

Curso de Python para Iniciantes – Ensino por Módulos

◆ Módulo 1 – Introdução ao Python

Aula 1: O que é Programação?

Conceito:

Programar é criar instruções que o computador entende para realizar tarefas. Python é uma linguagem simples, ideal para quem está começando.

Exercício:

Escreva, com suas palavras, o que é programar e por que Python é uma boa linguagem para iniciantes.

Aula 2: Instalando Python e Escrevendo o Primeiro Código

Conceito:

- Instale o Python no site: <https://www.python.org>
- Use o IDLE (instalado com o Python) ou um editor como o VS Code.
- Primeiro programa:

```
python
CopiarEditar
print("Olá, mundo!")
```

Exercício:

Escreva um programa que mostre seu nome e uma saudação.

Aula 3: Variáveis e Tipos de Dados

Conceito:

- Variáveis armazenam informações.
- Tipos básicos:
 - int: números inteiros
 - float: números com ponto
 - str: textos
 - bool: verdadeiro/falso

```
python
CopiarEditar
```

```
nome = "Carlos"
idade = 30
altura = 1.75
estudante = True
```

Exercício:

1. Crie variáveis com seu nome, idade, altura e se é estudante.
 2. Mostre tudo com `print()`.
-

Aula 4: Entrada de Dados

Conceito:

- `input()` serve para o usuário digitar dados.
- O que é digitado é sempre uma string (`str`), então convertemos com `int()` ou `float()` quando necessário.

```
python
CopiarEditar
nome = input("Qual seu nome? ")
idade = int(input("Qual sua idade? "))
print("Olá", nome, "você tem", idade, "anos.")
```

Exercício:

Peça ao usuário nome, idade e altura. Mostre uma frase com essas informações.

Aula 5: Operadores Matemáticos

Conceito:

- Soma: +
- Subtração: -
- Multiplicação: *
- Divisão: /
- Divisão inteira: //
- Resto: %
- Potência: **

Exercício:

Peça dois números ao usuário e mostre:

- A soma
- A diferença
- O produto
- O quociente da divisão inteira
- O resto da divisão
- O resultado da potência

◆ Módulo 2 – Condições e Decisões

Aula 6: Estrutura condicional `if`

Conceito:

Usamos `if`, `elif` e `else` para tomar decisões no código.

```
python
CopiarEditar
idade = int(input("Qual sua idade? "))
if idade >= 18:
    print("Você é maior de idade.")
else:
    print("Você é menor de idade.")
```

Exercício:

Peça a idade e diga se a pessoa pode votar ($\text{idade} \geq 16$).

Aula 7: Operadores lógicos e relacionais

Conceito:

- Relacionais: `>`, `<`, `==`, `!=`, `>=`, `<=`
- Lógicos: `and`, `or`, `not`

```
python
CopiarEditar
idade = 20
tem_carteira = True
if idade >= 18 and tem_carteira:
    print("Pode dirigir")
```

Exercício:

Peça a idade e se a pessoa tem carteira (responda "sim" ou "não"). Diga se pode dirigir.

Aula 8: Projeto 1 – Calculadora Simples

Conceito:

Criar uma mini calculadora com 2 números e operação (+, -, *, /).

Exemplo:

```
python
CopiarEditar
n1 = float(input("Primeiro número: "))
n2 = float(input("Segundo número: "))
op = input("Operação (+, -, *, /): ")
```

```
if op == "+":
    print(n1 + n2)
elif op == "-":
    print(n1 - n2)
elif op == "*":
    print(n1 * n2)
elif op == "/":
    print(n1 / n2)
else:
    print("Operação inválida")
```

Exercício:

Implemente o código acima. Teste com diferentes valores.

◆ Módulo 3 – Laços de Repetição

Aula 9: Laço `while`

Conceito:

Repete um bloco enquanto uma condição for verdadeira.

```
python
CopiarEditar
contador = 1
while contador <= 5:
    print("Contando:", contador)
    contador += 1
```

Exercício:

Crie um programa que conte de 1 a 10.

Aula 10: Laço `for`

Conceito:

Usado para percorrer um intervalo de valores com `range()`.

```
python
CopiarEditar
for i in range(1, 6):
    print(i)
```

Exercício:

Peça um número e mostre a tabuada de 1 a 10 com `for`.

Aula 11: Projeto 2 – Jogo de Adivinhação

Conceito:

Use `while` e `if` para criar um jogo simples.

```
python
CopiarEditar
import random
numero = random.randint(1, 10)
palpite = 0

while palpite != numero:
    palpite = int(input("Adivinhe o número (1-10): "))
    if palpite < numero:
        print("Mais...")
    elif palpite > numero:
        print("Menos...")
print("Acertou!")
```

Exercício:

Implemente e jogue.

◆ Módulo 4 – Listas e Funções

Aula 12: Listas

Conceito:

Estruturas que guardam vários valores.

```
python
CopiarEditar
frutas = ["maçã", "banana", "uva"]
print(frutas[0]) # maçã
```

Exercício:

Crie uma lista com 3 filmes favoritos. Mostre cada um na tela.

Aula 13: Funções

Conceito:

Funções organizam o código e podem ser reutilizadas.

```
python
CopiarEditar
def saudacao(nome):
    print("Olá, ", nome)

saudacao("Ana")
```

Exercício:

Crie uma função que receba dois números e retorne a soma.

Aula 14: Projeto 3 – Lista de Compras

Conceito:

Combinar `lista`, `input()` e `while`.

Exemplo:

```
python
CopiarEditar
compras = []
while True:
    item = input("Digite um item (ou 'sair'): ")
    if item == "sair":
        break
    compras.append(item)

print("Sua lista:", compras)
```

Exercício:

Implemente e personalize com emojis ou frases divertidas.