


PHẦN 1: LẬP TRÌNH PYTHON		
	NỘI DUNG:	TRANG
1	BIẾN TRONG PYTHON	
2	CÁC KIỂU DỮ LIỆU và TOÁN TỬ	
3	CẤU TRÚC ĐIỀU KHIỂN, VÒNG LẶP	
4	<b>HÀM TRONG PYTHON</b>	
5	LÀM VIỆC VỚI FILE TRONG PYTHON	
6	MODULE, XỬ LÝ NGOẠI LỆ	
7	LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG	
Tạ Quang Chiều - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275		97/112

97

HÀM (FUNCTION) TRONG PYTHON		
		
Tạ Quang Chiều - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275		98/112

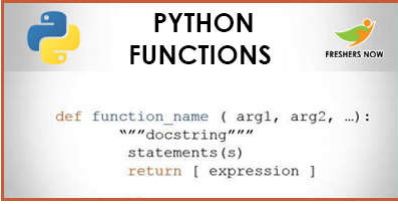
98

A

D

**Hàm trong Python là gì?**

- Hàm trong Python là một nhóm các câu lệnh trong chương trình được tổ chức chung với nhau để thực hiện một chức năng hay một nhiệm vụ cụ thể nào đó.
- Sử dụng hàm giúp phân rã chương trình từ một chương trình lớn, phức tạp thành các phần cụ thể nhỏ hơn giúp dễ quản lý, tổ chức, nâng cao khả năng tái sử dụng và chia sẻ công việc.




```
def function_name ( arg1, arg2, ...):
    """docstring"""
    statements(s)
    return [ expression ]
```

```
1 #xây dựng hàm trong python
2 def hello_MDC(str):
3     #Hiển thị câu chào
4     print('Hi ', str,', How are you?')
5     print('Have a nice day!')
```

```
1 #xử dụng hàm đã xây dựng
2 hello_MDC('Tùng Dương')
```

Hi Tùng Dương , How are you?  
Have a nice day!



Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

99/112

99

A

D

**Cú pháp:**

```
def Tên_hàm(các_tham_số):
    "function_docstring"
    Các câu lệnh xử lý bên trong hàm
    return [kết quả trả về]
```

- Từ khóa **def** được sử dụng để bắt đầu phần định nghĩa hàm.
- Tên\_hàm** được đặt theo quy tắc như tên biến.
- Các tham số** được truyền vào bên trong các dấu ngoặc đơn.
- Ở cuối là dấu hai chấm ":".
- Các lệnh xử lý bên trong hàm**: là lệnh để được thực thi.
- Kết quả trả về** cho hàm được thực hiện thông qua lệnh **return**

**Lưu ý:** Hàm không bắt buộc phải có tham số truyền vào hay kết quả trả về

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

100/112

100

A
D
A
I
Cú pháp:

- Trước khi bắt tay vào xây dựng 1 hàm trong Python, cần phải tự trả lời các câu hỏi:
  - Hàm này sử dụng nhằm mục đích gì?
  - Hàm này nhận đầu vào là gì?
  - Hàm này trả kết quả ra là gì?

```

1  #Xây dựng hàm tính n!
2  #1)Hàm này dùng để làm gì? - Để tính n!
3  #2)Hàm này nhận dữ liệu vào là gì? - Một số nguyên dương N
4  #3)Hàm trả kết quả là gì? - Một số nguyên dương là tích của 1*2*...*N
5  def giai_thua(n):
6      #Nhóm câu lệnh xử lý bên trong hàm
7      tích=1
8      for i in range(1,n+1):
9          tích=tích*i
10     #kết quả trả về cho hàm
11     return tích

```

```

1  n = int(input('Nhập vào một số nguyên N:'))
2  print(n,'!=',giai_thua(n))

```

Nhập vào một số nguyên N:10  
10 != 3628800

EXAMPLE

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
101/112

101

A
D
A
I
Gọi hàm

- Hàm sau khi được xây dựng, có thể thực hiện lời gọi hàm ở nơi nào cần dùng đến.

```

1  n = int(input('Nhập vào một số nguyên N:'))
2  print(n,'!=',giai_thua(n))

```

Nhập vào một số nguyên N:10  
10 != 3628800

```

1  #Gọi hàm giai_thua đã xây dựng
2  #Tính 12!
3  print('12! = ', giai_thua(12))

```

12! = 479001600

EXAMPLE

- Các lệnh mà chúng ta đã được học và sử dụng trước đây như: print(), input(), type(), int(), float(), str()... Đây thực chất là các hàm được Python định nghĩa sẵn.

