

OS LAB EXPERIMENT:- 3

Shell Scripting

Nithin Jose

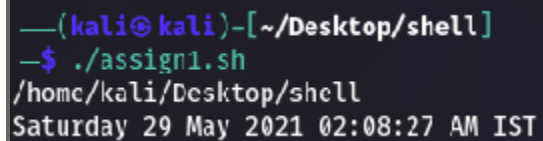
R4 46

1. Write a shell script program to display the name of the current working #directory and date.

```
#!/bin/bash
```

```
pwd
```

```
Date
```

A terminal window with a dark background. The prompt is `—(kali@kali)-[~/Desktop/shell]`. The user enters `—$./assign1.sh`. The output is `/home/kali/Desktop/shell` followed by `Saturday 29 May 2021 02:08:27 AM IST` on the next line.

```
—(kali@kali)-[~/Desktop/shell]  
—$ ./assign1.sh  
/home/kali/Desktop/shell  
Saturday 29 May 2021 02:08:27 AM IST
```

2.

Write a shell script program to perform arithmetic operation on two numbers

```
#!/bin/bash
```

```
read -p "Enter the 1st number: " a
```

```
read -p "Enter the 2nd number: " b
```

```
read -p "Enter the operation you want to perform (+ or - or x or /): " o
```

```
echo -n "$a $o $b = "
```

```
if [ $o == 'x' ]
```

```
then
```

```
    echo $((a * b))
```

```
elif [ $o == '-' ]
```

```
then
```

```
    echo $((a - b))
```

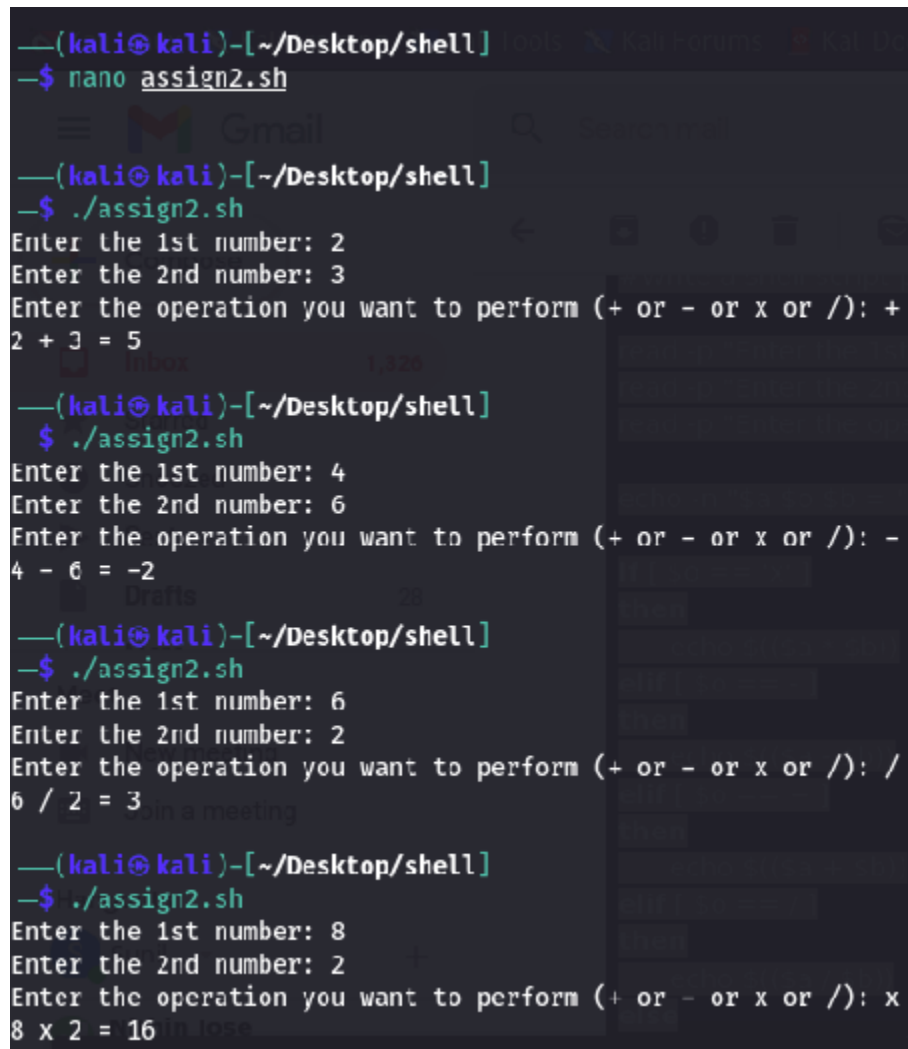
```
elif [ $o == '+' ]
```

```
then
```

```

        echo $(( $a + $b ))
    elif [ $o == / ]
    then
        echo $(( $a / $b ))
    else
        echo "$o is not a valid operation"
    fi

