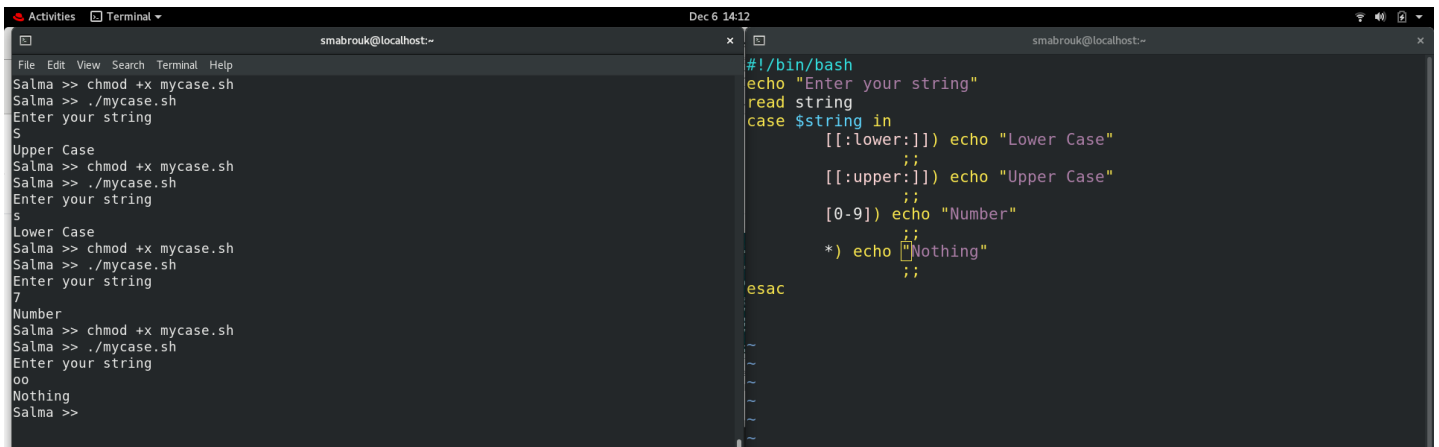


Bash script lab 3

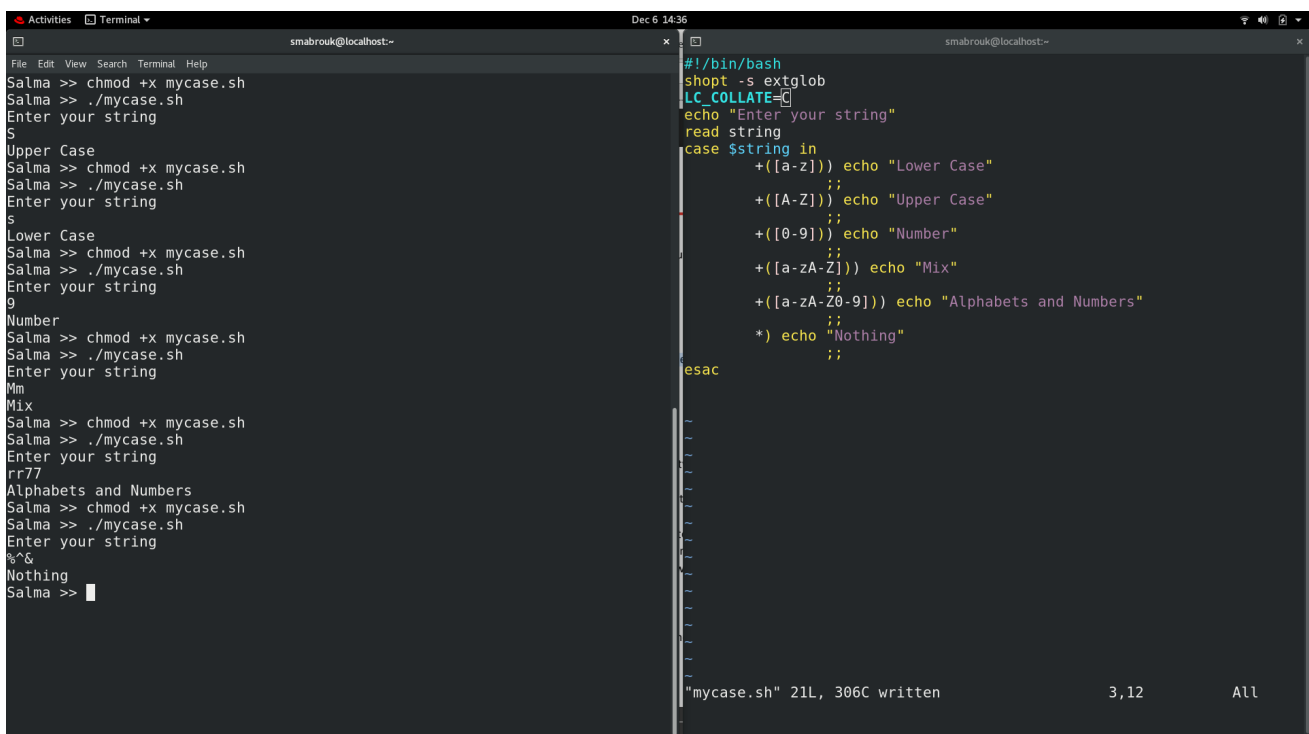
1. Write a script called mycase, using the case utility to checks the type of character entered by a user:
 - Upper Case.
 - Lower Case.
 - Number.
 - Nothing.



```
#!/bin/bash
echo "Enter your string"
read string
case $string in
    [[:lower:]] ) echo "Lower Case"
    ;;
    [[:upper:]] ) echo "Upper Case"
    ;;
    [0-9] ) echo "Number"
    ;;
    *) echo "Nothing"
    ;;
esac
```

Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
s
Upper Case
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
S
Lower Case
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
7
Number
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
oo
Nothing
Salma >>

2. Enhanced the previous script, by checking the type of string entered by a user:
 - Upper Cases.
 - Lower Cases.
 - Numbers.
 - Mix.
 - Nothing.

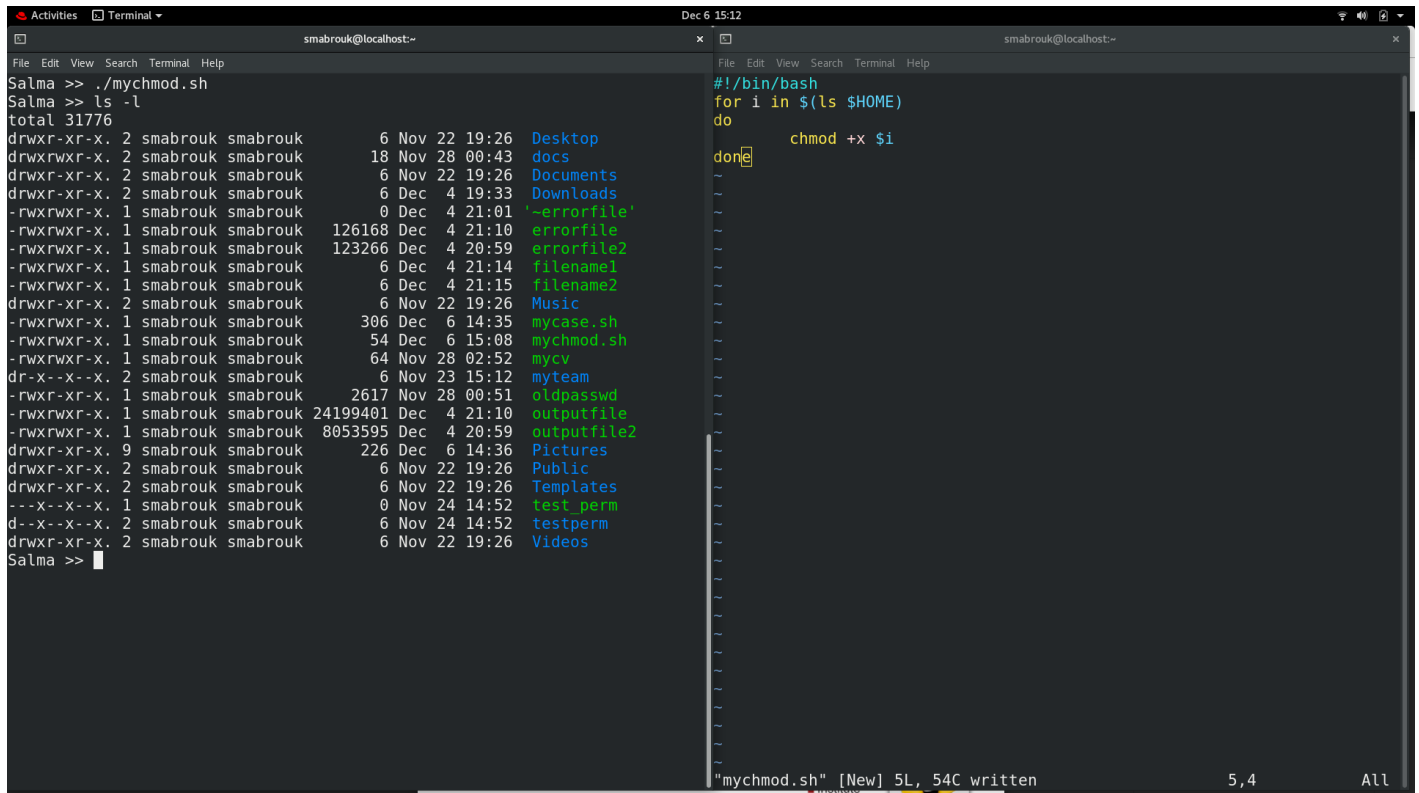


```
#!/bin/bash
shopt -s extglob
LC_COLLATE=C
echo "Enter your string"
read string
case $string in
    +([a-z])) echo "Lower Case"
    ;;
    +([A-Z])) echo "Upper Case"
    ;;
    +([0-9])) echo "Number"
    ;;
    +([a-zA-Z])) echo "Mix"
    ;;
    +([a-zA-Z0-9])) echo "Alphabets and Numbers"
    ;;
    *) echo "Nothing"
    ;;
esac
```

Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
s
Lower Case
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
S
Upper Case
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
7
Number
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
Mm
Mix
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
rr77
Alphabets and Numbers
Salma >> chmod +x mycase.sh
Salma >> ./mycase.sh
Enter your string
%^&
Nothing
Salma >>

"mycase.sh" 21L, 306C written 3,12 All

3. Write a script called mychmod using for utility to give execute permission to all files and directories in your home directory.

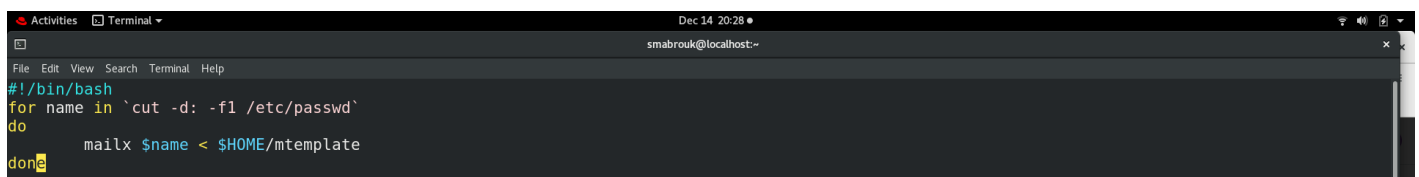


```
Salma >> ./mychmod.sh
Salma >> ls -l
total 31776
drwxr-xr-x. 2 smabrouk smabrouk      6 Nov 22 19:26 Desktop
drwxrwxr-x. 2 smabrouk smabrouk    18 Nov 28 00:43 docs
drwxr-xr-x. 2 smabrouk smabrouk      6 Nov 22 19:26 Documents
drwxr-xr-x. 2 smabrouk smabrouk      6 Dec  4 19:33 Downloads
-rwxrwxr-x. 1 smabrouk smabrouk      0 Dec  4 21:01 '-errorfile'
-rwxrwxr-x. 1 smabrouk smabrouk 126168 Dec  4 21:10 errorfile
-rwxrwxr-x. 1 smabrouk smabrouk 123266 Dec  4 20:59 errorfile2
-rwxrwxr-x. 1 smabrouk smabrouk      6 Dec  4 21:14 filename1
-rwxrwxr-x. 1 smabrouk smabrouk      6 Dec  4 21:15 filename2
drwxr-xr-x. 2 smabrouk smabrouk      6 Nov 22 19:26 Music
-rwxrwxr-x. 1 smabrouk smabrouk   306 Dec  6 14:35 mycase.sh
-rwxrwxr-x. 1 smabrouk smabrouk    54 Dec  6 15:08 mychmod.sh
-rwxrwxr-x. 1 smabrouk smabrouk    64 Nov 28 02:52 mycv
dr-x--x--x. 2 smabrouk smabrouk      6 Nov 23 15:12 myteam
-rwxr-xr-x. 1 smabrouk smabrouk   2617 Nov 28 00:51 oldpasswd
-rwxrwxr-x. 1 smabrouk smabrouk 24199401 Dec  4 21:10 outputfile
-rwxrwxr-x. 1 smabrouk smabrouk 8053595 Dec  4 20:59 outputfile2
drwxr-xr-x. 9 smabrouk smabrouk    226 Dec  6 14:36 Pictures
drwxr-xr-x. 2 smabrouk smabrouk      6 Nov 22 19:26 Public
drwxr-xr-x. 2 smabrouk smabrouk      6 Nov 22 19:26 Templates
---x--x--x. 1 smabrouk smabrouk      0 Nov 24 14:52 test_perm
d--x--x--x. 2 smabrouk smabrouk      6 Nov 24 14:52 testperm
drwxr-xr-x. 2 smabrouk smabrouk      6 Nov 22 19:26 Videos

Salma >>

#!/bin/bash
for i in $(ls $HOME)
do
    chmod +x $i
done
```

