Python Booleans

(Kiểu logic trong Python)

Booleans represent one of two values: True or False.

>> Kiểu logic chỉ có 1 trong 2 giá trị:True hoặc False.

#Kiểu logic biểu diễn một trong hai giá trị: True hoặc False

Boolean Values

Giá trị logic

In programming you often need to know if an expression is True or False.

>>Trong lập trình bạn cần biết một biểu thức có giá trị là True hay false.

#Trong lập trình, thường, bạn cần biết một biểu thức có giá trị là True hay False.

You can evaluate any expression in Python, and get one of two answers, True or False.

>> Bạn có thể tính được giá trị bất kỳ biểu thức nào trong Python và nhận được câu trả lời là True hoặc False. (Đúng hay sai)

When you compare two values, the expression is evaluated and Python returns the Boolean answer:

>> Khi bạn so sánh hai giá trị, biểu thức được tính toán và Python trả về câu trả lời kiểu logic:

Example

print(10 > 9)  
print(10 == 9)  
print(10 < 9)

When you run a condition in an if statement, Python returns True or False:

>>Khi bạn thực hiện (Chạy) một điều kiện trong câu lệnh If, Python trả về True hoặc False:

Example

Print a message based on whether the condition is True or False:

>>In ra một thông báo dựa trên điều kiện là True hay False:

#In ra một thông báo dựa trên/dựa vào điều kiện là True hoặc False

a = 200  
b = 33  
  
if b > a:  
  print("b is greater than a")  
else:  
  print("b is not greater than a")

Evaluate Values and Variables

>> Đánh giá các giá trị và các biến

The bool() function allows you to evaluate any value, and give you True or False in return,

>>Hàm bool() cho phép bạn đánh giá bất kỳ giá trị nào, và trả về cho bạn True hay False

#

Example

Evaluate a string and a number:

>>Đánh giá một chuỗi và một số:

#

print(bool("Hello"))  
print(bool(15))

Example

Evaluate two variables:

>> Đánh giá Hai biến

#

x = "Hello"  
y = 15  
  
print(bool(x))  
print(bool(y))

Most Values are True

(Hầu hết các giá trị là True)

Almost any value is evaluated to True if it has some sort of content.

>>Hầu như các giá trị được đánh giá là True nếu nó có một số nội dung được chọn.

#Hầu như các giá trị được đánh giá là True nếu nó có chứa nội dung [nào đó].

Any string is True, except empty strings.

>>Bất kỳ string nào cũng đều là True, ngoại trừ chuỗi rỗng

#

Any number is True, except 0.

>>Bất kỳ number nào cũng là True, ngoại trừ số 0

#

Any list, tuple, set, and dictionary are True, except empty ones.

>>Bất kỳ list, tuple, set, và dictionary đều là True, ngoại trừ giá trị nào rỗng

#Bất kỳ list, tuple, set và dictionary đều là True, trừ những cái [list, tuple, set và dictionary] rỗng.

Example

The following will return True:

>>Kết quả dưới đây sẽ trả về True:

bool("abc")  
bool(123)  
bool(["apple", "cherry", "banana"])

Some Values are False

>>Một vài giá trị là False

In fact, there are not many values that evaluate to False, except empty values, such as (), [], {}, "", the number 0, and the value None. And of course the value False evaluates to False.

Thực tế, không nhiều giá trị được đánh giá là False, ngoại trừ giá trị trống (rỗng), như là (), [],{},””, số 0, và giá trị None. Và tất nhiên giá trị False được đánh giá là False

#

Example

The following will return False:

>>Kết quả dưới đây sẽ trả về False

#

bool(False)  
bool(None)  
bool(0)  
bool("")  
bool(())  
bool([])  
bool({})

One more value, or object in this case, evaluates to False, and that is if you have an object that is made from a class with a \_\_len\_\_ function that returns 0 or False:

Một giá trị hoặc đối tượng khác trong trường hợp này, được đánh giá là False, và đó là nếu bạn có một đối tượng được tạo ra từ một class có hàm \_len\_ trả về 0 hoặc False

#Thêm một giá trị hay đối tượng đánh giá trả về False trong trường hợp này, và đó là khi bạn có một đối tượng được tạo ra từ một class với hàm \_\_len\_\_ trả về 0 hoặc False

#phần này ai học OOP rồi dễ hiểu, chưa học chắc khó hình dung

Example

class myclass():  
  def \_\_len\_\_(self):  
    return 0  
  
myobj = myclass()  
print(bool(myobj))

Functions can Return a Boolean

>> Hàm có thể trả về giá trị logic

You can create functions that returns a Boolean Value:

>> Bạn có thể tạo ra các hàm mà trả về một giá trị kiểu logic:

#

Example

Print the answer of a function:

>> In ra câu trả lời của hàm:

#

def myFunction() :  
  return True  
  
print(myFunction())

You can execute code based on the Boolean answer of a function:

Bạn có thể thực thi lệnh dựa trên câu trả lời logic của một hàm:

#

Example

Print "YES!" if the function returns True, otherwise print "NO!":

>>In “Yes!” nếu hàm trả về True, ngược lại in “No!”:

#In "YES!" nếu hàm trả về True, ngược lại in "NO!":

def myFunction() :  
  return True  
  
if myFunction():  
  print("YES!")  
else:  
  print("NO!")

Python also has many built-in functions that return a boolean value, like the isinstance() function, which can be used to determine if an object is of a certain data type:

Ngoài ra Python cũng có nhiều hàm tích hợp trả về một giá trị logic, như hàm isinstance(), hàm này có thể sử dụng để xác định một đối tượng là thuộc kiểu dữ liệu nhất định nào không:

#

Example

Check if an object is an integer or not:

>>Kiểm tra một đối tượng là số nguyên hay không:

x = 200  
print(isinstance(x, int))

#Dịch phần lý thuyết bài học, phần test tự làm nha

Test Yourself With Exercises

Bài kiểm tra:

Exercise:

The statement below would print a Boolean value, which one?

>> Câu lệnh bên dưới có thể in ra một giá trị logic không? Giá trị đó là gì?

print(10 > 9)

TL: có - True