Program :

```
echo Please enter the no. :
read a
b=`expr $a % 2`
if [ b -eq 0 ]
then
echo $a is even.
else
echo $a is odd.
fi
```

Output :

```
"pr_2.sh" 10L, 114B written
[root@localhost d2]# sh pr_2.sh
Please enter the no. :
3
3 is odd.
[root@localhost d2]#
```

Q. write a shell script program to print the table of the given number.

Program :

```
echo Please enter the no. to print table :
read a
i=1
echo Table of $a is :
while [ $i -le 10 ]
do
t=`expr $i \* $a`
echo $a X $i = $t
i=`expr $i + 1`
done
```

Q. write a shell script program for addition of two numbers.

Program :

```
echo Please enter the first no. :
read a
echo Please enter the second no. :
read b
sum=`expr $a + $b`
echo Sum is $sum
~
```

Output :

```
~
"pr_2.sh" 7L, 121B written
[root@localhost d2]# sh pr_2.sh
Please enter the first no. :
10
Please enter the second no. :
20
Sum is 30
[root@localhost d2]#
```

expr : basic arithmetic operation is used for ...

```
~
"pr_2.sh" 11L, 157B written
[root@localhost d2]# sh pr_2.sh
please enter the no. to print table :
5
Table of 5 is :
5 X 1 = 5
5 X 2 = 10
5 X 3 = 15
5 X 4 = 20
5 X 5 = 25
5 X 6 = 30
5 X 7 = 35
5 X 8 = 40
5 X 9 = 45
5 X 10 = 50
[root@localhost d2]#
```

PRACTICAL 4

AIM:-Write a shell script to validate the entered date. (eg. Date format is dd-mm-yyyy).

 # Input: -

```
#!/bin/bash
d=`date +%m-%d-%Y`
echo $d #DD-MM-YYYY
echo " Please Enter Date "
read D
echo " Please Enter Month "
read M
echo " Please Enter Year "
read Y
if [ `expr $Y % 4` -eq 0 ]
 then
Mayur Pandya Division: - 4B18
echo "$Y is a leap year"
else
echo "$Y is not a leap year"
fi
```

#Output:-

PRATICAL – 6

AIM: write a shell script to say good morning/good afternoon/good evening as you login to system.

Code:

```
a=$(date +%H)
if [ $a -lt 12 ];then
b="good morning"
elif [ $a -lt 18 ];then
b="good afternoon"
else
b="good evening"
fi
echo $b
```

```
a=$(date +%H)
if [ $a -lt 12 ];then
b="good morning"
elif [ $a -lt 18 ];then
b="good afternoon"
else
b="good evening"
fi
echo $b
```

```
~
"time.sh" [New] 9L, 129B written
[root@localhost ~]# sh time.sh
good evening
[root@localhost ~]#
```

PRACTIAL-9

AIM:printing the patterns using for loop.

Code:

```
rows=5
for ((i=1; i<=rows; i++))
do
 for ((j=1; j<=i; j++))
 do
  echo -n "* "
 done
 echo
Done
```

```
rows=5
for ((i=1; i<rows; i++))
do
  for ((j=1; j<=i; j++))
  do
echo -n "*"
 done
   echo
done
```

```
~
"a.sh" 10L, 97B written
[root@localhost ~]# sh a.sh
*
**
***
****
[root@localhost ~]#
```

2303031050656

Q. write a shell script program to check which number is grater from the given two numbers.

Program :

```
echo Please enter the first no. :
read a
echo Please enter the second no. :
read b
if [ $a -gt $b ]
then
echo a is grater.
else
echo b is grater.
fi
```

Output :

```
"pr_2.sh" 11L, 150B written
[root@localhost d2]# sh pr_2.sh
Please enter the first no. :
15
Please enter the second no. :
10
a is grater.
[root@localhost d2]#
```

Q. write a shell script program to check whether number is even or odd.

PRACTIAL-8

AIM:finding out biggest number from given three numbers supplied as command line arguments.

