S20_HW_IOS203

The supervisor: Eng.Numan Falloh

Linux Platform 1

Prepared by

alaa_110562 Alaa Mohammad Ajelo (آلاء محمد عجيلو)

Table of Contents

EXERCISE (1) **2**

ANSWER (1) **2**

1) USER CREATION: 2
2) THE SCRIPT: 3
3) THE LOCATION: 3
4) OUTPUT: 4

EXERCISE (2) **5**

ANSWER (2) **5**

1) THE SCRIPT: 5
2) OUTPUT: 6

EXERCISE (3) **10**

ANSWER (3) **10**

1) THE FIRST SCRIPT: [PUT THIS SCRIPT IN FILE (DELETE.SH)] 10
2) THE FIRST OUTPUT: 11
3) THE SECOND SCRIPT: [PUT THIS SCRIPT IN FILE (DELETE_2.SH)] 12
4) THE SECOND OUTPUT: 13

EXERCISE (4) **14**

ANSWER (4) **14**

1) THE SCRIPT: [PUT THIS SCRIPT IN FILE (APPEND.SH)] 14
2) THE REQUIRED FILES: 16
3) OUTPUT: 17

EXERCISE (5) **18**

ANSWER (5) **19**

1) THE FIRST SCRIPT: 19
   - ✓ Directory Creation 19
   - ✓ Output of direcrory tree: 19
2) THE SECOND SCRIPT: 20
   - ✓ Files Creation in the directory "Drafts" [in Terminal] 20
   - ✓ Output of folder [Drafts]: 21
3) THE THIRD SCRIPT: 21
   - ✓ Shell script that adds an extension ".sh" to all the files previously created 21
   - ✓ [in Terminal] 22
   - ✓ Output of folder [Drafts]: 22
4) THE FOURTH SCRIPT: 23
   - ✓ Move the converted files to the "Scripts" directory. [in Terminal] 23
   - ✓ Output 24
5) THE TREE: 25

with regards 26

Exercise (1)

Create a new user account as your SVU account, ex. Abc_88888, and login with it. You should have a welcome message when you login. The message should contain the statement "Good TIME Abc_88888", where:

- the word "TIME" should be replaced with "Day" when it's before 12:00 p.m. or "Evening" otherwise.

- and the word "Abc_88888" with your username.

The message should contain also:

- Your UID number,

- The path to your home directory,

- Your login shell,

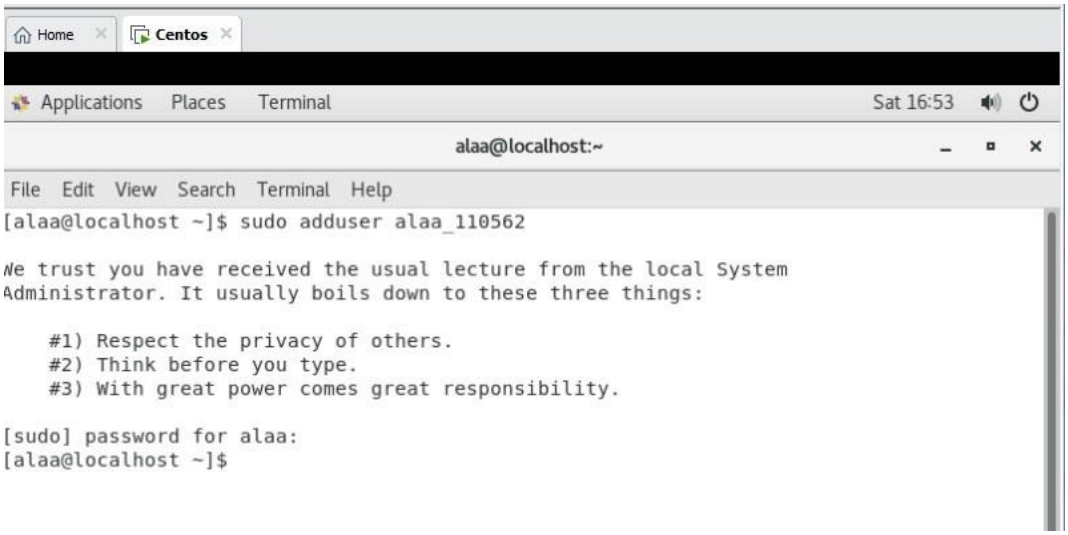
- Currently logged users, and

- The current timing displayed like this: Nov 8, 2020 at 23:45

Note: you should test it at least two times to proof your solution

Answer (1)

1) **User creation:**



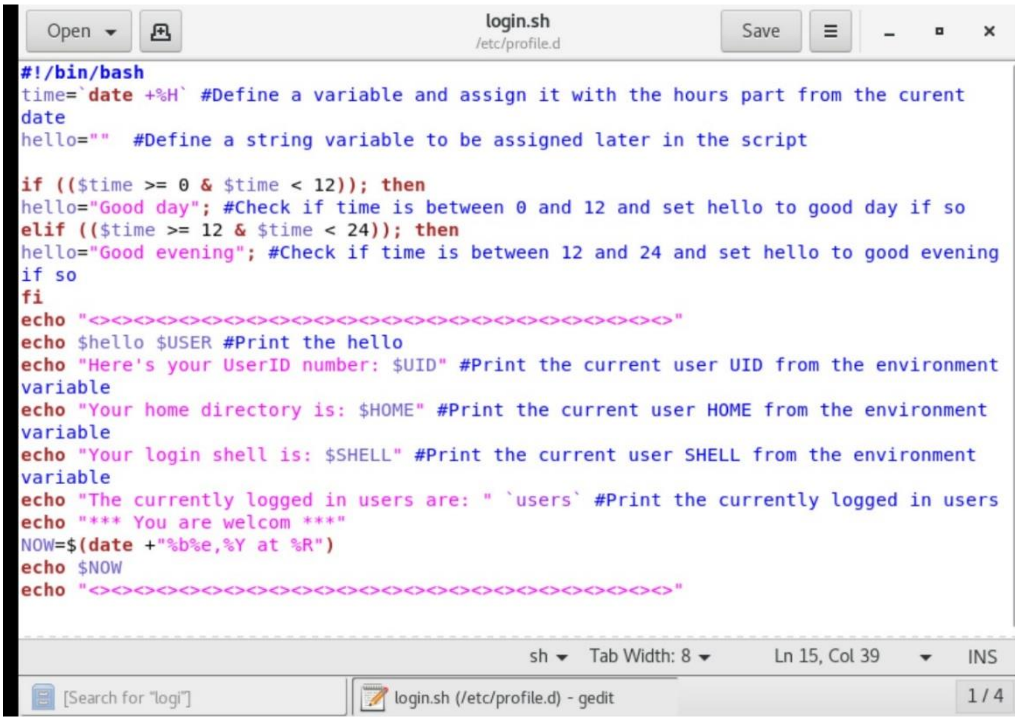
2) The script:

```
#!/bin/bash
time=`date +%H` #Define a variable and assign it with the hours part from the curent date
hello=""  #Define a string variable to be assigned later in the script
if (($time >= 0 & $time < 12)); then
hello="Good day"; #Check if time is between 0 and 12 and set hello to good day if so
elif (($time >= 12 & $time < 24)); then
hello="Good evening"; #Check if time is between 12 and 24 and set hello to good evening if so
fi
echo "<><><><><><><><><><><><><><><><><><><><><><><><><><>"
echo $hello $USER #Print the hello
echo "Here's your UserID number: $UID" #Print the current user UID from the environment variable
echo "Your home directory is: $HOME" #Print the current user HOME from the environment variable
echo "Your login shell is: $SHELL" #Print the current user SHELL from the environment variable
echo "The currently logged in users are: "  `users` #Print the currently logged in users
echo "*** You are welcom ***"
NOW=$(date +"%b%e, %Y at %R")
echo $NOW
echo "<><><><><><><><><><><><><><><><><><><><><><><><><><>"
```

3) The location:

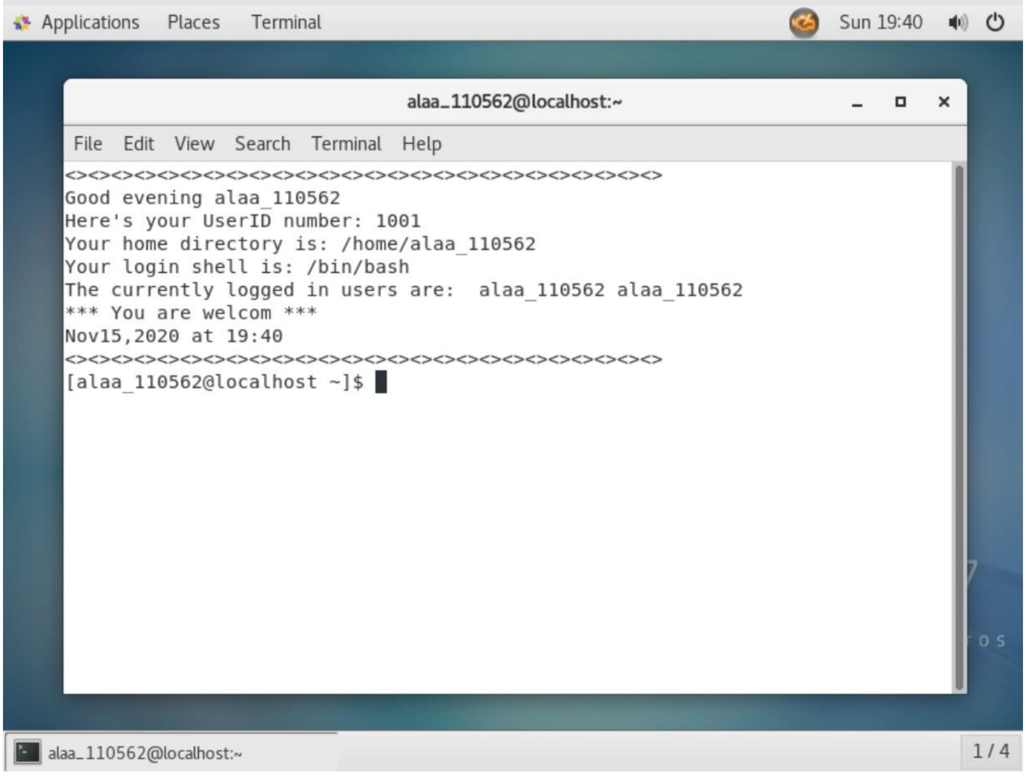
Save the previous script in the following location: /etc/profile.d

This way it will be applied on login for all of the users.