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
102/112

102

A D  
A I

## Lệnh return (1)

- Lệnh **return** <kết quả trả về> được sử dụng để trả kết quả xử lý thông qua tên hàm.
- Lệnh return có thể có hoặc không.
- Trong trường hợp **không cung cấp <kết quả trả về>**, thì hàm **return** này sẽ trả về **None**. Nói cách khác, lệnh return được sử dụng để thoát khỏi định nghĩa hàm.

```

1 #xây dựng hàm trong python
2 def hello_MDC(str):
3     #Hiển thị câu chào
4     print('Hi ', str, ', How are you?')
5     print('Have a nice day!')
```

```

1 #Xây dựng hàm tính n!
2 def giai_thua(n):
3     tích=1
4     for i in range(1,n+1):
5         tích=tích*i
6     #kết quả trả về cho hàm
7     return tích
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
103/112

103

A D  
A I

## Lệnh return (2)

Lệnh return() có thể trả về 1 hay nhiều kết quả, nếu có nhiều hơn một kết quả thì ngăn cách nhau bởi **dấu phẩy**.

```

1 #Hàm tính tổng, hiệu, tích và thương
2 def all_ab(a,b):
3     tong=a+b
4     hieu=a-b
5     tích=a*b
6     thương=a/b
7     #hàm trả về
8     return tong, hieu, tích, thương
```

```

1 a=10
2 b=6
3 #Lấy kết quả trả về khi thực hiện hàm
4 tong,hieu,tích,thương = all_ab(a,b)
5
6 #Lưu ý: Thứ tự trả về theo đúng thứ tự đã viết trong
7 #câu lệnh return
8 print('Tổng ',a,'+',b,'=',tong)
9 print('Hiệu ',a,'-',b,'=',hieu)
10 print('Tích ',a,'*',b,'=',tích)
11 print('Thương ',a,'/',b,'=',thương)
```

Tổng 10 + 6 = 16

Hiệu 10 - 6 = 4

Tích 10 \* 6 = 60

Thương 10 / 6 = 1.6666666666666667

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
104/112

104

A D  
A I

## Tham số truyền vào hàm (1)

- Tham số bắt buộc
- Tham số có mặc định (Default parameter)
- Tham số có độ dài biến (Variable-Length Parameter)

➤ **Tham số bắt buộc**

```

1 #Xây dựng hàm
2 #Hàm giai_thu
3 def giai_thua
4     tích=1
5     for i in
6         tích=
7     #kết quả
8     return ti

```


```

1 #Gọi hàm giai_thua đã xây dựng
2 print('12! = ', giai_thua(12))

12! = 479001600

1 #Gọi hàm giai_thua đã xây dựng
2 #Khi không truyền vào tham số
3 print('12! = ', giai_thua())

```



```

Traceback (most recent call last)
<ipython-input-14-01226097b9b9> in <module>
      1 #Gọi hàm giai_thua đã xây dựng
      2 #Khi không truyền vào tham số
----> 3 print('12! = ', giai_thua())

TypeError: giai_thua() missing 1 required positional argument: 'n'

```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

105/112

105

A D  
A I

## Tham số truyền vào hàm (2)

➤ **Tham số mặc định cho hàm:**

- Để hạn chế trường hợp báo lỗi khi gọi hàm không cung cấp tham số thì trong Python cũng cung cấp cho chúng ta **thiết lập giá trị mặc định của tham số** khi khai báo hàm. Bằng cách sử dụng dấu **=** với cú pháp như sau:

