

Aula 04 – Listas

Listas

Listas são conjuntos de valores mutáveis, mais especificamente uma sequência de valores indexados. Ou seja, cada valor em uma lista possui um índice único que vai de 0 até n-1, sendo n o número de elementos da lista. Pense em uma lista temática e faça operações similares às seguintes para incorporar bem os conceitos.

```
batatas = ['rústicas', 'sotê', 'palito', 'smile']
mais_batatas = ['assada', 'recheada']

batatas[0] #operação de acesso
batatas[1:2] #operação de fatiamento ou slicing
batatas = batatas + mais_batatas #operação de concatenação
batatas2 = batatas[:] #operação de cópia - sintaxe 1
batatas2 = batatas.copy() #operação de cópia - sintaxe 2
batatas2 = batatas #operação de referencição
batatas.append('estragada') #adição de um único elemento na lista
batatas.extend(['crocante', 'cremosa']) #adição de vários elementos a partir de uma lista referência
batatas.pop(2) #recupera um elemento da lista e o apaga dela
batatas.insert(2, 'doce') #insere um elemento em uma posição específica
batatas.remove('cremosa') #remove um elemento específico
del batatas[2:3] #remove um intervalo da lista
batatas.sort() #organiza em ordem alfabética
```

A forma mais comum de percorrer uma lista é usando um loop for:

```
queijos = ['cheddar', 'prato', 'suiço']
opcao_queijo = 0

print("Queijo:")
for queijo in queijos:
    opcao_queijo += 1
    print('{0} - '.format(opcao_queijo), queijo)
```

Podemos também fazer um for aninhado para fazer combinações de elementos de duas listas:

```
def combinacoes_comidas():
    adjetivos = ['suculenta', 'deliciosa', 'horrorosa']
    comidas = ['banana', 'feijoadada', 'vaca atolada']

    for adj in adjetivos:
        for com in comidas:
            print(adj, com)

combinacoes_comidas()
```

Palavras especiais que usamos quando estamos lidando com listas:

- **Redução:** padrão de processamento que atravessa uma lista e congrega todos os elementos em um único resultado (ex. soma de todos os valores de uma lista).
- **Mapeamento:** padrão de processamento que atravessa uma sequência e executa uma operação em cada elemento. (ex. “somar um” em todos os elementos de uma lista).
- **Filtragem:** padrão de processamento que atravessa uma lista e seleciona os elementos que satisfazem algum critério (ex. selecionar somente os valores pares).
- **Alias:** Uma circunstância onde duas ou mais variáveis se referem ao mesmo objeto.

List comprehension ou compreensão por listas é uma das maiores vantagens do Python. Além de ser mais “pytônico” escrever o for usando list comprehension, leva menos tempo (curiosidade: ver conceito de Big O’). Veja:

```
import time

lista_base = list(range(2500000))

#Teste 01 - For convencional
t_inicial = time.time()
for i in lista_base:
    lista_base[i] = lista_base[i] + 1
t_final = time.time()
print("For convencional levou:", t_final - t_inicial, "segundos")

#Teste 02 - For com List Comprehension
t_inicial = time.time()
lista_base = [elemento+1 for elemento in lista_base]
t_final = time.time()
print("For com list comprehension levou:", t_final - t_inicial, "segundos")
```

Podemos também aninhar fors no formato list comprehension:

```
def list_comprehension_aninhado():

    cores_singular = [
        ['amarelo', 'laranja', 'beje'],
        ['preto', 'branco', 'cinza'],
        ['vermelho', 'roxo', 'rosa']
    ]

    #achatamento (flattening)
    cores_plural = [cor[i] + 's' for cor in cores_singular for i in range(3)]

    #transposta
    cores_plural = [[cor[i] + 's' for cor in cores_singular] for i in range(3)]

    print(cores_plural)

list_comprehension_aninhado()
```

Projetinho – Parte I

Faça um programa para automatizar a anotação de pedidos de sanduíche do Subway (sem calcular o valor total). Considere cada conjunto de ingredientes (tipo do pão, carne do lanche, saladas, queijo, adicionais, molhos) como uma lista separada.

Exercícios

Podemos nos basear nos exercícios do *Pense em Python* para fixação:

<https://penseallen.github.io/PensePython2e/10-listas.html>

Ou nos exercícios do Python.org: <https://wiki.python.org.br/ListaDeExercicios>

Dicionários

Veremos com mais detalhes em alguma das próximas aulas.

Projetinho – Parte II

Desenvolva agora a parte de precificação do pedido do subway utilizando um dicionário para conectar o preço de cada ingrediente e contabilizar o valor final. Consulte o website do Subway ou o iFood para fazer o protótipo o mais fiel possível.