

# LISTA DE EXERCÍCIOS

## Funções

1- Faça um programa, com uma função que necessite de três argumentos, e que forneça a soma desses três argumentos.

---

```
In [ ]: def soma(num1, num2, num3):  
        res = num1 + num2 + num3  
        print(res)  
  
        # Programa Principal  
        soma(1, 2, 3)  
        soma(9, -1, 0)
```

2-Escreva uma função que receba dois números inteiro e imprima o menor número.

---

```
In [ ]: def menor(num1, num2):  
        if num1 < num2:  
            print(num1)  
        else:  
            print(num2)  
  
        # Programa Principal  
        menor(6, 9)  
        menor(3, 1)
```

3- Escreva uma função que, dado um número real passado como parâmetro, retorne a parte inteira e a parte fracionária desse número.

---

```
In [ ]: def parte_num(num):  
        inteira = int(num)  
        frac = num - inteira  
        return inteira, frac  
  
        # Programa Principal  
        print(parte_num(3.5))
```

4- Elabore uma função que receba uma lista contendo N valores e retorne o maior elemento da lista e o número de vezes que esse elemento apareceu na lista.

---

```
In [ ]: def rep_maior(lista):
    maior = lista[0]
    for num in lista[1:]:
        if maior < num:
            maior = num
    cont = lista.count(maior)
    return maior, cont

# Programa Principal
minha_lista = [4, 5, 6, 7, 7, 1, 7]
print(rep_maior(minha_lista))
```

5- Escreva uma função que receba 3 notas de um aluno e uma letra. Se a letra for "A" a função retorna a média aritmética das notas do aluno, se for "P", a sua média ponderada (pesos: 5, 3 e 2) e se for "H", a sua média harmônica.

---

```
In [ ]: def media(n1, n2, n3, letra):
    letra = letra[0].upper() # somente a primeira letra
    if letra == "A":
        # Média Aritmética
        media = (n1 + n2 + n3)/3
    elif letra == "P":
        # Média Ponderada
        media = (n1*5 + n2*3 + n3*2)/10
    elif letra == "H":
        # Média Harmônica
        media = 3/(1/n1 + 1/n2 + 1/n3)
    else:
        return "Opção inválida!"
    return media

# Programa Principal
print(media(5, 7, 6, "a"))
print(media(5, 7, 6, "p"))
print(media(5, 7, 6, "h"))
print(media(5, 7, 6, "xxx"))
```

6- Faça uma função que leia 1 número e, em seguida, retorne uma frase que diga se o número é:

- positivo ou negativo;
- inteiro ou decimal.
- par ou ímpar;

obs.: O número só é par ou ímpar se for inteiro.

---

```
In [ ]: def verifica_num(num):
    if num - int(num) == 0:
```

```

inteiro_decimal = "Inteiro"
# PAR OU ÍMPAR
if num % 2 == 0:
    par_impar = "Par"
else:
    par_impar = "Impar"
else:
    inteiro_decimal = "Decimal"
    # PAR OU ÍMPAR
    par_impar = ""
# POSITIVO OU NEGATIVO
if num > 0:
    sinal = "Positivo"
elif num < 0:
    sinal = "Negativo"
else:
    sinal = "Neutro"

return f"{num} é {par_impar} {sinal} {inteiro_decimal}."

# Programa Principal
print(verifica_num(5))
print(verifica_num(-1))
print(verifica_num(2.5))

```

7- Faça uma função que retorne o reverso de um número inteiro informado. Por exemplo: 127 -> 721.

---

```

In [ ]: def reverso(num):
        return int(str(num)[::-1])

# Programa Principal
print(reverso(77589))
print(reverso(321))

```

8- Crie uma função que receba um número natural e retorne o seu fatorial. Por exemplo: 3! --> 1 \* 2 \* 3 = 6

---

```

In [ ]: def fatorial(num):
        if num >= 0:
            fat = 1
            for n in range(2, num + 1):
                fat = n * fat
            return fat
        return "Número inválido!"

# Programa Principal
print(fatorial(5))
print(fatorial(3))

```

9- Crie uma função que receba por parâmetro a altura e o sexo de uma pessoa, calcule o seu peso ideal e imprima uma mensagem com essa informação na tela.

1. Para homens:  $72 * \text{altura} - 60$
2. Para mulheres:  $62 * \text{altura} - 50$

```
In [ ]: def peso_ideal(altura, sexo):
        sexo = sexo[0].lower() # 1ª letra
        if sexo == "m":
            ideal = 72 * altura - 60
        elif sexo == "f":
            ideal = 62 * altura - 50
        else:
            return "Sexo inválido!"
        return f"Peso ideal: {ideal:.2f}kg"

# Programa Principal
print(peso_ideal(1.73, "f"))
print(peso_ideal(1.73, "m"))
```

10- Crie uma função que receba a altura e a largura e imprima um retângulo com as bordas completas e o meio aberto:

Exemplo:

```
retangulo(3, 10)
```

```
#####
#       #
#####
```

```
In [ ]: def retangulo(altura, largura):
        print('#' * largura)
        for _ in range(altura-2):
            print('#' + ' ' * (largura-2) + '#')
        print('#' * largura)

# Programa Principal
retangulo(3, 3)
print() # separar
retangulo(3, 10)
```