Python Booleans

Booleans represent one of two values: True or False.

Booleans đại diện cho 2 giá trị: đúng hoặc sai.

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

Boolean Values

Giá trị Boolean

#

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

Trong lập trình, bạn thường cần biết một biểu thức là Đúng hay Sai.

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

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

Bạn có thể đánh giá bất kỳ biểu thức nào trong Python và nhận được một trong hai câu trả lời, Đúng hoặc Sai.

#True và False là các giá trị đặc biệt, khỏi dịch có lẽ tốt hơn

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 đánh giá và Python trả về câu trả lời Boolean:

#

Example

print(10 > 9) # True  
print(10 == 9) # False  
print(10 < 9) # False

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

Khi bạn 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 tin nhắn dựa trên điều kiện là Đúng hay Sai:

#In ra một tin nhắn/thông điệp dựa trên/dựa vào điều kiện là True hay 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 giá trị và 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 là True hoặc False,

#

Example

Evaluate a string and a number:

Đánh giá một chuỗi và một số:

#

print(bool("Hello")) #True  
print(bool(15)) #True

Example

Evaluate two variables:

Đánh giá hai biến:

#

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

Most Values are True

Hầu hết các giá trị đều đúng

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

Hầu hết mọi giá trị đều được đánh giá là True nếu nó có một số loại nội dung.

#

Any string is True, except empty strings.

Bất kỳ chuỗi nào là True, ngoại trừ các chuỗi rỗng.

#

Any number is True, except 0.

Bất kỳ số nào là True, ngoại trừ 0.

#

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

Mọi danh sách, bộ, bộ và từ điển đều Đúng, ngoại trừ những danh sách trống.

#Bất cứ dữ liệu danh sách, tuple, set, và từ điển đều True, trừ những dữ liệu [danh sách, tuple, set, và từ điển] rỗng.

#tuple và set là kiểu dữ liệu tương tự như kiểu dữ liệu danh sách (list)

Example

The following will return True:

Phần sau đây sẽ trả về True:

#

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

Some Values are False

Một số giá trị là sai

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.

Trong thực tế, không có nhiều giá trị đánh giá thành Sai, ngoại trừ các giá trị trống, chẳng hạn như (), [], {}, "", số 0 và giá trị Không có. Và tất nhiên giá trị False đánh giá là False.

#Trong thực tế/thực ra, không có nhiều giá trị đánh giá thành False, ngoại trừ các giá trị trống/rỗng, chẳng hạn như (), [], {}, "", số 0 và giá None. Và tất nhiên giá trị False đánh giá là False.

#None: đây là giá trị đặc biệt trong Python, sau này sẽ gặp, khỏi dịch có lẽ tốt hơn

Example

The following will return False:

Phần sau 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 từ một lớp 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)) #True #đâu ra giá trị True này?

Functions can Return a Boolean

Các hàm có thể trả về một Boolean

You can create functions that returns a Boolean Value:

Bạn có thể tạo các hàm trả về Giá trị Boolean:

#

Example

Print the answer of a function:

In câu trả lời của một hàm:

#

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

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

Bạn có thể thực thi mã dựa trên câu trả lời Boolean của một hàm:

#

Example

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

In "CÓ!" nếu hàm trả về giá trị True, nếu không thì in "NO!":

#In "YES!" nếu hàm trả về giá trị 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:

Python cũng có nhiều hàm tích hợp trả về giá trị boolean, như hàm isinstance (), có thể được sử dụng để xác định xem một đối tượng có thuộc một kiểu dữ liệu nhất định hay không:

#

Example

Check if an object is an integer or not:

Kiểm tra xem một đối tượng có phải là số nguyên hay không:

#

x = 200  
print(isinstance(x, int)) # True