


# Introdução à linguagem Python

Python é uma linguagem de programação orientada a objetos. Sua essência foi criada para ter comandos e estruturas simples, de fácil leitura e compreensão. A seguir, conheça as principais características dessa linguagem.

 [Se desejar, baixe a versão em texto do objeto.](#)

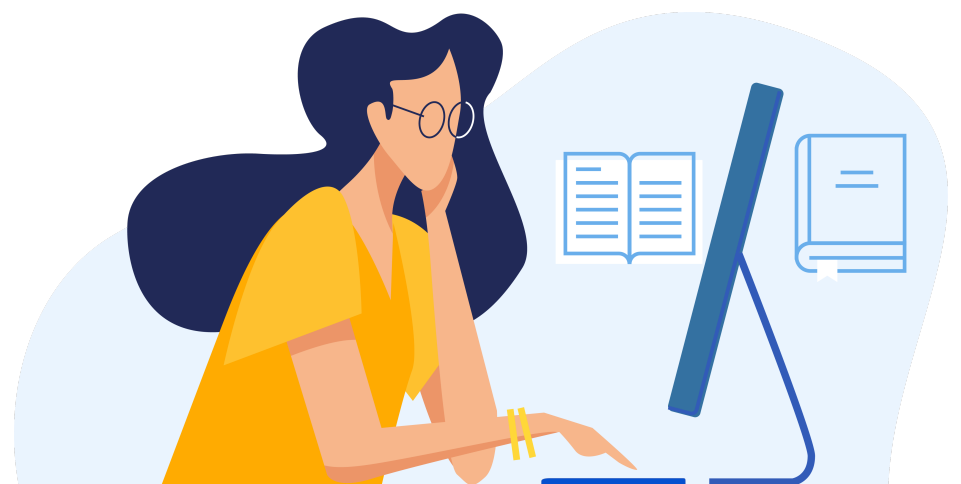
Agora que você já conheceu um pouco mais sobre a ferramenta Python vamos aprofundar nessa linguagem de programação.

## Variáveis e tipos básicos de dados em Python

A construção de um algoritmo envolve entrada, processamento e saída. Para que o processamento ocorra, é necessário um meio de armazenar valores temporariamente, razão pela qual surge o conceito de variáveis. Uma variável é um espaço alocado na memória RAM.

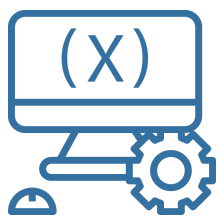
Muitas linguagens de programação possuem variáveis com tipo primitivo, como **int**, ou **string** – em Python esses tipos primitivos não existem.

Na linguagem Python, o tipo de variável é identificado no momento que se atribui um valor a ela. Portanto, uma variável refere-se a um valor.



Fonte: Shutterstock.

### Saiba mais



Na verdade, em Python não há a necessidade de definir estaticamente o tipo de variável, como em outras linguagens de programação.

Banin (2018) explica que “O modelo de dados do Python (*Python Data Model*, em inglês) adota como paradigma que todo dado em um programa escrito com Python é representado por um objeto”.

Todo objeto Python tem três aspectos:

- » Um identificador.
- » Um tipo.
- » Um conteúdo.

Os códigos a seguir exibem o resultado para três objetos: **int**, **str** e **float** (tipos de variáveis), sem declaração do tipo.

```
print('olá')  
# string  
print(10)  
# inteiro  
print(3.141592)  
# float
```



Fonte: Shutterstock.

