

1. Write a shell script to display your LOGIN NAME and HOME directory.

```
(kali㉿kali)-[~/LabQ09Dec]
$ logname
kali

(kali㉿kali)-[~/LabQ09Dec]
$ pwd
/home/kali/LabQ09Dec
```

2. Write a shell script to display menu like “1. Date, 2. Cal, 3. Ls, 4. Pwd, 5. Exit” and execute the commands depending on user choice.

```
(kali㉿kali)-[~]
$ mkdir LabQ09Dec

(kali㉿kali)-[~]
$ cd LabQ09Dec

(kali㉿kali)-[~/LabQ09Dec]
$ nano q1.sh

(kali㉿kali)-[~/LabQ09Dec]
$ chmod a+x q1.sh

(kali㉿kali)-[~/LabQ09Dec]
$ ./q1.sh
Fri Dec 9 05:49:09 AM EST 2022
December 2022
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 31

q1.sh
/home/kali/LabQ09Dec

(kali㉿kali)-[~/LabQ09Dec]
$ cat q1.sh
date
cal
ls
pwd
exit
```

3. Write a shell script to accept the name from the user and check whether user entered name is file or directory. If name is file display its size and if it is directory display its contents.

4. Write a shell script to determine whether a given number is prime or not

```
GNU nano 6.4 q3.sh
#!/bin/bash
echo -e "Enter Number : \c"
read n
for((i=2; i<=$n/2; i++))
do
    ans=$(( n%i ))
    if [ $ans -eq 0 ]
    then
        echo "$n is not a prime number."
        exit 0
    fi
done
echo "$n is a prime number."
```

```
(kali@kali)-[~/LabQ09Dec]
$ ./q3.sh
Enter Number : 7
7 is a prime number.
```

5. Write a program to find the greatest of three numbers

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q4.sh
echo "Enter num 1"
read num1
echo "Enter num 2"
read num2
echo "Enter num 3"
read num3

if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]
then
    echo $num1
elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]
then
    echo $num2
else
    echo $num3
fi
```

```
(kali@kali)-[~/LabQ09Dec]
$ ./q4.sh
Enter num 1
40
Enter num 2
80
Enter num 3
50
80
```

6. Write a program to find whether a given year is a leap year or not

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q6.sh
#!/bin/bash
echo "Program to check for leap year or not"
read -p "Enter the year: " year
if [ $((year%400)) -eq 0 ]
then
    echo "$year is a leap year"
elif [ $((year%100)) -eq 0 ]
then
    echo "$year is not a leap year"
elif [ $((year%4)) -eq 0 ]
then
    echo "$year is a leap year"
else
    echo "$year is not a leap year"
fi
```

```
(kali@kali)-[~/LabQ09Dec]
$ ./q6.sh
Program to check for leap year or not
Enter the year: 2022
2022 is not a leap year
```

7. Write a program to find out the area of a circle

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q7.sh
echo "Enter the radius : "
read r
area=$(echo "3.14*$r*$r" | bc )
echo "area of the circle is " $area
```

```
(kali㉿kali)-[~/LabQ09Dec]
$ nano q7.sh

(kali㉿kali)-[~/LabQ09Dec]
$ ./q7.sh
Enter the radius :
50
area of the circle is          7850.00
```

8. Write a program to find out the area of a rectangle

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q8.sh *
#!/bin/sh
echo "Enter the width and height of rectangle:"
read width
read height
echo "Area of the rectangle is:"
expr $width \* $height
```

```
(kali㉿kali)-[~/LabQ09Dec]
$ ./q8.sh
Enter the width and height of rectangle:
30
80
Area of the rectangle is:
2400
```

9. Write a program to find whether a given number is positive or negative

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q9.sh
echo "Enter a Number"
read num

if [ $num -lt 0 ]
then
    echo "Negative"
elif [ $num -gt 0 ]
then
    echo "Positive"
else
    echo "Neither Positive Nor Negative"
fi
```

```
(kali㉿kali)-[~/LabQ09Dec]
$ ./q9.sh
Enter a Number
-8
Negative

(kali㉿kali)-[~/LabQ09Dec]
$ ./q9.sh
Enter a Number
5
Positive
```

10. Write a program to print the table of a given number

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q10.sh *
#!/bin/bash
echo "Enter the number -"
read n

i=1
while [ $i -le 10 ]
do
res=`expr $i \* $n`

echo "$n * $i = $res"
((++i))
done
```

```
(kali@kali)-[~/LabQ09Dec]
$ ./q10.sh
Enter the number -
6
6 * 1 = 6
6 * 2 = 12
6 * 3 = 18
6 * 4 = 24
6 * 5 = 30
6 * 6 = 36
6 * 7 = 42
6 * 8 = 48
6 * 9 = 54
6 * 10 = 60
```

11. Write a program to find the factorial of given number.

```
kali@kali: ~/LabQ09Dec
GNU nano 6.4 q11.sh *
#!/bin/bash
read -p "Enter a number" num
fact=1
for((i=2;i<=num;i++))
{
fact=$((fact*i))
}
echo $fact
```

```
(kali@kali)-[~/LabQ09Dec]
$ ./q11.sh
Enter a number5
120
```

12. Write a program to find given number of terms in the Fibonacci series

```
File Actions Edit View Help
GNU nano 6.4 q12.sh
echo "How many number of terms to be generated ?"
read n
function fib
{
  x=0
  y=1
  i=2
  echo "Fibonacci Series up to $n terms : "
  echo "$x"
  echo "$y"
  while [ $i -lt $n ]
  do
    i=`expr $i + 1 `
    z=`expr $x + $y `
    echo "$z"
    x=$y
    y=$z
  done
}
r=`fib $n`
echo "$r"
```

```
File Actions Edit View Help
(kali@kali) ~/LabQ09Dec
$ ./q12.sh
zsh: ./q12.sh: bad interpreter: bin/bash: no such file or directory
(kali@kali) ~/LabQ09Dec
$ nano q12.sh
(kali@kali) ~/LabQ09Dec
$ ./q12.sh
How many number of terms to be generated ?
10
./q12.sh: 3: function: not found
Fibonacci Series up to 10 terms :
0
1
1
2
3
5
8
13
21
34
```

