

16 - Função Lambda e List e Dict Comprehension - Exercícios

July 20, 2022

1 Funções Lambda

- ```
[]: # Exercício 1 - Escreva uma função lambda que adiciona 5 a um número passado
 ↪ como argumento para a função

[]: # Exercício 2 - Escreva uma função utilizando "def" e dentro dela utilize
 ↪ lambda.
 # A função deve se chamar "dividir_por" e a partir dela deve ser possível criar
 ↪ "funções filhas", exemplo: "dividir_por_2", "dividir_por_3"
 # O número da função dividir_por vai ser o denominador.

[]: # Exercício 3 - Escreva uma função lambda que receba uma palavra como parâmetro
 ↪ e retorne True se essa palavra começar com a letra "a" ou "A"

[]: # Exercício 4 - Escreva uma função lambda para ordenar a lista abaixo em ordem
 ↪ alfabética
 nomes = ['Jeniffer', 'Diego', 'Penélope', 'Thomas', 'Bruna', 'Emerson']

[]: # Exercício 5 - Escreva uma função lambda para ordenar a lista do exercício
 ↪ anterior pela última letra do nome

[]: # Exercício 6 - Escreva uma função lambda para ordenar de forma descendente a
 ↪ lista abaixo pelo segundo item das tuplas
 notas = [('Inglês', 8), ('Geografia', 5), ('Matemática', 10), ('Português', 7),
 ↪ ('História', 3)]
```

## 2 List/Dict Comprehension

Utilize list/dict comprehension nos exercícios abaixo

- ```
[ ]: # Exercício 7 - Crie uma lista idêntica a lista abaixo
      numeros = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

[ ]: # Exercício 8 - Eleve ao quadrado todo número da lista "numeros" abaixo e
      ↪ retorne uma nova lista
```

```
[ ]: # Exercício 9 - O mesmo que o exercício 8, mas acrescente na nova lista, apenas  
    ↳ se o número elevado ao quadrado for maior que 30.
```

```
[ ]: # Exercício 10 - Crie um dicionário a partir da lista abaixo, com a mesma chave  
    ↳ e valor (exemplo: "x":"x")  
estados=['SP', 'RJ', 'BA']
```

```
[ ]: # Exercício 11 - Crie um dicionário com as chaves e valores abaixo.  
keys = ['Inglês', 'Português', 'Matemática', 'Geografia']  
values = [7, 4, 9, 8]
```

```
[ ]: # Exercício 12 - Crie um dicionário a partir do range abaixo, onde cada número  
    ↳ do range é a chave e cada número dividido por 100 é o valor.  
rng = range(100, 10000, 500)
```