



TRILHA DE APRENDIZADO

Revisão Pythonica

- Lógica de programação
- Apresentação do Python
- Conceitos básicos da linguagem
- Instalação: IDE e Python 3.9
- Introdução as libs

Bibliotecas essenciais

- Logging
- Comunicação com API's
- Conexão com bancos de dados
- Entre outras libs p/ soluções criativas

Dicas e truques

- Onde e como procurar soluções
- Boas práticas
- Agilizando o desenvolvimento
- Funções prontas

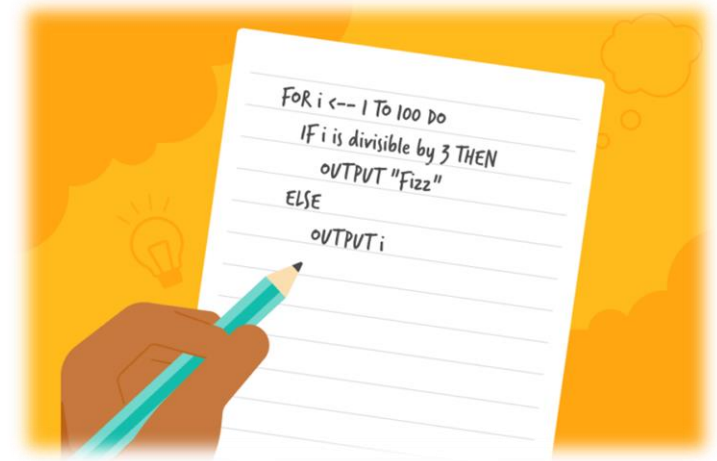
Afinal, o que é um algoritmo?



Sequência de passos
para atingir um objetivo

Representação de algoritmos

- Descrição narrativa
- Pseudocódigo
- Fluxograma
- Linguagens de programação

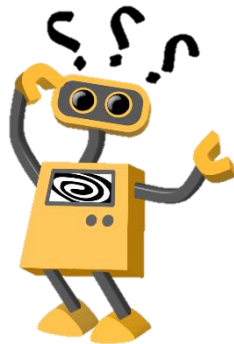


```
FOR i <-- 1 TO 100 DO  
  IF i is divisible by 3 THEN  
    OUTPUT "Fizz"  
  ELSE  
    OUTPUT i
```

Descrição narrativa

1. Ler 2 valores (x e y)
2. Verificar se x e y são iguais
3. Se x for igual a y, mostrar a msg "Valores Iguais!"
4. Se x for diferente de y, mostrar a msg "Valores Diferentes!"
5. Fim

* Algoritmo em **linguagem natural**

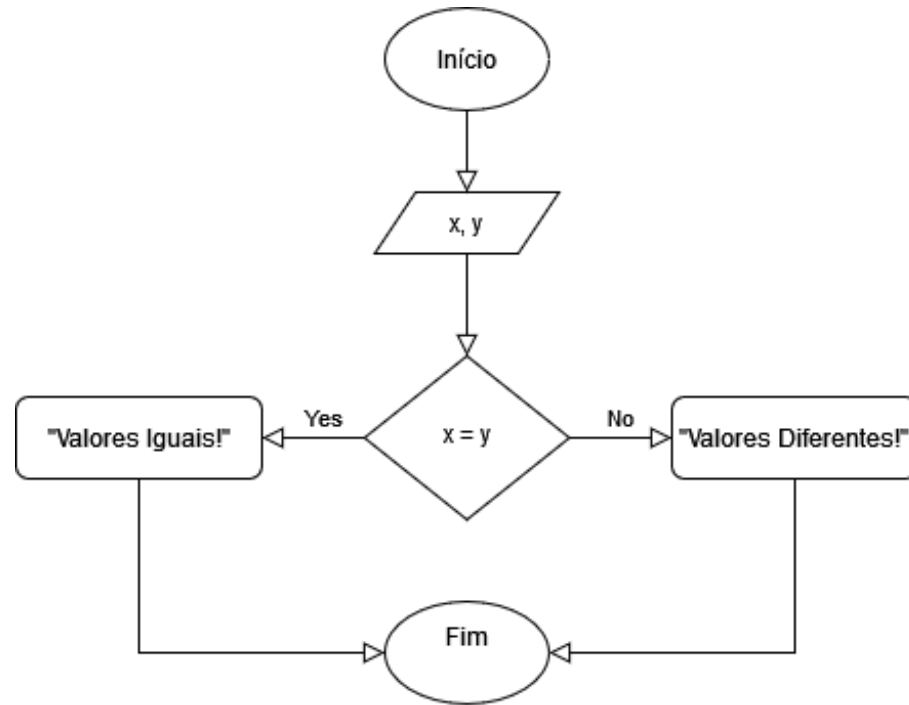


Pseudocódigo

1. inteiro x, y
2. leia(x, y)
3. se (x = y) então:
 mostrar("Valores Iguais!")
4. senão:
 mostrar("Valores diferentes!")
5. fim



