Ex. No: 4 **Date: 11-09-2023**

SHELL PROGRAMMING

Aim:

To learn the Shell Programming with following examples in Linux.

Shell Programming:

A shell is a command-line interpreter and typical operations performed by shell scripts include file manipulation, program execution, and printing text.

1. To find whether the number is odd or even:

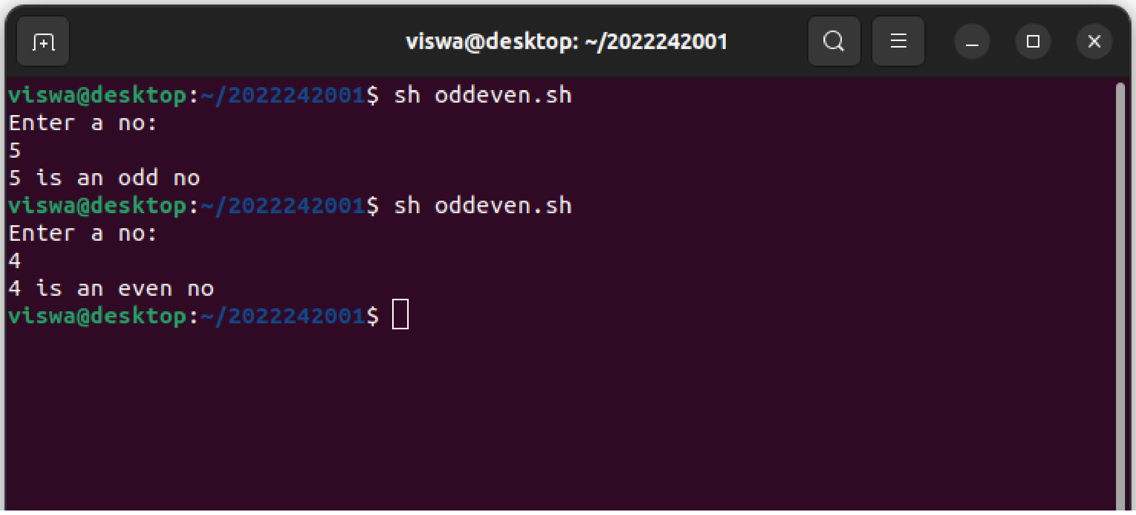
Aim:

To write shell program to check whether a number is either odd or even.

Program:

```
echo "Enter a no: "
read num
rem=$(($num%2))
if [ $rem -eq 0 ]
then
 echo "$num is an even no"
else
 echo "$num is an odd no"
fi
```

Output:



```
viswa@desktop:~/2022242001$ sh oddeven.sh
Enter a no:
5
5 is an odd no
viswa@desktop:~/2022242001$ sh oddeven.sh
Enter a no:
4
4 is an even no
viswa@desktop:~/2022242001$ 
```

Result:

Thus, the shell program is executed and outputs are verified.

2. To generate the Fibonacci series:

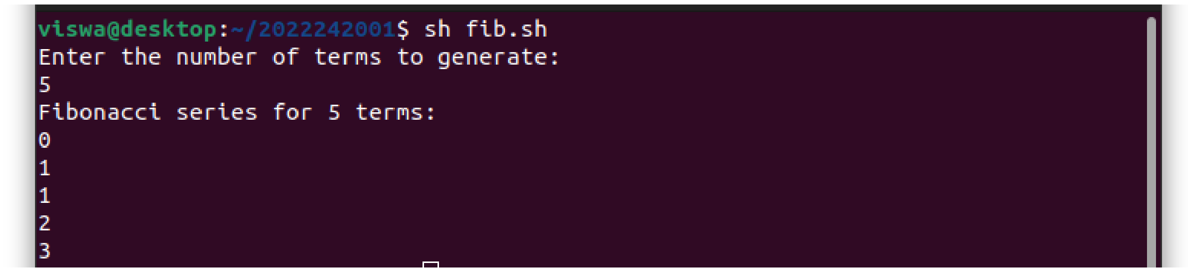
Aim:

To write shell program to generate the Fibonacci series.

Program:

```
echo "Enter the number of terms to generate: "
read term
a=0
b=1
x=2
echo "Fibonacci series for $term terms: "
echo "$a"
echo "$b"
while [ $x -lt $term ]
do
x=$(($x+1))
c=$(($a+$b))
echo "$c"
a=$b
b=$c
done
```

Output:



```
viswa@desktop:~/2022242001$ sh fib.sh
Enter the number of terms to generate:
5
Fibonacci series for 5 terms:
0
1
1
2
3
```

Result:

Thus, the shell program is executed and outputs are verified.

