



Introdução à Linguagem Python

Paradigmas de Linguagens de Programação

Rômulo Souza Fernandes
Ausberto S. Castro Vera

5 de setembro de 2022



Copyright © 2022 Rômulo Souza Fernandes e Ausberto S. Castro Vera

UENF - UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE DARCY RIBEIRO

CCT - CENTRO DE CIÊNCIA E TECNOLOGIA

LCMAT - LABORATÓRIO DE MATEMÁTICAS

CC - CURSO DE CIÊNCIA DA COMPUTAÇÃO

Primeira edição, Maio 2019



Sumário

1	Introdução	5
1.1	História da linguagem Python	5
1.2	Áreas de Aplicação da Linguagem	6
1.2.1	Big Data	6
1.2.2	Orientação a objetos	6
1.2.3	Outras	7
	Bibliografia	9
	Index	11



1. Introdução

O Python é uma linguagem orientada a objetos de alto nível que possui uma sintaxe simples e objetiva, assim colaborando para a fácil compreensão do código-fonte e permitindo que a linguagem seja produtiva. O Python contém várias estruturas de alto nível, como hora, data, dicionários, listas, complexos, entre outras estruturas, contém um amplo conjunto de módulos disponíveis para utilização, frameworks que podem ser acrescentados, possui ferramentas de outras linguagens atuais, como persistência, unidades de teste, geradores, introspecção e metaclasses, além de ter disponíveis diversas bibliotecas, como IPython, Matplotlib, mIPy, NumPy, Pandas, SciPy, ScraPy, entre outras bibliotecas conhecidas.

O Python é uma linguagem multiparadigma, suportando a programação orientada a objetos, modular e funcional. A linguagem Python foi criada na Holanda, no ano de 1990, por Guido van Rossum, no Instituto Nacional de Pesquisa para Matemática e Ciência da Computação. [Bor14]

A linguagem Python é de código aberto, porém o criador Guido van Rossum possui a função central de decidir a evolução da linguagem. O Python se popularizou e se tornou a linguagem de desenvolvimento de aplicações mais indicada para iniciantes, assim sendo aconselhada como primeira linguagem de programação. [Per16]

1.1 História da linguagem Python

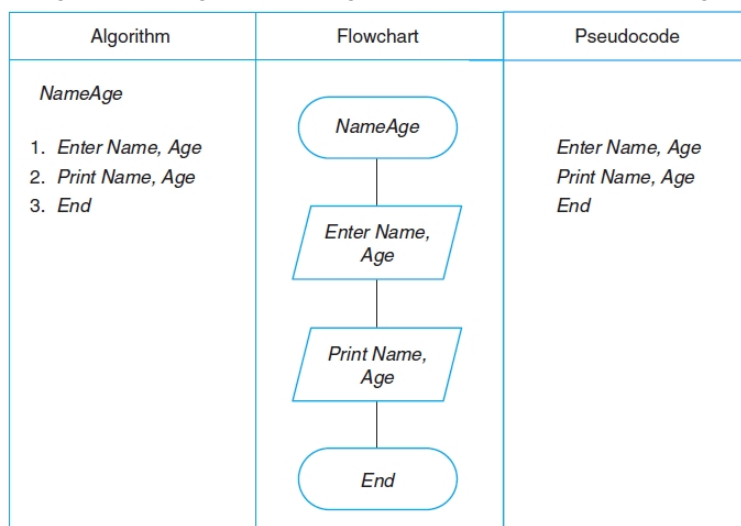
O intuito de Guido van Rossum era criar uma linguagem que pudesse suprir suas exigências, assim criando o Python, com base na linguagem ABC, mas solucionando as incoerências encontradas por ele na linguagem. O Python tinha como usuários principais os engenheiros e físicos.

A seguir alguns aspectos históricos da linguagem Python, baseados em [Per16] e [Bor14] :

- O Holandês Guido van Rossum foi o autor principal da linguagem Python. O autor trabalhava no CWI (Centrum Wiskunde & Informatica), localizada em Amsterdã na Holanda.
- O nome Python não veio da espécie de serpente e sim do seriado de comédia preferido do autor da linguagem, chamado Monty Python's Flying Circus.
- A versão 0.9.0 do Python foi lançado em 1991, incluindo manipulação de exceções, classes, listas e strings. Incluía também alguns aspectos de programação funcional como lambda, maps, filter e reduce.

- No ano de 1995, o autor da linguagem continuou seu trabalho sobre Python na Corporation for National Research Initiatives (CNRI) em Reston, Virginia, USA.
- Em Maio de 2000, Guido van Rossum e o grupo de desenvolvimento do Python se mudaram para BeOpen.com, assim formando a equipe BeOpen PythonLabs.
- A versão 1.6 do Python foi lançada em 5 de setembro de 2000.
- A versão 2.0 do Python foi lançada em 16 de outubro de 2000.
- A versão 3.0 do Python foi lançada em 3 dezembro de 2008.

