

DAY 03

FUNCTIONS

```
# EXAMPLE for normal syntax
def greet(name):
    print("Hello, " + name + ". Good morning!")
greet('Tahir')

Hello, Tahir. Good morning!
```

FUNCTIONAL arguments

1.POSITIONAL ARGUMENTS

```
def add(a,b):
    return(a+b)
add(5,3)

8
```

2.KEYWORD ARGUMENTS

```
def greet(name,message):
    print(message + ", " + name + "!")
greet(name='Tahir',message='Good Morning')

Good Morning, Tahir!
```

3.DEFAULT ARGUMENTS

```
def greet(name,message='Hello'):
    print(message + ", " + name + "!")
greet('Tahir')
greet('HERO','HI')

Hello, Tahir!
HI, HERO!
```

4.VARIABLE LENGTH ARGUMENT a)Positional argument

```
def sum_numbers(*args):
    total=0
```

```

    for num in args:
        total+=num
    return total
print(sum_numbers(1,2,3,4,))
10

# keyword variable length agrgument
def display_info(**info):
    for key, value in info.items():
        print(f"{key}: {value}")

display_info(name="Tahir", age=19, city="Tumkur")

name: Tahir
age: 19
city: Tumkur

```

RETURN STATEMENT

```

def square(num):
    return num*num
square(5)
25

result=square(7)#Thala for a reason
print('Square is',result)

Square is 49

```

MODULES

1.Importing Modules

```

# math module
import math
print(math.sqrt(16))
4.0

# Import specific module
from math import pi,sin
print(pi)
print(sin(math.radians(90)))

```

```
3.141592653589793
1.0
```

2.IMPORT AND USE THE MODULE IN ANOTHER FILE

```
def greet(name):
    print("Hello, " + name )

def add(a,b):
    return a+b

import my_module

my_module.greet('Tahir')
result=my_module.add(5,3)

Hello, Tahir
```

Hands on Practice

1.Prime number by using function

```
def is_prime(num):
    if num <=1:
        return False
    for i in range(2, int(number**0.5) + 1):
        if num % i == 0:
            return False
    return True
number = int(input("Enter a number: "))
if is_prime(number):
    print(number, "is a prime number.")
else:
    print(number, "is not a prime number.")

Enter a number: 2
2 is a prime number.
```

2.Fibonacci sequence by a function

```
def fibonacci(n):
    sequence=[]
    a,b=0,1
    for _ in range(n):
        sequence.append(a)
        a,b=b,a+b
    return sequence
```

```
terms=int(input("Enter the number of terms: "))
print("Fibonacci sequence:",fibonacci(terms))
```

```
Enter the number of terms: 5
Fibonacci sequence: [0, 1, 1, 2, 3]
```

3.use the math module to solve a problem

```
import math

angle=float(input("Enter the angle in degrees: "))
radians=math.radians(angle)
print("Sine of angle:",math.sin(radians))
print("Cosine of angle:",math.cos(radians))
```

```
Enter the angle in degrees: 90
Sine of angle: 1.0
Cosine of angle: 6.123233995736766e-17
```

4.Factorial using Recursion

```
def factorial(n):
    if n==0:
        return 1
    else:
        return n*factorial(n-1)
number=int(input("Enter a number: "))
print("Factorial of",number,"is",factorial(number))
```

```
Enter a number: 5
Factorial of 5 is 120
```

5.Fibonacci series using functions

```
def fibonacci(n):
    a,b=0,1
    for _ in range(n):
        print(a,end=" ")
        a,b=b,a+b
count=int(input("Enter the number of terms: "))
fibonacci(count)
```

```
Enter the number of terms: 5
0 1 1 2 3
```

6.check if a number is prime

```

def is_prime(num):
    if num<=1:
        return False
    for i in range(2,int(num**0.5)+1):
        if num%i==0:
            return False
    return True
number=int(input("Enter a number: "))
if is_prime(number):
    print(number,"is a prime number.")
else:
    print(number,"is not a prime number.")

```

Enter a number: 5
5 is a prime number.

7.Reverse a String using a Function

```

def reverse_string(s):
    return s[::-1]
text=input("Enter a string: ")
print("Reversed string:",reverse_string(text))

```

Enter a string: car
Reversed string: rac

8.Find GCD of Two Numbers Using a Function

```

def gcd(a,b):
    while b:
        a,b=b,a%b
    return a
num1=int(input("Enter first number: "))
num2=int(input("Enter second number: "))
print("GCD:",gcd(num1,num2))

```

Enter first number: 12
Enter second number: 24
GCD: 12