

# LÓGICA E LINGUAGEM DE PROGRAMAÇÃO

Curso Técnico Subsequente em Informática  
Lucas Sampaio Leite



# Combinação de operadores aritméticos e atribuições

Operador	Exemplo	Operação
=	x = 5	x recebe 5
+=	x += 5	Adiciona 5 a x
-=	x -= 3	Decrementa 3 de x
*=	x *= 2	Multiplica x por 2
/=	x /= 4	Divide x por 4
**=	x **= 2	x elevado a 2
%=	x %= 2	Resto de x por 2
//=	x //= 3	Quociente inteiro de x por 3

## Operadores aritméticos com strings

- O operador + executa uma concatenação de strings, ou seja, une as strings pelas extremidades.
- O operador \* também funciona em strings; ele executa a repetição da string.

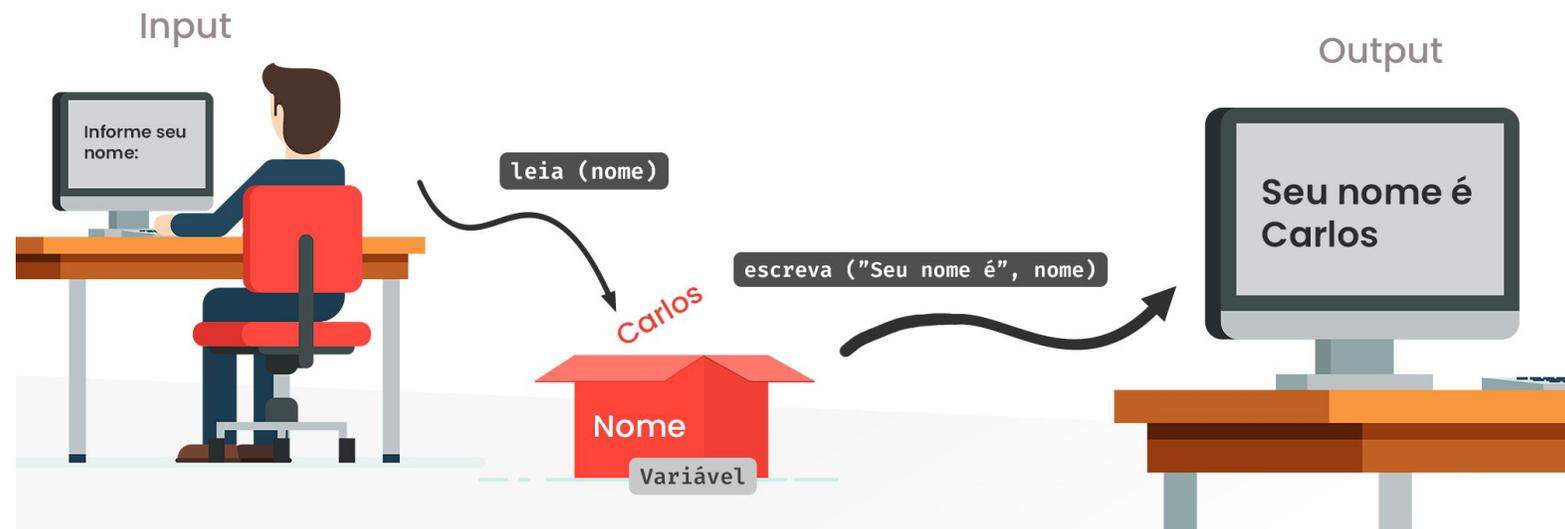
```
instituicao = "IF Baiano"  
campus = "Senhor do Bonfim"  
curso = "Técnico Integrado em Informática"  
resultado = instituicao + "/" + campus + "/" + curso  
print(5*"=" + resultado + 5*"=")
```



```
=====IF Baiano/Senhor do Bonfim/Técnico Integrado em Informática=====
```

## Comandos de entrada e saída

- A função `input()` é utilizada para solicitar ao usuário de um programa para digitar dados via teclado. (Comando de entrada)
- A função `print()` é utilizada para imprimir mensagens e/ou valores de variáveis na tela para o usuário. (Comando de saída)



## Comandos de entrada e saída

- Comandos de entrada e saída:

```
1 nome = input('Digite o seu nome: ')
2 idade = int(input('Digite a sua idade: '))
3 print('Olá, {}! Você tem {} anos.'.format(nome, idade))
4 print(f'Olá, {nome}! Você tem {idade} anos.')
5 nova_idade = idade + 25
6 print('Daqui a 25 anos, sua idade será {} anos'.format(nova_idade))
7 print(f'Daqui a 25 anos, sua idade será {nova_idade} anos')
```



