



BÀI 3

Kiểu dữ liệu Logical và Câu lệnh If else

Tóm Tắt Nội Dung

Trong bài học này chúng ta sẽ đi tìm hiểu lần lượt các nội dung

- 1 Tìm hiểu về toán tử Logical trong Python
- 2 Tìm hiểu về câu lệnh điều kiện rẽ nhánh if else
- 3 Câu lệnh **pass** trong Python

3.2 Toán tử Logic trong Python



Dữ liệu **Logic** là gì ?

Dữ liệu logic (bool) là loại dữ liệu chỉ có 2 giá trị Đúng (True) và Sai (False).
Dữ liệu logic được dùng khi mô tả các điều kiện hoặc phép so sánh số hoặc chữ

Hàm **bool()** sẽ kiểm tra xem giá trị của đối số truyền vào là True hay False

```
print(3 > 2)  
# kết quả cho ra một số : True
```

```
x = 2 > 3  
print(bool(x)) # kết quả cho ra một số : False
```

3.2 Toán tử Logic trong Python



Toán tử So Sánh

Toán tử	Ý Nghĩa	Ví dụ	Diễn giải Phép so sánh
==	So sánh bằng	<code>x == y</code>	x có bằng y hay không ?
!=	So sánh không bằng	<code>x != y</code>	x khác y hay không ?
>	So sánh lớn hơn	<code>x > y</code>	x lớn hơn y hay không
>=	Lớn hơn hoặc bằng	<code>x >= y</code>	x lớn hơn hoặc bằng y hay không
<	So sánh bé hơn	<code>x < y</code>	x bé hơn hoặc bằng y hay không
<=	Bé hơn hoặc bằng	<code>x <= y</code>	x bé hơn hoặc bằng y hay không

Toán tử so sánh trên sẽ tạo ra kiểu dữ liệu Logic: True hoặc False

```
print(3 > 2)
# kết quả cho ra một số : True
```

3.2 Toán tử Logic trong Python



Toán tử Logic

Ví dụ: Cho $x = 5$, $y = 8$

Toán tử	Ý Nghĩa	Ví dụ	Kết quả
and	và	$x > 3$ and $y > 5$	True and True = True
or	Hoặc	$x > 3$ or $y > 9$	True and False = False
not	Phủ định	not($x > 3$)	not(True) = False

Bảng định nghĩa giá trị của toán tử **and** và **or**

x	and	y	Kết quả
True	and	True	True
True	and	False	False
False	and	False	False
False	and	True	False

x	or	y	Kết quả
True	or	True	True
True	or	False	True
False	or	False	False
False	or	True	True

3.3 Câu lệnh if, else, elif



Từ thực tế đến ngôn ngữ lập trình

“**Nếu** trời tạnh mưa thì tôi sẽ được **đi chơi**,
Còn nếu không tôi sẽ ở nhà **học bài**”

Trong phát biểu trên chúng ta có 2 hành động là “đi chơi” và “học bài”. Việc thực hiện 2 hành động này phụ thuộc vào điều kiện của thời tiết sau từ khóa “nếu”

Trong các ngôn ngữ lập trình các lệnh thực hiện theo điều kiện như vậy sẽ được thể hiện bằng từ khóa **if**

Câu lệnh **if** mô tả bằng ngôn ngữ con người như sau :

Nếu <điều kiện đúng> thì
<thực hiện khối lệnh này>

3.3 Câu lệnh if, else, elif



Cú pháp if ngắn

Câu lệnh "if" được sử dụng để kiểm tra một điều kiện và thực thi một khối mã chỉ khi điều kiện đó là đúng.

if <điều kiện kiểm tra>:

<khối lệnh thực thi khi điều kiện đúng>

Điều kiện kiểm tra: Trả về một giá trị logic (True/False) và kết thúc bằng dấu :

Khối lệnh của if: Thụt vào một khoảng bằng nhau tối thiểu một dấu cách so với if. Điều này rất quan trọng, nếu không đúng lệnh sẽ không chạy được.

```
x = 15
if x > 10:
    print("x lớn hơn 10")
#Output: x lớn hơn 10
```

```
x = 7
if x > 10:
    print("x lớn hơn 10")
```

3.3 Câu lệnh if, else, elif



Cú pháp if đầy đủ

if <điều kiện kiểm tra>:

<khối lệnh thực thi khi điều kiện đúng >

else:

<khối lệnh thực thi khi điều kiện sai>

Điều kiện kiểm tra: Trả về một giá trị logic (True/False) và kết thúc bằng dấu :

else: sau else có thêm dấu :

```
x = 15
if x > 10:
    print("x lớn hơn 10")
else:
    print("x nhỏ hơn 10")
```


3.3 Câu lệnh if, else, elif



Cú pháp if-elif-else

Câu lệnh "if-elif-else" được sử dụng để kiểm tra nhiều điều kiện khác nhau và thực thi các khối mã tương ứng. Cú pháp của câu lệnh if-elif-else như sau:

if <điều kiện 1>:

<khối lệnh thực thi khi điều kiện 1 đúng >

elif <điều kiện 2>:

<khối lệnh thực thi khi điều kiện 2 đúng >

...

else:

<khối lệnh thực thi khi không có điều kiện nào đúng>

3.3 Câu lệnh if, else, elif



Cú pháp if-elif-else

Câu lệnh "if-elif-else" được sử dụng để kiểm tra nhiều điều kiện khác nhau và thực thi các khối mã tương ứng. Cú pháp của câu lệnh if-elif-else như sau:

Ví dụ:

```
x = 10
if x < 5:
    print("x nhỏ hơn 5")
elif x < 10:
    print("x nhỏ hơn 10")
else:
    print("x lớn hơn hoặc bằng 10")
```

Kết quả: x lớn hơn hoặc bằng 10

3.3 Câu lệnh if, else, elif



Câu lệnh pass trong Python

Từ khóa **pass** được sử dụng để đánh dấu một khối mã rỗng hoặc tạm thời không có nội dung. Nó không làm gì cả và được sử dụng để bỏ qua một phần của mã trong trường hợp bạn không muốn thực hiện bất kỳ hành động nào tại thời điểm đó, nhưng cú pháp yêu cầu phải có một khối mã

```
if x < 5:  
    pass #Không muốn làm gì khi điều kiện đúng  
  
for i in range(10):  
    pass # không có hành động cụ thể trong mỗi lần lặp  
  
def my_function():  
    pass #Không muốn làm gì khi gọi hàm
```

3.4 Câu lệnh if, else, elif



Nguồn tham chiếu

- ◆ Dữ liệu Logic

https://www.w3schools.com/python/python_booleans.asp

- ◆ Tổng quan về if, else, elif

https://www.w3schools.com/python/python_conditions.asp

Tổng kết lại bài 3

1. Nắm được cách sử dụng lệnh `input()`
2. Nắm được toán tử Logical trong Python
3. Cách sử dụng câu lệnh điều kiện rẽ nhánh `if else`
4. Nắm được cách sử dụng vòng lặp `while` trong Python
5. Nắm được cách sử dụng vòng lặp `for` trong Python