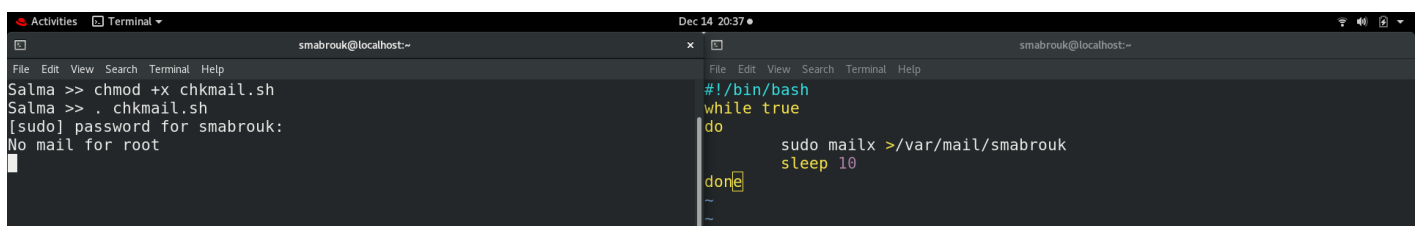
4. Write a script called mybackup using for utility to create a backup of only files in your home directory.
5. Write a script called mymail using for utility to send a mail to all users in the system. Note: write the mail body in a file called mtemplate.



```
Salma >> ./mymail.sh

#!/bin/bash
for name in $(cut -d: -f1 /etc/passwd)
do
    mailx $name < $HOME/mtemplate
done
```

6. Write a script called chkmail to check for new mails every 10 seconds. Note: mails are saved in /var/mail/username.



```
Salma >> chmod +x chkmail.sh
Salma >> ./chkmail.sh
[sudo] password for smabrouk:
No mail for root

#!/bin/bash
while true
do
    sudo mailx >/var/mail/smabrouk
    sleep 10
done
```

