

FUNTIONS AND MODULES

Funtions

```
#function syntax
def greet(name):
    print('hello,'+name+"!")
greet('alice')

hello,alice!
```

Function arguements

Positional arguements

```
def add(a,b):
    return a+b
print(add(6,9))

15

#keyword argument
def great(name,message):
    print(message+', '+name+'!')
great(name='alice',message='haiiii')

haiiii,alice!
```

Default argument

```
#default argument
def greet(name,message='hello'):
    print(message+', '+name+'!')
greet('alice',)
greet('bob','hlo')

hello,alice!
hlo,bob!
```

variable length arguemnet

```
def sum_numbers(*numbers):
    return sum(numbers)
print(sum_numbers(1,2,3,4,9))

19

def sum_numbers(**numbers):
    for key,value in numbers.items():
```

```
    print(key,value)
sum_numbers(name='guru',age=23,city='tumkur')

name guru
age 23
city tumkur
```

RETURN STATEMENT

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

25

result=square(5)
print('the square of the number is',result)

the square of the number is 25
```

MODULES

import math

```
import math
print(math.sqrt(16))

4.0

from math import pi,sin
print(pi)
print(sin(math.radians(90)))

3.141592653589793
1.0

def greet(name):
    print('hello,'+name)
def add(a,b):
    return a+b
import my_module
my_module.greet('alice')
print(my_module.add(2,3))

hello,alice
5
hello,alice
5
```

HANDS ON PRACTICE

To check the 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('the number is prime.')
else:
    print('the number is not prime')

enter a number7
the number is prime.
```

Fibonacci series

```
def fibonacci(n):
    sequence=[]
    a,b=0,1
    for _ in range(n):
        sequence.append(a)
        a,b=b,a+b
    return sequence
n=int(input('enter the number of terms'))
print('fibonacci sequence:',fibonacci(n))

enter the number of terms10
fibonacci sequence: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

math module

```
import math

angle = float(input('enter an angle in degrees'))
radian = math.radians(angle)
print('sine of angle:', math.sin(radian))
print('cosine of angle:', math.cos(radian))

enter an angle in degrees45
sine of angle: 0.7071067811865475
cosine of angle: 0.7071067811865475
```

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 is: ', factorial(number))

enter a number4
factorial is:  24
```

fibonacci series using funtions

```
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:'))
fibonacci(count)

enter the number:12
0 1 1 2 3 5 8 13 21 34 55 89
```

Find GCD of two numbers using a funtion

```
def gcd(a,b):
    while b:
        a,b=b,a%b
    return a

num1=int(input('enter a number:'))
num2=int(input('enter a number:'))
print('GCD;',gcd(num1,num2))

enter a number:2
enter a number:5
GCD; 1
```

Reverse a string using a funtion

```
def reverse_string(s):
    return[s[::-1]]
text=input('enter a string')
print('reversed string;'reverse_string(text))
```