```



```

—(kali㉿kali)-[~/Desktop/shell]
—$ nano assign2.sh

—(kali㉿kali)-[~/Desktop/shell]
—$ ./assign2.sh
Enter the 1st number: 2
Enter the 2nd number: 3
Enter the operation you want to perform (+ or - or x or /): +
2 + 3 = 5

—(kali㉿kali)-[~/Desktop/shell]
$ ./assign2.sh
Enter the 1st number: 4
Enter the 2nd number: 6
Enter the operation you want to perform (+ or - or x or /): -
4 - 6 = -2

—(kali㉿kali)-[~/Desktop/shell]
—$ ./assign2.sh
Enter the 1st number: 6
Enter the 2nd number: 2
Enter the operation you want to perform (+ or - or x or /): /
6 / 2 = 3

—(kali㉿kali)-[~/Desktop/shell]
—$ ./assign2.sh
Enter the 1st number: 8
Enter the 2nd number: 2
Enter the operation you want to perform (+ or - or x or /): x
8 x 2 = 16

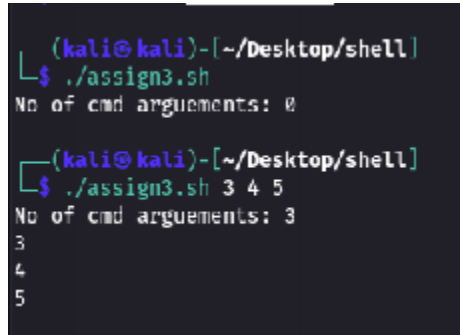
```

3. Write a shell script program to pass arguments to the program and display the count of arguments and content.

```
#!/bin/bash
```

```
echo "No of cmd arguments: $#"
```

```
for i in $@
do
    echo "$i"
done
```

A terminal window with a dark background and light-colored text. The prompt is (kali@kali)-[~/Desktop/shell]. The first command is ./assign3.sh, which outputs 'No of cmd arguments: 0'. The second command is ./assign3.sh 3 4 5, which outputs 'No of cmd arguments: 3' followed by three lines of '3', '4', and '5' respectively.

```
(kali@kali)-[~/Desktop/shell]
└─$ ./assign3.sh
No of cmd arguments: 0

(kali@kali)-[~/Desktop/shell]
└─$ ./assign3.sh 3 4 5
No of cmd arguments: 3
3
4
5
```

4.

Write a shell script program to test whether 3 arguments are being passed #to it

```
#!/bin/bash
```

```
if [ $# == 3 ]
then
    echo -n "[PASSED]"
else
    echo -n "[FAILED]"
fi

echo " $# Arguments are passed"
```

```

(kali@kali)-[~/Desktop/shell]
$ ./assign4.sh
[FAILED] 0 Arguments are passed

(kali@kali)-[~/Desktop/shell]
$ ./assign4.sh 4 5 6
[PASSED] 3 Arguments are passed

(kali@kali)-[~/Desktop/shell]
$ ./assign4.sh 4 3
[FAILED] 2 Arguments are passed

```

5

Write a shell script program to check whether two strings sent as command line arguments are same or not using test command.

```
#!/bin/bash
```

```
[[ $1 == $2 ]] && echo "Equal" || echo "Not Equal"
```

```

(kali@kali)-[~/Desktop/shell]
$ ./assign5.sh Hello Hello
Equal

(kali@kali)-[~/Desktop/shell]
$ ./assign5.sh Within Jose
Not Equal

(kali@kali)-[~/Desktop/shell]
$ ./assign5.sh Within Within
Equal

```

6.

a) Let argument be a filename. Write a shell script program to check whether the file exist or not.

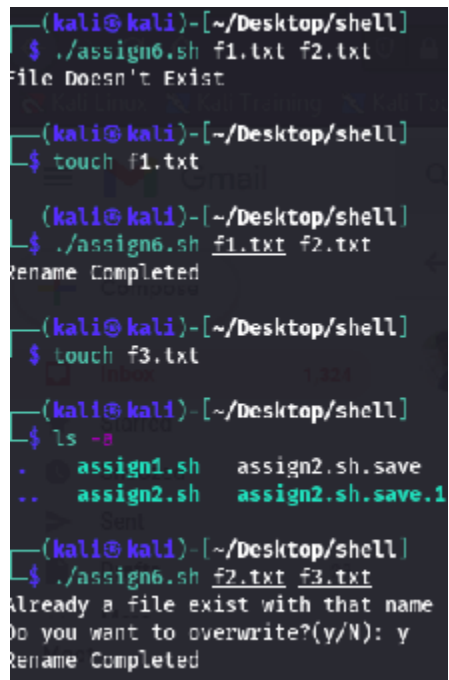
b) Modify the above program to rename a filename with another name passed as argument, Also check number of parameters being passed.

```
#!/bin/bash
```

```

if [[ -f $1 ]];then
    if [[ -f $2 ]];then
        echo "Already a file exist with that name"
        read -p "Do you want to overwrite?(y/N): " choice
        case "$choice" in
            "y"|"YES"|"Y"|"yes")
                mv $1 $2
                echo "Rename Completed";;
            *);;
        esac
    else
        mv $1 $2
        echo "Rename Completed"
    fi
else
    echo "File Doesn't Exist"
fi