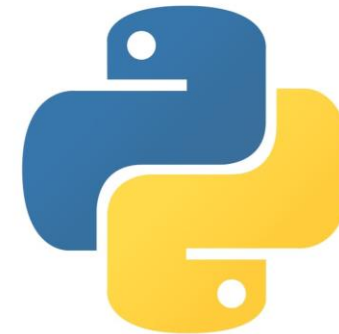
Fluxograma



* **Organizar** o raciocínio lógico

Linguagem de programação

1. $x, y = 1, 2$
2. `if x == y:`
3. `print("Valores Iguais!")`
4. `else:`
5. `print("Valores Diferentes!")`



* Possui **regras** (palavras-chave, símbolos e sequências específicas)

Python: Origem, objetivo e vantagens x desvantagens

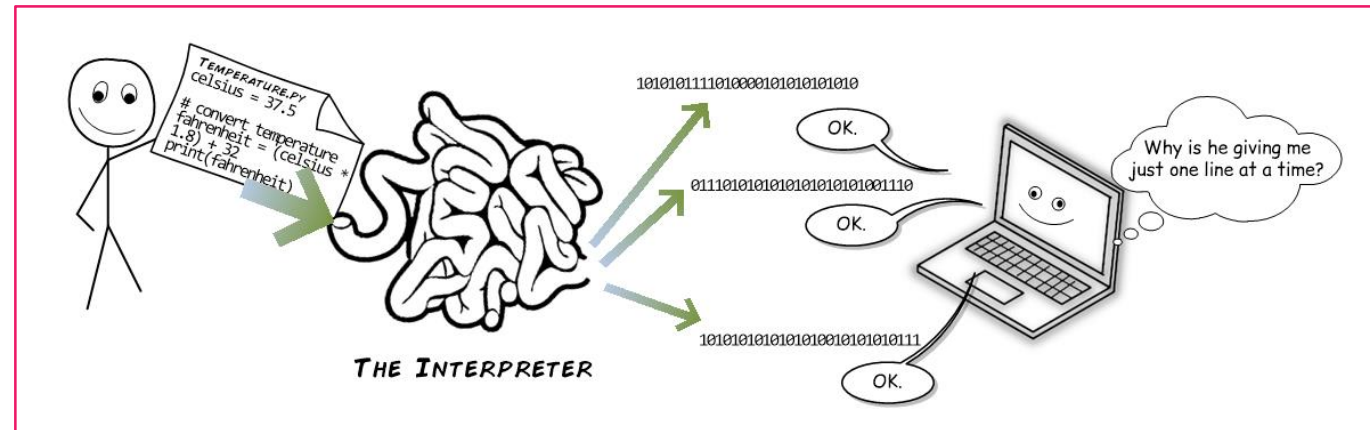
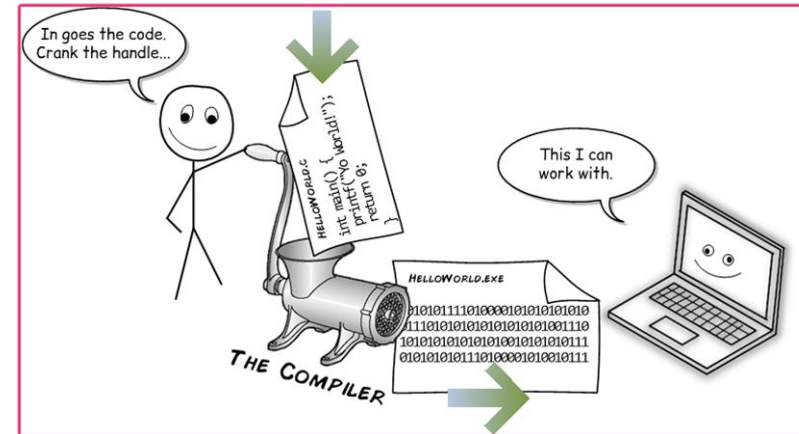
- Linguagem de **propósito geral**
- Multiplataforma, simples e intuitiva
- **Comunidade ativa** com atualizações constantes