Os dados digitados pelos usuários sempre serão do tipo <str>

## Comandos de entrada e saída

- Comandos de entrada e saída:

```
1 nome = input('Digite o seu nome: ')\n2 idade = int(input('Digite a sua idade: '))\n3 print('Olá, {}! Você tem {} anos.'.format(nome, idade))\n4 print(f'Olá, {nome}! Você tem {idade} anos.')\n5 nova_idade = idade + 25\n6 print('Daqui a 25 anos, sua idade será {} anos'.format(nova_idade))\n7 print(f'Daqui a 25 anos, sua idade será {nova_idade} anos')
```



```
Olá, Lucas! Você tem 31 anos.\nOlá, Lucas! Você tem 31 anos.\nDaqui a 25 anos, sua idade será 56 anos.\nDaqui a 25 anos, sua idade será 56 anos.
```

# Comandos de entrada e saída

```
1 nome = input("Digite o seu nome: ")
2 idade = int(input("Digite a sua idade: "))
3 altura = float(input("Digite a sua altura "))
4
5 print(nome, type(nome))
6 print(idade, type(idade))
7 print(altura, type(altura))
```



```
Digite o seu nome: Lucas
Digite a sua idade: 18
Digite a sua altura 1.72
Lucas <class 'str'>
18 <class 'int'>
1.72 <class 'float'>
```

# Comandos de entrada e saída

```
1 print('Alguns exemplos de impressão em Python, ',end='')
2 print('o end no fim da linha anterior vai unir estas duas linhas.')
3 print('A barra invertida seguida de n em Python \n faz essa String saltar uma linha')
4 print('A barra invertida seguida de t em Python \t faz essa String saltar uma tabulação')
5 print('Lucas','Sampaio','Leite', sep="_")
6 print(1, 2, 3, sep="*", end="= \n")
```



```
Alguns exemplos de impressão em Python, o end no fim da linha anterior vai unir estas duas linhas.
A barra invertida seguida de n em Python
faz essa String saltar uma linha
A barra invertida seguida de t em Python          faz essa String saltar uma tabulação
Lucas_Sampaio_Leite
1*2*3=
```

## Comandos de entrada e saída

- Pode-se usar aspas triplas (""" ou """) para criar uma string em várias linhas e, em seguida, imprimi-la. Isso pode ser útil quando você tem um bloco de texto grande para imprimir:

```
1 mensagem = """
2 Esta é a primeira linha.
3 Esta é a segunda linha.
4 E aqui está a terceira linha.
5 """
6
7 print(mensagem)
```



```
● lucas@lucas-Inspiron-5548:~/Dr
  /Logica/codes/teste.py

Esta é a primeira linha.
Esta é a segunda linha.
E aqui está a terceira linha.
```

## Comandos de entrada e saída

- Formatar número de casas de ponto flutuante:

```
PI = 3.141592653589793238462643383279

print("PI com 2 casas decimais {:.2f}".format(PI))
print("PI com 3 casas decimais {:.3f}".format(PI))
print("PI com 10 casas decimais {:.10f}".format(PI))

print(f"PI com 2 casas decimais {PI:.2f}")
print(f"PI com 3 casas decimais {PI:.3f}")
print(f"PI com 10 casas decimais {PI:.10f}")
```



```
PI com 2 casas decimais 3.14
PI com 3 casas decimais 3.142
PI com 10 casas decimais 3.1415926536
PI com 2 casas decimais 3.14
PI com 3 casas decimais 3.142
PI com 10 casas decimais 3.1415926536
```

## Comandos de entrada e saída

```
1 x = 5; y = 10
2 print('O valor de x é {} e o de y é {}'.format(x,y))
3
4 print('Eu amo {0} e {1}'.format('pão', 'manteiga'))
5 print('Eu amo {1} e {0}'.format('pão', 'manteiga'))
6 print('Oi {nome}, {cumprimento}'.format(cumprimento = 'Bom dia!', nome = 'Lucas'))
7
8 x = 12.3456789
9 print('The value of x is %.2f' %x)
10 print('The value of x is %.4f' %x)
```

Pouco utilizado!!!

## Exercícios rápidos

1. Crie um programa em Python que solicite ao usuário e receba o valor do lado para calcular a área de um quadrado;
2. Crie um programa em Python que solicite ao usuário e receba os valores da base e altura para calcular a área de um triângulo;
3. Crie um programa em Python que que solicite ao usuário e receba o valor do raio para calcular a área, perímetro e diâmetro de um círculo. Declare  $\pi$  como constante.
4. Faça um Programa que peça 2 números inteiros e um número real. Calcule e mostre:
  - a) o produto do dobro do primeiro com metade do segundo .
  - b) a soma do triplo do primeiro com o terceiro.
  - c) o terceiro elevado ao cubo.

Mais exercícius...

<https://wiki.python.org.br/EstruturaSequencial>



# Dúvidas



# LÓGICA E LINGUAGEM DE PROGRAMAÇÃO

Curso Técnico Subsequente em Informática  
Lucas Sampaio Leite