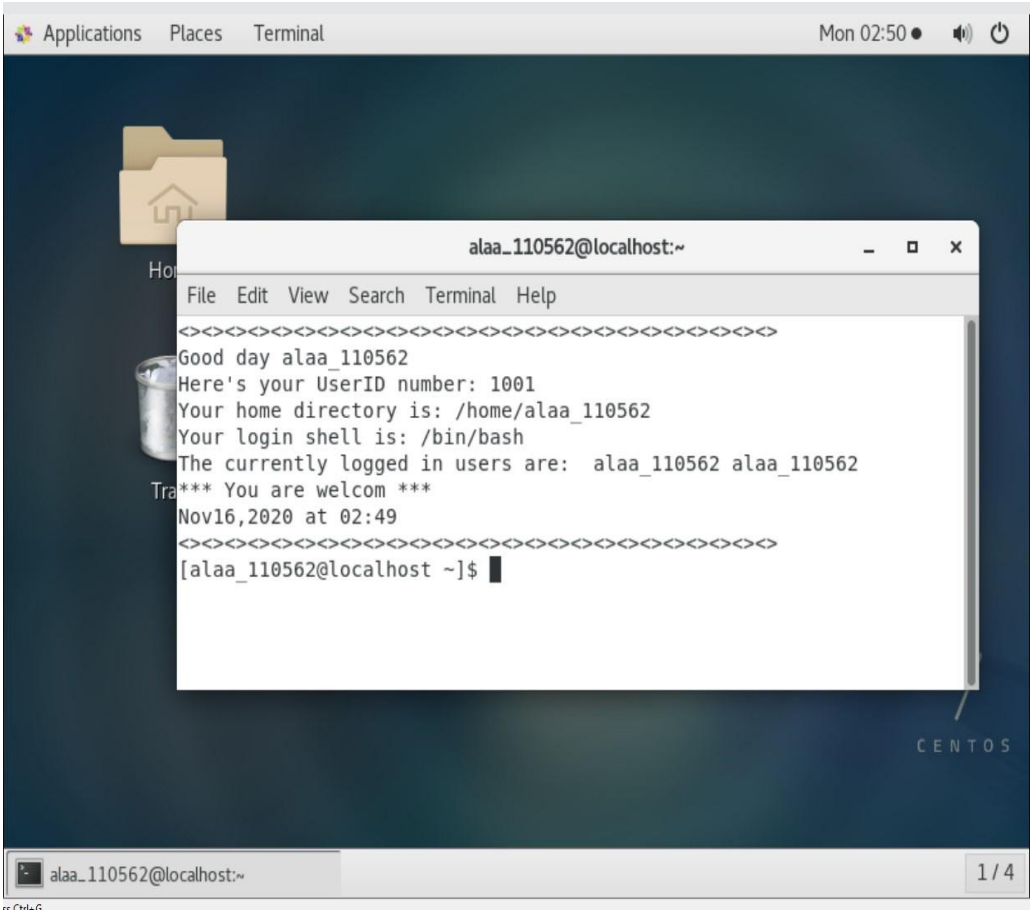


```
#!/bin/bash
time=`date +%H` #Define a variable and assign it with the hours part from the curent
date
hello=""  #Define a string variable to be assigned later in the script

if (($time >= 0 & $time < 12)); then
hello="Good day"; #Check if time is between 0 and 12 and set hello to good day if so
elif (($time >= 12 & $time < 24)); then
hello="Good evening"; #Check if time is between 12 and 24 and set hello to good evening
if so
fi
echo "<><><><><><><><><><><><><><><><><><><><><><><><><><>"
echo $hello $USER #Print the hello
echo "Here's your UserID number: $UID" #Print the current user UID from the environment
variable
echo "Your home directory is: $HOME" #Print the current user HOME from the environment
variable
echo "Your login shell is: $SHELL" #Print the current user SHELL from the environment
variable
echo "The currently logged in users are: " `users` #Print the currently logged in users
echo "*** You are welcom ***"
NOW=$(date +"%b%e,%Y at %R")
echo $NOW
echo "<><><><><><><><><><><><><><><><><><><><><><><><><><>"
```

4) Output:



```
<><><><><><><><><><><><><><><><><><><><><><><><><><>
Good evening alaa_110562
Here's your UserID number: 1001
Your home directory is: /home/alaa_110562
Your login shell is: /bin/bash
The currently logged in users are:  alaa_110562 alaa_110562
*** You are welcom ***
Nov15,2020 at 19:40
<><><><><><><><><><><><><><><><><><><><><><><><><><>
[alaa_110562@localhost ~]$ 
```



```
<><><><><><><><><><><><><><><><><><><><><><><><><><>
Good day alaa_110562
Here's your UserID number: 1001
Your home directory is: /home/alaa_110562
Your login shell is: /bin/bash
The currently logged in users are:  alaa_110562 alaa_110562
*** You are welcom ***
Nov16,2020 at 02:49
<><><><><><><><><><><><><><><><><><><><><><><><><><>
[alaa_110562@localhost ~]$ █
```

Exercise (2)

Write a shell script that takes a valid directory path and then:

- It checks the validity of the input directory, and respond with suitable error message if not.

- It prints the count of files and subdirectories in it.

- It gets the total count of the word “bash” in all the script files and across files present in subdirectories.

Note: you should prepare the directory with required data to test your execution on it.

Answer (2)

1) **The script:**

```
#! bin\bash
clear
echo "It's a shell script that takes a valid directory path"
#It checks the validity of the input directory, and respond with suitable error message if not.
read -p "Enter your directory:" DIRECTORY
```

if [-d "$DIRECTORY"]; then

Will enter here if $DIRECTORY exists, even if it contains spaces

echo "**** It's a valid directory path ****"

echo "Number of directories in $DIRECTORY = "$(find $DIRECTORY/* -type d | wc -l)

echo "Number of Files in $DIRECTORY = "$(find $DIRECTORY/* -type f | wc -l)

It prints the count of files and subdirectories in it

#[use find the scan the dir tree and wc will do the rest]

echo "Number of word bash in $DIRECTORY = "$(grep -rcP '^bash$' $DIRECTORY | wc -w)

#gets the total count of the word "bash" in all the script files and across files present in subdirectories (only the count).

else

echo "It's not a valid directory path"

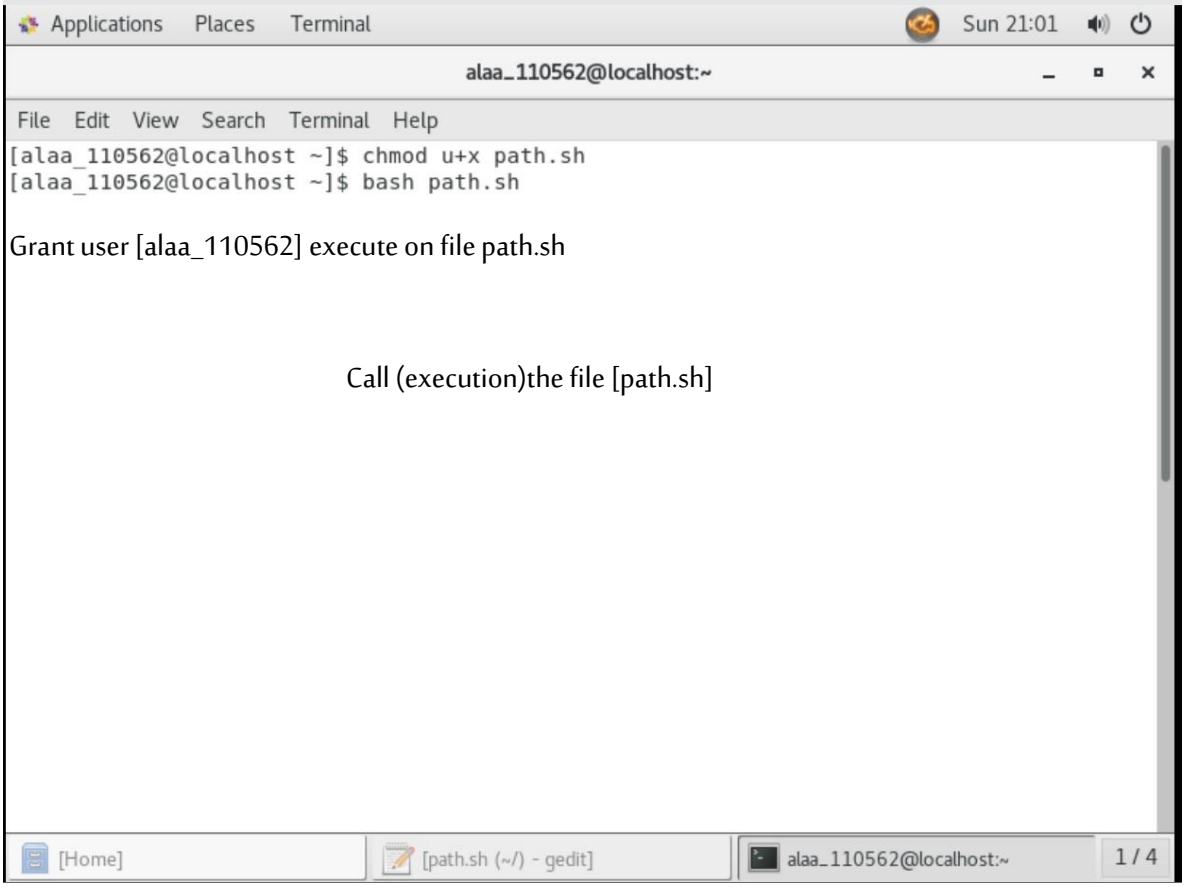
fi

put this script in file (path.sh) and call (execution) it in terminal by use [bash path.sh]