## Exemplos

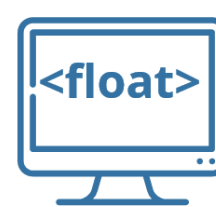
Exemplos de códigos que atribuem valores à variável.



```
>>> valor = 10  
>>> type(valor)  
<class 'int'>
```

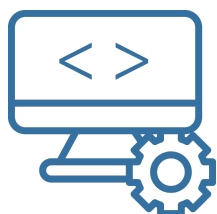


```
>>> valor = 'nome'  
>>> type(valor)  
<class 'str'>
```



```
>>> valor = 10.5  
>>> type(valor)  
<class 'float'>>
```

### Saiba mais



Em outras linguagens de programação, tais objetos seriam variáveis primitivas. Por serem objetos, tais tipos possuem métodos e comportamentos que outras linguagens não suportam – por exemplo, multiplicar uma string por um valor inteiro.

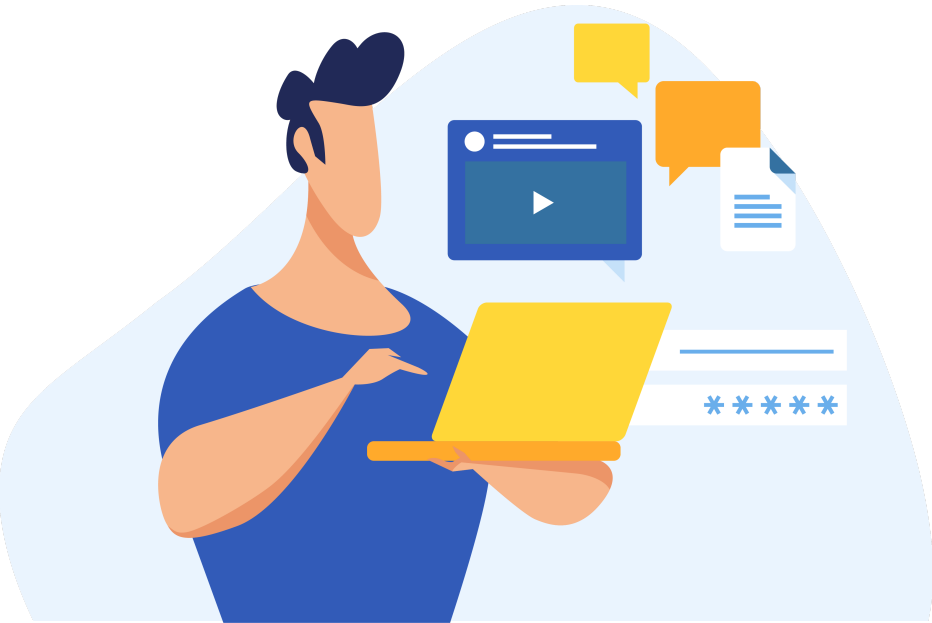
## Operadores numéricos

Como já destacamos, todo objeto em Python possui um identificador (o nome), um tipo e o conteúdo. Diferentes tipos de objetos vão suportar diferentes operações. Cada uma destas deve ser escolhida de acordo com o problema a ser resolvido.



Fonte: Shutterstock.

Os tipos numéricos, naturalmente, suportam operações matemáticas entre si, devendo ser respeitada a ordem de precedência dos operadores:

