

# Introdução à Programação com Python

Bem-vindo(a) ao mundo da programação! Esta apostila foi feita para ser seu guia prático na jornada de aprendizado da linguagem **Python**. Vamos do zero ao seu primeiro programa de forma divertida e passo a passo.

---

## 1. Conhecendo os Materiais: Variáveis e Tipos de Dados

Em programação, os dados são a matéria-prima. Para guardá-los, usamos **variáveis**, que são como "caixas" na memória do computador. Cada caixa tem um nome (o nome da variável) e guarda um tipo de dado.

### Tipos de Dados Essenciais

- **int (inteiro):** Para números inteiros, sejam eles positivos ou negativos.  
Python  
idade = 25  
numero\_de\_alunos = 30
- **float (ponto flutuante):** Para números com casas decimais. Lembre-se de usar o ponto (.), não a vírgula.  
Python  
altura = 1.85  
preco = 9.99
- **str (string):** Para textos. Sempre coloque o texto entre aspas simples ( ' ') ou duplas ( " ").  
Python  
nome = "João da Silva"  
email = 'joao.silva@email.com'
- **bool (booleano):** Para valores lógicos que só podem ser **True** (verdadeiro) ou **False** (falso).  
Python  
esta\_chovendo = True  
usuario\_logado = False

## 2. As Ferramentas: Operadores e Funções de Interação

Com os materiais em mãos, precisamos das ferramentas certas para manipulá-los.

### Operadores Matemáticos

Estes são os símbolos que usamos para fazer cálculos.

Operador	Significado	Exemplo	Resultado
+	Adição	10 + 5	15
-	Subtração	10 - 5	5
*	Multiplicação	10 * 5	50
/	Divisão	10 / 5	2.0
**	Potência	2 ** 3	8
//	Divisão Inteira	10 // 3	3
%	Módulo (resto)	10 % 3	1

### Funções print(), input() e Conversão de Tipos

- **print():** Exibe informações na tela.
- **input():** Permite que o usuário digite algo. **Atenção:** O valor digitado é sempre uma string.
- **int(), float(), str():** Funções para converter entre os tipos de dados. Essenciais para fazer cálculos com dados do usuário.

Python

```
# Exemplo de conversão
peso = float(input("Digite seu peso em kg: "))
altura = float(input("Digite sua altura em metros: "))

imc = peso / (altura ** 2)

# Usando f-string para formatar a saída
print(f"Seu IMC é de {imc:.2f}")
```

### 3. Lógica e Decisões: if, else, elif

Para que seu código seja inteligente, ele precisa tomar decisões. Essa é a função da **lógica condicional**.

#### Operadores Relacionais e Lógicos

- **Relacionais:** ==, !=, >, <, >=, <= (comparam valores, retornam True ou False).
- **Lógicos:** and, or, not (combinam ou invertem resultados lógicos).

#### O Comando if

O **if** testa uma condição. Se ela for **True**, o bloco de código indentado abaixo é executado. Se for **False**, ele é ignorado.

Python

```
idade = 19
if idade >= 18:
    print("Você pode entrar.")
```

#### O Comando else

O **else** complementa o if. Ele executa um bloco de código se a condição do if for **False**.

Python

```
idade = 17
if idade >= 18:
    print("Você pode entrar.")
else:
    print("Entrada não permitida.")
```

## O Comando elif

O **elif** (abreviação de "else if") permite testar uma nova condição se a anterior falhou. É útil para criar uma sequência de verificações.

Python

```
nota = 8.5
if nota >= 9:
    print("Aprovado com louvor!")
elif nota >= 7:
    print("Aprovado.")
else:
    print("Reprovado.")
```

---

## 4. Repetições e Ciclos: while e break

Muitas tarefas em programação são repetitivas. Para isso, usamos laços (**loops**).

