**Loop Statements:**

**============**

**Day-01**

**11-03-2025**

**==============**

-> Loop is called as "repeat"

also called as "iteration"

when we want to execute the certain statements in program want to execute again and again, we can use "loop statements".

-> The block which we can execute again and again called as "loop"

-> There are two types of loop statements:

 1) while loop

 2) for loop

**for loop:**

**-----------**

for:

----

 -> for is a keyword

 which we can use to define the for loop.

in:

---

 -> in is also a keyword

 can use for the membership check.

range():

--------

 -> range() is one of the pre-defined function in python

 -> we can use the range() to generate values from the specified index.

 Syntax:

 range(start, stop)

 here:

 start ==>starting point from which value the sequence can be get generate

 default value ==> 0

 Ex: range(10) ==> range(0,10)

 stop ==> until which value the sequence to be generate

 range(1,10) ==> 1 to 10-1 ==> 1 to 9

 -> range() with three parameters:

 Syntax:

 range(start, stop, step)

 ==> to generate the values from the specified with the difference, we can use above syntax.

 Ex: range(10,101,10) ==> 10, 20, 30, 40, 50, 60, 70, 80, 90, 100

print(range(10))

print(range(1,10))

print(range(1,101,10))

print(list(range(10)))

print(list(range(1,10)))

print(list(range(10,101,10)))

Syntax of the for loop:

--------------------------

 for variable in range(start, stop, step):

 statement-1

 statement-2

**# generate the values from 1 to 10 using for loop**

for i in range(1,11):

 print(i,end = "\t")

**# Write a python program using for loop to find the sum of n natural numbers.**

'''

natural number: 1, 2, 3,.. n

sum ==> 1 + 2+ 3+ ...n

'''

n = int(input("Enter the n:"))

sum\_naturals = 0

for i in range(1,n+1):

 sum\_naturals = sum\_naturals + i # 1 3

print("The sum of",n,"natural numbers is = ",sum\_naturals)

**Day-02**

**12-04-2025**

**=================**

**'''**

**Write a python program using for loop to find the sum of even natural numbers and also find**

**the sum of odd natural numbers separately.**

**'''**

'''

Natural numbers which are evenly divided with '2' are called as "Even natural numbers

Natural numbers which are not evenly divided with '2' are called as "Odd natural numbers.

1) generate 'n' natural numbers using for loop

2) check and separate even and odd natural numbers

3) find the sum of even natural numbers

4) find the sum of odd natural numbers.

'''

n = int(input("Enter the value for n:"))

s\_even = 0

s\_odd = 0

for i in range(1,n+1):

 if i % 2 == 0:

 s\_even = s\_even + i

 else:

 s\_odd = s\_odd + i

print("The sum of even natural numbers = ",s\_even)

print("The sum of odd natural numbers = ",s\_odd)

**'''**

**Write a python program to print the multiplication table of a given number.**

**'''**

'''

11 X 1 = 11

11 X 2 = 22

11 X 3 = 33 ...

'''

number = int(input("Enter a value:"))

for i in range(1,11):

 print(number," X ",i," = ",number\*i)

**# Write a program in python to accept a number as an input.**

**# And find the factorial of that number.**

'''

n = 7

n! ==> 7! ==> 7 X 6 X 5 X 4 X 3 X 2 X 1 => 5040

range(10) ==> 0 to 9

range(1,100) ==> 1 t0 99

range(100,10) ==> will not work

range(100,10,-1) ==> 100 to 11

'''

number = int(input("Enter a value:")) # 7

factorial = 1

for i in range(number,0,-1):

 factorial = factorial \* i # 7 42

print("The factorial of",number,"is = ",factorial)

**Transfer Statements:**

**==============**

-> Two transfer statements:

 1) break

 2) continue

1) break:

----------

 -> when we need to get the immediate termination from the loop, we can use "break".

 -> break is a keyword.

for i in range(1,10):

 if i == 6:

 break

 print(i)

2) continue:

--------------

 -> "continue" is a keyword

 -> when we want to stop the specific iteration and continue with remaining iterations as normal, then we can use "continue".

for i in range(1,10):

 if i == 6:

 continue

 print(i)

**# Write a python program to check whether the given number is prime number or not**

'''

prime number: When a number have two factors 1 and itself those are called as "Prime numbers".

