



# Introdução à Lógica de Programação

Portugol para Python.

# Seu primeiro Programa

- Em **Portugol** quando queremos mostrar algo na tela, “imprimir” para o usuário, usamos a palavra reservada “**ESCREVA**”.
- Em **Python** para executar a mesma ação usamos a palavra reservada “**PRINT**”

```
nenhum elemento  
1 print('Olá, turma!')
```

# Variáveis e Tipos de Dados

- Aprendemos que **variáveis** são alocações na **memória do computador**, onde podemos armazenar um valor de maneira **temporária** e manipular os dados de maneira **eficiente**.
- Tipos de dados, no decorrer do curso conhecemos alguns deles:
  - **Inteiro**
  - **Cadeia/String/Caracteres**
  - **Real/Ponto Flutuante**
  - **Real/Booleano**
- Existem alguns outros que podemos encontrar pelo caminho como:
  - **Lista/Arrays**
  - **Tupla**
  - **Dicionário/Objetos**
  - **Conjunto**

```
nenhum elemento
1  # Inteiros
2  x = 5
3
4  # Flutuantes
5  y = 3.14
6
7  # Strings
8  name = "Alice"
9
10 # Booleanos
11 is_student = True
12
```

# Operações básicas

- Com o **Python** é possível fazer operações matemáticas básicas:

```
nenhum elemento
1  a = 10
2  b = 3
3
4  # Adição
5  print(a + b)  # 13
6
7  # Subtração
8  print(a - b)  # 7
9
10 # Multiplicação
11 print(a * b)  # 30
12
13 # Divisão
14 print(a / b)  # 3.333...
15
16 # Divisão inteira
17 print(a // b)  # 3
18
19 # Módulo
20 print(a % b)  # 1
21
```

# Estruturas de Controle

- Python tem várias estruturas de controle para permitir a criação de loops e a execução de código condicional.
- **If, Elif e Else:**

nenhum elemento

```
1  idade = 18
2  if idade >= 18:
3      print("Você é um adulto.")
4  else:
5      print("Você é menor de idade.")
6
```

# Estruturas de Controle

- Condicionais com Mais de Uma condição.
- Para fazer condicionais com mais de uma condição, você pode utilizar operadores lógicos como ``and``, ``or`` e ``not``

- **and**: Todas as condições devem ser verdadeiras.
- **or**: Pelo menos uma das condições deve ser verdadeira.
- **not**: Inverte o valor da condição.

## AND

nenhum elemento

```
1  idade = 25
2  e_estudante = True
3
4  if idade >= 18 and e_estudante:
5      print("Você é um adulto e também é estudante.")
6  |
```

# Estruturas de Controle

- Condicionais com Mais de Uma condição.
- Para fazer condicionais com mais de uma condição, você pode utilizar operadores lógicos como ``and``, ``or`` e ``not``

- **and**: Todas as condições devem ser verdadeiras.
- **or**: Pelo menos uma das condições deve ser verdadeira.
- **not**: Inverte o valor da condição.

OR

nenhum elemento

```
1  idade = 25
2  e_estudante = False
3
4  if idade >= 18 or e_estudante:
5      print("Você é adulto ou é estudante (ou ambos).")
6  |
```

# Estruturas de Controle

- Condicionais com Mais de Uma condição.
- Para fazer condicionais com mais de uma condição, você pode utilizar operadores lógicos como ``and``, ``or`` e ``not``

- **and**: Todas as condições devem ser verdadeiras.
- **or**: Pelo menos uma das condições deve ser verdadeira.
- **not**: Inverte o valor da condição.

NOT

```
nenhum elemento
1  e_estudante = False
2
3  if not e_estudante:
4      print("Você não é um estudante.")
5
```



# Estruturas de Controle

- Condicionais com Mais de Uma condição.
- Para fazer condicionais com mais de uma condição, você pode utilizar operadores lógicos como ``and``, ``or`` e ``not``

- **and**: Todas as condições devem ser verdadeiras.
- **or**: Pelo menos uma das condições deve ser verdadeira.
- **not**: Inverte o valor da condição.

## Combinando AND, OR E NOT

nenhum elemento

```
1  idade = 25
2  e_estudante = False
3  esta_empregado = True
4
5  if (idade >= 18 and esta_empregado) or e_estudante:
6      print("Você é adulto empregado ou um estudante.")
7  else:
8      print("Você não é adulto empregado nem estudante.")
9
```

# Estruturas de Controle

- Condicionais com Mais de Uma condição.
  - Para fazer condicionais com mais de uma condição, você pode utilizar operadores lógicos como ``and``, ``or`` e ``not``
- **and**: Todas as condições devem ser verdadeiras.
  - **or**: Pelo menos uma das condições deve ser verdadeira.
  - **not**: Inverte o valor da condição.

```
nenhum elemento
1  nome = "Alice"
2  idade = 25
3  e_estudante = True
4  esta_empregado = False
5
6  if idade >= 18 and e_estudante:
7      status = "adulto e estudante"
8  elif idade >= 18 and esta_empregado:
9      status = "adulto e empregado"
10 elif idade < 18:
11     status = "menor de idade"
12 else:
13     status = "adulto"
14
15 mensagem = f"Olá, {nome}. Você é {status}."
16 print(mensagem)
17
```

# Estruturas de Controle

- Python tem várias estruturas de controle para permitir a criação de loops e a execução de código condicional.
- **For Loop:**

```
nenhum elemento
1  for i in range(5):
2      print(i)
3
```

# Estruturas de Controle

- Python tem várias estruturas de controle para permitir a criação de loops e a execução de código condicional.
- **While Loop:**

nenhum elemento

```
1  contador = 0
2  while contador < 5:
3      print(contador)
4      contador += 1
5
```