

CURSO DE PYTHON



PARA INICIANTES





AULA 02: VARIÁVEIS E TIPOS DE DADOS



Variáveis são como "caixinhas" que guardam informações que você pode usar mais tarde. Cada caixinha tem um nome para você lembrar o que guardou nela. Por exemplo, você pode ter uma variável chamada idade que guarda o número 25.

Em Python, você **não precisa** dizer que tipo de dado está guardando. O Python descobre isso automaticamente. Isso significa que você pode simplesmente escrever `idade = 25`, e o Python sabe que é um número.

Tipos de dados são os diferentes tipos de informações que você pode guardar nessas "caixinhas". Os principais tipos de dados em Python são:

Inteiros(int) • Float • String(str) • Booleano(bool)

```
33 self.fingerprints = []
34 self.logdups = True
35 self.debug = debug
36 self.logger = logging
37
38 if path:
39     self.file = open
40     self.file.seek(
41     self.fingerprints
42
43 @classmethod
44 def from_settings(cls,
45 debug = settings.
46 return cls(job_di
47
48 def request_seen(self
49 fp = self.request
50 if fp in self.fi
51     return True
52 self.fingerprint
53 if self.file:
54     self.file.w
55
56 def request_fingerp
57     return request
```

REGRAS DE NOMENCLATURA

Para nomear nossas variáveis são implantadas algumas regras para que exista uma padronização desses nomes. As regras que são exigidas são as seguintes:


- Podem conter letras, números e underscores (_).
- Devem começar com uma letra ou underscore, mas nunca com um número.
- Sensíveis a maiúsculas e minúsculas (ex: Nome e nome são variáveis diferentes).
- Evitar palavras reservadas do Python (ex: def, class, etc.).

Exemplo de nomenclatura correta



```
nome_correto = "Ana"
```

Exemplo de nomenclatura incorreta



```
nomeincorreto123 = 30 #Inicia com  
número, inválido
```

Tipos de dados - introdução

O python suporta alguns **tipos de dados**, que são formas de classificar o tipo de valor que pode estar dentro de uma variável.

Eles ajudam o Python a entender **como operar com os dados e como armazená-los de maneira eficiente**. Cada tipo de dado define quais operações podem ser realizadas sobre ele, além de como o valor é armazenado na memória.

Int (Números inteiros)

Float (Números decimais)

String (Caracteres/Texto)

Bool (Verdadeiro ou Falso)

Vamos explorar os principais tipos usados no dia a dia da programação.



Operadores básicos

Soma (+): Adiciona dois valores

```
10 + 5 #resultado: 15
```

Multiplicação (*): multiplica dois valores

```
10 * 5 #resultado: 50
```

Subtração (-): Subtrai um valor de outro

```
10 - 5 #resultado: 5
```

Divisão (/): Divide um valor por outro

```
10 / 5 #resultado: 2
```

Operadores de comparação

Igual (==): Verifica se dois valores são iguais

```
10 == 5 #resultado: False
```

Diferente (!=): verifica se dois valores são diferentes

```
10 != 5 #resultado: True
```

Maior que (>): Verifica se um valor é maior que outro

```
10 > 5 #resultado: True
```

Menor que (<): Verifica se um valor é menor que outro

```
10 < 5 #resultado: False
```

inteiros (int)

Um número inteiro representa um valor numérico sem partes decimais, ou seja, números inteiros positivos, negativos ou zero.

Em Python, não há limite para o tamanho de um inteiro, desde que tenha memória suficiente.

Exemplos de inteiros: -10, 0, 42.

A dark-themed Python code editor window with a title bar containing three colored dots (red, green, yellow) and the Python logo in the top right corner. The code inside is `idade = 25 #exemplo de int`, where the number 25 is highlighted in pink.

```
idade = 25 #exemplo de int
```

Float

Um número de ponto flutuante (float) representa números reais com casas decimais. É usado para armazenar valores fracionários. Em Python, float suporta uma ampla faixa de valores com precisão limitada pelas casas decimais.

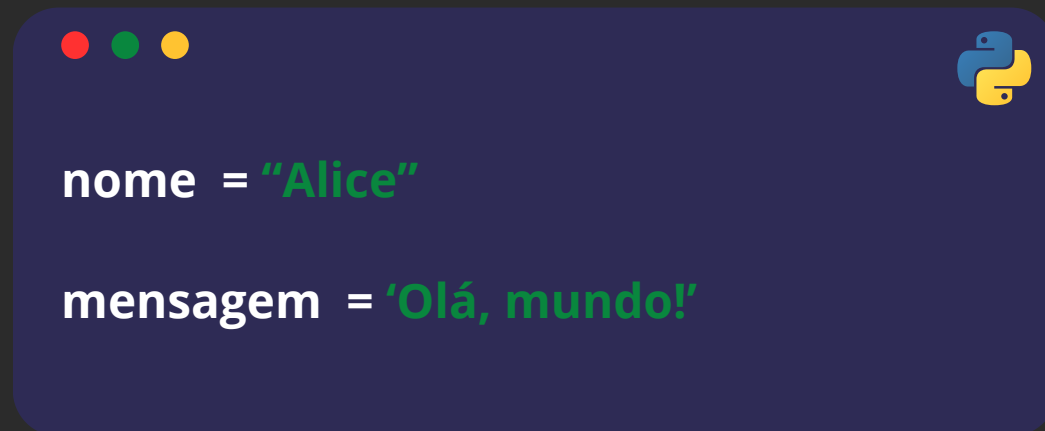
Exemplos de numero real: 3.14 ou -0.001.

