

Rotinas e Funções

{

- Interactive Help
- Docstrings
- Argumentos opcionais
- Escopo de variáveis
- Retorno de resultados

}



Rotinas e Funções

{ Interactive Help

O "Interactive Help" (Ajuda Interativa) é uma ferramenta integrada muito útil para obter informações sobre os módulos, classes, funções, keywords e outros aspectos da língua Python diretamente no terminal.



Rotinas e Funções

{ Interactive Help

Documentação da Função/Classe/Módulo: Mostra a string de documentação (docstring) que fornece uma visão geral do que a função/classe/módulo faz.

Assinatura da Função/Método: Apresenta a assinatura da função ou método, que inclui o nome da função, parâmetros e valores padrão, se houver.

Informações Adicionais: Para classes, por exemplo, pode mostrar a lista de métodos disponíveis.

}



Rotinas e Funções

{ Docstrings

Docstrings em Python são strings literais que aparecem logo após a definição de uma função, método, classe ou módulo e são usadas para documentar o código.

Elas são acessíveis de dentro do programa, o que as torna muito úteis para entender e utilizar corretamente o código que se está a trabalhar.

As docstrings são uma parte importante do estilo de escrita idiomático de Python e são encorajadas como uma prática de documentação.

}

Rotinas e Funções

{ Docstrings

```
def soma(a, b):  
    """  
    -> Calcula e imprime a soma de dois números.  
    param: a -- primeiro número  
    param: b -- segundo número  
    return: Sem retorno.  
    """  
    print(a + b)
```

}

Rotinas e Funções

{ Parâmetros opcionais

```
def soma(a, b, c):  
    print(a + b + c)
```

```
soma(2, 4, 6)
```

```
soma(2, 4)
```

```
}
```

Rotinas e Funções

{ Parâmetros opcionais

```
def soma(a, b, c=0):  
    print(a + b + c)
```

```
soma(2, 4, 6)
```

```
soma(2, 4)
```

}

Rotinas e Funções

{ Parâmetros opcionais

```
def soma(a=0, b=0, c=0):  
    print(a + b + c)
```

```
soma(2, 4, 6)
```

```
soma(2, 4)
```

```
soma()
```

}

Rotinas e Funções

{

Escopo de variáveis

```
def escopo(b):
    b+=5
    c=3
    print(f'A dentro vale {a}')
    print(f'B dentro vale {b}')
    print(f'C dentro vale {c}')
```

```
a = 6
escopo(a)
```

escopo local

c 3

b 11

a 6

}



escopo global

Rotinas e Funções

{

Escopo de variáveis

```
def escopo(b):
    b+=5
    c=3
    print(f'A dentro vale {a}')
    print(f'B dentro vale {b}')
    print(f'C dentro vale {c}')
```

```
a = 6
escopo(a)
print(f'B fora vale {b}')
```

escopo local

c 3

b 11

a 6

}

escopo global

Rotinas e Funções

{

Escopo de variáveis

```
def escopo(b):  
    a=8  
    b+=5  
    c=3  
    print(f'A dentro vale {a}')  
    print(f'B dentro vale {b}')  
    print(f'C dentro vale {c}')
```

```
a = 6  
escopo(a)  
print(f'A fora vale {a}')
```

escopo local

c 3

b 11

a 8

a 6

}

Rotinas e Funções

{

Escopo de variáveis

```
def escopo(b):
    global a
    a=8
    b+=5
    c=3
    print(f'A dentro vale {a}')
    print(f'B dentro vale {b}')
    print(f'C dentro vale {c}')
```

```
a = 6
escopo(a)
print(f'A fora vale {a}')
```

escopo local

c 3

b 11

a 8

}

escopo global

Rotinas e Funções

{ Retorno de resultados

```
def soma(a=0, b=0, c=0):  
    s = a + b + c  
    print(f' A soma é {s}')  
  
soma(2, 4, 6)  
  
soma(2, 4)  
  
soma()
```

Esta função apenas pode realizar uma determinada ação como mostrar algo no ecrã de forma dinâmica, mas não pode passar valores.

