



---

# OPRATING SYSTEM

---

Assignment No : 2



## Shell Script

Name: Onkar Anil Waghmode

Div: SY-CS-D

Roll No: 72

PRN: 12210334

# shellProgramming

1. Perform **Arithmetic operations** using shell script

Code:

```
#!/bin/bash

addition() {
    result=$((num1 + num2))
    echo "Result: $result"
}

subtraction() {
    result=$((num1 - num2))
    echo "Result: $result"
}

multiplication() {
    result=$((num1 * num2))
    echo "Result: $result"
}

division() {
    if [ $num2 -eq 0 ]; then
        echo "Error: Cannot divide by zero!"
    else
        result=$((num1 / num2 | bc))
        echo "Result: $result"
    fi
}

echo "Enter first number: "
read num1
echo "Enter second number: "
read num2

echo "Select an operation:"

read choice
```



# shellProgramming

```
case $choice in
    +) addition ;;
    -) subtraction ;;
    *) multiplication ;;
    /) division ;;
    *) echo "Invalid oprator. Please try again." ;;
esac
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./arithmaticoprations.sh
Enter first number:
24
Enter second number:
12
Select an operation:
+
Result: 36
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./arithmaticoprations.sh
Enter first number:
24
Enter second number:
12
Select an operation:
-
Result: 12
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./arithmaticoprations.sh
Enter first number:
24
Enter second number:
12
Select an operation:
*
Result: 288
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./arithmaticoprations.sh
Enter first number:
24
Enter second number:
12
Select an operation:
/
Result: 2
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./arithmaticoprations.sh
Enter first number:
24
Enter second number:
12
Select an operation:
%
Invalid oprator. Please try again.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

# shellProgramming

2. Write shell program to print 1 to 5 number using simple **FOR LOOP**

Code:

```
#!/bin/bash

echo "for loop print number 1 to 5"

for n in 1 2 3 4 5
do
    echo $n
done
```

Output :

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./forloop.sh
for loop print number 1 to 5
1
2
3
4
5
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

3. Write shell program to print table of given number using **CTYPE FOR LOOP**

Code:

```
#!/bin/bash

echo "Enter a number to display its multiplication table: "
read number

echo "Multiplication table of $number:"
for ((i = 1; i <= 10; i++)); do
    result=$((number * i))
    echo "$number x $i = $result"
done
```



# shellProgramming

Output :

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./ctypeloop.sh
Enter a number to display its multiplication table:
5
Multiplication table of 5:
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
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

4. Write shell script for square of number 1-5 using **RANGEBASED LOOP**

Code:

```
#!/bin/bash

echo "square of number 1-5 using rangebased loop"

for n in {1..5}
do
    sq=$((n * n))
    echo "$n ==> $sq"
done
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./rangebasedloop.sh
square of number 1-5 using rangebased loop
1 ==> 1
2 ==> 4
3 ==> 9
4 ==> 16
5 ==> 25
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

# shellProgramming

5. Write shell script for calculating sum of numbers till given range using **UNTIL LOOP**

code

```
#!/bin/bash

echo "Enter the upper limit: "
read limit

calculate_sum() {
    upper_limit=$1
    sum=0
    number=1

    until [ $number -gt $upper_limit ]; do
        sum=$((sum + number))
        ((number++))
    done

    echo "Sum of numbers from 1 to $upper_limit is: $sum"
}

calculate_sum "$limit"
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./untillloop.sh
Enter the upper limit:
10
Sum of numbers from 1 to 10 is: 55
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```



# shellProgramming

6. Write shell script for Fibonacci series using **WHILE LOOP**

code

```
#!/bin/bash

echo "enter number for fibonnci terms : "
read n

generate_fibonacci() {
    num_terms=$1

    a=0
    b=1
    counter=1

    while [ $counter -le $num_terms ]; do
        echo -n "$a "

        next=$((a + b))
        a=$b
        b=$next

        ((counter++))
    done
    echo
}

generate_fibonacci "$n"
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./whileloop.sh
enter number for fibonnci terms :
10
0 1 1 2 3 5 8 13 21 34
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

# shellProgramming

7. Write shell script for printing "shellProgramming" string till the number given by user using  
**Infinite loop**

## code

```
#!/bin/bash

echo "Infinite loop"
echo "enter number for break the loop"
read number
n=1;
for (( ; ; ));
do
if [ $n -eq $number ];then
    break;
fi
    echo "shell Programming"
    ((n=n+1))
done
```

## output :

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./infinitemloop.sh
Infinite loop
enter number for break the loop
5
shell Programming
shell Programming
shell Programming
shell Programming
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```