Figura 1.1: Algoritmo, Diagrama de fluxo, e Pseudo-código



Fonte: (SH12, p. 93)

1.2 Áreas de Aplicação da Linguagem

Esta linguagem é utilizada e aplicada nas seguintes áreas: !!!!! As aqui mostradas são exemplos!!!

1.2.1 Big Data

Python fornece um grande número de bibliotecas para trabalhar em Big Data. Você também pode trabalhar – em termos de desenvolvimento de código – usando Python para Big Data muito mais rápido do que qualquer outra linguagem de programação. Esses dois aspectos estão permitindo que desenvolvedores em todo o mundo adotem o Python como a linguagem de escolha para projetos de Big Data.

Para obter um conhecimento aprofundado sobre o Python juntamente com seus vários aplicativos, você pode se inscrever para o treinamento on-line ao vivo do Python com suporte 24 horas por dia, 7 dias por semana e acesso vitalício.

É extremamente fácil lidar com qualquer tipo de dados em python. Vamos estabelecer isso com um exemplo simples. Você pode ver no instantâneo abaixo que o tipo de dados de 'a' é string e o tipo de dados de 'b' é inteiro. A boa notícia é que você não precisa se preocupar em lidar com o tipo de dados. Python já cuidou disso.

1.2.2 Orientação a objetos

A linguagem de programação Python é considerada orientada a objeto porque os valores são sempre armazenados em objetos. Em linguagens de programação diferentes de Python, os valores de certos tipos não são armazenados em entidades abstratas, como objetos, mas explicitamente na memória.

O termo classe é usado para se referir aos tipos cujos valores são armazenados em objetos. Como cada valor em Python é armazenado em um objeto, cada tipo Python é uma classe.

Essa técnica de manipular dados, onde os dados são armazenados em objetos e os métodos são invocados sobre objetos, é chamada programação orientada a objeto (POO). A POO é uma técnica poderosa para organização e desenvolvimento de código.

1.2.3 Outras



Referências Bibliográficas

- [Bor14] Luiz Eduardo Borges. *Python para desenvolvedores: aborda Python 3.3*. Novatec Editora, São Paulo, SP, Brasil, 2014. Citado na página 5.
- [Per16] Ljubomir Perkovic. *Introdução à computação usando Python: um foco no desenvolvimento de aplicações*. Rio de Janeiro, RJ, 2016. Citado na página 5.
- [SH12] Maureen Sprankle and Jim Hubbard. *Problem solving and programming concepts*. Pearson Education, Inc., 9 edition, 2012. Citado na página 6.

Disciplina: *Paradigmas de Linguagens de Programação 1970*

Linguagem: *LinguagemXYZabcd*

Aluno: *Rômulo Souza Fernandes*

Ficha de avaliação:

Aspectos de avaliação (requisitos mínimos)	Pontos
Introdução (Máximo: 01 pontos) <ul style="list-style-type: none"> • Aspectos históricos • Áreas de Aplicação da linguagem 	
Elementos básicos da linguagem (Máximo: 01 pontos) <ul style="list-style-type: none"> • Sintaxe (variáveis, constantes, comandos, operações, etc.) • Cada elemento com exemplos (código e execução) 	
Aspectos Avançados da linguagem (Máximo: 2,0 pontos) <ul style="list-style-type: none"> • Sintaxe (variáveis, constantes, comandos, operações, etc.) • Cada elemento com exemplos (código e execução) • Exemplos com fonte diferenciada (listing) 	
Mínimo 5 Aplicações completas - Aplicações (Máximo : 2,0 pontos) <ul style="list-style-type: none"> • Uso de rotinas-funções-procedimentos, E/S formatadas • Uma Calculadora • Gráficos • Algoritmo QuickSort • Outra aplicação • Outras aplicações ... 	
Ferramentas (compiladores, interpretadores, etc.) (Máximo : 1,0 pontos) <ul style="list-style-type: none"> • Ferramentas utilizadas nos exemplos: pelo menos DUAS • Descrição de Ferramentas existentes: máximo 5 • Mostrar as telas dos exemplos junto ao compilador-interpretador • Mostrar as telas dos resultados com o uso das ferramentas • Descrição das ferramentas (autor, versão, homepage, tipo, etc.) 	
Organização do trabalho (Máximo: 01 ponto) <ul style="list-style-type: none"> • Conteúdo, Historia, Seções, gráficos, exemplos, conclusões, bibliografia • Cada elemento com exemplos (código e execução, ferramenta, nome do aluno) 	
Uso de Bibliografia (Máximo: 01 ponto) <ul style="list-style-type: none"> • Livros: pelo menos 3 • Artigos científicos: pelo menos 3 (IEEE Xplore, ACM Library) • Todas as Referências dentro do texto, tipo [ABC 04] • Evite Referências da Internet 	
Conceito do Professor (Opcional: 01 ponto)	
<p style="text-align: right;">Nota Final do trabalho:</p>	

Observação: Requisitos mínimos significa a metade dos pontos