A dark-themed Python code editor window with a title bar containing three colored dots (red, green, yellow) and the Python logo in the top right corner. The code inside is `preco = 19.99 #exemplo de float`, where the number 19.99 is highlighted in pink.

```
preco = 19.99 #exemplo de float
```

String

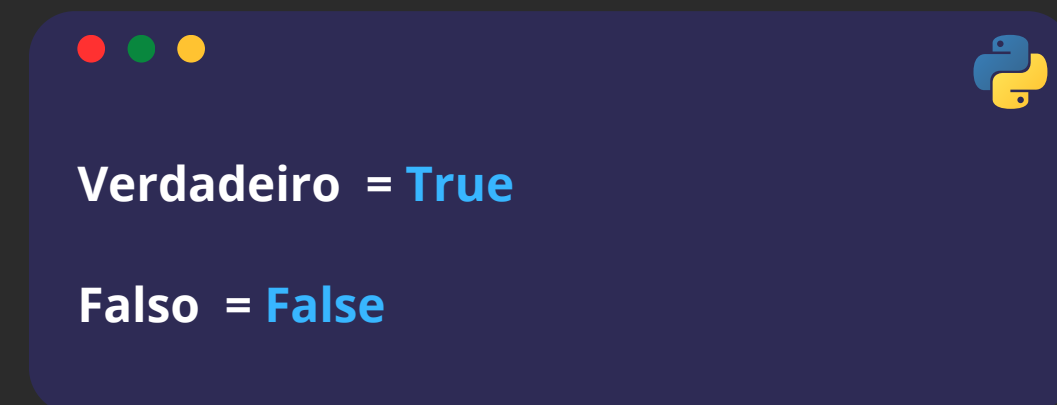
Uma string é uma sequência de caracteres usada para representar texto. Ela pode incluir letras, números, espaços e símbolos. Strings em Python são delimitadas por aspas simples ('texto') ou aspas duplas ("texto").

A dark-themed Python code editor window with a title bar containing three colored circles (red, green, yellow) and a Python logo icon in the top right corner. The code inside shows two lines: 'nome = "Alice"' and 'mensagem = 'Olá, mundo!''.

```
nome = "Alice"  
mensagem = 'Olá, mundo!'
```

Booleano (Bool)

O tipo booleano representa valores lógicos: Verdadeiro ou Falso. Ele é usado para expressar condições e resultados de comparações. Booleans são comumente usados em estruturas de controle, como condicionais, para decidir o fluxo do programa.


A dark-themed Python code editor window with a title bar containing three colored circles (red, green, yellow) and a Python logo icon in the top right corner. The code inside shows two lines: 'Verdadeiro = True' and 'Falso = False'.

```
Verdadeiro = True  
Falso = False
```


Lista

Uma lista em Python é uma coleção ordenada e mutável de itens, que podem ser de diferentes tipos (números, strings, etc.). As listas são definidas entre colchetes [], e os itens são separados por vírgulas.

- Mutável: elementos podem ser alterados.
- Indexada: acessamos os itens usando índices, começando em 0.
- Aceita elementos duplicados.



```
minha_lista = [1, 2, "Python", True]

frutas = ["maçã", "banana", "laranja"]


numeros = [1, 2, 3, 4, 5]
```

AGORA QUE VOCÊ
APRENDEU SOBRE
VARIÁVEIS E OS TIPOS DE
DADOS, AQUI ESTÁ UM
DESAFIO SIMPLES PARA
VOCÊ PRATICAR!

DESAFIO!

- Crie uma variável chamada nome e armazene nela o seu nome (como uma string).
- Crie uma variável chamada idade e armazene sua idade (como um número inteiro).
- Crie uma variável chamada altura e armazene sua altura (como um número float).
- Crie uma variável chamada é_estudante e armazene nela se você é ou não estudante (use True ou False).
- Exiba todas as variáveis no console usando o comando print.

Exemplo de como seu código deve ficar:



```
nome = "Ana"
idade = 25
altura = 1.68
é_estudante = True

print("Nome:", nome)
print("Idade:", idade)
print("Altura:", altura)
print("É estudante:", é_estudante)
```

***Desafio extra:** Mude os valores das variáveis e execute o código novamente. Isso te ajuda a entender como as variáveis funcionam na prática!

PARABÉNS!

Essa foi a sua SEGUNDA aula do curso de introdução a linguagem PYTHON.

Conteúdo:

- Variáveis armazenam dados e possuem nomes.
- Python tem vários tipos de dados: inteiros, floats, strings, booleanos, listas, entre outros.
- Compreender os tipos de dados é essencial para escrever código eficaz.

Próxima aula: Operadores e expressões.