2) **Output:**

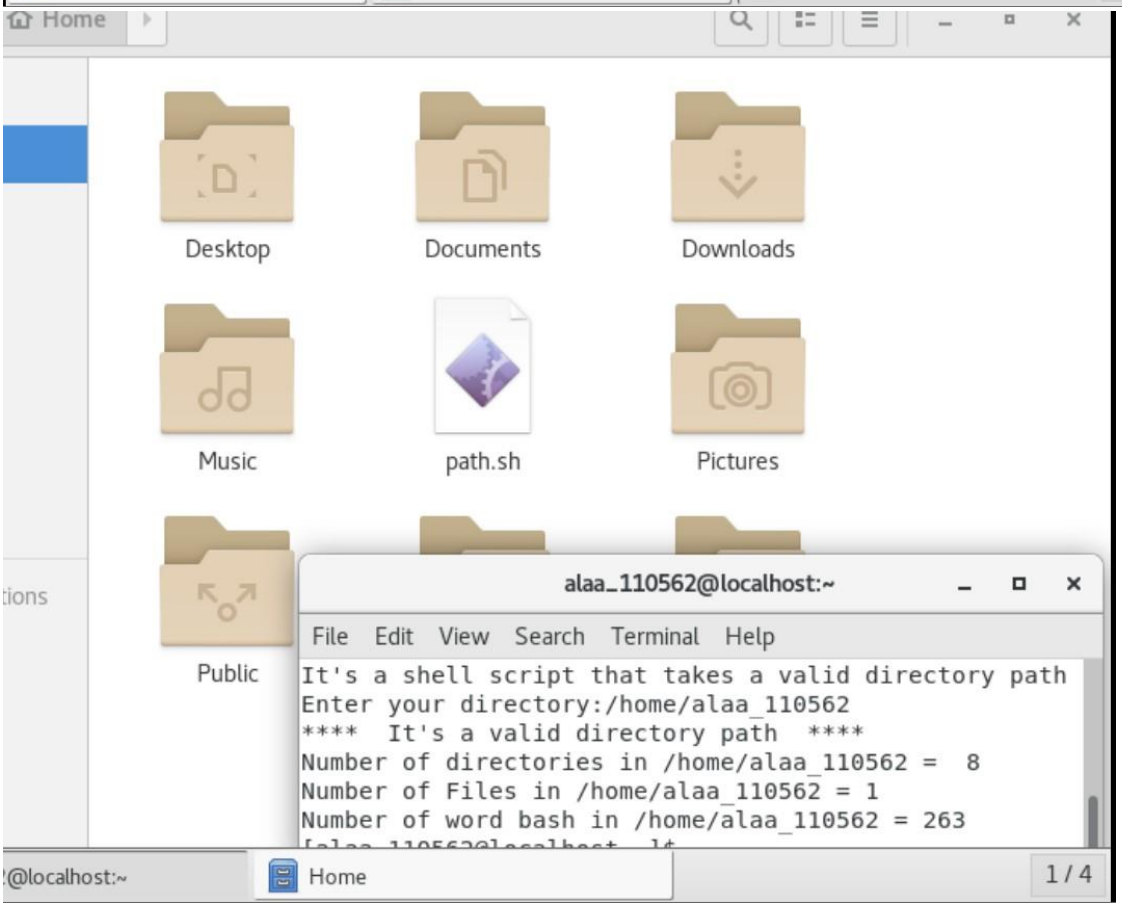
Open path.sh Save

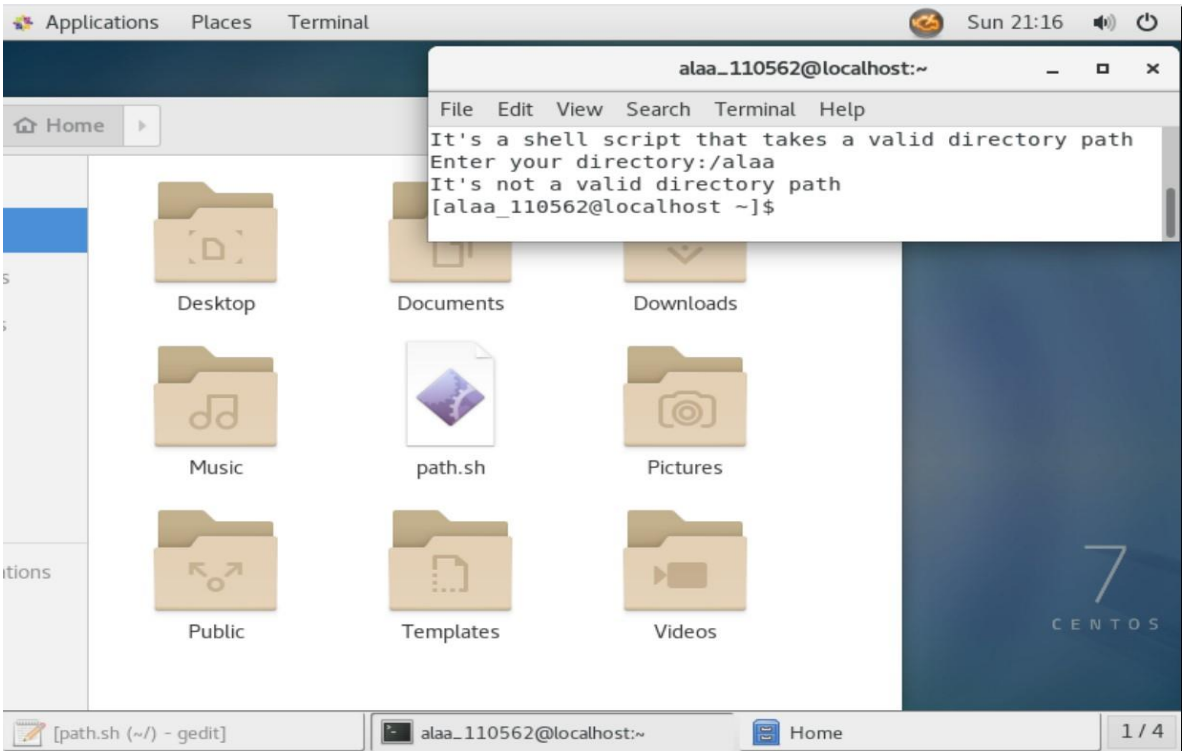
```
#! bin\bash
clear
echo "It's a shell script that takes a valid directory path"
#It checks the validity of the input directory, and respond with suitable error message
if not.
read -p "Enter your directory:" DIRECTORY

if [ -d "$DIRECTORY" ]; then
    # Will enter here if $DIRECTORY exists, even if it contains spaces
echo "****  It's a valid directory path  ****"

echo "Number of directories in $DIRECTORY =  "$(find $DIRECTORY/* -type d | wc -l)

echo "Number of Files in $DIRECTORY = "$(find $DIRECTORY/* -type f | wc -l)

echo "Number of word bash in $DIRECTORY = "$(grep -rcP '^bash$' $DIRECTORY | wc -w)

else
echo "It's not a valid directory path"
fi
```



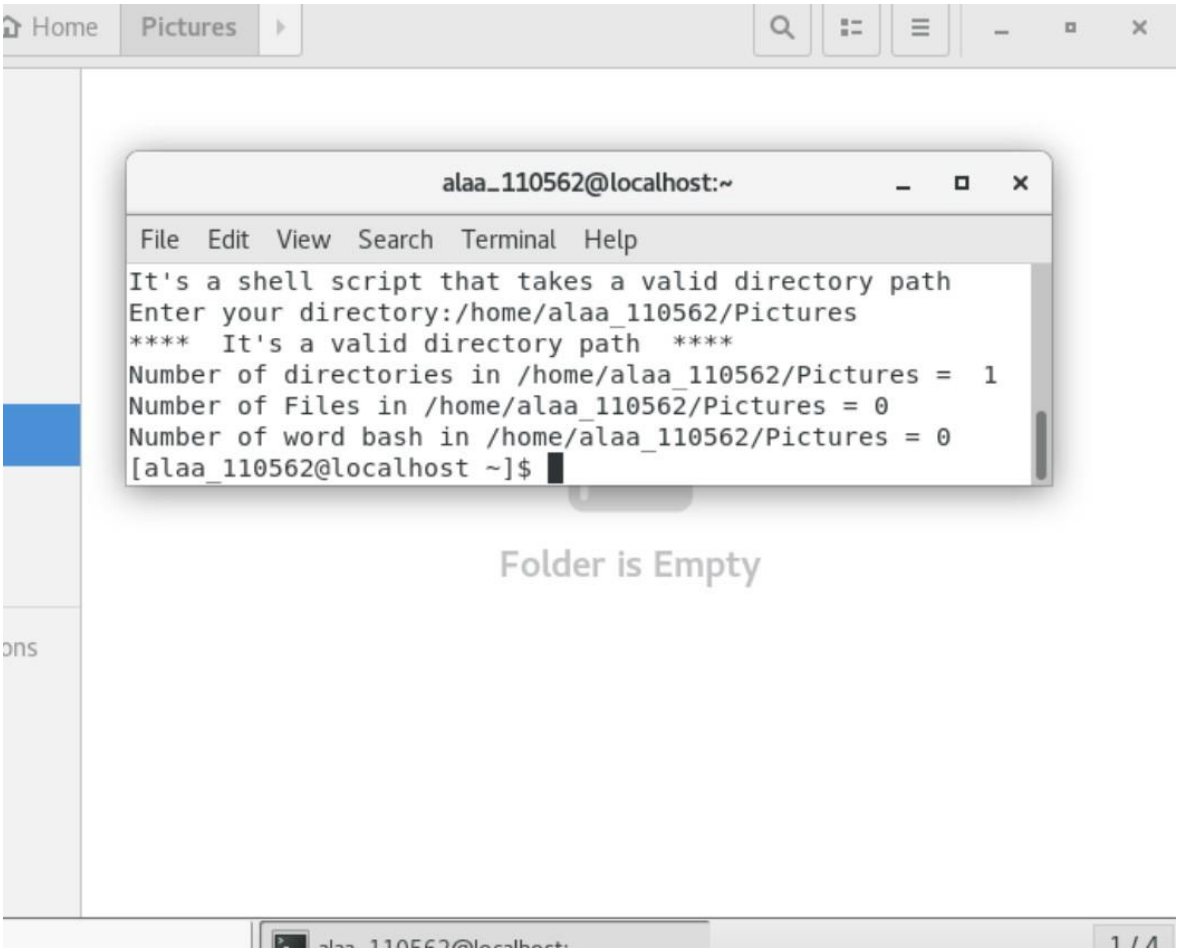
Grant user [alaa_110562] execute on file path.sh

