

16. Biến toàn cục, và địa phương (Global and Local Variables)

Trong phần này ta sẽ phân tích qua các ví dụ:

Trong ví dụ 1, ta định nghĩa hàm `func()` trước cả đặt biến toàn cục `s = "hello world"`. Hàm `func()` cần hiểu `s` sẽ là một biến toàn cục ở đâu đó bên ngoài hàm `func()`

```
def func():  
    print s  
s = "hello world"  
func()
```

Output:

```
hello world
```

Trong ví dụ 2, ta thấy trong `func()` có thêm một biến địa phương `s`, `func()` sẽ nhận giá trị của biến địa phương này thay vì biến toàn cục phía bên ngoài. Ta thấy rằng hai biến toàn cục và biến địa phương cùng tên trong hàm `func()` không ảnh hưởng gì đến nhau.

```
def func():  
    s = "it's me"  
    print s  
s = "hello world"  
func()  
print "s = ",s
```

Output:

```
it's me
```

```
s = hello world
```

Trong ví dụ 3, ta thấy chương trình báo lỗi “`UnboundLocalError: local variable 's' referenced before assignment`”. Python “giả định” rằng chúng ta muốn có một biến địa phương do việc gán bên trong của hàm `func_err()`, do đó câu lệnh `print` đầu tiên đưa ra thông báo lỗi này. Bất kỳ biến được thay đổi hoặc tạo ra bên trong một hàm là biến địa phương, nếu nó không được khai báo như một biến toàn cục.

```
def func_err():  
    print s  
    s = "it's me"  
    print s  
  
s = "hello world"  
func_err()
```

Output

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

File "<stdin>", line 2, in `func_err`

`UnboundLocalError: local variable 's' referenced before assignment`

Ví dụ 4. Sửa lỗi cho ví dụ 3, để nói với Python, chúng ta muốn sử dụng biến toàn cục, chúng ta

phải sử dụng từ khóa "global", như trong ví dụ 4 sau:

```
def func_fix():  
    global s  
    print s  
    s = "it's me"  
    print s
```

```
s = "hello world"  
func_fix()
```

Output:

hello world

it's me

Ví dụ 5 mô tả biến địa phương không thể truy cập sau khi ra khỏi scope của nó.

```
def func():  
    ss = "it's me"  
    print ss
```

```
func()  
print ss
```

Output:

it's me

Traceback (most recent call last):

File "C:/Users/cuongtran/PycharmProjects/BigData/course.py", line 6, in <module>

print ss

NameError: name 'ss' is not defined

Ví dụ 6: Tổng hợp của việc sử dụng biến địa phương và biến toàn cục

```
def func(x,y):  
    global a  
    a = 1  
    x = y+1  
    b = 2  
    print a,b,x,y
```

```
a,b,x,y = 3,4,5,6
```

```
func(7,8)
```

```
print a,b,x,y
```

Output

1 2 9 8

1 4 5 6

Kết Luận

Như vậy cái tên nói lên tất cả. Biến toàn cục và biến địa phương khác nhau ở chính phạm vi mà nó được sử dụng trong chương trình. Biến địa phương sẽ chỉ được sử dụng bên trong phạm vi hàm và không thể truy cập được khi ra khỏi phạm vi đó. Ngược lại biến toàn cục được sử dụng trong toàn bộ chương trình, được đặt bên ngoài hàm và nếu một hàm muốn truy cập vào biến toàn cục thì cần khai báo lại biến trong hàm với từ khóa “global”.