7. What is the output of the following script

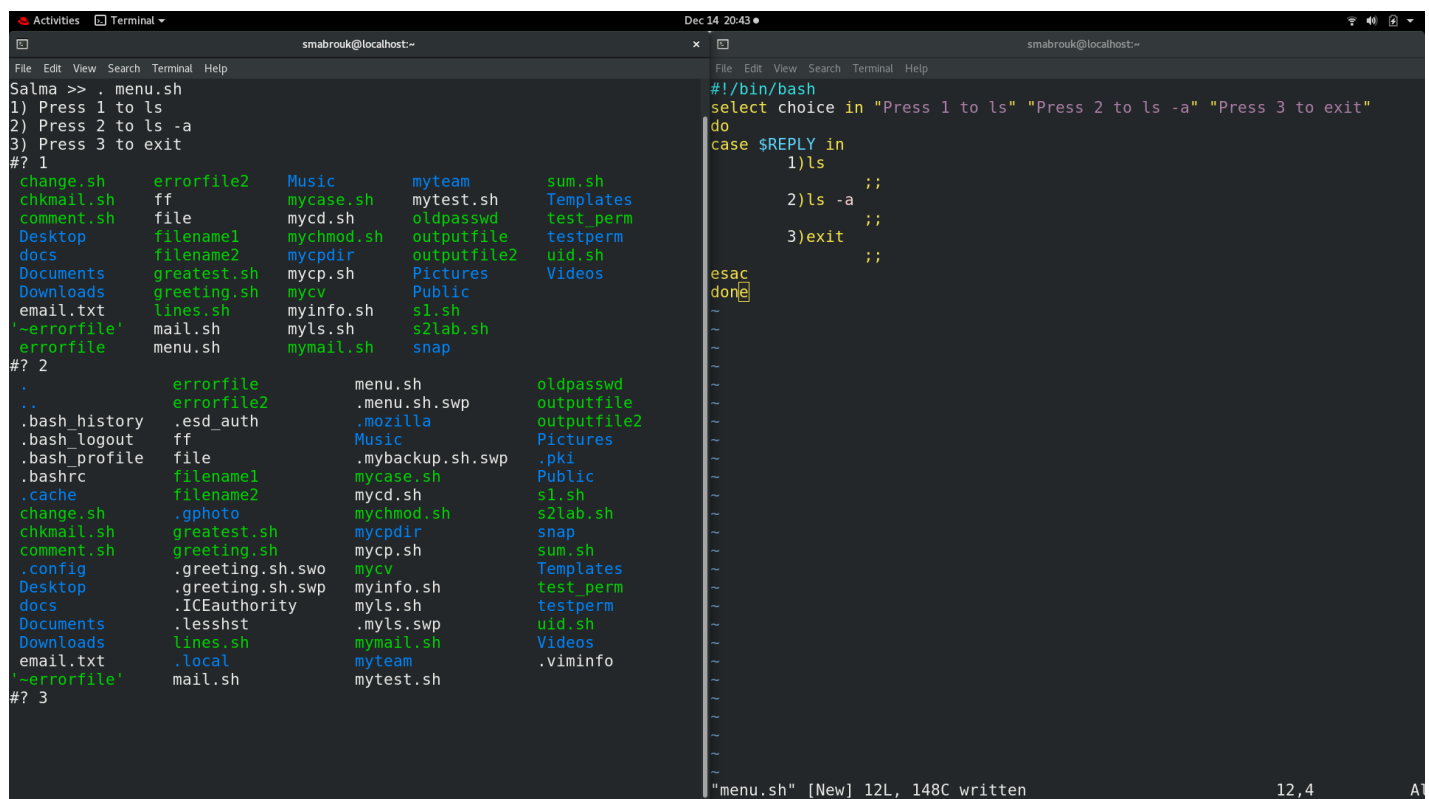
```
typeset -i n1
typeset -i n2
n1=1
n2=1
while test $n1 -eq $n2
do
n2=$n2+1
print $n1
if [ $n1 -gt $n2 ]
then
break
else
continue
fi
n1=$n1+1
print $n2
done
```