3. To find factorial of a number:

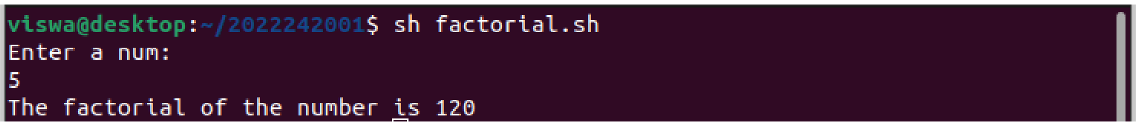
Aim:

To write shell program to find the factorial of a number given

Program:

```
echo "Enter a num: "
read num
fact=1
while [ $num -gt 1 ]
do
fact=$((fact*num))
num=$((num-1))
done
echo "The factorial of the number is $fact"
```

Output:



```
viswa@desktop:~/2022242001$ sh factorial.sh
Enter a num:
5
The factorial of the number is 120
```

Result:

Thus, the shell program is executed and outputs are verified.

4. To find a string whether it is palindrome or not:

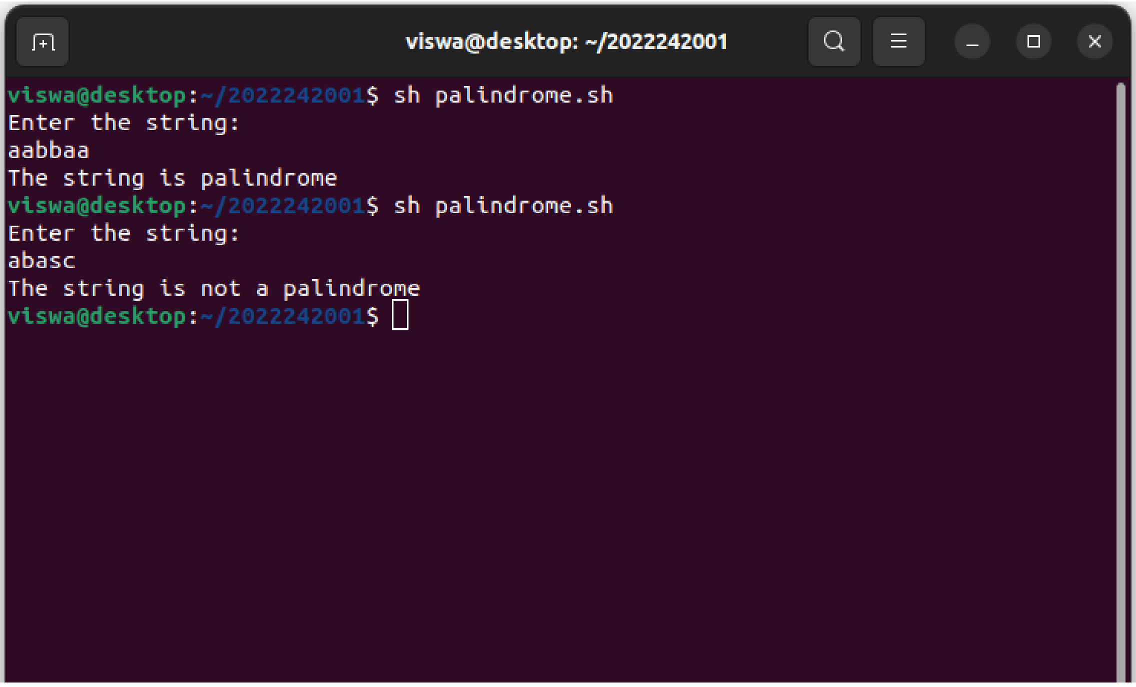
Aim:

To write shell program to find a string is palindrome or not.

Program:

```
echo "Enter the string: "
read str
echo $str>temp
reverse="$(rev temp)"
if [ $str = $reverse ]
then
echo "The string is palindrome"
else
echo "The string is not a palindrome"
fi
```

Output:



```
viswa@desktop:~/2022242001$ sh palindrome.sh
Enter the string:
aabbaa
The string is palindrome
viswa@desktop:~/2022242001$ sh palindrome.sh
Enter the string:
abasc
The string is not a palindrome
viswa@desktop:~/2022242001$
```

Result:

Thus, the shell program is executed and outputs are verified.