```



```

(kali@kali)-[~/Desktop/shell]
$ ./assign6.sh f1.txt f2.txt
File Doesn't Exist

(kali@kali)-[~/Desktop/shell]
$ touch f1.txt

(kali@kali)-[~/Desktop/shell]
$ ./assign6.sh f1.txt f2.txt
Rename Completed

(kali@kali)-[~/Desktop/shell]
$ touch f3.txt

(kali@kali)-[~/Desktop/shell]
$ ls -la
.  assign1.sh  assign2.sh.save
.. assign2.sh  assign2.sh.save.1

(kali@kali)-[~/Desktop/shell]
$ ./assign6.sh f2.txt f3.txt
Already a file exist with that name
Do you want to overwrite?(y/N): y
Rename Completed

```

7.

Write a shell script program to test whether a string is present in a file or not

```

(kali@kali)-[~/Desktop/shell]
$ ./assign7.sh string.txt Revenant
Found

(kali@kali)-[~/Desktop/shell]
$ ./assign7.sh string.txt Photon
No found

```

8.

Write a shell script program to copy contents of file1 to file2 . If file 2 exists then append the content of file1 to its original content

```
#!/bin/bash
```

```
if [[ -f $2 ]];then
    cat $1 >> $2
```

```
else
    cp $1 $2
```

```
fi
```

```

(kali@kali)-[~/Desktop/shell]
$ ./assign8.sh f1.txt f2.txt

(kali@kali)-[~/Desktop/shell]
$ cat f2.txt
This is file 2
This is filename 1

(kali@kali)-[~/Desktop/shell]
$ ./assign8.sh f1.txt f3.txt

(kali@kali)-[~/Desktop/shell]
$ cat f3.txt
This is filename 1

```

9.

Write a shell script program that take first 10 lines from first line and rest 10 lines from the second file and move them in a third file. The filename will be sent as command line arguments.

```
#!/bin/bash
```

```
head $1 > $3
```

```
tail $1 >> $3
```

```

(kali@kali)~/Desktop/shell
$ vi c.txt

(kali@kali)~/Desktop/shell
$ vi a.txt

(kali@kali)~/Desktop/shell
$ vi b.txt

(kali@kali)~/Desktop/shell
$ ./assign9.sh a.txt b.txt c.txt

(kali@kali)~/Desktop/shell
$ cat c.txt
Line 1
Line 2
Line 3
Line 4
Line 5
cddcd
dcdcdc
dcdcdcd
cdcdcdcdc
cdcdcdcd
rgrgrgrg
rgrgr
grgr
grgrgr
grgrgrg
rgrgrg
rgrgrg
rgrg
rgrg

```

10.

Write a shell script program to show number of files in the current directory beginning with a specified character

```
#!/bin/bash
```

```
ls | grep "^$1"
```

```
echo 'Number of files starting with the string '$1' are: '
```

```
ls | grep -c "^$1"
```

```
Applications  Places  Terminal  Jun 1 07:39  kali@kali: ~/Desktop/shell

(kali@kali)-[~/Desktop/shell]
$ ./assign10.sh f
f1.txt
f2.txt
f3.txt
Number of files starting with the string f are:
./assign10.sh: line 7: unexpected EOF while looking for matching `"'
./assign10.sh: line 8: syntax error: unexpected end of file

(kali@kali)-[~/Desktop/shell]
$ nano assign10.sh

(kali@kali)-[~/Desktop/shell]
$ ./assign10.sh f
f1.txt
f2.txt
f3.txt
Number of files starting with the string f are:
./assign10.sh: line 7: unexpected EOF while looking for matching `"'
./assign10.sh: line 8: syntax error: unexpected end of file

(kali@kali)-[~/Desktop/shell]
$ nano assign10.sh

(kali@kali)-[~/Desktop/shell]
$ ./assign10.sh f
f1.txt
f2.txt
f3.txt
Number of files starting with the string f are:
3

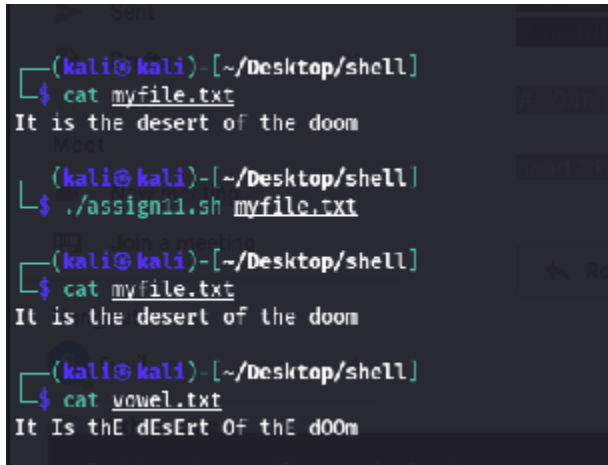
(kali@kali)-[~/Desktop/shell]
$
```