The output = 1

8. Create the following menu using select utility then while utility.:

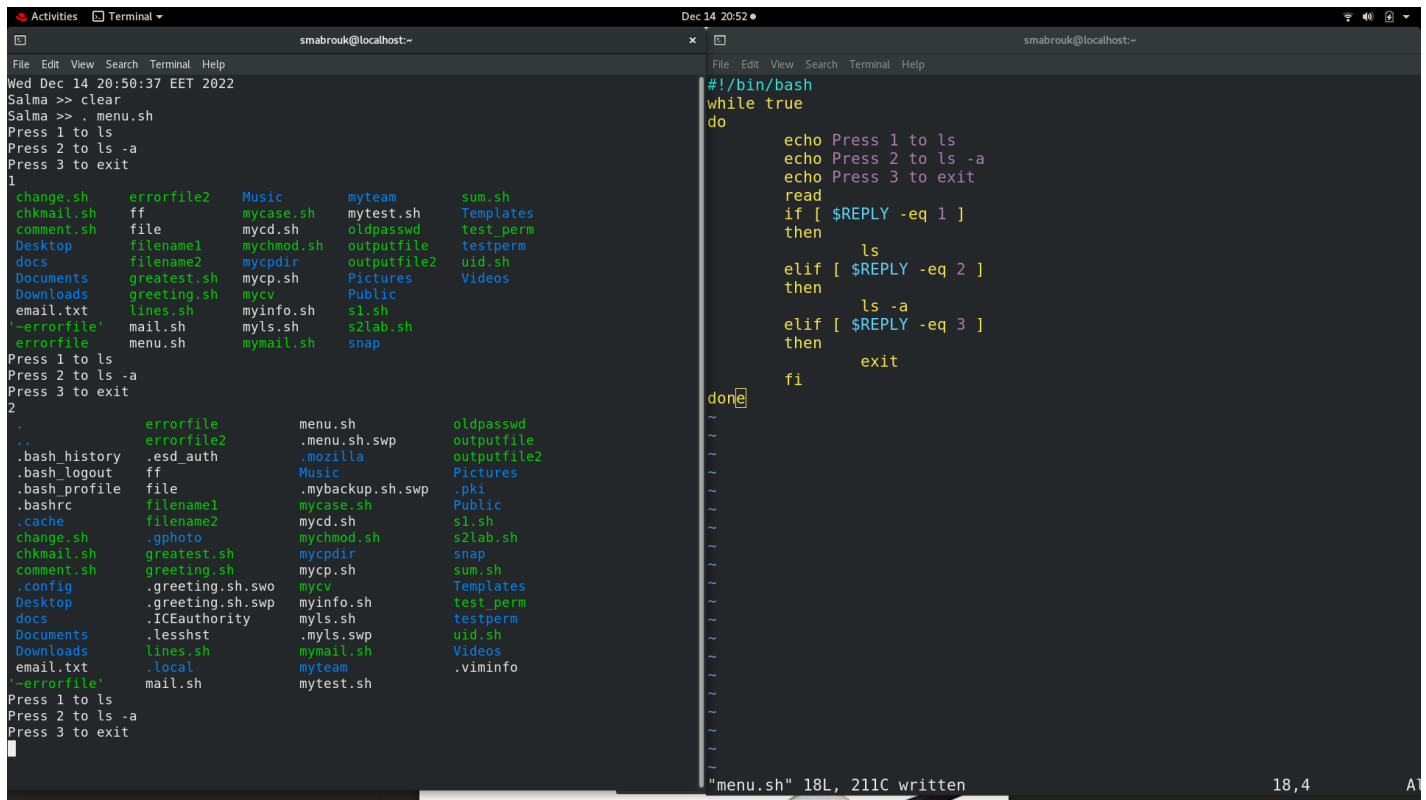
- Press 1 to ls
- Press 2 to ls -a
- Press 3 to exit

Using select utility:



```
Salma >> . menu.sh
1) Press 1 to ls
2) Press 2 to ls -a
3) Press 3 to exit
#? 1
change.sh      errorfile2    Music          myteam         sum.sh
chkmail.sh     ff             mycase.sh     mytest.sh     Templates
comment.sh     file          mycd.sh       oldpasswd     test perm
Desktop        filename1     mychmod.sh    outputfile    testperm
docs           filename2     mycpdir       outputfile2   uid.sh
Documents      greatest.sh  mycp.sh       Pictures      Videos
Downloads      greeting.sh  mycv          Public
email.txt      lines.sh     myinfo.sh     s1.sh
'-errorfile'   mail.sh      myls.sh       s2lab.sh
errorfile      menu.sh      mymail.sh     snap
#? 2
.              errorfile      menu.sh        oldpasswd
..            errorfile2     .menu.sh.swp  outputfile
.bash_history .esd_auth     .mozilla      outputfile2
.bash_logout  ff            Music         Pictures
.bash_profile file          .mybackup.sh.swp .pk1
.bashrc       filename1     mycase.sh     Public
.cache        filename2     mycd.sh       s1.sh
change.sh     .gphoto      mychmod.sh    s2lab.sh
chkmail.sh    greatest.sh  mycpdir       snap
comment.sh    greeting.sh  mycp.sh       sum.sh
.config       .greeting.sh.swo mycv          Templates
Desktop      .greeting.sh.swp myinfo.sh     test perm
docs         .ICEauthority  myls.sh       testperm
Documents    .lessht       .myls.swp     uid.sh
Downloads    lines.sh      mymail.sh     Videos
email.txt    .local        myteam        .viminfo
'-errorfile' mail.sh       mytest.sh
#? 3
```