5. To display a greeting message using time (using if):

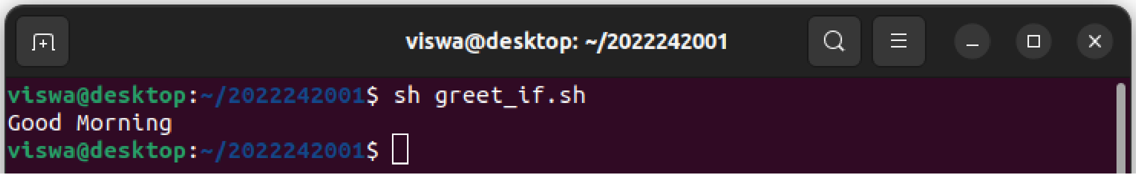
Aim:

To write shell program to display a greeting message based on time.

Program:

```
hour="$(date +%H)"
if [ $hour -ge 0 -a $hour -lt 12 ]
then
echo "Good Morning"
elif [ $hour -ge 12 -a $hour -lt 18 ]
then
echo "Good Afternoon"
elif [ $hour -ge 18 -a $hour -lt 20 ]
then
echo "Good Evening"
else
echo "Good Night"
fi
```

Output:



```
viswa@desktop:~/2022242001$ sh greet_if.sh
Good Morning
viswa@desktop:~/2022242001$ 
```

Result:

Thus, the shell program is executed and outputs are verified.

6. To display a greeting message using time (using case):

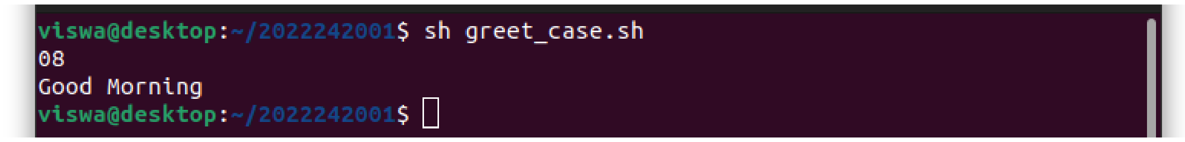
Aim:

To write shell program to display a greeting message based on time.

Program:

```
hour="$(date +%H)"
echo $hour
case $hour in
0[1-9]|1[0-2])
echo "Good Morning"
;;
1[3-6])
echo "Good Afternoon"
;;
1[7-9])
echo "Good Evening"
;;
*)
echo "Good Night"
esac
```

Output:



```
viswa@desktop:~/2022242001$ sh greet_case.sh
08
Good Morning
viswa@desktop:~/2022242001$ 
```

Result:

Thus, the shell program is executed and outputs are verified.

7. To add the digits of the given number:

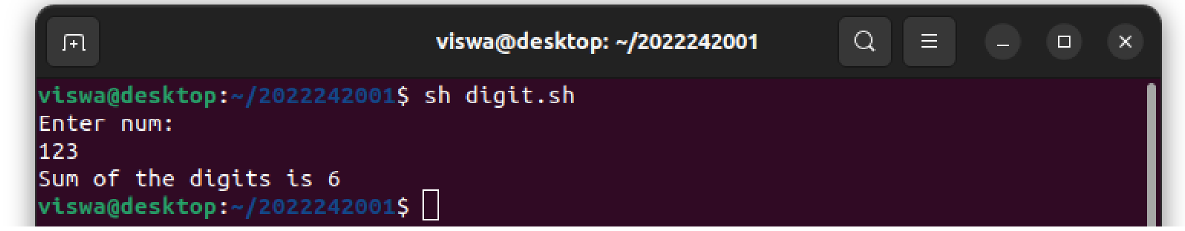
Aim:

To write shell program to add the digits of the given number.

Program:

```
echo "Enter num: "
read num
sum=0
while [ $num -gt 0 ]
do
rem=$(($num%10))
sum=$(($sum+rem))
num=$((num/10))
done
echo "Sum of the digits is $sum"
```

Output:



```
viswa@desktop:~/2022242001$ sh digit.sh
Enter num:
123
Sum of the digits is 6
viswa@desktop:~/2022242001$ 
```

Result:

Thus, the shell program is executed and outputs are verified.

Results:

Thus, the Shell Programming were learnt.