13. Write a program to calculate gross salary if the DA is 40%, HRA is 20% of basic salary. Accept basic salary from user and display gross salary (Result can be floating point value).

```
File Actions Edit View Help
GNU nano 6.4 q13.sh
#!/bin/bash
echo "enter the basic salary:"
read basal
grossal=$(( echo "$basal+((40/100)*$basal)+((20/100)*$basal)" | bc -l ))
echo "The gross salary : $grossal"
```

```
(kali@kali) ~/LabQ09Dec
$ ./q13.sh
enter the basic salary:
15000
The gross salary : 24000.000000000000000000000000
```

14. Write a shell script to accept a filename as argument and displays the last modification time if the file exists and a suitable message if it doesn't exist.

```
File Actions Edit View Help
GNU nano 6.4 q14.sh
echo -n "Enter a filename to see last modification time : "
read fileName
# make sure file exists
if [ ! -f $fileName ]
then
    echo "$fileName not a file"
    exit 1
fi
# use stat command to display
echo "$fileName was last modified on $(stat -c %x $fileName)"
```

```
(kali㉿kali)-[~/LabQ09Dec]
$ ./q14.sh
Enter a filename to see last modification time : q1.sh
q1.sh was last modified on 2022-12-09 05:49:09.280642776 -0500
```

15. Write a shell script to display only hidden file of current directory.

```
File Actions Edit View Help
GNU nano 6.4 q15.sh *
#!/bin/bash
echo "This is to HIDE any file."
echo "Enter the name of the file in the current directory:"
read file
# a (.) or period sign is used to hide any file in linux
mv $file .$file
echo ".....The has been hide successfully....."
```

16. Write a shell script to display only executable files of current directory.

```
GNU nano 6.4 q16.sh
for filess in *
do
    if [ -d $filess ]
    then
        echo $filess
    fi
done
```

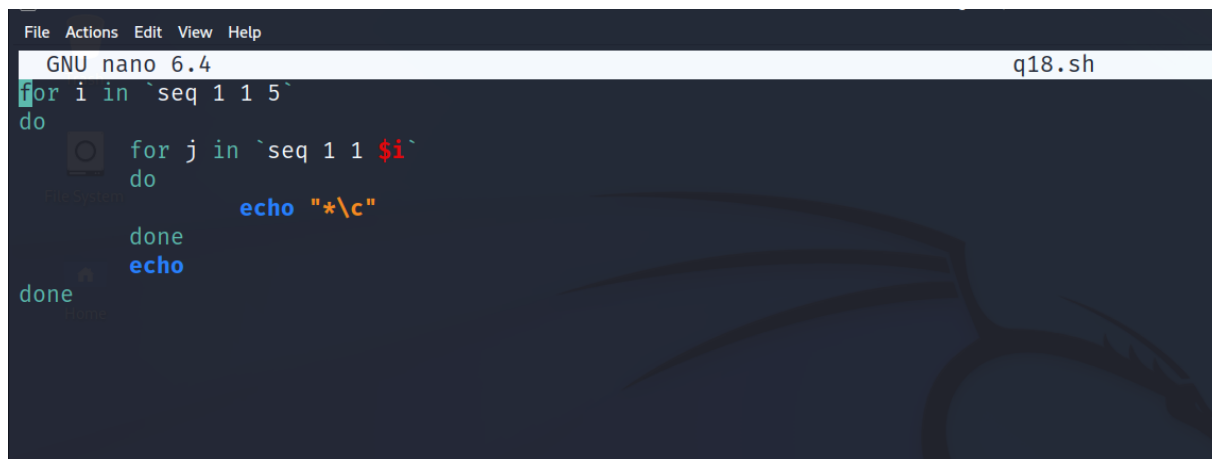
```
(kali㉿kali)-[~/LabQ09Dec]
$ ./q16.sh
LabQ10Dec
```

17. Accept the two file names from user and append the contents in reverse case of first file into second file

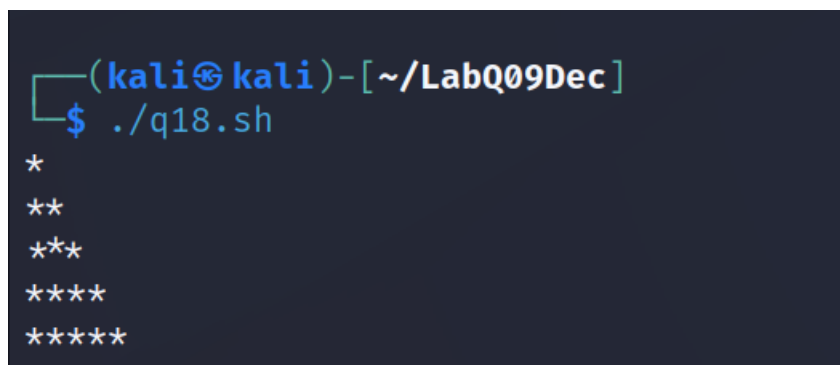
18. Print the following pattern.

*

**



```
File Actions Edit View Help
GNU nano 6.4 q18.sh
for i in `seq 1 1 5`
do
    for j in `seq 1 1 $i`
    do
        echo "*\c"
    done
done
echo
```



```
(kali@kali)-[~/LabQ09Dec]
$ ./q18.sh
*
**
***
****
*****
```