



IF e ELSE

Condições Python e instruções If

Python suporta as condições lógicas usuais da matemática:

- Igual a: `a == b`
- Diferente de: `a != B`
- Menor que: `a < b`
- Menor ou igual a: `a <= b`
- Maior que: `a > b`
- Maior ou igual a: `a > = b`

Essas condições podem ser usadas de várias maneiras, mais comumente em "instruções if" e loops.

Uma "declaração if" é escrita usando a palavra-chave `if`.

Exemplo:

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

```
#Resultado
b é maior que a
```

Indentação

Python depende de indentação (espaço em branco no início de uma linha) para definir o escopo no código. Outras linguagens de programação costumam usar chaves para esse propósito.

Elif

A palavra-chave `elif` é a forma de dizer "se as condições anteriores não eram verdadeiras, tente esta condição".

Exemplo:

```
a = 33
b = 33
if b > a:
    print("b é maior que a")
elif a == b:
    print("a e b são iguais")

#Resultado
a e b são iguais
```

Else

A palavra-chave `else` captura qualquer coisa que não seja capturada pelas condições anteriores.

Exemplo:

```
a = 200
b = 33
if b > a:
    print("b é maior que a")
elif a == b:
    print("a e b são iguais")
else:
    print("a é maior que b")

#Resultado:
a é maior que b
```



Nota: Você também pode ter um `else` sem `elif`

Short Hand If

Se você tiver apenas uma instrução para executar, pode colocá-la na mesma linha da instrução `if`.

Exemplo:

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

Short Hand If ... Else

Se você tem apenas uma instrução para executar, uma para if e outra para else, você pode colocar tudo na mesma linha:

Exemplo:

```
a = 2
b = 330
print("A") if a > b else print("B")
```



Nota: Essa técnica é conhecida como **Operadores Ternários** ou **Expressões Condicionais** .

Você também pode ter várias instruções else na mesma linha:

Exemplo:

```
a = 330
b = 330
print("A") if a > b else print("=") if a == b else print("B")
```

And

A palavra-chave `and` é um operador lógico, e é usado para combinar declarações condicionais.

Exemplo:

```
a = 200
b = 33
c = 500
if a > b and c > a: print("Ambas condições são Verdadeiras")
```

```
#Resultado:  
Ambas condições são Verdadeiras
```

Or

A palavra-chave `or` é um operador lógico e também é usada para combinar declarações condicionais.

Exemplo:

```
a = 200  
b = 33  
c = 500  
if a > b or a > c: print("Pelo menos uma das condições é Verdadeira") #a > b  
  
#Resultado:  
Pelo menos uma das condições é Verdadeira
```

Aninhando If

Você pode ter instruções `if` dentro de instruções `if`, isso é chamado de instruções aninhadas `if` (nested if).

Exemplo

```
x = 41  
if x > 10:  
    print("Acima de 10 ")  
if x > 20:  
    print("e também de 20,")  
if x > 50:  
    print("e também de 50.")  
else:  
    print("mas não acima de 50")  
  
#Resultado:  
Acima de 10  
e também de 20,  
mas não acima de 50
```

A declaração pass

Declarações `if` não podem estar vazias, mas se por algum motivo você tiver uma declaração `if` sem conteúdo, adicione a

declaração `pass` para evitar um erro.

```
a = 33
b = 200
if b > a:
    pass
print("O erro no if vazio foi contornado")

#Resultado:
O erro no if vazio foi contornado
```