```
def ten_ham(param = defaultValue):
    # code
```

- Trong đó: **defaultValue** là giá trị mặc định của tham số đó mà bạn muốn gán.

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

106/112

106

A

D

Tham số truyền vào hàm (3)

➤ Ví dụ:

- Hàm **sum()** gọi khi không truyền giá trị và có truyền giá trị

```

1  # Phan dinh nghia ham o day
2  def sum( arg1=5, arg2=7 ):
3      # Cong hai tham so va tra ve ket qua."
4      total = arg1 + arg2
5      return total;
6
7  # Goi ham sum() co du 2 tham so truyen vao
8  print(sum( 10, 20 ))
9
10 # Goi ham sum() khong co tham so truyen vao
11 print(sum())
12
13 #Khi code tren duoc thuc thi se cho ket qua:

```

30  
12

EXAMPLE

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

107/112

107

A

D

Tham số truyền vào hàm (4)

➤ Tham số có độ dài biến (Variable-Length Parameter)

- Trên thực tế, không phải lúc nào chúng ta cũng biết được chính xác số lượng biến truyền vào trong hàm. Chính vì thế trong Python có cũng cấp cho chúng ta khai báo một **param** đại diện cho các biến truyền vào hàm bằng cách thêm dấu **\*** vào trước param đó.

■ Ví dụ:

```

1  def get_sum(*num):
2      tmp = 0
3      # duyet cac tham so
4      for i in num:
5          tmp += i
6      return tmp
7
8  result = get_sum(1, 2, 3, 4, 5)
9  print(result)
10
11 #Khi code tren duoc thuc thi se cho ket qua:

```

15

EXAMPLE

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

108/112

108

A

D

A

I

## Tham số truyền vào hàm (5)

**Phạm vi của biến trong hàm.**

- Khi một biến được khai báo ở trong hàm thì nó chỉ có thể được sử dụng ở trong hàm đó thôi.

➤ Ví dụ:

```

1 def say_hello():
2     a = "Hello"
3     print(a)
4
5 print(a)
6 # Lỗi: name 'a' is not defined

```

```

-----
NameError                                Traceback (most recent call last)
<ipython-input-10-b25341c3aba9> in <module>
      3     print(a)
      4
----> 5 print(a)
      6 # Lỗi: name 'a' is not defined

NameError: name 'a' is not defined

```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

109/112

109

A

D

A

I

# THỰC HÀNH

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

110/112

110

**A**  
A

**D**  
I

## Bài 17\_a: Viết hàm cho các bài đã thực hiện

- 1) Viết hàm **greeting()**: Trả về câu chào với tham số truyền vào là chuỗi họ tên và năm sinh (**Xem lại bài tập số 2**)
- 2) Viết hàm **rabbit\_count()**: tính số thỏ trong rừng khi truyền vào số tháng (**Xem lại bài tập số 3**)
- 3) Viết hàm **count\_mark()**: trả về số sinh viên học lại và tổng số sinh viên trong lớp với tham số truyền vào là một danh sách bảng điểm (**Xem lại bài tập số 5 ý 1, 2**)




Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275
111/112

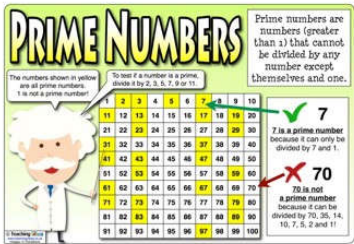

111

**A**  
A

**D**  
I

## Bài 17\_b: Viết hàm cho các bài đã thực hiện

- 4) Viết hàm **bmi\_show()**: Trả về nhận xét dựa vào chỉ số BMI đã tính với 2 tham số truyền vào là chiều cao, cân nặng (**Xem lại bài tập số 9**)
- 5) Viết hàm **cal\_point()**: Trả về điểm trung bình hệ 10 và hệ 4 của một học sinh khi truyền vào danh sách điểm (**Xem lại bài tập số 12 ý 2**)
- 6) Viết hàm **list\_prime()**: trả danh sách các số nguyên tố trong khoảng từ 1 đến n với tham số truyền vào là n (**Xem lại bài tập số 14**)

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275
112/112

112