

Para saber mais: História

História do Python

Python é uma linguagem de programação interpretada, orientada a objetos, de alto nível e com semântica dinâmica.

A simplicidade do Python reduz a manutenção de um programa. Python suporta módulos e pacotes (conteúdo que veremos ao longo do curso), que encoraja a programação modularizada e reuso de códigos.

É uma das linguagens que mais tem crescido devido sua compatibilidade (roda na maioria dos sistemas operacionais) e capacidade de auxiliar outras linguagens. Programas como Dropbox, Reddit e Instagram são escritos em Python.

Python também é a linguagem mais popular para análise de dados e conquistou a comunidade científica.

Breve história da linguagem

Python foi criada em 1990 por [Guido Van Rossum](https://pt.wikipedia.org/wiki/Guido_van_Rossum) (https://pt.wikipedia.org/wiki/Guido_van_Rossum) no Centro de Matemática Stichting na Holanda como uma sucessora da linguagem [ABC](https://pt.wikipedia.org/wiki/ABC_(linguagem_de_programa%C3%A7%C3%A3o)) ([https://pt.wikipedia.org/wiki/ABC_\(linguagem_de_programa%C3%A7%C3%A3o\)](https://pt.wikipedia.org/wiki/ABC_(linguagem_de_programa%C3%A7%C3%A3o))). Guido é lembrado como o principal autor de Python mas outros programadores ajudaram com muitas contribuições.

A linguagem ABC foi desenhada para uso de não programadores, mas logo de início mostrou certas limitações e restrições. A maior reclamação dos primeiros alunos não programadores dessa linguagem era a presença de regras arbitrárias que as linguagens de programação haviam estabelecido tradicionalmente - muita coisa de baixo nível ainda era feita e não agradou o público.

Guido então se lançou na tarefa de criar uma linguagem de script simples que possuísse algumas das melhores propriedades da ABC. Listas Python, dicionários, declarações básicas e uso obrigatório de indentação - conceitos que aprenderemos neste curso - diferenciam Python da linguagem ABC.

Você provavelmente já ouviu ou leu em algum lugar que Python é uma linguagem interpretada ou uma linguagem de script. Em certo sentido, também é verdade que Python é tanto uma linguagem interpretada quanto uma linguagem compilada. Um compilador traduz linguagem Python em linguagem de máquina - código Python é traduzido em um código intermediário que deve ser executado por uma máquina virtual conhecida como PVM (Python Virtual Machine).

O interpretador faz esta 'tradução' em tempo real para código de máquina, ou seja, em tempo de execução. Já o compilador traduz o programa inteiro em código de máquina de uma só vez e então o executa, criando um arquivo que pode ser rodado (executável). O compilador gera um relatório de erros (casos eles existam) e o interpretador interrompe a tradução quando encontra um primeiro erro.

Mais informações sobre Python?

Para mais informações, acesse a [documentação oficial do Python neste link](https://docs.python.org/pt-br/3/) (<https://docs.python.org/pt-br/3/>) ou a [apostila de Python da Caelum neste link](https://www.caelum.com.br/apostila-python-orientacao-objetos/o-que-e-python/#breve-historia) (<https://www.caelum.com.br/apostila-python-orientacao-objetos/o-que-e-python/#breve-historia>).