Frequentemente queremos usar o resultado de uma função como entrada para outra, ou guardá-la numa variável.

Este tipo de funções não podem ser integradas com outras, pois o resultado da função é uma outra função ou ação e não pode ser passado diretamente como argumento ou valor.

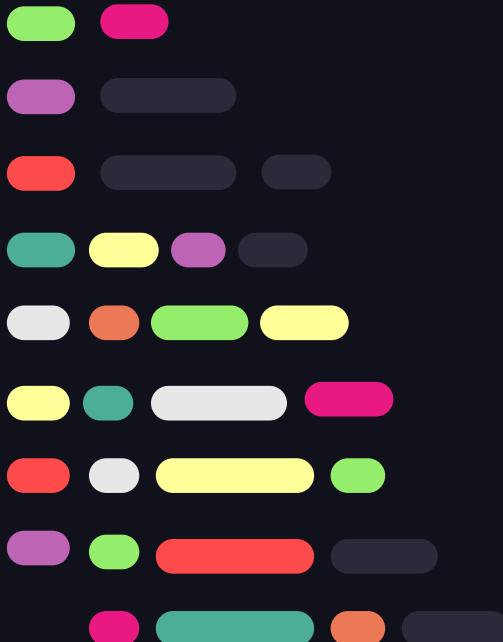
Rotinas e Funções

{ Retorno de resultados

```
def soma(a=0, b=0, c=0):  
    s = a + b + c  
    return s  
  
r1 = soma(2, 4, 6)  
r2 = soma(2, 4)  
r3 = soma()  
  
print(f' Os resultados deram: {r1}, {r2} e {r3}')
```

}

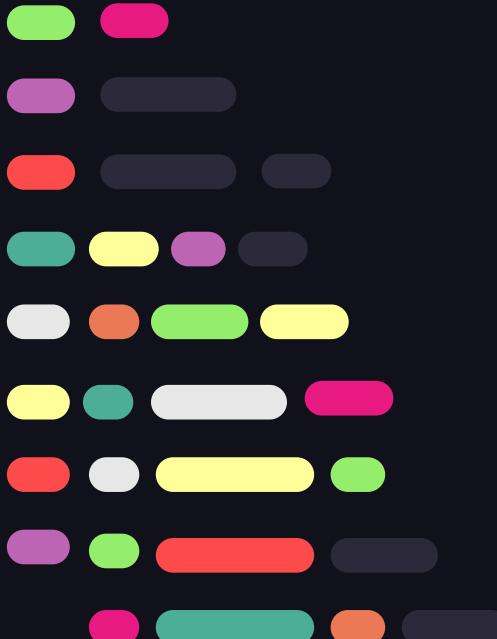
PRÁTICA! Exercício 35



Crie um programa com uma função chamada `fatorial()`, que receba dois parâmetros: o primeiro será o número a calcular o fatorial e o segundo será opcional e lógico que indique se será exibido no ecrã ou não o processo de cálculo do fatorial.



PRÁTICA! Exercício 36



Desenvolva um programa que permita ao utilizador calcular o seu Índice de Massa Corporal (IMC). O programa deve solicitar ao utilizador a sua altura e o seu peso. De seguida, utilizando uma função, deve calcular o IMC e o programa deve exibir uma mensagem com base no valor do IMC calculado.

- IMC abaixo de 18,5: Abaixo do peso
- IMC entre 18,5 e 24,9: Peso normal
- IMC entre 25,0 e 29,9: Sobre peso
- IMC entre 30,0 e 34,9: Obesidade grau 1
- IMC entre 35,0 e 39,9: Obesidade grau 2
- IMC acima de 40,0: Obesidade grau 3 (obesidade mórbida)

