function :-

for code reusability we use function. There are 2 types of function

i) User-defined function :-

here we take the function as per user requirement

Ex:- def addi(a,b):
    print(a+b)
addi(2,3)
addi(4,6)

o/p:- 5
10

Syntax:-

def function_name(parameters):
    return statement (optional)

ii) Built-in function :-

These functions come along python software

Example :-

input()
type()

let discuss some Examples on User-defined function

Ex-1:-

```
def f1():
    print("f1");
def f2():
    print("f2")
def f3():
    print("f3")
```

```
f3()
f2()
f1()
```

o/p:- f3
f2
f1

Ex-2:-

```
def f1():
    print("f1")
    f2()
def f2():
    print("f2")
def f3():
    print("f3")
    f1()
f3()
f2()
f1()
```

o/p:- f3
f1
f2

Ex-3:-

```
def f1():
    print("f1")
    f2()
```

```
def f2():
    print("f2")
    f3()
def f3():
    print("f3")
    f1()
f3()
```

o/p:- It continuously Execute in a recursive / repeatative manner as there is no ending manner.

Nested function:-

It means nothing but the examples we discuss but where we just called the function inside the ~~inside~~ function, but here now we also want to define function inside the function.

Ex:-

```
def f1():
    print("f1")
    def f2():
        print("f2")
        def f3():
            print("f3")
        f3()
    f2()
f1()
```

O/p:- f1
f2
f3

Ex:-

```
def f1():
    print("f1")
    def f2():
        print("f2")
        def f3():
            print("f3")
        f3()
    f2()
```

O/p:- [will be Empty ie no output]