

# Mateus Sousa do Carmo

## Atividade

### Questão 1

Escreva um programa que solicite ao usuario uma palavra e verifique se ela é um palíndromo.

Resposta abaixo

```
In [ ]: def Palindromo(palavra:str)->None:
        print(f'{palavra} , {palavra[::-1]}')
        if (palavra == palavra[::-1]):
            print('Palavra é um palindromo')
        else:
            print('Palavra não é um palindromo')
```

```
In [ ]: Palindromo(input('Escreva uma palavra :'))
```

```
tenet , tenet
Palavra é um palindromo
```

### Questão 2

Escreva um programa que leia números digitados pelo usuario e pare quando o número 0 for digitado. No final , imprima a media dos numeros digitados

Resposta abaixo

```
In [ ]: def MediaDosNumeros():
        numero = None
        ListaDeNumeros = list()
        while (numero != 0):
            #Caso o usuario digite um valor diferente de numeros
            #Isso me dara um erro , irei trata-lo para ignorar esses erro
            try:
                #entrada de valores
                numero = float(input('Digite numeros diferente de 0: '))

                if (numero == 0.0): break

                ListaDeNumeros.append(numero)

            except Exception:
                print('Valor digitado Invalido')

        Quantidade = (len(ListaDeNumeros))
        Soma = sum(ListaDeNumeros)
        Resultado = Soma/Quantidade
        Valores = " ".join(map(str,ListaDeNumeros))

        print(f'Valor digitados : {Valores}')
        print(f'Media dos numeros : {Resultado}')

MediaDosNumeros()
```

Valor digitado Invalido  
Valor digitados : 2.0 3.0 4.0 5.0  
Media dos numeros : 3.5

### Questão 3

Dada um string 'python é um linguagem poderosa', crie um lista com as palavras separadas.

Resposta abaixo

```
In [ ]: def ListaComPalavrasSeparadas(palavra:str): return palavra.split(' ')
```

```
In [ ]: ListaComPalavrasSeparadas('Python é uma linguagem poderosa')
```

```
Out[ ]: ['Python', 'é', 'uma', 'linguagem', 'poderosa']
```

### Questão 4

Dada uma lista de strings ['banana','maçã','laranja','abacaxi'], crie uma nova lista com palavras que tem mais de 5 letras e outra lista com as palavras que terminam com a letra 'a'.

Resposta abaixo

```
In [ ]: Lista = ['banana', 'maçã', 'laranja', 'abacaxi']

ListaA = []
ListaB = []

for palavra in Lista:
    if len(palavra)>5:
        ListaA.append(palavra)
    if palavra[len(palavra)-1]=='a':ListaB.append(palavra)

ListaA,ListaB
```

```
Out[ ]: (['banana', 'laranja', 'abacaxi'], ['banana', 'laranja'])
```

### Questão 5

Escreva uma função que receba uma lista de números e retorne outra lista com os números primos presentes.

Resposta abaixo

```
In [ ]: # usando o loop dentro de uma lista junto com a condição
def RetornaListaNumerosImpares(ListaDeNumeros):
    return [ i for i in ListaDeNumeros if i%2 != 0]
```

```
In [ ]: ListaDeNumeros = [1,2,3,4,5,6,7,8,9,10,11,16,19]
```

```
In [ ]: RetornaListaNumerosImpares(ListaDeNumeros)
```

```
Out[ ]: [1, 3, 5, 7, 9, 11, 19]
```

### Questão 6

Escreva uma função que receba uma lista de números e retorne outra lista com os numeros primos presentes.

Resposta abaixo

```
In [ ]: def RetornaNumerosPrimos(ListaDeNumeros):
        ListaDePrimos = list()

        for numero in ListaDeNumeros:
            booleano = True
            QuantidadeDeDivisores = 0

            if ( numero == 1 ) : booleano = False

            for i in reversed([ n for n in range(1,numero+1)]):

                if numero%i == 0 : QuantidadeDeDivisores +=1
                if QuantidadeDeDivisores > 2: booleano = False

            if (booleano): ListaDePrimos.append(numero)

        return ListaDePrimos
```

```
In [ ]: RetornaNumerosPrimos([i for i in range(1,55)])
```

```
Out[ ]: [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53]
```

## Questão 7

Escreva uma função que receba duas lista de numeros e retorne outra lista com os elements que estão presente em apenas uma lista.

Resposta abaixo

```
In [ ]: def DiferencaDeLista( ListaA:list , ListaB:list , diferenca = [], iter=1):

        for valores in ListaA:
            try:
                if ListaB.index(valores) != 0:pass
            except ValueError:
                diferenca.append(valores)

        if iter ==2: return diferenca

        # Usando recursividade
        # trocando os valores
        else:
            return DiferencaDeLista(ListaB,ListA,diferenca,2)
```

```
In [ ]: DiferencaDeLista([1,2,3,31,10],[1,2,3,4,5,6],[],1)
```

```
Out[ ]: [31, 10, 4, 5, 6]
```

## Questão 8

Dada uma lista de números inteiros, escreva uma função para encontrar o segundo maior valor na lista.

Resposta abaixo

```
In [ ]: def RetornaSegundoMaior(Lista:list):
        print(f'Lista:{list(reversed(sorted(Lista)))}')

        Lista.remove(max(Lista))
        segundo = max(Lista)
```

```
print(f'Segundo Maior {segundo}')
```

```
In [ ]: RetornaSegundoMaior([1,2,4,5,10,99])
```

```
Lista:[99, 10, 5, 4, 2, 1]  
Segundo Maior 10
```

## Questão 9

Crie uma função que receba uma lista de tuplas , cada uma contendo o nome e a idade de uma pessoa , e retorne a lista ordenada pelo nome das pessoas em ordem alfabética.

Resposta abaixo

```
In [ ]: def OrdemAlfabetica(ListaDeTupla):  
        return sorted(ListaDeTupla)
```

```
In [ ]: OrdemAlfabetica(  
        [('w',10),('b',20),('a',12),('g',41),('x',10),('b',18)]  
        )
```

```
Out[ ]: [('a', 12), ('b', 18), ('b', 20), ('g', 41), ('w', 10), ('x', 10)]
```

## Questão 10

Dada uma lista contendo números inteiros, como você encontraria o maior número eo menor número dessa lista em uma única passagem?

Resposta abaixo

```
In [ ]: lista = [1 , 4 , 5, 10,200, 31, 112]  
        print(max(lista),min(lista))
```

```
200 1
```