11.

```
# Write a shell script program to read a line from a file and store them into  
# another file after converting all the vowels from first file into uppercase.
```

```
#!/bin/bash
```

```
head "$1" -n 1 | tr 'aeiou' 'AEIOU' > vowel.txt
```



```
(kali@kali)-[~/Desktop/shell]
└─$ cat myfile.txt
It is the desert of the doom

(kali@kali)-[~/Desktop/shell]
└─$ ./assign11.sh myfile.txt

(kali@kali)-[~/Desktop/shell]
└─$ cat myfile.txt
It is the desert of the doom

(kali@kali)-[~/Desktop/shell]
└─$ cat vowel.txt
It Is thE dEsErT Of thE dOom
```

12.

```
# Write a shell script program that accept the name of the user and prints the  
# entire name in reverse and also print the length of the entire name
```

```
#!/bin/bash
```

```
read -p "Enter your name: " name
```

```
len=${#name}
```

```
reverse=""
```

```
for ((i = $len - 1; i >= 0; i--))
```

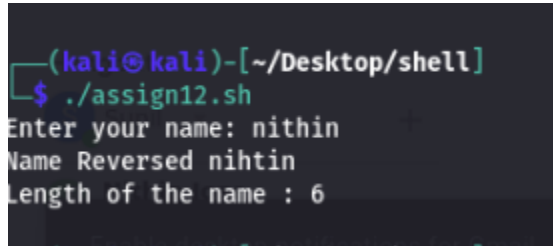
```
do
```

```
reverse="$reverse${name:i:1}"
```

done

```
echo "Name Reversed $reverse"
```

```
echo "Length of the name : ${len}"
```



```
(kali@kali)~[~/Desktop/shell]
$ ./assign12.sh
Enter your name: nithin
Name Reversed nihtin
Length of the name : 6
```


13. Consider a file school.dat with the following fields: roll no, name and marks. Write a shell script program to sort the file in descending order of marks

```
#!/bin/bash
```

```
file="students.txt"
```

```
head -n 1 $file
```

```
tail -n +2 $file | sort -s -r -nk3
```



```
(kali@kali)~[~/Desktop/shell]
$ ./assign13.sh
roll no      name      marks
4           Rjan      43
6           Babu      39
1           Anu       34
10          George    29
7           Lesli     12
11          Madhavan   3
```

14. Write a shell script program to display sum of first n numbers

```
#!/bin/bash
```

```
read -p "Enter a value for n: " n
```

```
sum=0
```

```
for (( i=1 ; i <= n ; i++ ))
```

```
do
```

```
    sum=$((sum + i))
```

```
done
```

```
echo "Sum: $sum"
```

```
—(kali@kali)~[~/Desktop/shell]
—$ ./assign14.sh
Enter a value for n: 10
Sum: 55
```

15.

Menu driven program to display the day when inputting 1 to 7

```
#!/bin/bash
```

```
while :
```

```
do
```

```
    read -p "Enter a number in between 1 and 7 (including them) I will return you the  
day(any other to quit): " day
```

```
    case $day in
```

```
        1) echo "Monday";;
```

```
        2) echo "Tuesday";;
```

```
        3) echo "Wednesday";;
```

```
        4) echo "Thursday";;
```

```
        5) echo "Friday";;
```

```
        6) echo "Saturday";;
```

```
        7) echo "Sunday";;
```

```
        *) echo "Quitting..."
```

```
    exit;;
```

```
    esac
```

```
done
```

```
—(kali@kali)~[~/Desktop/shell]
—$ ./assign15.sh
Enter a number in between 1 and 7 (including them) I will return you the day(any other to quit): 5
Friday
Enter a number in between 1 and 7 (including them) I will return you the day(any other to quit): 6
Saturday
Enter a number in between 1 and 7 (including them) I will return you the day(any other to quit): 3
Wednesday
Enter a number in between 1 and 7 (including them) I will return you the day(any other to quit):
```