Code:

```
echo "Enter three numbers: "
read num1 num2 num3
if [ $num1 -ge $num2 ] && [ $num1 -ge $num3 ]; then
  largest=$num1
elif [ $num2 -ge $num1 ] && [ $num2 -ge $num3 ]; then
  largest=$num2
else
  largest=$num3
fi
echo "The largest number is: $largest"
```

```
echo "Enter three numbers:"
read num1 num2 num3
if [ $num1 -ge $num2 ] && [ $num1 -ge $num3 ];then
largest=$num1
elif [ $num2 -ge $num1 ] && [ $num2 -ge $num3 ]; then
largest=$num2
else
largest=$num3
fi
echo " the largest number is:$largest"
```

```
"a.sh" 11L, 243B written
[root@localhost ~]# sh a.sh
Enter three numbers:
10 20 30
 the largest number is:30
[root@localhost ~]#
```

PRACTIAL-7

AIM:write a c program to create a child process.

Code:

```
#include<stdio.h>
#include<sys/types.h>
#include<unistd.h>
Int main()
{
Fork()
Fork()
Printf(" using fork() system call");
Return 0;
}
```

```
#include<stdio.h>
#include<sys/types.h>
#include<unistd.h>
int main()
{
    fork();
    fork();
    printf("using fork() system call");
return 0;
}
~
~
```

a

```
~
~
"abc.c" 10L, 148B written
[root@localhost ~]# gcc abc.c -o abc
[root@localhost ~]# ./abc
using fork() system call[root@localhost ~]# using fork() system callusing fork()
 system callusing fork() system call
```


2303031050656

PRATICAL-10

AIM: shell script to determine whether given file exist or not.

Code:

```
Echo plz enter file name

Read a

If [ -f $a ]

Then

Echo file exists

Else

Touch $a

Fi
```

```
echo plz enter file name
read a
if [ -f $a ]
then
echo file exists
else
touch $a
fi
```

```
"t1.txt" 8L, 84B written
[root@localhost ~]# sh t1.txt
plz enter file name
t1.txt
file exists
[root@localhost ~]#
```

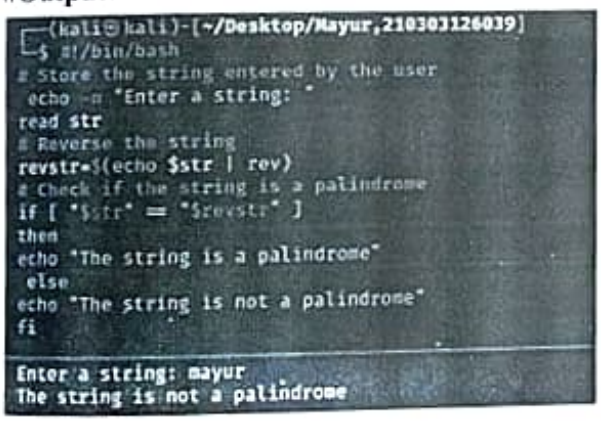
PRACTICAL 5

AIM:- Write a shell script to check entered string is palindrome or not

 # Input: -

```
#!/bin/bash
# Store the string entered by the user
 echo -n "Enter a string: "
read str
# Reverse the string
revstr=$(echo $str | rev)
# Check if the string is a palindrome
if [ "$str" == "$revstr" ]
then
echo "The string is a palindrome"
 else
echo "The string is not a palindrome"
fi
```

#Output:-



PRACTICAL 3

AIM:-Write a Shell script to print the given numbers sum of all digits.

 # Input: -

```
#!/bin/bash
echo "Enter a number"
 read num
sum=0
while [ $num -gt 0 ]
do
mod=$((num % 10))
sum=$((sum + mod))
num=$((num / 10))
done
echo $sum
```

#Output:-