

“Colinha” de Programação em Python

ELC106 - Algoritmo e Programação - UFSM (2024a)

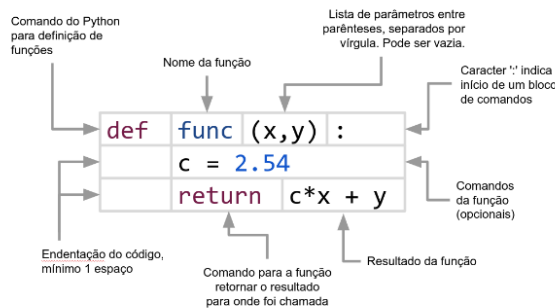
Profa Andrea Schwertner Charão | Monitora: Vanessa Eduarda Neumann de Oliveira

Operadores aritméticos: usados para realizar operações matemáticas com valores numéricos

Símbolo	Descrição	Exemplo
+	Adição de valores	$a + b$
-	Subtração de valores	$a - b$
*	Multiplicação de valores	$a * b$
/	Divisão de valores reais	a / b
//	Divisão de valores inteiros	$a // b$
**	Exponenciação de valores	$a ** b$
%	Resto da divisão (módulo)	$a \% b$

Ordem de prioridade: $()$, $**$, $*/\%///$, $+$ -

Sintaxe de definição de função: estrutura que determina a *função* de cada bloco do código



Ex: uma função que realiza operações aritméticas para converter temperaturas.

1) Início com a palavra-chave “def”, nomeio a função como “converter” e estabeleço o parâmetro “temp”

```
def converter(temp):
```

2) Crio a variável “resultado” que vai conter o resultado da conversão de acordo com uma fórmula

```
    resultado = ((temp-32)*(5/9))
```

↳ a **indentação** (o espaço abaixo de def) indica que a linha seguinte ainda faz parte da função

3) Com “return”, comando o retorno do resultado, ou seja, o retorno da conversão (não o retorno da função!)

```
    return resultado
```

Sintaxe de uso de função: refere-se à maneira como você *chama* uma função que foi definida anteriormente.

Retomando o exemplo:

```
def converter(temp):  
    resultado = ((temp-32)*(5/9))  
    return resultado
```

4) Por fim, adiciono uma linha que não faz mais parte da função. Uso a função “print” para imprimir o resultado na tela, chamo a função nomeada “converter” e substituo o parâmetro anterior “temp” pelo valor real da temperatura em Fahrenheit que deverá ser convertida para Celsius.

```
def converter(temp):  
    resultado = ((temp-32)*(5/9))  
    return resultado  
  
print(converter(120))
```

! Até então, havia sido criado apenas o bloco de *definição* da função, com sua variável e sua fórmula. O bloco final dá uso a elas. Note como a indentação (recuo) estabelece a hierarquia e ordem entre os blocos.

Saída do código:

```
48.88888888888889
```