1. Shell Script to display the first 10 natural numbers.

Expected Output :

1 2 3 4 5 6 7 8 9 10

#!/bin/bash

# Loop from 1 to 10 and print each number

for ((i=1; i<=10; i++))

do

echo -n "$i " # Print number followed by a space

done

echo "" # Print a new line at the end

2. Shell Script to compute the sum of the first 10 natural numbers.

Expected Output :

The first 10 natural number is :

1 2 3 4 5 6 7 8 9 10

The Sum is : 55

#!/bin/bash

# Initialize variables

sum=0

numbers=""

# Loop from 1 to 10 to compute sum and create a string of numbers

for ((i=1; i<=10; i++))

do

sum=$((sum + i)) # Compute sum

numbers="$numbers $i" # Add number to the string

done

# Print the first 10 natural numbers

echo "The first 10 natural numbers are : $numbers"

# Print the sum

echo "The Sum is : $sum"

3. Shell Script to display n terms of natural numbers and their sum.

Test Data : 7

Expected Output :

The first 7 natural number is :

1 2 3 4 5 6 7

The Sum of Natural Number upto 7 terms : 28

#!/bin/bash

# Read the value of n from the user

echo "Enter the value of n:"

read n

# Initialize variables

sum=0

numbers=""

# Loop from 1 to n to compute sum and create a string of numbers

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

do

sum=$((sum + i)) # Compute sum

numbers="$numbers $i" # Add number to the string

done

# Print the first n natural numbers

echo "The first $n natural numbers are : $numbers"

# Print the sum

echo "The Sum of Natural Numbers up to $n terms : $sum"

4. Shell Script to read 10 numbers from the keyboard and find their sum and average.

Test Data :

Input the 10 numbers :

Number-1 :2

...

Number-10 :2

Expected Output :

The sum of 10 no is : 55

The Average is : 5.500000

#!/bin/bash

# Initialize variables

sum=0

# Read 10 numbers from the user

echo "Input the 10 numbers :"

for ((i=1; i<=10; i++))

do

echo -n "Number-$i : "

read num

sum=$((sum + num)) # Compute sum

done

# Calculate average

average=$(echo "scale=6; $sum / 10" | bc)

# Print the sum and average

echo "The sum of 10 numbers is : $sum"

echo "The Average is : $average"

5. Shell Script to display the cube of the number up to an integer.

Test Data :

Input number of terms : 5

Expected Output :

Number is : 1 and cube of the 1 is :1

Number is : 2 and cube of the 2 is :8

Number is : 3 and cube of the 3 is :27

Number is : 4 and cube of the 4 is :64

Number is : 5 and cube of the 5 is :125

#!/bin/bash

# Read the number of terms from the user

echo -n "Input number of terms: "

read n

# Loop through numbers from 1 to n and calculate their cubes

echo "Expected Output:"

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

do

cube=$((i \* i \* i))

echo "Number is : $i and cube of $i is : $cube"

done

6. Shell Script to display the multiplication table for a given integer.

Test Data :

Input the number (Table to be calculated) : 15

Expected Output :

15 X 1 = 15

...

...

15 X 10 = 150

#!/bin/bash

# Read the number for which the multiplication table is to be calculated

echo -n "Input the number (Table to be calculated) : "

read num

# Display the multiplication table

echo "Expected Output :"

for ((i=1; i<=10; i++))

do

# Calculate the product

product=$((num \* i))

# Display the multiplication expression

echo "$num X $i = $product"

done

7. Shell Script to display the multiplier table vertically from 1 to n.

Test Data :

Input upto the table number starting from 1 : 8

Expected Output :

Multiplication table from 1 to 8

1x1 = 1, 2x1 = 2, 3x1 = 3, 4x1 = 4, 5x1 = 5, 6x1 = 6, 7x1 = 7, 8x1 = 8

...

1x10 = 10, 2x10 = 20, 3x10 = 30, 4x10 = 40, 5x10 = 50, 6x10 = 60, 7x10 = 70, 8x10 = 80

#!/bin/bash

# Read the number up to which the multiplication table is to be displayed

echo -n "Input upto the table number starting from 1 : "

read n

# Display the multiplication table

echo "Expected Output : Multiplication table from 1 to $n"

for ((i=1; i<=10; i++))

do

for ((j=1; j<=$n; j++))

do

# Calculate the product

product=$((j \* i))

# Display the multiplication expression

echo -n "$j x $i = $product"

# Add comma and space if not the last number in the row

if [ $j -ne $n ]; then

echo -n ", "

else

echo

fi

done

done

8. Shell Script to display the n terms of odd natural numbers and their sum.

Test Data

Input number of terms : 10

Expected Output :

The odd numbers are :1 3 5 7 9 11 13 15 17 19

The Sum of odd Natural Number upto 10 terms : 100

#!/bin/bash

# Read the number of terms

echo -n "Input number of terms : "

read n

# Initialize variables

sum=0

odd\_numbers=""

# Generate odd numbers and calculate their sum

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

do

# Calculate the odd number

odd=$((2 \* $i - 1))

# Add the odd number to the list

odd\_numbers="$odd\_numbers $odd"

# Add the odd number to the sum

sum=$((sum + odd))

done

# Display the odd numbers and their sum

echo "The odd numbers are :$odd\_numbers"

echo "The Sum of odd Natural Number upto $n terms : $sum"

9. Shell Script to display a pattern like a right angle triangle using an asterisk.

The pattern like :

\*

\*\*

\*\*\*

\*\*\*\*

#!/bin/bash

# Read the number of rows

echo -n "Enter the number of rows: "

read rows

# Loop through each row

for ((i=1; i<=rows; i++))

do

# Loop to print '\*' in each row

for ((j=1; j<=i; j++))

do

echo -n "\* "

done

# Move to the next line

echo ""

done

10. Shell Script to display a pattern like a right angle triangle with a number.

The pattern like :

1

12

123

1234

#!/bin/bash

# Read the number of rows

echo -n "Enter the number of rows: "

read rows

# Loop through each row

for ((i=1; i<=rows; i++))

do

# Loop to print numbers in each row

for ((j=1; j<=i; j++))

do

echo -n "$j "

done

# Move to the next line

echo ""

done

11. Shell Script to make such a pattern like a right angle triangle with a number which will repeat a number in a row.

The pattern like :

1

22

333

4444

#!/bin/bash

# Read the number of rows

echo -n "Enter the number of rows: "

read rows

# Loop through each row

for ((i=1; i<=rows; i++))

do

# Loop to print numbers in each row

for ((j=1; j<=i; j++))

do

echo -n "$i "

done

# Move to the next line

echo ""

done

12. Shell Script to make such a pattern like a right angle triangle with the number increased by 1.

The pattern like :

1

2 3

4 5 6

7 8 9 10

#!/bin/bash

# Read the number of rows

echo -n "Enter the number of rows: "

read rows

# Initialize a variable to keep track of the current number

number=1

# Loop through each row

for ((i=1; i<=rows; i++))

do

# Loop to print numbers in each row

for ((j=1; j<=i; j++))

do

echo -n "$number "

# Increment the number for the next iteration

((number++))

done

# Move to the next line after each row

echo ""

done