



Introdução à Linguagem Python

Paradigmas de Linguagens de Programação

Ricardo Willian Pontes da Silva

Ausberto S. Castro Vera

24 de setembro de 2022



Copyright © 2022 Ricardo Willian Pontes da Silva 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	Aspectos históricos da linguagem Python	5
1.2	Áreas de Aplicação da Linguagem	6
1.2.1	Big Data	6
1.2.2	Orientação a objetos	7
1.2.3	Processamento de imagens	7
	Bibliografia	9
	Index	11



1. Introdução

A linguagem de Programação Python, criada pelo holandês Guido Van Rossum em meados das décadas de 1980 à 1990, se caracteriza por ser do tipo *Very High Level Languages* (VHLL), ou seja, isso torna a linguagem em alto nível de abstração, porém não voltada apenas para a programação profissional, mas também se destacando em instituições de ensino e por indivíduos autodidatas que desejam ingressar no Python como primeira linguagem. Desta forma, Python também se destaca em detrimento das demais linguagens de programação existentes no mercado pela simplicidade em sua sintaxe, código aberto e também por ser executável em multi-plataformas.

Com a evolução da programação, os paradigmas das linguagens também foram se modificando com relação ao tempo, com a linguagem Python não foi diferente, tendo como base o paradigma orientado à objetos, mas também ainda permitindo a criação de códigos na forma procedural, abrangendo diferentes tipos de aplicações (de grandes e pequenas proporções) e desenvolvedores (Profissionais do nível básico ao avançado). Com isso, Python possibilita a migração de um paradigma para outro de forma natural e espontânea por parte do programador.

1.1 Aspectos históricos da linguagem Python

Python surgiu como uma possível solução para as deficiências das linguagens de programação mais aquecidas da época, como C, tendo como filosofia principal a fácil aplicação e intuitiva.

A seguir, menciona-se alguns aspectos históricos da linguagem Python, baseados em [\[Per16\]](#), [\[Sev15\]](#), [\[Lab\]](#):

- Criado por Guido van Rossum, um holandês de aproximadamente 26 anos em 1982, quando trabalhava no CWI (Centrum Wiskunde & Informatica), em Amsterdã, Holanda.
- Após pensar na linguagem de programação, sua nomeação veio com base no padrão exercido pela organização que seu criador atuava, que se dava por nomenclaturas de animais. Desta forma, Van Rossum, que era fã de um seriado de comédia da BBC *Monty Python's Flying Circus* nomeou-a como Python. .
- Python 0.9.0 foi lançado em 1991. Esta versão incluía manipulação de exceções, classes,

listas e strings. Também foram incluídos alguns aspectos de programação funcional: lambda, maps, filtros e reduce.

- Em 1995, Guido continuou seu trabalho sobre Python na Corporation for National Research Initiatives (CNRI) em Reston, Virginia, USA.
- Python 1.6 foi lançado no CNRI em
- No ano 2000, Guido e a equipe de desenvolvimento principal do Python foram para BeOpen.com para formar a equipe BeOpen PythonLabs.
- Python 2.0 foi lançado no ano 2000
- Em 2001, foi formada a PSF (Python Software Foundation), uma organização sem fins lucrativos que mantém e coordena o uso da linguagem. Atualmente, o PSF é suportado por grandes empresas como Google, Microsoft e Globo.com, que também utilizam Python em seus sistemas.
- Python 3.0 foi lançado em dezembro de 2008

Figura 1.1: Logo da Linguagem Python



Fonte: (VRDJ95)

1.2 Áreas de Aplicação da Linguagem

A linguagem Python, diferente da grande maioria em uso atualmente, não possui uma aplicação em específico, podendo ser encontrada no desenvolvimento *Backend/Frontend* como são divididas atualmente áreas mais fomentadas da computação, mas também sendo utilizada em meios como processamento de imagem, aplicações *Mobile*, *Data Science* e inúmeras outras.

Vale destacar também que, como a linguagem de programação Python é muito versátil, diversas companhias utilizam de sua estrutura para aplicar em seus serviços que serão fornecidos ao cliente. Podemos citar empresas que utilizam-se da linguagem Python: Dropbox, Yahoo, Intel, Cisco, HP, IBM e etc. Como citado por [dSS19]

A linguagem Python também pode ser integrada com diversas outras linguagens de programação, como Java, JavaScript e C. Desta forma, abrangendo ainda mais o seu uso e o perfil do profissional ao qual irá ser responsável pela aplicação.