### O Comando while

O **while** executa um bloco de código **enquanto** uma condição for **True**. Para evitar loops infinitos, é crucial que algo no bloco de código mude a condição para False em algum momento.

Python

```
contador = 0
while contador < 3:
    print("Contando...")
    contador += 1 # contador = contador + 1
print("Fim da contagem.")
```

### O Comando break

O **break** é um comando que serve para **interromper a execução de um loop imediatamente**, mesmo que a condição do while ainda seja verdadeira.

Python

```
while True: # Loop infinito por padrão
    senha = input("Digite a senha: ")
    if senha == "123":
        print("Acesso liberado!")
        break # A senha está certa, saímos do loop
    else:
        print("Senha incorreta.")
```

---

## 5. Exercícios e Desafios

Hora de colocar a mão na massa! Resolva os exercícios abaixo para praticar o que aprendeu. Eles evoluem em complexidade.

1. **Conversor de Moedas Simples (Variáveis e input):**
  - Crie um programa que peça ao usuário o valor em reais (float).
  - Calcule o valor equivalente em dólar, sabendo que 1 dólar = R\$ 5,00.
  - Imprima o resultado.
2. **Par ou Ímpar (Operadores e if-else):**
  - Peça ao usuário para digitar um número inteiro.
  - Use o operador de módulo (%) para verificar se o número é par (o resto da divisão por 2 é 0).
  - Imprima se o número é "Par" ou "Ímpar".
3. **Maior de Idade (Aninhamento de if):**
  - Peça ao usuário o nome e a idade.
  - Se a idade for maior ou igual a 18, imprima: "Olá, [nome]! Você é maior de idade."
  - Se for menor, imprima: "Olá, [nome]! Você é menor de idade."
4. **Sistema de Login Básico (while e break):**
  - Defina um nome de usuário e uma senha corretos (ex: admin, 1234).
  - Use um loop while True para pedir ao usuário que digite o nome de usuário e a senha.
  - Se ambos estiverem corretos, imprima "Login bem-sucedido!" e use break para sair do loop.
  - Se estiverem incorretos, imprima "Login inválido. Tente novamente."
5. **Calculadora de Média (while e if):**

- Crie um programa que peça notas ao usuário.
- Use um while para continuar pedindo notas até que o usuário digite -1.
- Ao final, calcule e imprima a média das notas.

**6. Classificador de Triângulos (if-elif-else):**

- Peça ao usuário para digitar o comprimento de três lados de um triângulo.
- Use a lógica condicional para classificar o triângulo em:
  - "Equilátero" (todos os lados iguais)
  - "Isósceles" (dois lados iguais)
  - "Escaleno" (todos os lados diferentes)

**7. Contador de Vogais (while):**

- Peça ao usuário para digitar uma palavra.
- Use um loop while para percorrer a palavra (usando um índice).
- Conte quantas vogais (a, e, i, o, u) existem na palavra e imprima o total.

**8. Jogo de Adivinhação Melhorado (while, if-elif-else e break):**

- Defina um número secreto.
- Use um loop while com um contador de tentativas, limitando a 3 tentativas.
- A cada tentativa, diga se o palpite é maior ou menor que o número secreto.
- Se o usuário acertar, imprima "Parabéns, você acertou!" e use break.
- Se as 3 tentativas acabarem, imprima "Você perdeu. O número era [número secreto]."

**9. Validação de Dados (while):**

- Peça ao usuário que digite um número entre 1 e 10.
- Use um while para garantir que a entrada seja válida.
- Se o número não estiver entre 1 e 10, imprima uma mensagem de erro e continue pedindo até que a entrada seja correta.

**10. Calculadora de Média com Repetições (while, if-else e break):**

- Crie um programa que permita ao usuário calcular a média de uma turma.
- Peça a quantidade de alunos. Use um while para garantir que o número seja positivo.
- Em seguida, use outro while para pedir as notas de cada aluno (uma por uma).
- Calcule a média final da turma.