

Lógica de Programação com Python

```
In [5]: print(1+1)
```

2

```
In [9]: x = 2  
print(x+2)
```

4

Criando minha primeira função

```
In [8]: def saudacao():  
        nome = input('Qual é o seu nome? ')  
        print(f'Olá {nome}')
```

```
In [10]: saudacao()
```

Qual é o seu nome? bento
Olá bento

Criando outra função (com parâmetro)

```
In [11]: nome = 'Luiza'
```

```
In [12]: def saudacao_com_parametro(nome):  
        print(f'Olá {nome}')
```

```
In [13]: saudacao_com_parametro(nome)
```

Olá Luiza

```
In [14]: saudacao_com_parametro('ana')
```

Olá ana

Condicional

```
In [15]: idade = 20  
  
def verifica_se_pode_dirigir(age):  
    if age > 18:  
        print('Tem permissão para dirigir')  
    else:  
        print('Não tem permissão para dirigir')
```

```
In [16]: verifica_se_pode_dirigir(idade)
```

Tem permissão para dirigir

```
In [17]: verifica_se_pode_dirigir(15)
```

Não tem permissão para dirigir

```
In [18]: def verifica_se_pode_dirigir_sem_parametros():  
         age = input('Qual a sua idade? ')  
  
         if int(age) > 18: #colocar int(palavra) converte a palavra em número  
             print('Tem permissão para dirigir')  
         else:  
             print('Não tem permissão para dirigir')
```

```
In [20]: verifica_se_pode_dirigir_sem_parametros()
```

Qual a sua idade? 18

Não tem permissão para dirigir

```
In [21]: verifica_se_pode_dirigir_sem_parametros()
```

Qual a sua idade? 20

Tem permissão para dirigir

Lista

```
In [22]: idade = 22  
         idade
```

```
Out[22]: 22
```

```
In [23]: type(idade)
```

```
Out[23]: int
```

```
In [24]: nome = 'guilherme'
```

```
In [25]: type(nome)
```

```
Out[25]: str
```

```
In [26]: idades = [18, 22, 15, 50]  
         #         0   1   2   3  
         #         0   3  -2  -1
```

```
In [27]: idades[1]
```

```
Out[27]: 22
```

```
In [28]: idades[0:2]
```

Out[28]: [18, 22]

In [29]: `idades[0:3]`

Out[29]: [18, 22, 15]

In [30]: `idades[1:]`

Out[30]: [22, 15, 50]

In [31]: `idades[-1]`

Out[31]: 50

In [32]: `idades[-2]`

Out[32]: 15

In [33]: `idades[:-1]`

Out[33]: [18, 22, 15]

In [34]: `idades[::-1]` *# de trás para frente*

Out[34]: [50, 15, 22, 18]

In [35]: `idades[::-2]` *#de trás para frente de 2 em 2*

Out[35]: [50, 22]

In [36]: `idades[::2]` *#de 2 em 2*

Out[36]: [18, 15]

Laços e loops

```
In [37]: #for fora da função
def verifica_se_pode_dirigir(age):
    if age >= 18:
        print(f'{age} anos tem permissão para dirigir')
    else:
        print(f'{age} anos não tem permissão para dirigir')

for idade in idades:
    verifica_se_pode_dirigir(idade)
```

```
18 anos tem permissão para dirigir
22 anos tem permissão para dirigir
15 anos não tem permissão para dirigir
50 anos tem permissão para dirigir
```

```
In [38]: #for dentro da função
def verifica_se_pode_dirigir():
    for idade in idades:
        if idade >= 18:
            print(f'{idade} anos tem permissão para dirigir')
        else:
            print(f'{idade} anos não tem permissão para dirigir')
```

```
In [39]: verifica_se_pode_dirigir()

18 anos tem permissão para dirigir
22 anos tem permissão para dirigir
15 anos não tem permissão para dirigir
50 anos tem permissão para dirigir
```

Booleano

```
In [40]: idade = 18
```

```
In [41]: idade >= 18
```

```
Out[41]: True
```

```
In [42]: idade <= 18
```

```
Out[42]: True
```

```
In [43]: idade > 18
```

```
Out[43]: False
```

```
In [44]: idade < 18
```

```
Out[44]: False
```

```
In [45]: idade == 18
```

```
Out[45]: True
```

```
In [46]: idade != 15
```

```
Out[46]: True
```

```
In [47]: idade > 15
```

```
Out[47]: True
```

```
In [48]: idade < 15
```

Out[48]: False

In [49]: `idade >= 15`

Out[49]: True

In [50]: `idade <= 15`

Out[50]: False

In [51]: `idade == 15`

Out[51]: False

In [52]: `idade == 18`

Out[52]: True

In [53]:

```
permissoes = []
idades = [20, 14, 40]

def verifica_se_pode_dirigir(ages, permission):
    for idade in ages:
        if idade >= 18:
            print(f'{idade} anos tem permissão para dirigir')
            permission.append(True)
        else:
            print(f'{idade} anos não tem permissão para dirigir')
            permission.append(False)

verifica_se_pode_dirigir(idades, permissoes)
```

20 anos tem permissão para dirigir
14 anos não tem permissão para dirigir
40 anos tem permissão para dirigir

In [54]: `permissoes`

Out[54]: [True, False, True]

In [55]:

```
for permissao in permissoes:
    if permissao: #pode ser usado também if permissao == True
        print('Tem permissão para dirigir')
    else:
        print('Não tem permissão para dirigir')
```

Tem permissão para dirigir
Não tem permissão para dirigir
Tem permissão para dirigir

Tipos em uma lista

In [58]: `lista = ["bento", 31, True, "18"]`

```
for elemento in lista:
    print(f'O elemento {elemento} é do tipo', type(elemento))
```

0 elemento bento é do tipo <class 'str'>
0 elemento 31 é do tipo <class 'int'>
0 elemento True é do tipo <class 'bool'>
0 elemento 18 é do tipo <class 'str'>

Imports

Lista de notas (forma aleatória)

```
In [60]: from random import randrange
```

```
In [61]: randrange(0,11)
```

```
Out[61]: 3
```

```
In [63]: notas_matematicas = []

for nota in range(8):
    notas_matematicas.append(randrange(0,11))
```

```
In [64]: notas_matematicas
```

```
Out[64]: [7, 10, 1, 10, 10, 6, 7, 5]
```

```
In [73]: from random import randrange, seed
seed(10)
notas_matematicas = []
for nota in range(8):
    notas_matematicas.append(randrange(0,11))
```

```
In [74]: notas_matematicas
```

```
Out[74]: [9, 0, 6, 7, 9, 0, 3, 7]
```

```
In [75]: len(notas_matematicas)
```

```
Out[75]: 8
```

MatPlot Lib

```
In [78]: import matplotlib.pyplot as plt
```

```
In [83]: x = list(range(1,9))
y = notas_matematicas
```

In [84]: x

Out[84]: [1, 2, 3, 4, 5, 6, 7, 8]

In [85]: y

Out[85]: [9, 0, 6, 7, 9, 0, 3, 7]

In [96]:

```
plt.plot(x,y, marker='o')
plt.title("Notas de Matemática")
plt.xlabel("Provas")
plt.ylabel("Notas")
plt.show()
```