# shellProgramming

8. Write shell program to print given number is even or odd using **if-else control statement**

Code :

```
#!/bin/bash

echo "Enter a number: "
read number

if [  $$(number \% 2)$  -eq 0 ]; then
    echo "$number is an even number."
else
    echo "$number is an odd number."
fi
```

Ouput :

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./controlif.sh
Enter a number:
24
24 is an even number.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./controlif.sh
Enter a number:
23
23 is an odd number.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

# shellProgramming

9. Write shell program to print given character is Capital-Alphabet or Small-Alphabet or Number or Symbol using **control case (switch case)**

## code

```
#!/bin/bash

check_character_type() {
    case $1 in
        [a-z])
            echo "Small Alphabet"
            ;;
        [A-Z])
            echo "Capital Alphabet"
            ;;
        [0-9])
            echo "Number"
            ;;
        *)
            echo "Symbol"
            ;;
    esac
}

echo "Enter a character: "
read input

check_character_type "$input"
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./controlcase.sh
Enter a character:
A
Capital Alphabet
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./controlcase.sh
Enter a character:
b
Small Alphabet
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./controlcase.sh
Enter a character:
5
Number
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./controlcase.sh
Enter a character:
*
Symbol
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```



# shellProgramming

10. Write shell program to print given number is Prime or Not using **Command Line argument**

**code**

```
#!/bin/bash

is_prime() {
    num=$1

    if [ $num -lt 2 ]; then
        return 1
    fi

    for ((i = 2; i * i <= num; i++)); do
        if [ $((num % i)) -eq 0 ]; then
            return 1
        fi
    done

    return 0
}

is_prime "$1"

if [ $? -eq 0 ]; then
    echo "$1 is a prime number."
else
    echo "$1 is not a prime number."
fi
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./commandline.sh 10
10 is not a prime number.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./commandline.sh 17
17 is a prime number.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./commandline.sh 25
25 is not a prime number.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

# shellProgramming

11. Write shell program to print the sum of elements of array use **Array**, input array and sum it elements

## code

```
#!/bin/bash

array_sum() {
    local sum=0
    for element in "${array[@]}; do
        sum=$((sum + element))
    done
    echo $sum
}

echo "Enter the number of elements in the array: "
read n

echo "Enter the array elements: "
for ((i = 0; i < n; i++)); do
    read num
    array[i]=$num
done

result=$(array_sum)

echo "Sum of elements in the array: $result"
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./arraysum.sh
Enter the number of elements in the array:
5
Enter the array elements:
12
21
41
32
44
Sum of elements in the array: 150
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```



# shellProgramming

12. Write shell program to print the **Length of given String**

code

```
#!/bin/bash

echo "Enter a string: "
read str

len=${#str}

echo "Length of the string: $len"
```

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringlen.sh
Enter a string:
vitpune
Length of the string: 7
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringlen.sh
Enter a string:
solapur
Length of the string: 7
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringlen.sh
Enter a string:
vit
Length of the string: 3
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

13. Write shell program to print the **concatenated string** given two separate strings

code

```
#!/bin/bash

echo "Enter the first string: "
read string1

echo "Enter the second string: "
read string2

string1+=${string2}

echo "Concatenated string: $string1"
```

# shellProgramming

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringconcat.sh
Enter the first string:
vit
Enter the second string:
pune
Concatenated string: vitpune
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringconcat.sh
Enter the first string:
shell
Enter the second string:
Programming
Concatenated string: shellProgramming
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

14. Write shell program for **compare two given strings**

code

```
#!/bin/bash

echo "Enter the first string: "
read string1

echo "Enter the second string: "
read string2

if [ "$string1" = "$string2" ]; then
    echo "Both strings are equal."
else
    echo "The strings are not equal."
fi
```



# shellProgramming

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringcompare.sh
Enter the first string:
solapur
Enter the second string:
kolhapur
The strings are not equal.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringcompare.sh
Enter the first string:
pune
Enter the second string:
pune
Both strings are equal.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringcompare.sh
Enter the first string:
Pune
Enter the second string:
pune
The strings are not equal.
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```

15. Write shell program to print the **sub string** as start index and number of characters provide by user

code

```
#!/bin/bash

echo "Enter a string: "
read str

echo "Enter start index :"
read si

echo "Enter number of charecter :"
read ch

endin=$((si + ch))

echo "Substring from index $si to $endin: ${str:$si:$ch}"
```

# shellProgramming

```
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringslice.sh
Enter a string:
vitpune
Enter start index :
3
Enter number of charecter :
4
Substring from index 3 to 7: pune
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ ./stringslice.sh
Enter a string:
indiagate
Enter start index :
0
Enter number of charecter :
5
Substring from index 0 to 5: india
onkar2004@DESKTOP-U4B25LI:~/shellProgramming$ |
```