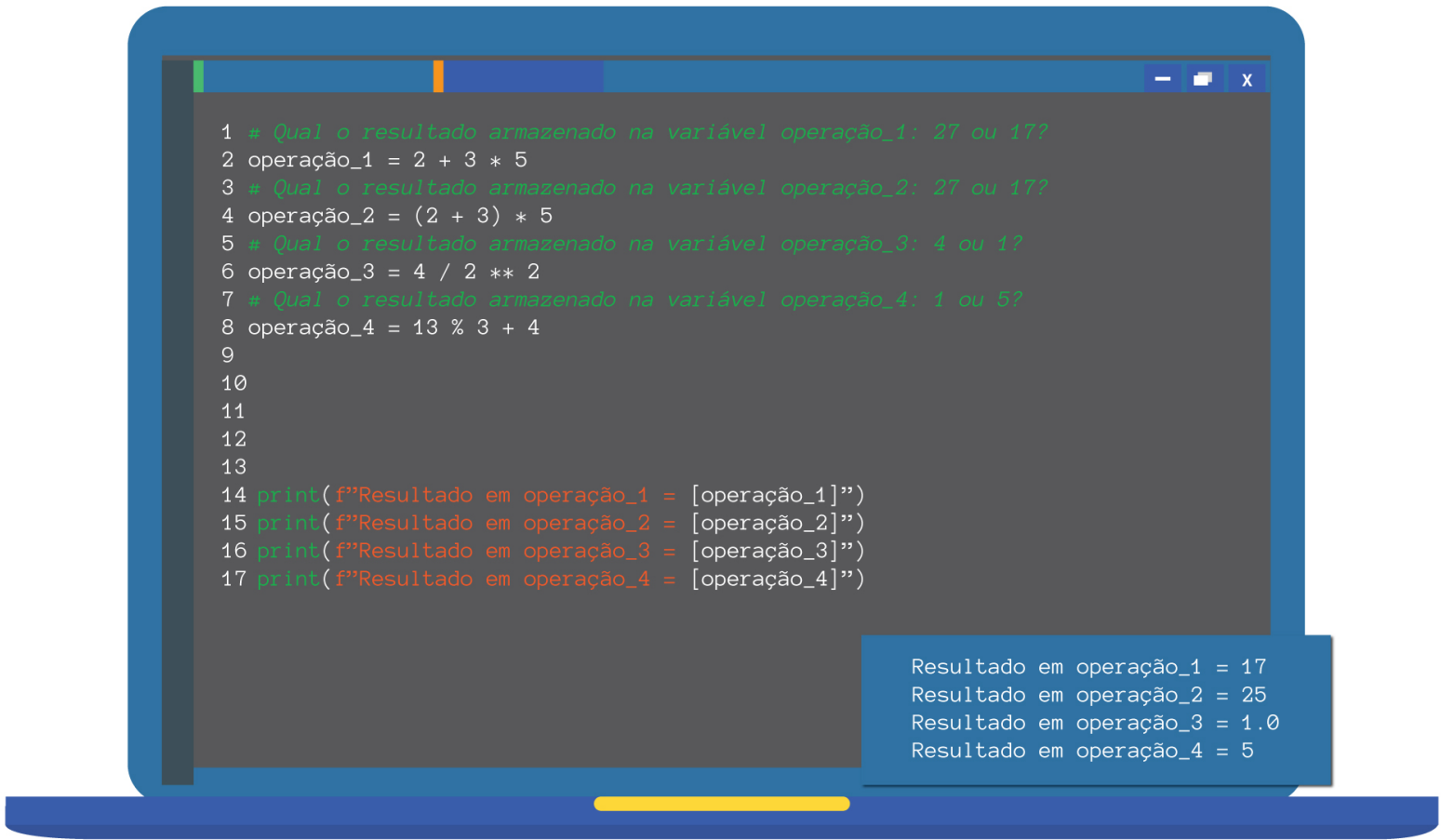


Fonte: Shutterstock.

- 1 Primeiro resolvem-se os parênteses, do mais interno para o mais externo.
- 2 Exponenciação.
- 3 Multiplicação e divisão.
- 2 Soma e subtração.

### Exemplos de operações matemáticas

Se a ordem de precedência não for respeitada, o resultado pode ser equivocado, conforme apresenta a ilustração seguinte.



Fonte: elaborada pela autora.

### Pesquise mais

Na página 17 do Capítulo 1 da obra: BANIN, S. L. Python 3 - conceitos e aplicações: uma abordagem didática. São Paulo: Érica, 2018, disponível na Biblioteca Virtual, o autor apresenta uma discussão sobre as versões da linguagem Python (Seção 1.2.2). É muito interessante realizar essa leitura, uma vez que não se deve mais utilizar as versões 2.X. Faça a leitura da seção e enriqueça seu relatório sobre essa importante linguagem de programação.



Fonte: Shutterstock.