# Aula 05 – Funções II: Avançando com Argumentos

Explorando técnicas avançadas de programação funcional em Python

```
.....
SOINGMANUAGO
                                                              E NORIGERIA
                                                                                                        BARRES BALL DEMONSTRA
                                                                                                                                      DORE ROOF SIDETON A
                                                                                                                                                                                THORNOCON TOWNS PROPERTIES OF THE DESPONDED ON LE LED HOLD HOLD TO A PERBER! DO
                                                                                                           direce el mos aguesta con contrara de un contrara de la contrara del contrara del contrara de la contrara del la contrara de la contrara del la contrara de 
                                                                                                                                                                      THE ROLL OF THE PARTY OF
                                                                                                                                                                   ALBERTAND OF MANAGERSARY BARREST
                                                                                                                                            ON SECONDERIES BUNDATED EMPIRE (BY CORRECTION 1/1)
                                                                                                     PERSONAL PA
                                                                                     THE CONTROL OF THE PARTY OF THE
                                                                                                                               SEAST VOR COLLEGE SEAST SUPPLY SOUS MAN STATE OF THE SEAST S
                                                                                                                     COLUMN TO THE SOUR CONTRACTOR SOURCE TO STAN THE SOURCE
                                                                                                                         - UNDINCOLDIGINA
                                                                                                                     estura punci andid
                                                                                                        SE AMORIGOGOGO HOR SUNETIMENTADO BRENTA DO UN GERRO NO REL ENTRAS Y INVESTIGIO
                                                              CHEODE I PADO
```

# Objetivos da Aula

### **Aprofundar Funções**

Focar em argumentos e modularização para código mais organizado

## Organizar Código

Estruturar programas em módulos funcionais reutilizáveis

## **Explorar Argumentos**

Dominar argumentos nomeados, padrão e funções como parâmetros

#### Prática Aplicada

Desenvolver menus funcionais e simulador bancário completo

# Revisão Rápida

# O que já sabemos sobre funções:



Declaração com def

Definindo funções usando a palavra-chave def



**Argumentos posicionais** 

Passando valores na ordem correta dos parâmetros



Retorno com return

Devolvendo valores processados pela função



Escopo local e global

Compreendendo onde as variáveis existem no código

```
II that weston(): {
 aredryttonin
    a nationed argumetts
    wwitinn (etenta lot())
      catton sky);
   (en fall servien darla);
    wattl natio when linel);
    f gol lexestmind)
    orie concliten andd lebt;
I en decembera angune?
```

# **Argumentos Nomeados**

# Vantagens principais:

- Permitem passar valores explicitamente pelo nome do parâmetro
- Aumentam significativamente a clareza do código
- Reduzem erros de ordem dos argumentos

# Exemplo prático:

```
def saudacao(nome, mensagem):
    print(f"{mensagem}, {nome}!")

# Usando argumentos nomeados
saudacao("Ana", "Bom dia")
saudacao(mensagem="Boa tarde", nome="João")
```

Dica: Use argumentos nomeados quando tiver muitos parâmetros!



# Argumentos Padrão

## Definição

Valores automáticos quando o usuário não passa o argumento específico

#### Flexibilidade

Tornam a função mais versátil e fácil de usar em diferentes situações

# Exemplo em ação:

```
def boas_vindas(nome, curso="Python"):
    print(f"Bem-vindo(a) {nome} ao curso de {curso}!")
```

# Usando valor padrão boas\_vindas("Carlos")

# Sobrescrevendo o padrão boas\_vindas("Maria", "JavaScript")

# Funções como Argumentos

## Conceito poderoso:

Em Python, funções também podem ser passadas como parâmetros para outras funções. Isso permite:

- Maior modularidade no código
- Reuso inteligente de lógica
- Programação mais dinâmica

# Exemplo prático:

```
def aplicar_operacao(a, b, operacao):
    return operacao(a, b)

# Usando funções lambda
print(aplicar_operacao(10, 5, lambda x, y: x + y))
print(aplicar_operacao(10, 5, lambda x, y: x * y))
```



# Modularização com Funções



#### **Blocos Reutilizáveis**

Separar o código em pequenos módulos funcionais que podem ser usados em diferentes partes do programa



# Facilita Manutenção

Evita repetição desnecessária e torna muito mais fácil fazer alterações e correções no código



#### Melhora Legibilidade

Código organizado em funções específicas é mais fácil de entender e debugar

**Exemplo prático:** Criar funções como menu\_principal(), depositar(), sacar() em vez de concentrar tudo no mesmo bloco gigante.

# utild by famp lines too, come laws that last t forfooms then titer); too coules fam let to taker tool; int the come not); time of Left. cloud in from farmental time is the commercing time (uny!) ]); tion best then they are time here to enhance (b, t) the Beam ten liner desition (built, 1), Sem a miertari. inction simu tin bert erriver: V/ on toll in Serviced head erized (); is inectage coeffect to they').

# **Atividades Práticas**

01

## Função de Boas-vindas

Criar uma função que receba um nome e mostre uma mensagem personalizada de boas-vindas

02

#### **Calcular Quadrado**

Criar uma função que calcule o quadrado de um número (com argumento e retorno claro)

03

## Saudação Padrão

Criar uma função com argumento padrão que exiba uma saudação (ex.: "Olá, visitante!")

04

#### Soma de Números

Criar uma função que receba dois números e exiba a soma deles de forma clara

# Prática Guiada

# Atividades em sala de aula:

1

#### Menu Funcional

Criar um sistema de menu interativo usando funções organizadas e modulares

2

## **Operações Matemáticas**

Implementar operações básicas: soma, subtração, multiplicação com interface amigável

3

#### Simulador Bancário

Desenvolver um sistema completo com: depositar valor, sacar valor, verificar saldo



# Resumo da Aula:

## **Argumentos Avançados**

Dominamos argumentos nomeados e padrão para código mais flexível

## Funções como Parâmetros

Aprendemos a passar funções como argumentos para maior modularidade

## Modularização

Compreendemos a importância de organizar código em blocos funcionais

## Aplicação Prática

Criamos menus funcionais e simuladores bancários reais

