

Lesson 1: Hello world and Python Syntax

Bài đầu tiên sẽ giới thiệu về ngôn ngữ python và chương trình huyền thoại Hello world cho python

Nội dung chính:

I. Viết chương trình hello world

II. Python syntax

1. Thụt lề`

2. Chú thích

3. Biến cục bộ

I. Hello world

Thực thi bằng 1 cái trình thực thi có tên là cpython (nó có thể coi là kết hợp của trình thông dịch (interpreter) và biên dịch (compiler), vì nó biên dịch mã Python thành bytecode trước khi thông dịch nó), nó được viết bằng C bởi ông Guido van Rossum

Dịch từng dòng lệnh sang ngôn ngữ máy (thực hiện tuần tự), dòng lệnh được dịch sang bytecode, sau đó bytecode tiếp tục được thực thi bởi VM

=> lỗi ở đâu thì dừng chương trình ở đấy

```
print('hello world!')  
print("hello")
```

```
hello world!  
hello
```

```
print('chuỗi 1\nchuỗi 2', end = "\n\n")  
print('chuỗi 3')
```

```
chuỗi 1  
chuỗi 2
```

```
chuỗi 3
```

II. Python Syntax

Giống như các ngôn ngữ tự nhiên, mỗi ngôn ngữ đều quy định những quy tắc trong cú pháp, VD như kết thúc câu bằng dấu chấm, kết thúc ý bằng dấu phẩy, cảm thán bằng

dấu !, cuối câu hỏi có dấu ?, v.v... Ngôn ngữ lập trình cũng thế, nó có những quy tắc cú pháp khác nhau. Với python, một số quy tắc cơ bản như sau:

1. Indentation

- Thụt đầu dòng là khoảng trắng đầu mỗi dòng
- Trong một số ngôn ngữ khác, nó chỉ dùng để format cho dễ đọc
- Còn trong python, nó để biểu thị một khối code (code block)

```
print('hello');print('hello 2')
```

```
if (True):{print('Đúng!!!'),print('aaaa')}
```

```
if (True):  
    if (True):  
        print('here', 'is', 'the', 'block', 'code')
```

```
hello  
hello 2  
Đúng!!!  
aaaa  
here is the block code
```

Phím tắt trong VS code: Ctrl +] để thụt dòng sang phải

Ctrl + [để thụt dòng sang trái

2. Comments

Chú thích giúp code dễ đọc, dễ hiểu. Trong python, dùng dấu # để biểu thị chú thích trên 1 dòng, dấu nháy 3 để chú thích nhiều dòng

```
'''  
Bài này dùng để gia'i thích về` comments  
Thêm 1 dòng nữa cũng cha' sao  
'''
```

```
# Đây là đoạn code để` print()  
print('ahihi') # Dòng này để` print() ra chữ ahihi  
print('ahaiah')
```

```
ahihi
```

Phím tắt trong vs code: Ctrl + /

3. Variables

Biến là cái để chứa giá trị dữ liệu. Nó trỏ đến vùng nhớ bất kì trong bộ nhớ và ghi dữ liệu vào đấy. Khi lấy ra nó chỉ cần trỏ đến địa chỉ ấy và lấy giá trị ra.

Một biến gồm những thành phần sau:

- Tên
- Giá trị
- Biến cục bộ và biến toàn cục

3.1. Tên biến

Quy tắc đặt tên biến:

- Tên biến phải bắt đầu bằng chữ cái hoặc ký tự gạch dưới
- Tên biến không được bắt đầu bằng số
- Tên biến chỉ được chứa các ký tự chữ và số và dấu gạch dưới (A-z, 0-9, và _)
- Tên biến có phân biệt chữ hoa chữ thường (age, Age và AGE là ba biến khác nhau)

Tên biến không trùng với một số cái mặc định trong python như:

and	exec	not
assert	finally	or
break	for	pass
class	from	print
continue	global	raise
def	if	return
del	import	try
elif	in	while
else	is	with
except	lambda	yield

```
# tên biến đúng
myname = "John"
my_name = "John"
_my_name = "John"
myName = "John"
MYName = "John"
myname2 = "John"

# Một số ví dụ tên biến sai
2myvar = "John" # số ở đầu
my-var = "John" # có dấu -
my var = "John" # có nút cách
and = 2 # trùng với từ khóa and
```

```
File "<ipython-input-46-14bdd6e64d97>", line 2
  2myvar = "John" # số ở đầu
    ^
```

SyntaxError: invalid syntax

Kinh nghiệm khi đặt tên biến

- Đặt tên biến phải thể hiện được giá trị mà nó lưu trữ, tránh các biến đặt vội như x, y, a, b, temp,... Nó sẽ hình thành thói quen sau này, giúp code clean và dễ đọc, dễ debug hơn.
- Đặt tên đúng, đủ, ngắn gọn, tránh các từ như data, info, list, ...
- Nên đặt tên biến bằng tiếng anh, có ý nghĩa rõ ràng

3.2. Giá trị biến

ví dụ khai báo biến

```
x = 1.52
age = 10
name = 'hieu'
a, b, c = 5, 4, 6
m = n = 4
```

Đầu ra biến

Dùng hàm print() để in giá trị ra

```
x = "Python"
y = "is"
z = "awesome"
print(x, y, z)
print(x + y + z)
```

Python is awesome
Pythonisawesome

in ra 10 chữ python

```
a = 'python'
print(a,a,a,a,a,a,a,a)
```

python python python python python python python python

3.3. Local và global variable

- Biến cục bộ (local variable) là biến được khai báo trong hàm và chỉ được sử dụng trong hàm
- Biến toàn cục (global variable) là biến được khai báo ở ngoài và được dùng trong cả ngoài và trong hàm.

```
x = 1
print(x)
x = 2
print(x)
```

1
2

```
global_var = 'global'
```

Định nghĩa 1 cái hàm

```
def myfunc():  
    hh = "local variable"  
    print(hh)  
    print(global_var)
```

```
myfunc() # gọi hàm
```

```
local variable  
global
```

```
print(x)
```

```
global variable
```

Thông thường, khi bạn tạo một biến bên trong một hàm, biến đó là cục bộ và chỉ có thể được sử dụng bên trong hàm đó.

Để tạo một biến toàn cục bên trong một hàm, bạn có thể sử dụng từ khóa toàn cục.

```
x = "awesome"
```

```
def myfunc():  
    global x  
    x = "fantastic"
```

```
myfunc() # thực thi hàm myfunc()
```

```
print("Python is " + x)
```

```
Python is fantastic
```