



BOOL

Os booleanos representam um de dois valores: `True` ou `False`.

Valores Booleanos

Na programação, você geralmente precisa saber se uma expressão é `True` ou `False`.

Você pode avaliar qualquer expressão em Python e obter uma de duas respostas, `True` ou `False`.

Quando você compara dois valores, a expressão é avaliada e o Python retorna a resposta booleana:

Exemplo:

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

```
#Resultado:
True
False
False
```

Quando você executa uma condição em uma instrução `if`, o Python retorna `True` ou `False`.

Exemplo:

```
a = 200
b = 33
if b > a:
    print("b é maior que a")
else:
    print("b não é maior que a")
```

```
#Resultado:
b não é maior que a
```

Avalie valores e variáveis

A função `bool()` permite que você avalie qualquer valor e dê a você `True` ou `False` em troca.

Exemplo:

```
print(bool("Olá"))
print(bool(15))

#Resultado:
True
True
```

Alguns valores são falsos

De fato, não existem muitos valores que avaliam o booleano `False`, exceto valores vazios, tais como `()`, `[]`, `{}`, `""`, o número `0` e o valor `None`. E, claro, o valor `False` avalia para `False`.

Exemplo:

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

#Resultado:
False
False
False
False
False
False
False
False
#Note que todos retornam False
```

Funções podem retornar um booleano

Você pode criar funções que retornam um valor booleano.

Exemplo:

```
def funcao() :  
    return True  
print(funcao())
```

```
#Resultado:  
True
```