1.2.1 Big Data

Com o avanço da tecnologia e a conectividade em praticamente todos os meios populacionais, é necessário um estudo e análise dos dados gerados por essa conexão. A área da tecnologia responsável por essa manipulação é chamada de análise de dados, que se tornará ainda mais necessária para que haja tomadas de decisões assertivas.

Por ser uma linguagem fortemente prototipada, de alto nível, orientada a objetos e dinâmica, o Python é uma das mais famosas linguagens para se trabalhar com manipulação de dados e desenvolvimento científico atuantes no mercado.[McK19]

Outro fator que contribui para o Python ser largamente utilizado nesse meio de aplicação quantitativa e analítica é a grande quantidade de bibliotecas existentes que auxiliam desde o

profissional mais iniciante ao mais experiente, com arquitetura robusta para suportar grandes aplicações. Como algumas dessas bibliotecas, podemos citar, NumPy, Pandas, Matplotlib dentre diversas outras

1.2.2 Orientação a objetos

Python é classificada como uma linguagem orientada a objetos, porém também é permissiva ao desenvolvimento de forma imperativa e funcional, ou seja, dizemos que Python é uma linguagem de multi-paradigmas. O paradigma orientado a objeto é baseado no conceito de classes, objetos, métodos e atributos, que visam representar o mundo real [Bar19].

Nesta linguagem de programação, tudo o que existe é um objeto. Atributos, classes, métodos, números e tudo o que existe em uma linguagem mesmo que não necessariamente orientada a objetos, se torna um objeto ao contrário da programação orientada a procedimentos.

Desta forma, a Orientação a Objetos na linguagem Python proporciona uma otimização do tempo, maior qualidade nos projetos e maior clareza de entendimento por parte do programador a qual está desenvolvendo a aplicação. Tais pontos positivos se dão uma vez que a estrutura da Python visa retratar o mais próximo possível de como as coisas se agrupam na vida real.

1.2.3 Processamento de imagens

Pela sua sintaxe de simples entendimento, robustez para grades aplicações e agilidade, o Python é altamente utilizado na área da computação responsável por processar e extrair informações de imagens. Esse processo se faz necessário nos dias atuais uma vez que a automação e inteligência dos aparelhos está cada vez mais presente no cotidiano, como, redares inteligentes, seleção de alimentos, dentre diversas outras.

Para realizar essas operações e análises, Python conta com uma grande quantidade de bibliotecas funcionais e uma significativa comunidade para auxiliar o programador. Podemos citar bibliotecas como, NumPy (*Numeric Python*), Scipy (*Scientific Python*), Matplotlib. Ambas têm como função principal operar funções matemáticas de maneira ágil e eficiente, como matrizes e vetores. Vale destacar também que, tais áreas que envolvem tomadas de decisão como processamento de imagem, inteligência artificial e etc, são tratadas como a nova fase da computação, onde aparelhos serão capazes de se assemelhar com os seres humanos em atividades rotineiras. Desta forma, podemos definir a linguagem Python como sendo uma linguagem do futuro e que possibilitará ao programador se manter atualizado [Bar19].



Referências Bibliográficas

- [Bar19] Felipe Barelli. *Introdução a visão computacional uma abordagem prática com python e opencv*. Casa do código, 2019. Citado na página 7.
- [dSS19] Rogério Oliveira da Silva and Igor Rodrigues Sousa Silva. *Linguagem de Programação Python*, volume 10. 2019. Citado na página 6.
- [Lab] Josue Labaki. *Introdução a Python Modulo A*. Citado na página 5.
- [McK19] Wes McKinney. *Python para análise de dados*. Novatec Editora, May 2019. Citado na página 6.
- [Per16] Ljubomir Perkovic. *Introdução à Computação usando Python: um foco no desenvolvimento de Aplicações*. LTC Livros Técnicos e Científicos Editora Ltda, Rio de Janeiro, 2016. Citado na página 5.
- [Sev15] C. Severance. Guido van Rossum: The Early Years of Python. *Computer*, 48(2):7–9, Feb 2015. Citado na página 5.
- [VRDJ95] Guido Van Rossum and Fred L Drake Jr. *Python tutorial*. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, Scotts Valley, CA, 1995. Citado na página 6.

Disciplina: Paradigmas de Linguagens de Programação 2022

Linguagem: Python

Aluno: Ricardo Willian Pontes da Silva

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