



Cláudio OK

40fb3d2 · 3 semanas atrás



172 lines (123 loc) · 2.93 KB

Visualização

Código

Culpa

Cru



Operadores em Python

1. Operadores Aritméticos

Permite realizar operações matemáticas básicas entre valores numéricos.

- + (adição)
- - (subtração)
- * (multiplicação)
- / (divisão com resultado float)
- // (divisão inteira)
- % (resto da divisão)
- ** (exponenciação)

Exemplo:

```
a, b = 7, 3
print(a + b) # 10
print(a - b) # 4
print(a * b) # 21
print(a / b) # 2.333...
print(a // b) # 2
print(a % b) # 1
print(a ** b) # 343
```



2. Operadores de Atribuição

Combinam uma operação aritmética com a atribuição de valor à variável.

- `=`
- `+=`
- `-=`
- `*=`
- `/=`
- `//=`
- `%=`
- `**=`

Exemplo:

```
x = 5
x += 3 # equivalente a x = x + 3 → x passa a valer 8
x *= 2 # x = x * 2 → x passa a valer 16
```



3. Operadores de Comparação

Retornam valores booleanos (`True` ou `False`) comparando dois operandos.

- `==` (igualdade)
- `!=` (diferença)
- `>` (mais que)
- `<` (menos que)
- `>=` (maior ou igual)
- `<=` (menor ou igual)

Exemplo:

```
a, b = 10, 20
print(a == b) # False
print(a != b) # True
print(a < b)  # True
```



4. Operadores Lógicos

Permitem combinar expressões booleanas.

- and (E)
- or (OU)
- not (NÃO)

Exemplo:

```
idade = 25
saldo = 1500

# True se idade entre 18 e 65 E saldo >= 1000
aprovado = (idade >= 18 and idade <= 65) and (saldo >= 1000)
print(aprovado) # True
```



5. Operadores Bit a Bit

Operam diretamente sobre os bits de inteiros.

- & (E um pouquinho)
- | (OU um pouco)
- ^ (XOR um pouco)
- ~ (NÃO um pouco)
- << (deslocamento à esquerda)
- >> (deslocamento para a direita)

Exemplo:

```
x, y = 0b1010, 0b0101 # 10 e 5 em binário
print(bin(x & y))      # 0b0000
print(bin(x | y))      # 0b1111
print(bin(x ^ y))      # 0b1111
print(bin(~x))          # complemento de 1010 → -0b1011
print(bin(x << 2))      # 0b101000
print(bin(x >> 1))      # 0b0101
```



6. Operadores de Associação (Membros)

Verifiquem se um valor está presente em uma sequência (string, lista, tupla, etc.).

- `in`
- `not in`

Exemplo:

```
texto = "Olá, mundo!"  
print("mundo" in texto)      # True  
print(42 not in [1, 2, 3])  # True
```



7. Operadores de Identidade

Compare se duas variáveis referenciam o mesmo objeto na memória.

- `is`
- `is not`

Exemplo:

```
a = [1, 2, 3]  
b = a  
c = [1, 2, 3]  
  
print(a is b)      # True (mesma referência)  
print(a is c)      # False (objetos diferentes, apesar de iguais)
```



8. Precedência de Operadores

Determinar a ordem em que as operações são avaliadas:

1. Parênteses `()`
2. Exponenciação `**`
3. Multiplicação `*`, Divisão `/`, Divisão inteira `//`, Resto `%`
4. Adição `+`, Subtração `-`
5. Deslocamentos `<<`, `>>`
6. Um pouco `&`, `^`, `|`
7. Comparações
8. Operadores lógicos `not`, `and`, `or`
9. Atribuições `=`, `+=`, etc.