6 ==> 1, 2, 3, 6

7 ==> 1, 7

factors: if a number is successfully with any value, that value is considered as factor for the given

number.

smallest prime number is 2

'''

Solution-1:

-----------

number = int(input("Enter a value:"))

count = 0

for i in range(2,number):

 if number % i == 0:

 count = count + 1

 break

if count == 0:

 print(number,"is prime number")

else:

 print(number,"is not prime number")

Solution-2:

-------------

number = int(input("Enter a value:"))

count = 0

for i in range(2,(number//2)+1):

 if number % i == 0:

 count = count + 1

 break

if count == 0:

 print(number,"is prime number")

else:

 print(number,"is not prime number")

Solution-3:

-------------

number = int(input("Enter a value:"))

count = 0

for i in range(2,int(number \*\* 0.5)+1):

 if number % i == 0:

 count = count + 1

 break

if count == 0:

 print(number,"is prime number")

else:

 print(number,"is not prime number")

Solution-4:

-------------

import math

number = int(input("Enter a value:"))

count = 0

for i in range(2,int(math.sqrt(number))+1):

 if number % i == 0:

 count = count + 1

 break

if count == 0:

 print(number,"is prime number")

else:

 print(number,"is not prime number")

Assignment:

--------------

1) Write a python program to check whether the given number is perfect number or not.

Perfect number:

---------------

The sum of all factors of the number excluding itself is equals to given number that number is called as "perfect number".

Ex:

6 ==> 1, 2, 3, 6

1 + 2 + 3 == 6

6 ==> perfect number.

**Day-03**

**13-03-2025**

**======================**

**# Write a program using python to print fibonacci series with n-elements.**

'''

0 1 1 2 3 5 8 13 21 ....

t1 = 0

t2 = 1

t3 = t1 + t2 # 1

t1 = t2 # 1

t2 = t3 # 1

t4 = t1 + t2 # 2

t1 = t2 # 1

t2 = t4 # 2

t5 = t1 + t2 # 3 ...

'''

n = int(input("Enter length of fibonacci series:"))

term1 = 0

term2 = 1

if n == 1:

 print(term1)

elif n == 2:

 print(term1,"\t",term2)

else:

 print(term1,"\t",term2,end = "\t")

 for i in range(3,n+1):

 term3 = term1 + term2

 print(term3,end = "\t")

 term1 = term2

 term2 = term3

**Nested For loop:**

**===========**

-> writing of a for loop in another for loop is called as "nested for loop".

Syntax:

 for v1 in range(start, stop):# outer for loop

 for v2 in range(start1, stop1):# inner loop

 loop body statements

# nested for loop

problem-1:

------------

for i in range(6):

 for j in range(5):

 print(i + j,end = "\t")

 print()

problem-2:

------------

for i in range(1,7):

 for j in range(10,16):

 print(j-i,end = "\t")

 print()

problem-3:

------------

for i in range(10,100,10):

 for j in range(10,21,3):

 print(i+j,end = "\t")

 print()

**# How many number of iterations will perform on the below snippet?**

for i in range(5):

 for j in range(5):

 print("Hi")

-> total iteration = stop-value of outer loop X stop-value of inner loop

-> total iterations = 5 X 5 ==> 25

**# How many number of iterations will perform on the below snippet?**

count = 0

for i in range(1,10):# 9

 for j in range(1,6):# 5

 print("Hi")

 count = count + 1

print(count)

-> from outer for loop, d1 = stop - start ==> 10-1 ==> 9

-> from inner for loop, d2 = stop - start ==> 6 - 1 ==> 5

-> total iterations => d1 X d2 ==> 9 X 5 ==> 45

**# How many number of iterations will perform on the below snippet?**

count = 0

for i in range(1,100,25):# stop // step ==> 4

 for j in range(1,20,5):# 20 // 5 ==> 4

 print("Hi")

 count = count + 1

print(count)

-> from outer loop, f1 = stop // start ==> 100 // 25 ==> 4

-> from inner loop, f2 = stop // start ==> 20 // 5 ==> 4

-> total iterations ==> f1 X f2 ==> 4 X 4 ==> 16

**Day-04**

**14-03-2025**

**==================**

**# Write a python program to check whether the number is perfect number or not.**

n = 6

s = 0

for i in range(1,n//2+1):

 if n % i == 0:

 s = s + i

if s == n:

 print("Perfect number")

else:

 print("Number is not perfect number")

**# Write a python program to print all perfect numbers from the given range.**

start = int(input("Enter the start value for the range:"))

stop = int(input("Enter the stop value for the range:"))

for i in range(start,stop + 1):

 s = 0

 for j in range(1,i//2 + 1):

 if i % j == 0:

 s = s + j

 if s == i:

 print(i,end = "\t")

**# Write a python program to print all prime numbers from the given range.**

import math

start = int(input("Enter the start value:")) # 1

stop = int(input("Enter the stop value:")) # 100

for i in range(start,stop + 1): # i = 1

 if i == 1:

 continue

 flag = False

 for j in range(2,int(math.sqrt(i))+1):

 if i % j == 0:

 flag = True

 break

 if flag == False:

 print(i,end = "\t")