Call (execution)the file [path.sh]





Other directory



Other method to checks the validity of the input directory by using:

[Grep " input directory " $DIRECTORY]

write it in new script file [new-path] and execute it [bash new-path.sh]

```
#! bin\bash
clear
echo "It's a shell script that takes a valid directory path"
#It checks the validity of the input directory, and respond with suitable error message
if not.
read -p "Enter your directory:" DIRECTORY

grep "your directory" $DIRECTORY
```

```
alaa_110562@localhost:~
File  Edit  View  Search  Terminal  Help

It's a shell script that takes a valid directory path
Enter your directory:/home/alaa_110562
grep: /home/alaa_110562: Is a directory
[alaa_110562@localhost ~]$
```

```
alaa_110562@localhost:~
File  Edit  View  Search  Terminal  Help
It's a shell script that takes a valid directory path
Enter your directory:/lolo
grep: /lolo: No such file or directory
[alaa_110562@localhost ~]$
```

Exercise (3)

Write a shell script to delete the lines - in a given file - containing a word "le" if it appears between the 3th and 10th position (character).

To enhance your script, you should read the word and the positions as an input from the user, where the positions should be numbers, otherwise the script should halt and respond with suitable error message.

Note: you should prepare the file with required data to test your execution on it.

Answer (3)

1) **The first script: [put this script in file (delete.sh)]**

```
#! bin\bash
FILE="file.data"
i=0
echo "***************CAT file.data*************"
while IFS= read -r line  # reading each line
do
 i=$(( $i + 1 ))
 POS=$(echo "$line"|awk 'END{print index($0,"le")}' )

 if (( $POS > 2 & $POS < 10 )) ; then

   echo "$i/d"
  else
   echo $line
 fi
done < $FILE
```

Applications Places Text Editor

Open delete.sh ~/

```
#! bin\bash
FILE="file.data"
i=0
echo "***************CAT file.data*************"
while IFS= read -r line  # reading each line
do
  i=$(( $i + 1 ))
  POS=$(echo "$line"|awk 'END{print index($0,"le")}' )

  if (( $POS > 2 & $POS < 10 )) ; then

     echo "$i/d"
   else
     echo $line
  fi
done < $FILE
```

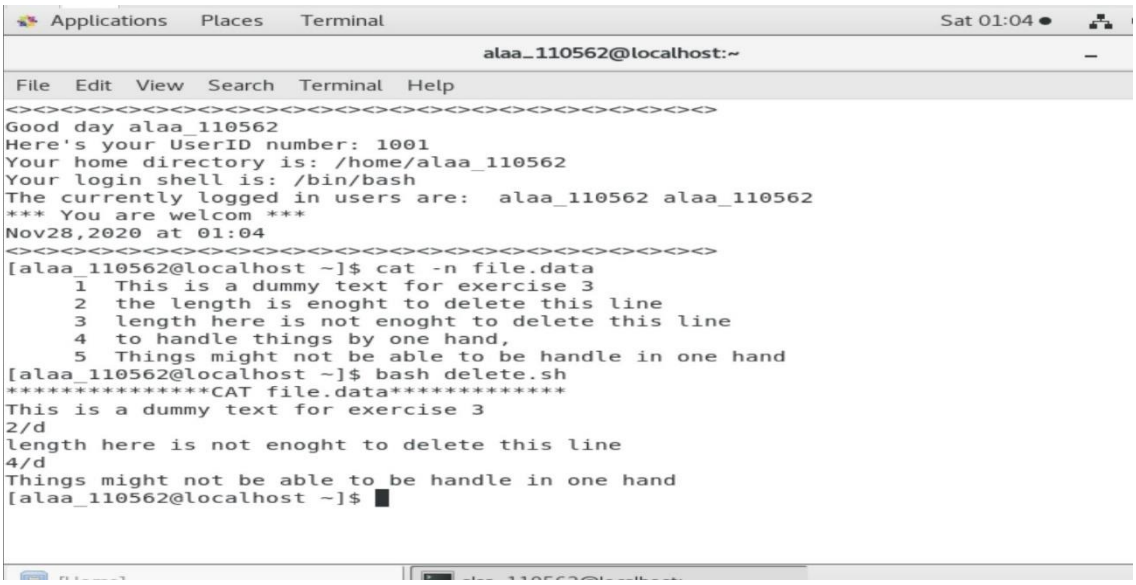
2) **The first output:**

File.data:

Open file.data ~/ Save

```
This is a dummy text for exercise 3
the length is enoght to delete this line
length here is not enoght to delete this line
to handle things by one hand,
Things might not be able to be handle in one hand
```

Output:



```
<><><><><><><><><><><><><><><><><><><><><><><><><><>
Good day alaa_110562
Here's your UserID number: 1001
Your home directory is: /home/alaa_110562
Your login shell is: /bin/bash
The currently logged in users are:  alaa_110562 alaa_110562
*** You are welcom ***
Nov28,2020 at 01:04
<><><><><><><><><><><><><><><><><><><><><><><><><><>
[alaa_110562@localhost ~]$ cat -n file.data
     1  This is a dummy text for exercise 3
     2  the length is enoght to delete this line
     3  length here is not enoght to delete this line
     4  to handle things by one hand,
     5  Things might not be able to be handle in one hand
[alaa_110562@localhost ~]$ bash delete.sh
***************CAT file.data*************
This is a dummy text for exercise 3
2/d
length here is not enoght to delete this line
4/d
Things might not be able to be handle in one hand
[alaa_110562@localhost ~]$ █
```

3) The second script: [put this script in file (delete_2.sh)]

```
#! bin\bash
FILE="file.data"
i=0
read -p "Search for:" STR
read -p "Start Position:" SP
read -p "End Position:" EP
if ((echo "$SP" | grep -qE '^[0-9]+$') && (echo "$EP" | grep -qE '^[0-9]+$')) ; then
echo "***************CAT file.data*************"
while IFS= read -r line  # reading each line
do
 i=$(( $i + 1 ))
 POS=$(echo "$line"|awk 'END{print index($0,"'"$STR"'")}')
 if (( POS >= SP && POS <= EP ));then
   echo "$i/d"
  else
   echo $line
 fi
done < $FILE
else
echo "error: SP OR EP is Not a number" >&2; exit 1
fi
```

Applications Places Text Editor Sat 00:48

Open delete_2.sh Save

```
#! bin\bash
FILE="file.data"
i=0
read -p "Search for:" STR
read -p "Start Position:" SP
read -p "End Position:" EP
if ((echo "$SP" | grep -qE '^[0-9]+$') & (echo "$EP" | grep -qE '^[0-9]+$')) ; then
echo "***************CAT file.data*************"
while IFS= read -r line  # reading each line
do
  i=$(( $i + 1 ))
  POS=$(echo "$line"|awk 'END{print index($0,"'$STR'")}')
  if (( POS >= SP && POS <= EP ));then
     echo "$i/d"
   else
     echo $line
  fi

done < $FILE

else
echo "error: SP OR EP Not an number" >&2; exit 1
fi
```

4) **The second output:**

Applications Places Terminal Sat 01:03

alaa_110562@localhost:~

File Edit View Search Terminal Help

```
[alaa_110562@localhost ~]$ cat -n file.data
     1  This is a dummy text for exercise 3
     2  the length is enoght to delete this line
     3  length here is not enoght to delete this line
     4  to handle things by one hand,
     5  Things might not be able to be handle in one hand
[alaa_110562@localhost ~]$ bash delete_2.sh
Search for:le
Start Position:t
End Position:5
error: SP OR EP is Not a number
[alaa_110562@localhost ~]$ bash delete_2.sh
Search for:le
Start Position:7
End Position:h
error: SP OR EP is Not a number
[alaa_110562@localhost ~]$ bash delete_2.sh
Search for:le
Start Position:3
End Position:10
***************CAT file.data*************
This is a dummy text for exercise 3
2/d
length here is not enoght to delete this line
4/d
Things might not be able to be handle in one hand
```

Exercise (4)

Write a script that receives four parameters (file names), and does the following tasks:

- Checks the number of arguments and halts with suitable error message when they are not 4.
- Checks whether those files exist or not,
- If a file exists, then append its content to the end of the file "content.txt", be sure to separate each content in the resulted file with couple of empty lines and the files names you are processing.

