

# Python Loops

# Python Program to Display the multiplication Table

- **Source Code**

- `# Multiplication table (from 1 to 10) in Python`
- `num = int(input("Display multiplication table of? "))`
- `# Iterate 10 times from i = 1 to 10`
- `for i in range(1, 11):`
- `print(num, 'x', i, '=', num*i)`

- **Output:**

- Display multiplication table of? 12
- 12 x 1 = 12
- 12 x 2 = 24
- 12 x 3 = 36
- 12 x 4 = 48
- 12 x 5 = 60
- 12 x 6 = 72
- 12 x 7 = 84
- 12 x 8 = 96
- 12 x 9 = 108
- 12 x 10 = 120

# Python Program to Find the Factorial of a Number

- **Source Code**

- `# Python program to find the factorial of a number provided by the user.`
- `#num = int(input("Enter a number: "))`
- `factorial = 1`
- `# check if the number is negative, positive or zero`
- `if num < 0:`
- `print("Sorry, factorial does not exist for negative numbers")`
- `elif num == 0:`
- `print("The factorial of 0 is 1")`
- `else:`
- `for i in range(1,num + 1):`
- `factorial = factorial*i`
- `print("The factorial of",num,"is",factorial)`

- **Output**

- `The factorial of 7 is 5040`

# Python Program to Print the Fibonacci sequence

- **Source Code:**

- `# Program to display the Fibonacci sequence up to n-th term`
- `nterms = int(input("How many terms? "))`
- `# first two terms`
- `n1, n2 = 0, 1`
- `count = 0`
- `# check if the number of terms is valid`
- `if nterms <= 0:`
- `print("Please enter a positive integer")`
- `elif nterms == 1:`
- `print("Fibonacci sequence upto",nterms,":")`
- `print(n1)`
- `else:`
- `print("Fibonacci sequence:")`
- `while count < nterms:`
- `print(n1)`
- `nth = n1 + n2`
- `# update values`
- `n1 = n2`
- `n2 = nth`
- `count += 1`

- **Output:**

- `How many terms? 7`
- `Fibonacci sequence:`
- `0`
- `1`
- `1`
- `2`
- `3`
- `5`
- `8`

# Python Program to Check Prime Number

- **Source Code**

- # Program to check if a number is prime or not
- num = int(input("Enter a number: "))
- # prime numbers are greater than 1
- if num > 1:
- # check for factors
- for i in range(2,num):
- if (num % i) == 0:
- print(num,"is not a prime number")
- break
- else:
- print(num,"is a prime number")
- # if input number is less than
- # or equal to 1, it is not prime
- else:
- print(num,"is not a prime number")
- **Output**
- Enter a number:407
- 407 is not a prime number

# Python Program to Add two Matrices

- `# This program is to add two given matrices. We are using the concept of nested lists to represent matrix`
- `M1 = [[1, 1, 1],`
- `[1, 1, 1],`
- `[1, 1, 1]]`
- `M2 = [[1, 2, 3],`
- `[4, 5, 6],`
- `[7, 8, 9]]`
- `# In this matrix we will store the sum of above matrices. We have initialized all the elements of this matrix as zero`
- `sum = [[0, 0, 0],`
- `[0, 0, 0],`
- `[0, 0, 0]]`
- `# iterating the matrix   rows: number of nested lists in the main list   columns: number of elements in the nested lists`
- `for i in range(len(M1)):`
- `for j in range(len(M1[0])):`
- `sum[i][j] = M1[i][j] + M2[i][j]`
- `# displaying the output matrix`
- `for num in sum:`
- `print(num)`
- **Output:**
- `[2, 3, 4]`
- `[5, 6, 7]`
- `[8, 9, 10]`

# Python continue statement

- **Example: Python continue**

- # Program to show the use of continue statement inside loops

- for val in "string":

- if val == "i":

- continue

- print(val)

- print("The end")

- **Output**

- s

- t

- r

- n

- g

- The end

**Thank You**