

INSTRUÇÕES DE CHAMADA A ROTINAS: FUNÇÕES

O fluxo de execução pode "saltar" para outra parte do código através de chamadas de funções, retornando ao ponto inicial após a execução da função.

► Uma **função** em Python, também chamada de **subprograma**, é um bloco de código reutilizável projetado para executar uma tarefa específica. As funções permitem modularizar o código, tornando-o mais estruturado, organizado e fácil de manter. Além disso, economizam tempo de programação por serem reutilizáveis.

```
def nome_da_funcao(parametros):  
    # Corpo da função  
    # Código que executa a tarefa  
    return resultado # Opcional
```

Componentes de uma Função

1. **Palavra-chave def** : Indica o início da definição de uma função.
2. **Nome da função** : Deve ser único e seguir as regras de identificadores do Python.
3. **Parâmetros (opcionais)** : Variáveis que recebem valores quando a função é chamada.
4. **Dois pontos** : : Indicam o fim do cabeçalho da função.
5. **Corpo da função** : Bloco de código indentado que executa a tarefa desejada.
6. **return (opcional)**: Encerra a execução da função e pode retornar um valor ao chamador.

EXEMPLO 1:

```
def saudacao(nome):  
    #Imprime uma saudação personalizada.  
    print(f"Olá, {nome}!")  
# Para usar essa função, é necessário passar um argumento (nome):  
saudacao("Maria") # Saída: Olá, Maria!
```

EXEMPLO 2:

```
def calcular_area(largura, altura=1):  
    """Calcula a área de um retângulo."""  
    return largura * altura  
  
# Chamadas de função  
area1 = calcular_area(5, 10) # Argumentos posicionais  
area2 = calcular_area(largura=5, altura=10) # Argumentos nomeados  
area3 = calcular_area(5) # Usa o valor padrão para altura (que é 1)  
  
print(area1, area2, area3) # Saída: 50 50 5
```

EXEMPLO 3:

```
def maximo(num1, num2):  
    if num1>num2:  
        return num1  
    else:  
        return num2  
  
# Chamadas de função  
print(maximo(2,3))  
print(maximo(9,7))
```

EXEMPLO 4:

```
def soma_dois_numeros():  
    num1 = float(input("Digite o primeiro número: "))  
    num2 = float(input("Digite o segundo número: "))  
    resultado = num1 + num2  
    print(f"A soma de {num1} e {num2} é {resultado}")  
  
# Chamada da função  
soma_dois_numeros()
```

VARIÁVEIS LOCAIS E GLOBAIS NUMA FUNÇÃO

- **Variável global**

é vista pelo programa principal e por todos os módulos (funções, métodos, procedimentos ...)
pertencentes a ele;

- ⇒ Criada fora de qualquer função.
- ⇒ Podem ser usada em todo o programa.
- ⇒ É acessível dentro e fora das funções.

- **Variável local**

é visível ao módulo em que ela foi declarada e aos módulos hierarquicamente inferiores.

- ⇒ Criada dentro de uma função
- ⇒ Só pode ser usada dentro dessa função
- ⇒ Deixa de existir quando a função termina

Podemos transformar uma variável local (dentro de uma função) numa variável global que possa ser usada pelo restante programa.

Exemplo:

Versão 1

```
num1 = 10
def quadrado():
    num1 = 3
    print(num1*num1)
    num1 += 2

# O que devolve o python nestas três instruções:
print(num1)
quadrado()
print(num1)
```

```
# devolve 10
# devolve 9
# devolve 10
```

Versão 2

```
num1 = 10
def quadrado():
    global num1
    print(num1*num1)
    num1 += 2

# O que devolve o python nestas três instruções:
print(num1)
quadrado()
print(num1)
```

```
# devolve 10
# devolve 100
# devolve 12
```