

Lista 02 - Prog 01 2024

Emilio Vital Brazil

Entrega: 06 de maio 2024

Todas as perguntas abaixo serão consideradas usando a sintaxes **Python 3.11**.

1. Observe o seguinte código e a sua saída de erro padrão:

```
1  def sum2list(l1: list, l2: list):
2      r = []
3      for n1 in l1:
4          c = []
5          r.append(c)
6          for n2 in l2:
7              c.append(n1*n2)
8      return r
9
10  mcc1 = sum2list(1, [3, 5, 6])
11  mcc2 = sum2list([1, 2], [3, 5, 6])
12  print(mcc1)
13  print(mcc2)
```

TypeError Traceback (most recent call last)

Cell In[3], line 10

```
7          c.append(n1*n2)
8      return r
—> 10 mcc1 = sum2list(1, [3, 5, 6])
11 mcc2 = sum2list([1, 2], [3, 5, 6])
13 print(mcc2)
```

Cell In[3], line 3, in sum2list(l1, l2)

```
1  def sum2list(l1: list, l2: list):
2      r = []
—> 3      for n1 in l1:
4          c = []
5          r.append(c)
```

TypeError: 'int' object is not iterable

- (a) Qual mudança precisa ser feita para o código rodar sem erro?
- (b) Qual saída padrão esperada?
- (c) Você acha que o nome *sum2list* representa bem o que a função faz? Renomeie e crie comentários para melhor documentar a função.

Para as seguintes questões não pode importar bibliotecas.

2. Crie uma função chamada *two_sum* que recebe uma lista de inteiros "nums" e um inteiro "target". A função retornar os índices de dois elementos da lista cuja soma seja igual a "target".

Por exemplo:

```
>>> two_sum([7, 3, 4, 2, 11, 15], 9)
[0, 3]
```

3. Crie uma função chamada *average* que recebe uma lista de números e retorne a média dos números da lista.

Por exemplo:

```
>>> average([.5, .25, .5, .25, 1.5, 0])
0.5
```

4. Crie uma função chamada *minus_num* que recebe uma lista de números e um número. A função retornar uma lista com os números da lista menos o número dado.

Por exemplo:

```
>>> minus_num([.5, .25, .5, .25, 1.5, 0], 0.5)
[.0, -.25, .0, -.25, 1.0, -.5]
```

5. Crie uma função chamada *std_deviation* que recebe uma lista de números e retorne o desvio padrão da lista. Tem que usar as funções criadas anteriormente: *average* e *minus_num*:

Por exemplo:

```
>>> std_deviation([.5, .25, .5, .25, 1.5, 0])
0.47871355387816905
```