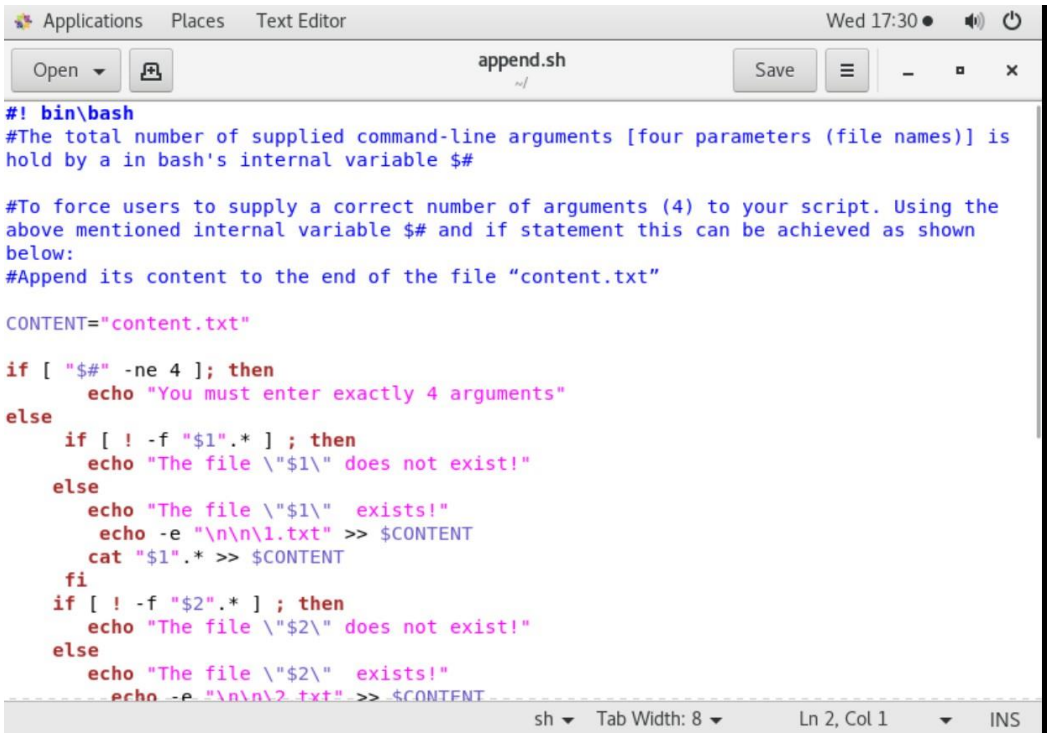
Note: you should prepare the files to handle with dummy data and the "content.txt".

Answer (4)

1) **The script: [put this script in file (append.sh)]**

```
#! bin\bash
#The total number of supplied command-line arguments [four parameters (file names)] is hold by a in bash's internal variable $#

#To force users to supply a correct number of arguments (4) to your script. Using the above mentioned internal variable $# and if statement this can be achieved as shown below:
#Append its content to the end of the file "content.txt"

CONTENT="content.txt"

if [ "$#" -ne 4 ]; then
    echo "You must enter exactly 4 arguments"
else
   if [ ! -f "$1".* ] ; then
    echo "The file \"$1\" does not exist!"
  else
    echo "The file \"$1\"  exists!"
     echo -e "\n\n\1.txt" >> $CONTENT
    cat "$1".* >> $CONTENT
   fi
  if [ ! -f "$2".* ] ; then
    echo "The file \"$2\" does not exist!"
  else
    echo "The file \"$2\"  exists!"
     echo -e "\n\n\2.txt" >> $CONTENT
```

```
    cat "$2".* >> $CONTENT
  fi
  if [ ! -f "$3".* ] ; then
    echo "The file \"$3\" does not exist!"
  else
    echo "The file \"$3\"  exists!"
    echo -e "\n\n\3.txt" >> $CONTENT
    cat "$3".* >> $CONTENT
  fi
  if [ ! -f "$4".* ] ; then
    echo "The file \"$4\" does not exist!"
  else
    echo "The file \"$4\"  exists!"
      echo -e "\n\n\4.txt" >> $CONTENT
    cat "$4".* >> $CONTENT
  fi
cat $CONTENT
fi
```



```
      cat "$1".* >> $CONTENT
    fi
    if [ ! -f "$2".* ] ; then
       echo "The file \"$2\" does not exist!"
    else
       echo "The file \"$2\"  exists!"
         echo -e "\n\n\2.txt" >> $CONTENT
        cat "$2".* >> $CONTENT
     fi
     if [ ! -f "$3".* ] ; then
       echo "The file \"$3\" does not exist!"
    else
       echo "The file \"$3\"  exists!"
        echo -e "\n\n\3.txt" >> $CONTENT
       cat "$3".* >> $CONTENT
     fi
     if [ ! -f "$4".* ] ; then
       echo "The file \"$4\" does not exist!"
    else
       echo "The file \"$4\"  exists!"
          echo -e "\n\n\4.txt" >> $CONTENT
        cat "$4".* >> $CONTENT
     fi
cat $CONTENT
fi
```

2) **The required files:**

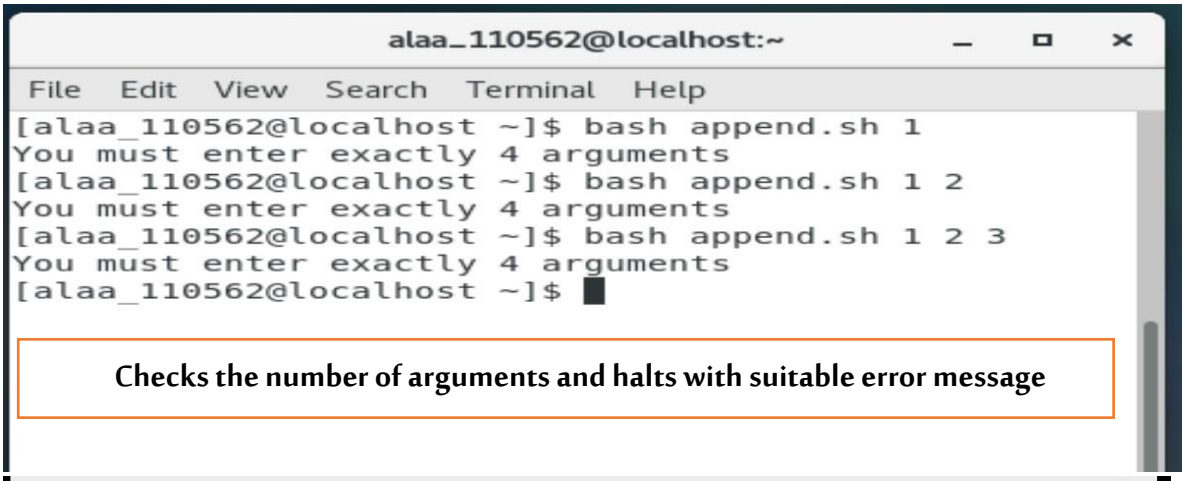
The content file:

```
Alaa Mohammad Ajelo
alaa_110562
```