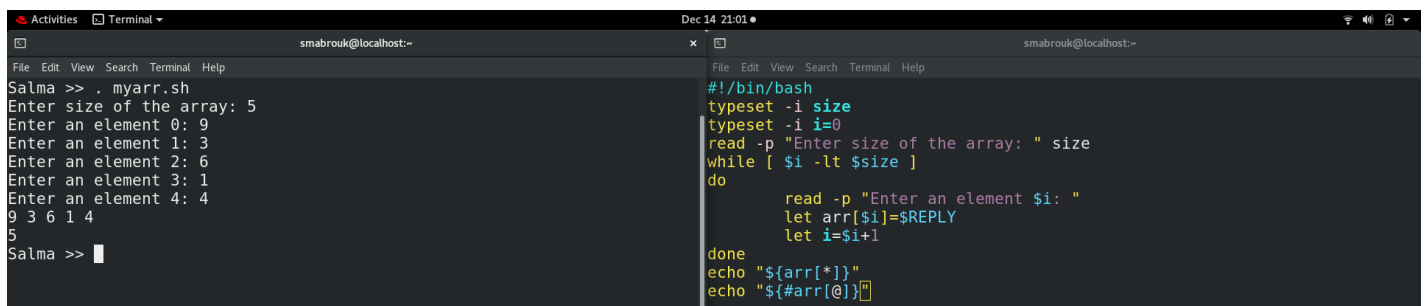
Using while utility:



The image shows two terminal windows. The left window displays a script menu with a list of files and directories. The right window shows the execution of a script that prompts the user to select an option and then lists the contents of the selected directory.

```
#!/bin/bash
while true
do
    echo Press 1 to ls
    echo Press 2 to ls -a
    echo Press 3 to exit
    read
    if [ $REPLY -eq 1 ]
    then
        ls
    elif [ $REPLY -eq 2 ]
    then
        ls -a
    elif [ $REPLY -eq 3 ]
    then
        exit
    fi
done
```

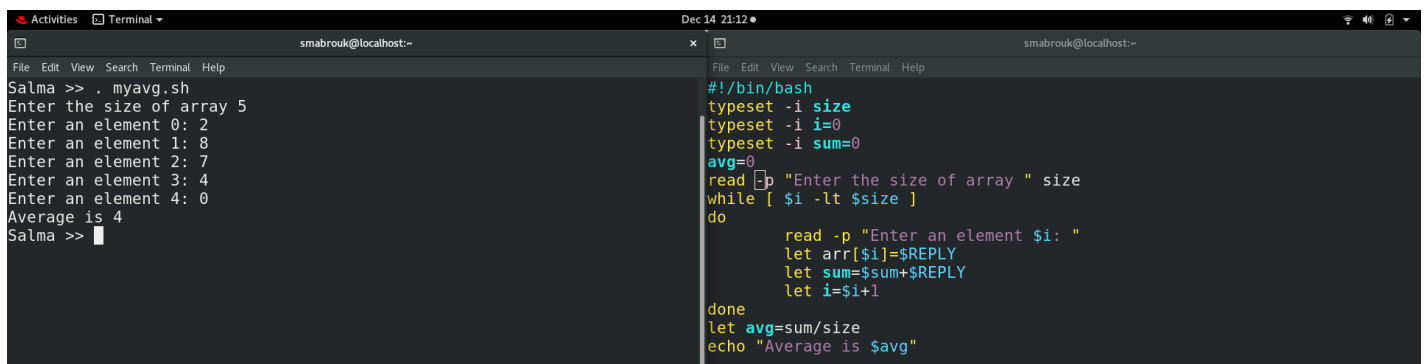
9. Write a script called myarr that ask a user how many elements he wants to enter in an array, fill the array and then print it.



The image shows two terminal windows. The left window shows the user running the myarr script and entering the size of the array and the elements. The right window shows the script code that prompts the user for the size and then fills the array with the entered elements.

```
#!/bin/bash
typeset -i size
typeset -i i=0
read -p "Enter size of the array: " size
while [ $i -lt $size ]
do
    read -p "Enter an element $i: "
    let arr[$i]=$REPLY
    let i=i+1
done
echo "${arr[*]}"
echo "${arr[@]}"
```

