13. Hàm (functions)

Hàm là một cấu trúc để tổ chức một chương trình. Chúng được biết đến với hầu hết các ngôn ngữ lập trình, đôi khi còn được gọi là các thủ tục (procedures) hoặc thủ tục con (subroutines). Các hàm được sử dụng để tái sử dụng mã ở nhiều nơi trong một chương trình.

Một hàm trong Python được định nghĩa bởi “def”. Cú pháp tổng quát sẽ như sau:

```
def function-name(Parameter list):

    statements, i.e. the function body
```

Danh sách tham số bao gồm không có hoặc nhiều tham số. Tham số được gọi là đối số, nếu hàm được gọi. Phần thân hàm được quy định phải tuân theo quy tắc khối, tức các đoạn mã cần viết thụt lại một TAB, 2 hoặc 4 SPACEs so với từ khóa “def” (xem lại bài 2_Chia khối và chú thích để biết thêm chi tiết). Phần thân chỉ được thực hiện khi hàm được gọi.

Thông số có thể là bắt buộc hoặc tùy chọn. Các tham số tùy chọn (nếu có) phải đặt sau các tham số bắt buộc.

Phần thân có thể trả về qua từ khóa “return”. Nó có thể đặt bất cứ nơi nào trong phần thân. “return” kết thúc thực thi hàm và trả lại kết quả cho hàm gọi đó. Nếu không có “return” trong mã của hàm, hàm sẽ kết thúc, khi chương trình đạt đến cuối của thân hàm.

Ví dụ:

```
def add(x, y):  #x & y la tham so bat buoc cua ham
   return x + y
def mul(x, y=1):  #x la tham so bat buoc, y la tuy chon
   return x * y
def mulOp(x, y=1, z=10):  #ham co 3 tham so
   return x * y * z
print "add(20, 30) = " , add(20, 30)  #buoc phai co hai doi so dau vao
print "mul(10, 2)", mul(10, 2)  #su dung ham dung 2 doi so
print "mul(10)", mul(10)  #su dung ham, ta chi can 1 doi so, doi so con lai se nhan gia tri mac dinh la 1
print "mulOp(2, z=3, y=1.5)",mulOp(2, z=3, y=1.5)  #co the nhap doi so tuong minh doi voi cac doi so tuy chon

Output:

add(20, 30) =  50

mul(10, 2) 20

mul(10) 10

mulOp(2, z=3, y=1.5) 9.0
```

Có rất nhiều tình huống trong lập trình, trong đó không thể xác định chính xác số lượng các thông số cần thiết. Một số tham số tùy ý (arbitrary parameter) có thể được sử dụng trong Python và được gọi là tham chiếu kiểu tuple. Một dấu hoa thị "*" được sử dụng ở phía trước của tên thông số cuối cùng để biểu thị nó như là tham chiếu kiểu tuple. Dấu hoa thị này không được nhầm lẫn với cú pháp C, nơi ký hiệu này được hiểu như là con trỏ.

```
def arbitrary(x, y, *more):  #
   print x, y, more

arbitrary(2, 3, 4, 5, 6, 7, 8)
```

```
Output:

2 3 (4, 5, 6, 7, 8)
```

Khi lập trình, bạn có thể không nhận thức được tất cả các trường hợp sử dụng có thể đối với đoạn mã, và bạn có thể muốn cung cấp nhiều tuỳ chọn hơn để làm việc với mô đun này. Chúng ta có thể truyền một số lượng các đối số biến cho một hàm bằng cách sử dụng *args và **kwargs trong mã.

Với *args đã được phân tích phía trên. Và dưới đây là một ví dụ thể hiện sự linh hoạt của nó.

```
def multiply(*args):
    z = 1
   for num in args:
        z *= num
    return z

print "multiply(1,2,3,4,5,6) = ", multiply(1, 2, 3, 4, 5, 6)  # co the dat bao nhieu do so tuy y nguoi dung
print "multiply() = ", multiply()  # doi so co the rong


Output:


multiply(1,2,3,4,5,6) =  720

multiply() =  1
```

Dấu hoa thị đôi “**” trước “kwargs” được sử dụng để đưa một dictionary với các đối số tùy chọn tới một hàm. Một lần nữa, hai dấu hoa thị (**) là yếu tố quan trọng ở đây, vì kwargs thường được sử dụng, mặc dù không bắt buộc.

Giống như *args, **kwargs có thể đưa bao nhiêu đối số mà bạn muốn cho hàm tùy ý. Tuy nhiên, **kwargs khác với *args là bạn sẽ cần phải chỉ đích danh từ khoá.

```
def    print_values(**kwargs):
    for key, value in kwargs.items():
        print("The value of {} is {}".format(key, value))
print_values(my_name="python", your_name="C++")


Output:


The value of my_name is python

The value of your_name is C++
```

Kết Luận

Về cơ bản, Hàm trong python cũng mang cùng ý nghĩa như hàm trong các ngôn ngữ lập trình cơ bản khác để người lập trình có thể tái sử dụng mã ở nhiều nơi trong chương trình. Ngoài ra để cung cấp nhiều tùy chọn hơn cho người dùng, hàm trong Python hỗ trợ thêm 2 kiểu đối số là *args và **kwargs. *args được dùng cho các tham chiếu kiểu tuple còn **kwargs được dùng cho các tham chiếu kiểu dictionary với điều kiện phải chỉ đích danh từ khóa. Chi tiết hơn về việc truyền đối số sẽ được chúng tôi trình bày trong bài tiếp theo trong loạt bài về Python cơ bản.