello World
tanya@tanya-VirtualBox:~/PES1UG19EC326/OSLAB_week3$ chmod 755 demo.sh
```