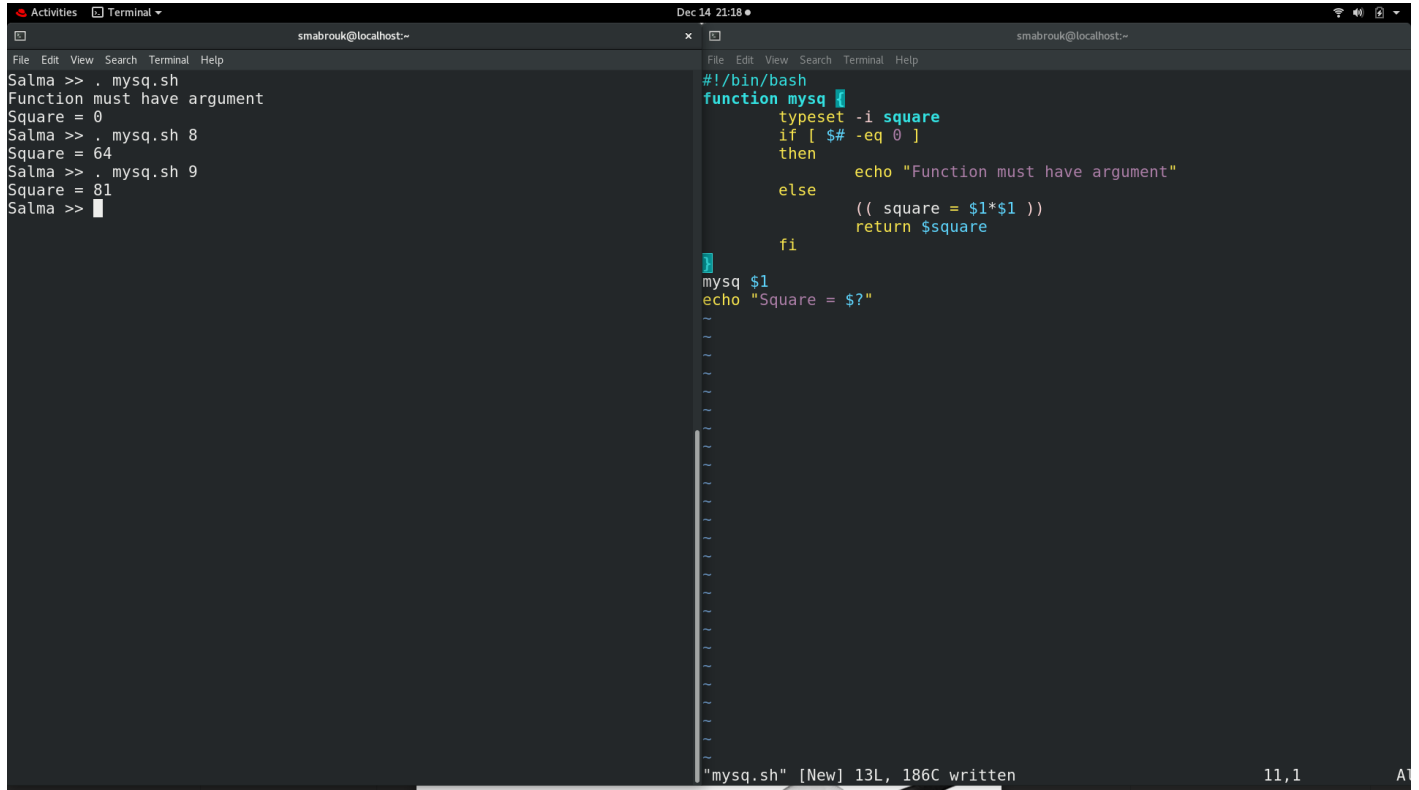
10. Write a script called myavg that calculate average of all numbers entered by a user. Note: use arrays



The image shows two terminal windows. The left window shows the user running the myavg script and entering the size of the array and the numbers. The right window shows the script code that prompts the user for the size and then calculates the average of the entered numbers.

```
#!/bin/bash
typeset -i size
typeset -i i=0
typeset -i sum=0
avg=0
read -p "Enter the size of array " size
while [ $i -lt $size ]
do
    read -p "Enter an element $i: "
    let arr[$i]=$REPLY
    let sum=sum+$REPLY
    let i=i+1
done
let avg=sum/size
echo "Average is $avg"
```

11. Write a function called mysq that calculate square if its argument.



The image shows a terminal window with two panes. The left pane shows the execution of a script named `mysq.sh`. The right pane shows the contents of the script being edited in a text editor.

```
Salma >> . mysq.sh
Function must have argument
Square = 0
Salma >> . mysq.sh 8
Square = 64
Salma >> . mysq.sh 9
Square = 81
Salma >>
```

```
#!/bin/bash
function mysq {
    typeset -i square
    if [ $# -eq 0 ]
    then
        echo "Function must have argument"
    else
        (( square = $1*$1 ))
        return $square
    fi
}

mysq $1
echo "Square = $?"
```

The status bar at the bottom of the text editor shows: "mysq.sh" [New] 13L, 186C written 11,1