

17, 18, 19 - Map, Filter e Reduce - Exercícios

August 17, 2022

1 Map, Reduce e Filter

```
[3]: from functools import reduce
```

```
[4]: # Exercício 1 - Utilize map para imprimir o quadrado dos números abaixo com 3
      ↪casas decimais.
      nums_float = [5.45, 2.57, 6.85, 9.77, 1.75, 8.88, 4.59]
      list(map(lambda x: round(x ** 2, 3), nums_float))
```

```
[4]: [29.703, 6.605, 46.922, 95.453, 3.062, 78.854, 21.068]
```

```
[5]: # Exercício 2 - Utilize reduce para imprimir na tela a multiplicação desses
      ↪números
      nums = [4, 6, 9, 23, 5]
      reduce(lambda num1, num2: num1 * num2, nums)
```

```
[5]: 24840
```

```
[7]: # Exercício 3 - Utilize filter para imprimir na tela nomes com menos de 5 letras
      nomes = ["Caio", "Pedro", "José", "Maria", "Fernando"]
      list(filter(lambda nome: len(nome) < 5, nomes))
```

```
[7]: ['Caio', 'José']
```

```
[8]: # Exercício 4 - Utilize map para mudar as letras abaixo. Se for minúscula
      ↪deve retornar maiúscula e vice-versa
      letras = ['a', 'b', 'C', 'd', 'E', 'F', 'g']
      list(map(lambda x: x.upper() if x == x.lower() else x.lower(), letras))
```

```
[8]: ['A', 'B', 'c', 'D', 'e', 'f', 'G']
```

```
[9]: # Exercício 5 - Utilize map com a lista de letras do exercício anterior
      # para retornar uma lista de tuplas com as letras em maiúsculas e minúsculas.
      ↪Ex.: 'a' -> ('A', 'a')
      list(map(lambda x: (x.upper(), x.lower()), letras))
```

```
[9]: [('A', 'a'),  
      ('B', 'b'),  
      ('C', 'c'),  
      ('D', 'd'),  
      ('E', 'e'),  
      ('F', 'f'),  
      ('G', 'g')]
```

```
[11]: # Exercício 6 - Utilize reduce para imprimir na tela a concatenação das  
      ↪ palavras abaixo, com um espaço ' ' entre elas  
      palavras = ['Eu', 'amo', 'python']  
      reduce(lambda palavra1, palavra2: palavra1 + ' ' + palavra2, palavras)
```

```
[11]: 'Eu amo python'
```

```
[12]: # Exercício 7 - Utilize filter para imprimir na tela apenas os números  
      ↪ positivos da lista abaixo  
      numeros = [5, -10, 55, 23, -16, -47, 20, -12]  
      list(filter(lambda num: num >= 0, numeros))
```

```
[12]: [5, 55, 23, 20]
```