Functions in Python

In [1]:
```
def hello():
    print('Good Evening')
```

In [3]:
```
def hello():
    print('Good Evening')
hello()
```

```
Good Evening
```

In [4]:
```
def hello():
    print('Good Evening')
Hello()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[4], line 3
      1 def hello():
      2     print('Good Evening')
----> 3 Hello()

NameError: name 'Hello' is not defined
```

In [5]:
```
def hello():
    print('Good Evening')
hello()

def hello():
    print('Good Evening')
hello()
def hello():
    print('Good Evening')
hello()
```

```
Good Evening
Good Evening
Good Evening
```

In [6]:
```
def hello():
    print('Good Evening')
hello()
hello()
hello()
```

```
Good Evening
Good Evening
Good Evening
```

We Will Define the Functions with Argument

Addition

In [7]:
```
def add(x,y):
    c=x+y
    print(c)
add(5,6)
```

```
11
```

In [8]:
```
def add(x,y,z):
    c=x+y+z
    print(c)
add(5,6)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[8], line 4
      2     c=x+y+z
      3     print(c)
----> 4 add(5,6)

TypeError: add() missing 1 required positional argument: 'z'
```

In [9]:
```
def add(x,y,z):
    c=x+y+z
    print(c)
add(5,6,7)
```

```
18
```

In [10]:
```
def add(x,y):
    c=x+y
    return c
add(5,6)
```

Out[10]: 11

In [11]:
```
def hello():
    print('Good Evening')
hello()

def add (x,y):
    c=x+y
    return c
add(5,6)
```

```
Good Evening
```

Out[11]: 11

Subtraction

In [13]:
```
def greet():
    print('Hello')
    print('Good Evening')
```

```
def add (x,y):
    c=x+y
    print(c)

def sub(x,y,z):
    d=x-y-z
    print(d)

greet()
add(5,4)
sub(10,2,4)
```

```
Hello
Good Evening
9
4
```

Why We Use The Return Function

In [14]:
```
def add_sub(x,y):
    c=x+y
    d=x-y
    return c,d
add_sub(4,5)
```

Out[14]: (9, -1)

In [20]:
```
def add_sub(x,y):
     c=x+y
     d=x-y
     return c,d

result,result1=add_sub(4,5)

print(result)
print(result1)
print(type(result))
print(type(result1))
```

```
9
-1
<class 'int'>
<class 'int'>
```

MUltiplication

In [21]:
```
def add_sub_mul(x,y):
    c=x+y
    d=x-y
    e=x*y
    return c,d,e
result,result1,result2=add_sub_mul(4,5)

print(result)
```

```
print(result1)
print(result2)

print(type(result))
print(type(result1))
print(type(result2))
```

```
9
-1
20
<class 'int'>
<class 'int'>
<class 'int'>
```

In [ ]:
```
# Type of Arguments Formal Arguments (We mention at the time of defination)

# Actual Arguments(Postional Argument, Keywoprd argument ,Default argument, variabl
length argument)
```

In [22]:
```
def update():
    x=8
    print(x)
update()
```

```
8
```

In [24]:
```
def add (a,b):

       c = a+b
       print(c)
add (5,6)
```

```
11
```

In [25]:
```
# Actual (Postional Argument)

def person(name,age):
    print(name)
    print(age)
person('qureshi',30)
```

```
qureshi
30
```

In [26]:
```
def person(name,age):
    print(name)
    print(age)
person('qureshi')
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[26], line 4
      2     print(name)
      3     print(age)
----> 4 person(         )

TypeError: person() missing 1 required positional argument: 'age'
```

In [27]:
```
def person(name):
    print(name)
    print(age+1)
person('qureshi',30)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[27], line 4
      2     print(name)
      3     print(age+1)
----> 4 person(         ,30)

TypeError: person() takes 1 positional argument but 2 were given
```

In [29]:
```
def person(name,age):
    print(name)
    print(age)
person(30,'qureshi')
```

```
30
qureshi
```

In [30]:
```
def person(name, age):
    print(name)
    print(age+1)
person(age=30,name='qureshi')
```

```
qureshi
31
```

In [31]:
```
# Default Argument
def person(name,age=18):
    print(name)
    print(age)
person('qureshi')
```

```
qureshi
18
```