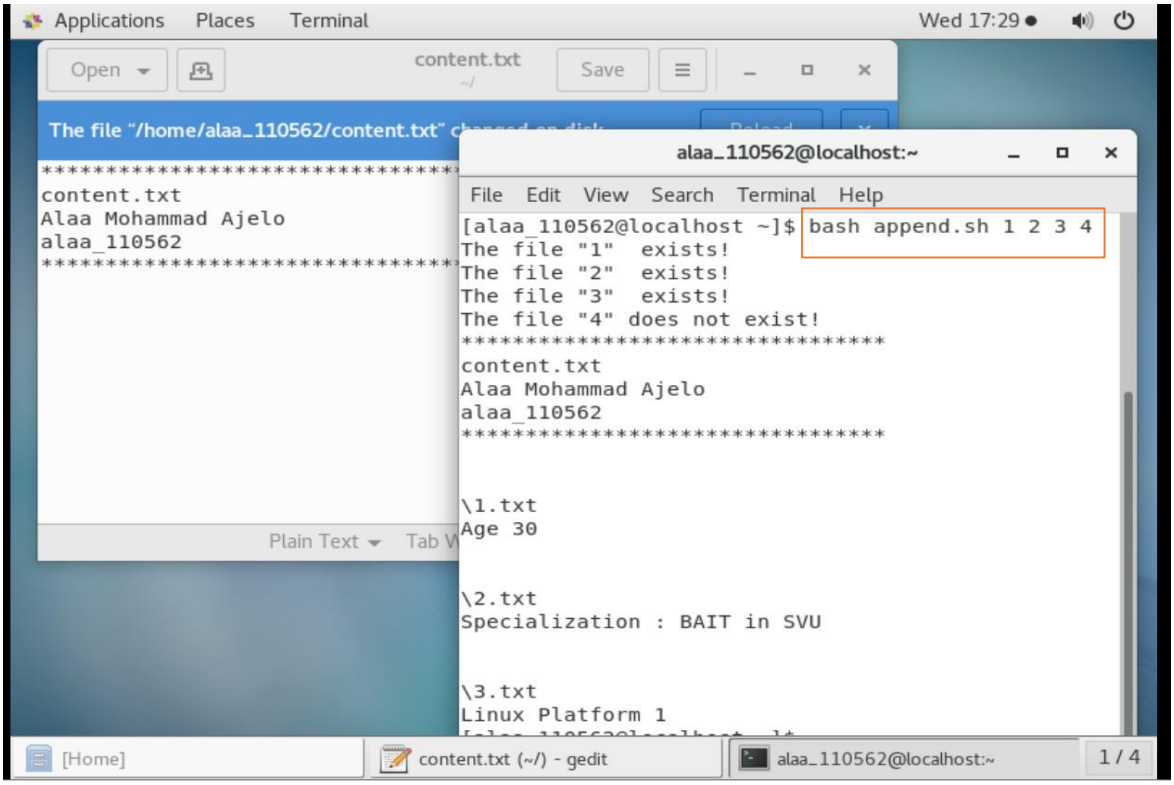
The three files:

```
Age 30
```

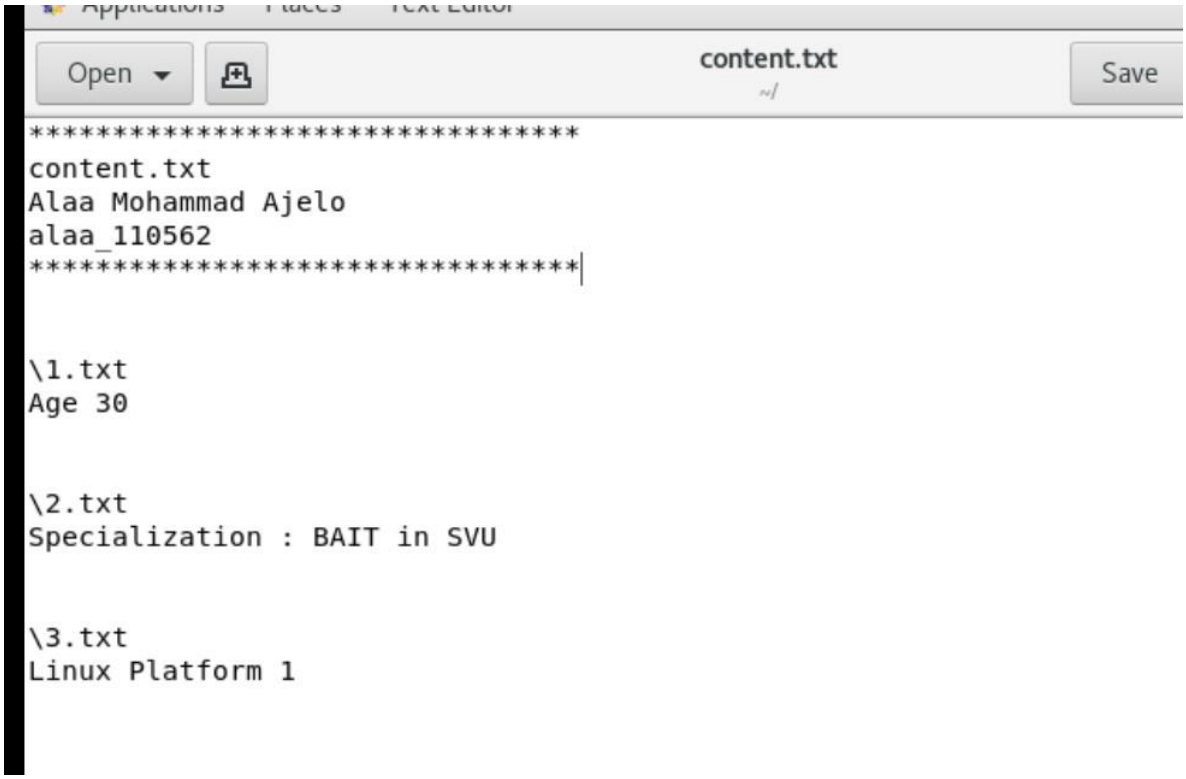
3) **Output:**



Checks the number of arguments and halts with suitable error message



After reload file (content.txt)



Exercise (5)

Using only one command, create the following directory tree.

```
├── Dir5
│   ├── Homeworks
│   │   ├── HW_F19
│   │   ├── HW_S19
│   │   └── HW_S20
│   ├── Operating\ Systems
│   │   ├── CentOs
│   │   └── Windows
│   └── Working
│       ├── Drafts
│       └── Scripts
```

Using only one command, and in the directory "Drafts", create set of 10 files with the following names pattern: script-X-file where X is a number between 0 and 9.

- Then write a shell script that adds an extension ".sh" to all the files previously created.

- Using only one command, move the converted files to the "Scripts" directory.

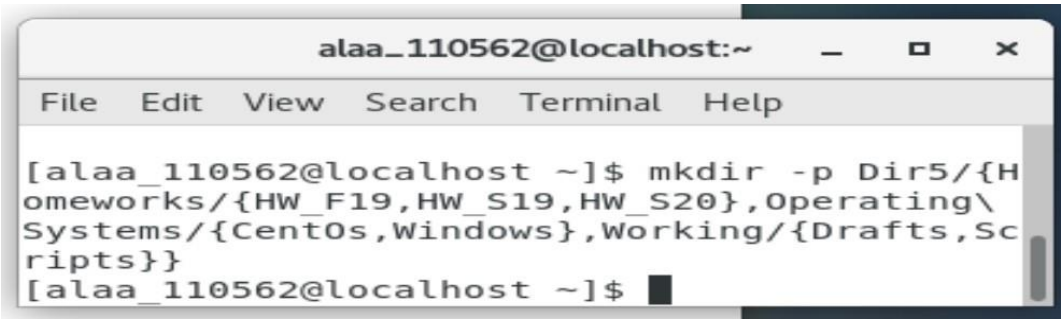
Note: Be sure to proof your steps one by one.

Answer (5)

1) **The first script:**

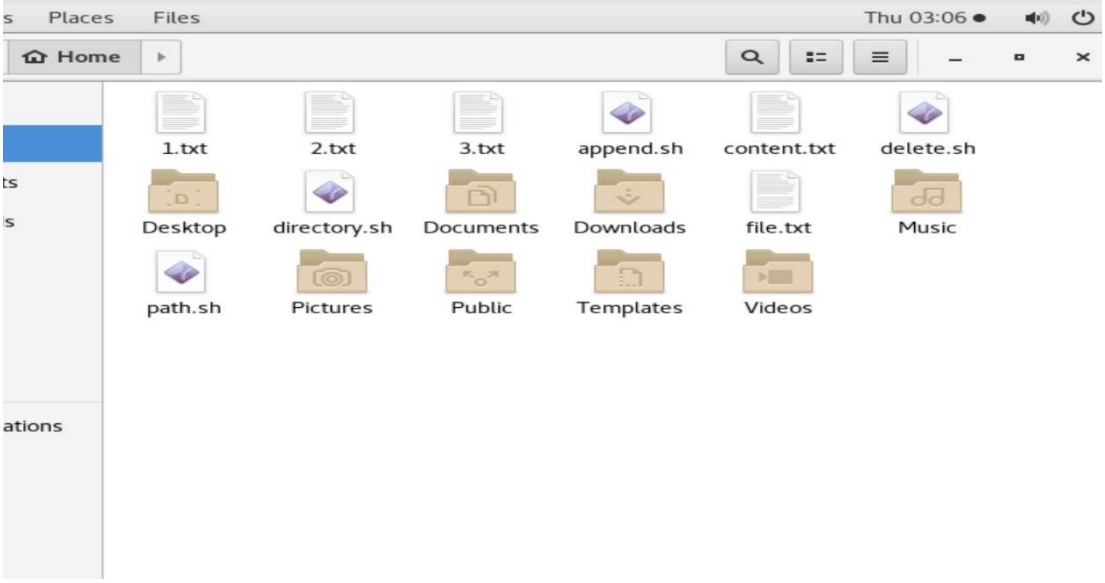
✓ Directory Creation

mkdir -p Dir5/{Homeworks/{HW_F19,HW_S19,HW_S20},Operating\ Systems/{CentOs,Windows},Working/{Drafts,Scripts}}

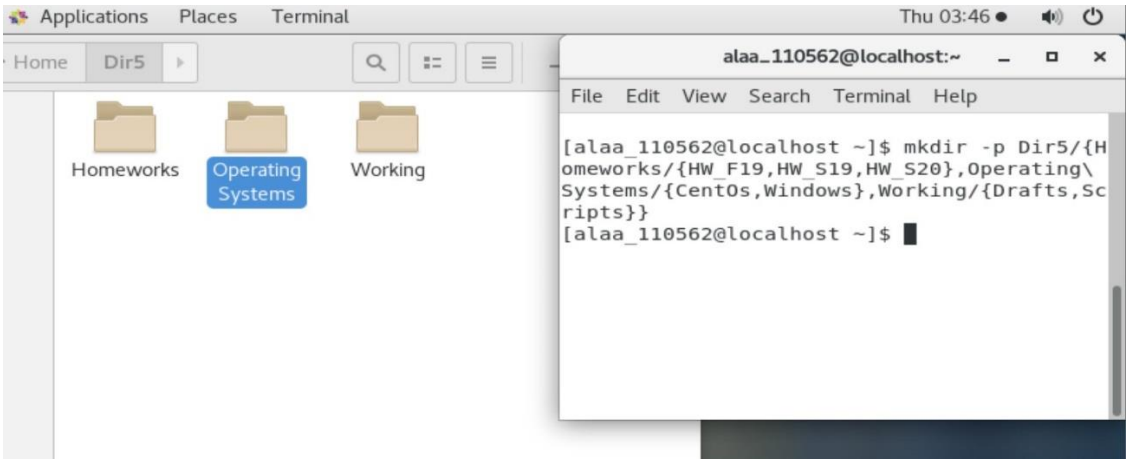


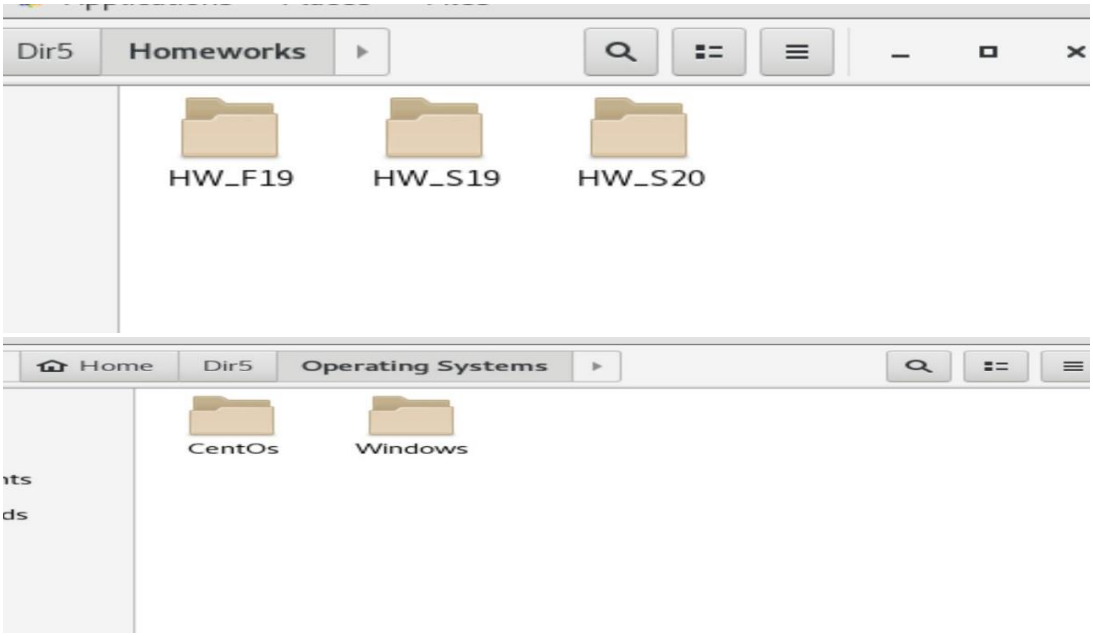
✓ Output of direcrory tree:

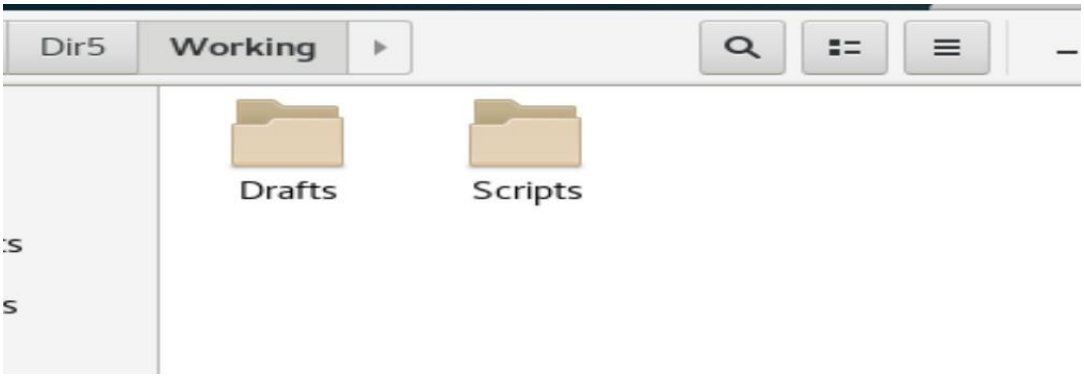
Before creation the previous tree



After creation







2) **The second script:**

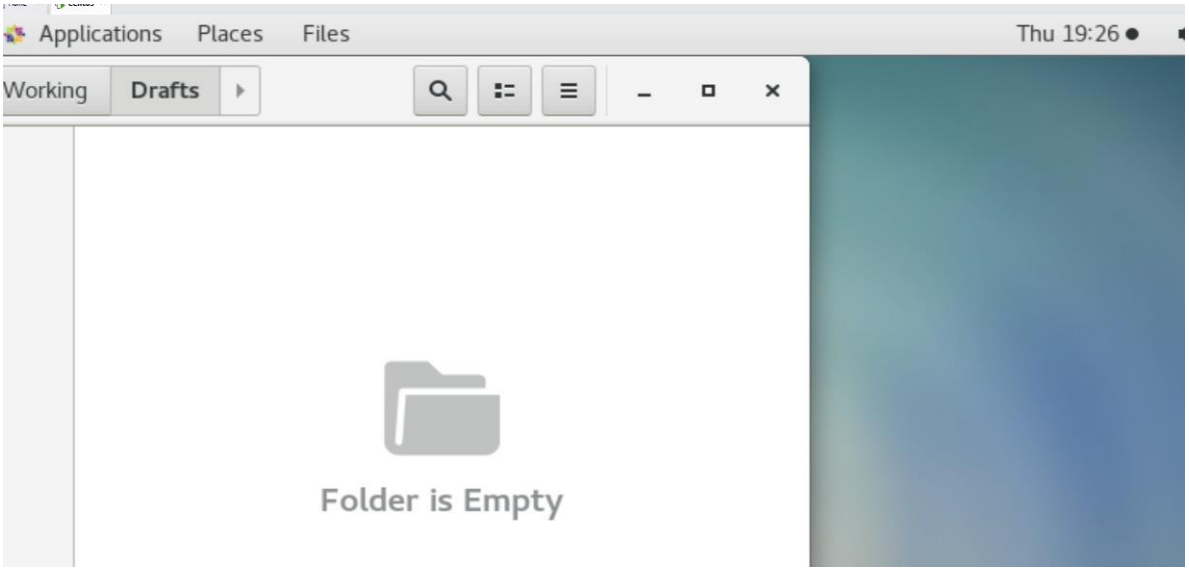
✓ Files Creation in the directory "Drafts" [in Terminal]

After change directory to Drafts

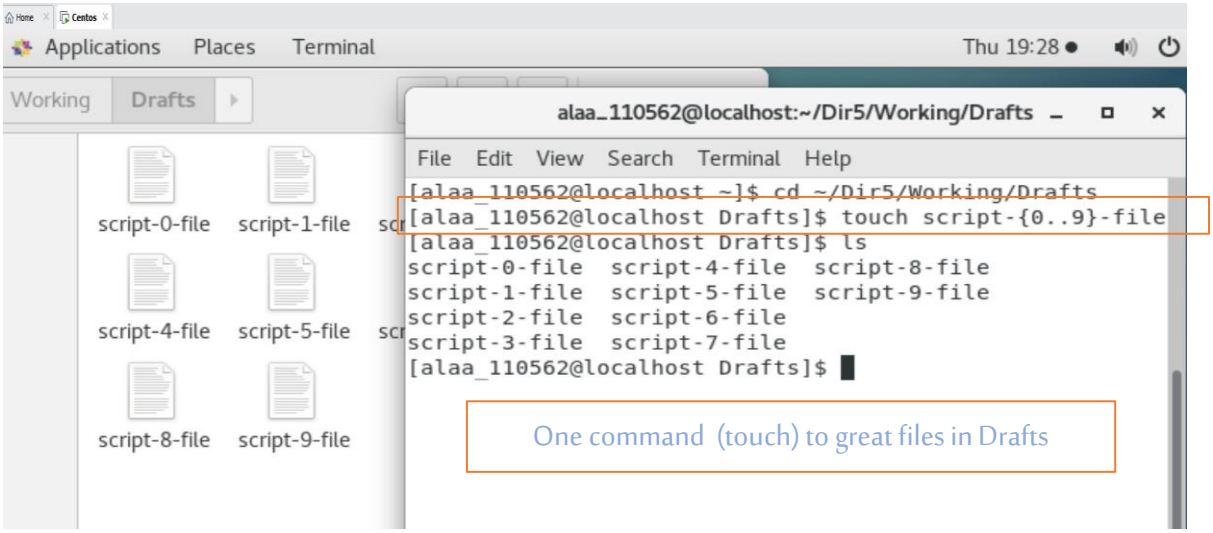
```
[alaa_110562@localhost Drafts]$ touch script-{0..9}-file  # Using touch to create a file
[alaa_110562@localhost Drafts]$ ls   # to list contents of a directory
script-0-file  script-4-file  script-8-file
script-1-file  script-5-file  script-9-file
script-2-file  script-6-file
script-3-file  script-7-file
[alaa_110562@localhost Drafts]$
```

✓ Output of folder [Drafts]:

Before creation the previous files



After creation



3) **The third script:**

✓ Shell script that adds an extension ".sh" to all the files previously created

[in script file (rename_files.sh)]

```
#!/bin/sh
cd $1
names_1=`ls`
for file in ${names_1}
do
mv ${file} ${file}.sh
done
```

```
#!/bin/sh
cd $1
names_1=`ls`
for file in ${names_1}
do
mv ${file} ${file}.sh
done
```

✓ [in Terminal]

[alaa_110562@localhost Drafts]$ cd ~/Dir5/Working/Drafts

[alaa_110562@localhost Drafts]$ ls

script-0-file script-2-file script-4-file script-6-file script-8-file

script-1-file script-3-file script-5-file script-7-file script-9-file

[alaa_110562@localhost Drafts]$ cd ~

[alaa_110562@localhost ~]$ bash rename_files.sh ~/Dir5/Working/Drafts

[alaa_110562@localhost ~]$ cd ~/Dir5/Working/Drafts

[alaa_110562@localhost Drafts]$ ls

script-0-file.sh script-3-file.sh script-6-file.sh script-9-file.sh

script-1-file.sh script-4-file.sh script-7-file.sh

script-2-file.sh script-5-file.sh script-8-file.sh

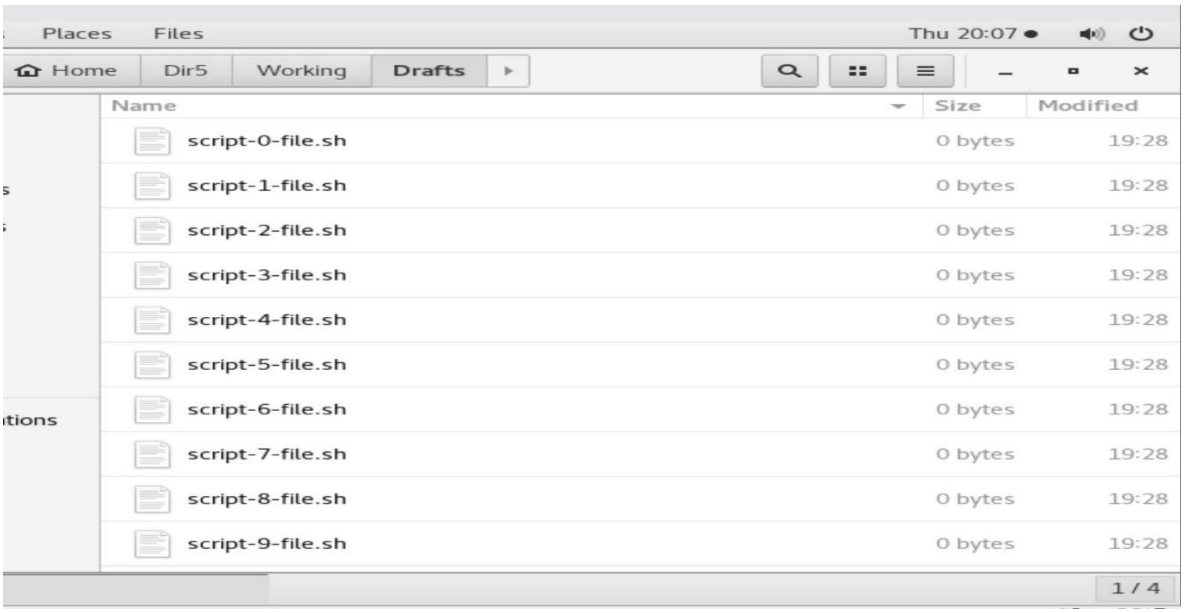
[alaa_110562@localhost Drafts]$

✓ Output of folder [Drafts]:

From terminal

```
[alaa_110562@localhost Drafts]$ cd ~/Dir5/Working/Drafts
[alaa_110562@localhost Drafts]$ ls
script-0-file  script-2-file  script-4-file  script-6-file  script-8-file
script-1-file  script-3-file  script-5-file  script-7-file  script-9-file
[alaa_110562@localhost Drafts]$ cd ~
[alaa_110562@localhost ~]$ bash rename_files.sh ~/Dir5/Working/Drafts
[alaa_110562@localhost ~]$ cd ~/Dir5/Working/Drafts
[alaa_110562@localhost Drafts]$ ls
script-0-file.sh  script-3-file.sh  script-6-file.sh  script-9-file.sh
script-1-file.sh  script-4-file.sh  script-7-file.sh
script-2-file.sh  script-5-file.sh  script-8-file.sh
[alaa_110562@localhost Drafts]$
```

From folder



4) The fourth script:

✓ Move the converted files to the "Scripts" directory. [in Terminal]

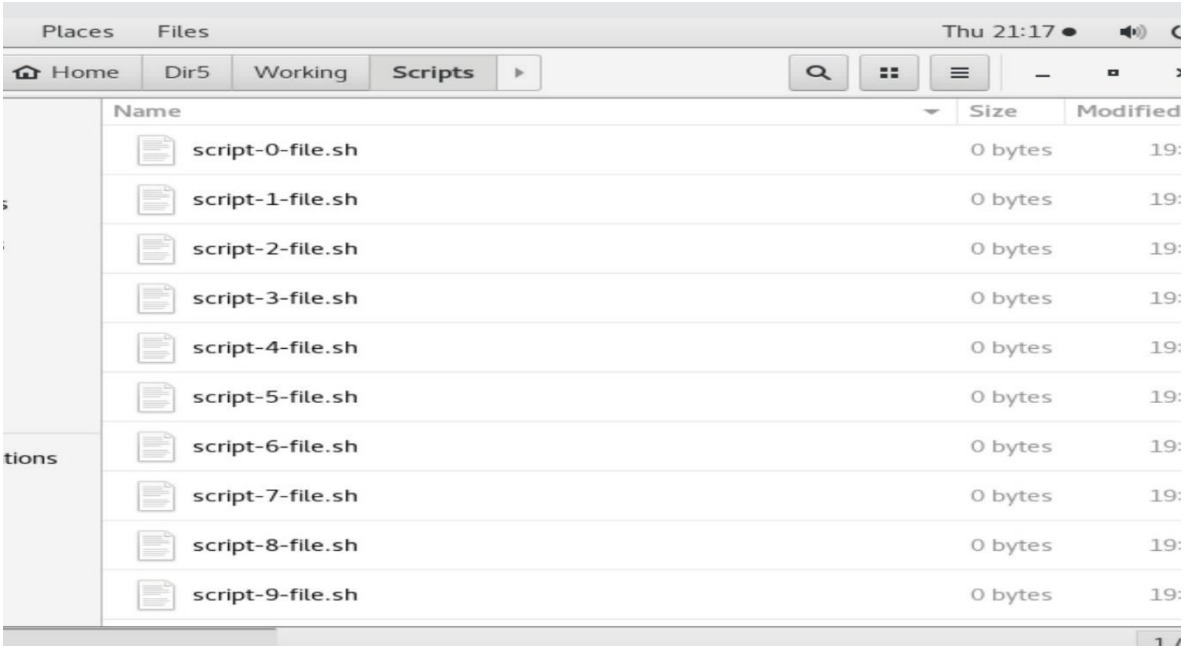
```
[alaa_110562@localhost Working]$ cd ~/Dir5/Working/Drafts
[alaa_110562@localhost Drafts]$ mv *.sh ~/Dir5/Working/Scripts
[alaa_110562@localhost Drafts]$ ls
[alaa_110562@localhost Drafts]$ cd ~/Dir5/Working/Scripts
[alaa_110562@localhost Scripts]$ ls
script-0-file.sh  script-3-file.sh  script-6-file.sh  script-9-file.sh
script-1-file.sh  script-4-file.sh  script-7-file.sh
script-2-file.sh  script-5-file.sh  script-8-file.sh
[alaa_110562@localhost Scripts]$
```

✓ Output

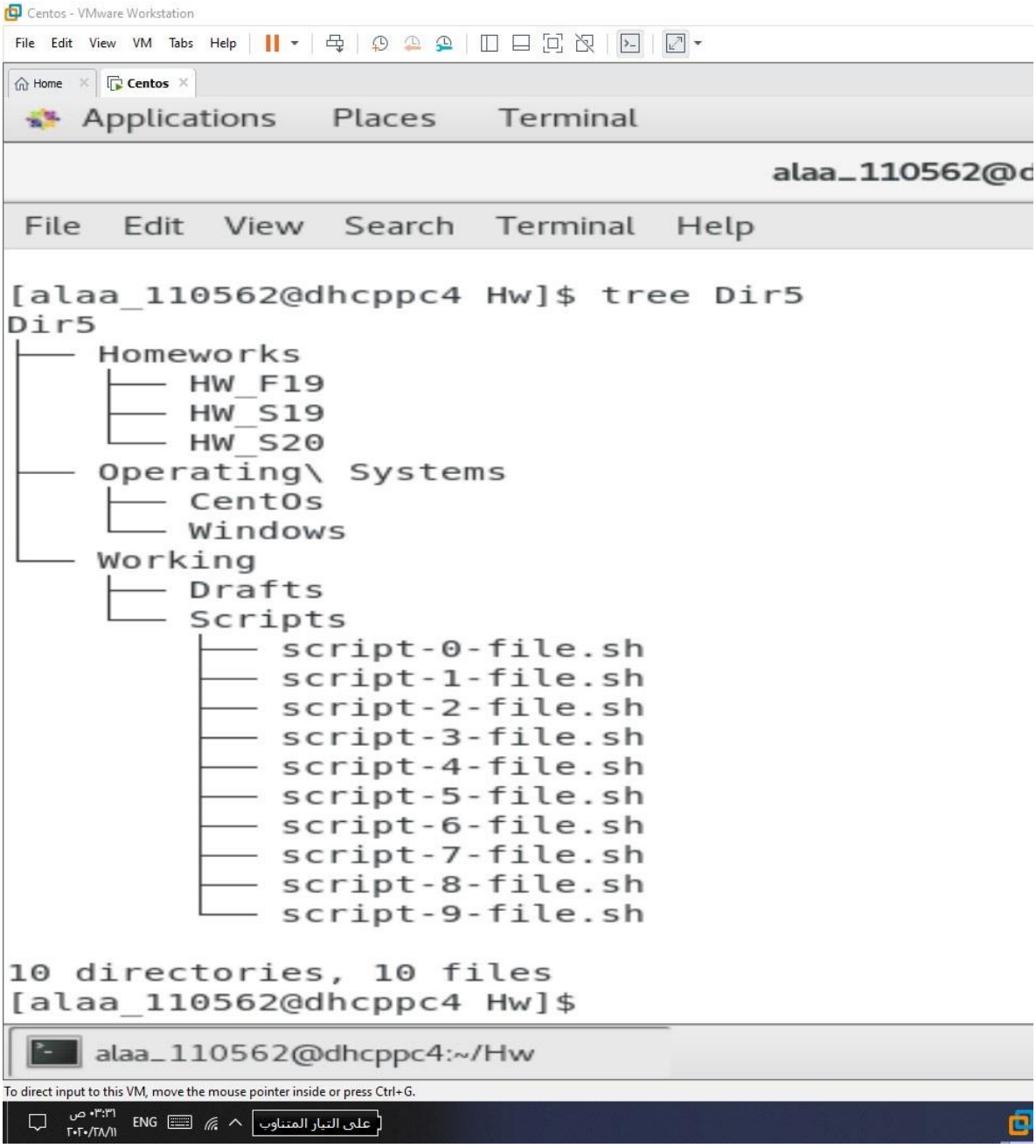
From terminal

```
alaa_110562@localhost:~/Dir5/Working/Scripts
File  Edit  View  Search  Terminal  Help
[alaa_110562@localhost Working]$ cd ~/Dir5/Working/Drafts
[alaa_110562@localhost Drafts]$  mv *.sh ~/Dir5/Working/Scripts
[alaa_110562@localhost Drafts]$ ls
[alaa_110562@localhost Drafts]$ cd ~/Dir5/Working/Scripts
[alaa_110562@localhost Scripts]$ ls
script-0-file.sh  script-3-file.sh  script-6-file.sh  script-9-file.sh
script-1-file.sh  script-4-file.sh  script-7-file.sh
script-2-file.sh  script-5-file.sh  script-8-file.sh
[alaa_110562@localhost Scripts]$
```

From folder



5) **The tree:**


Centos - VMware Workstation
File Edit View VM Tabs Help
Home
Centos
Applications Places Terminal
alaa_110562@d
File Edit View Search Terminal Help
[alaa_110562@dhcppc4 Hw]$ tree Dir5
Dir5
├── Homeworks
│ ├── HW_F19
│ ├── HW_S19
│ └── HW_S20
├── Operating\ Systems
│ ├── CentOs
│ └── Windows
└── Working
  ├── Drafts
  └── Scripts
    ├── script-0-file.sh
    ├── script-1-file.sh
    ├── script-2-file.sh
    ├── script-3-file.sh
    ├── script-4-file.sh
    ├── script-5-file.sh
    ├── script-6-file.sh
    ├── script-7-file.sh
    ├── script-8-file.sh
    └── script-9-file.sh
10 directories, 10 files
[alaa_110562@dhcppc4 Hw]$
alaa_110562@dhcppc4:~/Hw
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
ENG
على التيار المتناوب

with regards