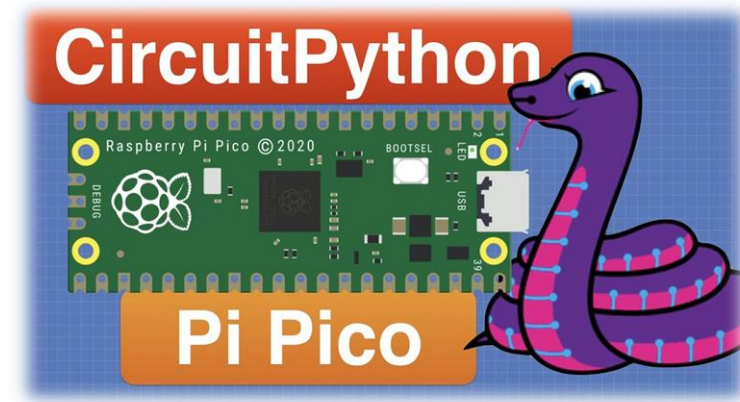
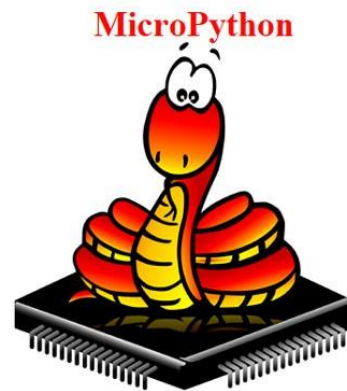


- Criada em **1982** por Guido Van Rossum
 - Inspirada na linguagem ABC
 - Homenagem ao **Mounty Python**

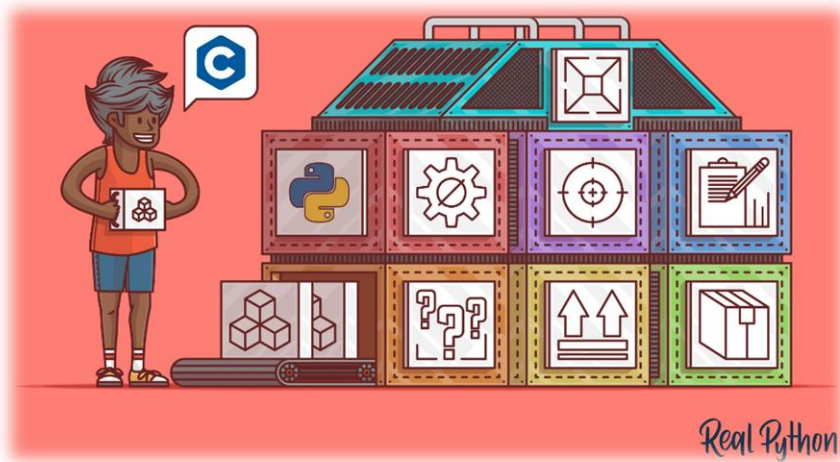
Desvantagem? Programação rápida e execução nem tanto...

- Linguagem interpretada e não compilada
- Verifica o tipo dos dados em tempo de execução





Melhorando o desempenho...



*Se necessário, não se limite a apenas uma linguagem!

Zen do Python – Tim Peters

Bonito é melhor que feio

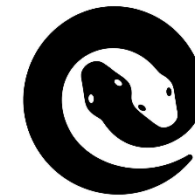
Explícito é melhor que implícito

Simples é melhor que complexo

Complexo é melhor que complicado

Legibilidade faz diferença

Agora é melhor que nunca





Lápis e papel na mão,
que chegou a hora da revisão

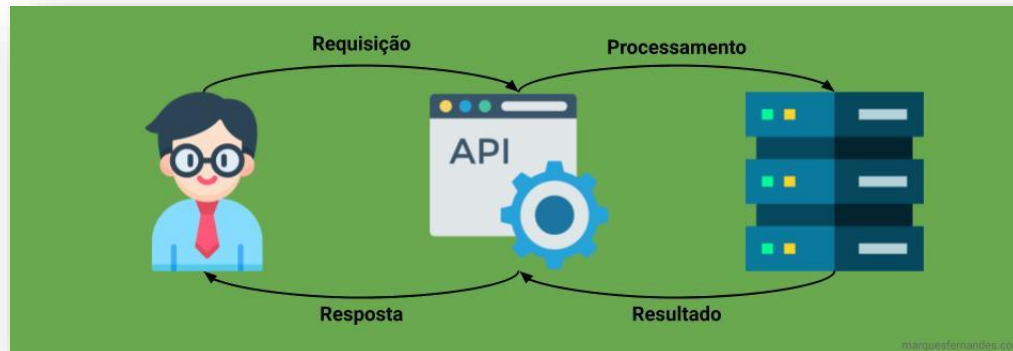




Bibliotecas

- Bibliotecas nativas
- Como instalar bibliotecas externas (pip)
- Bibliotecas mais usadas
- Como construir seus módulos
- Framework x Módulo





HTTP Status Codes



HTTP Methods



```
{
  "firstName": "Jonathan",
  "lastName": "Freeman",
  "loginCount": 4,
  "isWriter": true,
  "worksWith": ["Spantree Technology Group", "InfoWorld"],
  "pets": [
    {
      "name": "Lilly",
      "type": "Raccoon"
    }
  ]
}
```



Selenium WebDriver



Tesseract OCR

| Python



(x=243, y=1) ~ R:255 G:255 B:255



Estilo de código para trabalhar em equipe



Melhorando o desempenho do seu código







CONTEÚDO DA AULA
PARA DOWNLOAD