

Curso baseado no
livro de Eric
Mattes, **Python
Crash Course**


Python

Curso Intensivo

Prof. Cláudio Fleury
Abr-22

Cap.3 - Listas

Conteúdo

1. Listas
 2. Listas - Acessando Itens
 3. Listas - Alterando, Incluindo e Removendo Itens
 4. Organizando uma Lista
 5. Erro de Índice ao Manipular uma Lista
 6. Resumo
- 

Listas

- Uma coleção de itens em uma determinada ordem
- Os itens armazenados podem ser heterogêneos (tipos de dados diferentes)
- Em geral, as listas armazenam mais de um item, e por isso são nominadas com identificadores no plural: produtos, alunos, vendas etc.
- Em Python, os colchetes delimitam os itens de uma lista

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

▶ colar4

◀ ▶ import this

bikes.py

```
1 bikes = ['trek', 'cannondale', 'redline', 'specialized', 'rav']  
2 print(bikes)  
3  
4
```

```
['trek', 'cannondale', 'redline', 'specialized', 'rav']  
[Finished in 831ms]
```

Listas - Acessando Itens

- Listas são coleções ordenadas de itens - podemos acessar seus elementos indicando a posição (índice) do item desejado
- O primeiro elemento da lista tem **índice 0** (e não 1)
- Para acessar o ***n***-ésimo elemento da lista use o índice ***n*-1**
- O último elemento da lista tem índice **len(lista)-1**, ou **-1** (índice negativo, não precisa saber a qtd de itens da lista)

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS


▶ colar4

```
import this
bikes.py
1 bikes = ['trek', 'cannondale', 'redline', 'specialized', 'rav']
2 print(bikes)
3 print(bikes[0])
4 print(bikes[0].title())
5 print(bikes[-1].title())
```

```
['trek', 'cannondale', 'redline', 'specialized', 'rav']
trek
Trek
Rav
[Finished in 194ms]
```

Exercícios

Tente estes programas de curtos para obter alguma experiência com listas em Python. Você pode criar uma nova pasta para cada capítulo de exercícios para se mantê-los organizados.

- 1. Nomes:** Armazenar os nomes de alguns de seus amigos em uma lista chamada **nomes**. Imprimir o nome de cada pessoa ao acessar cada elemento na lista, um de cada vez.
 - 2. Saudações:** Usando a lista criada no exercício anterior, imprima o nome de cada pessoa e uma mensagem para eles. O texto de cada mensagem deve ser o mesmo, mas cada mensagem deve ser personalizado com a nome da pessoa.
 - 3. Lista Própria:** Pense no seu modo favorito de transporte: moto, carro, bike, ônibus,... e crie uma lista para armazená-los. Use sua lista para imprimir uma série de afirmações sobre esses itens, tais como 'O transporte mais barato é ônibus', 'O transporte mais rápido é moto.', ...
- 

Listas - Alterando/Incluindo/Removendo Itens

- A maioria das listas são dinâmicas - operações de inclusão, alteração e remoção de itens são frequentes
- Modificação: `nome_lista[indice] = novo_valor`
- Inclusão
 - Ao final da lista: `nome_lista.append(item)`
 - Numa posição específica: `nome_lista.insert(indice,item)`
- Remoção:
 - Do último item da lista: `nome_lista.pop()`
 - De um item em posição conhecida: `del nome_lista[indice]`
ou: `nome_lista(indice)`
 - De um valor específico da lista: `nome_lista.remove(valor)`

O método **append()** facilita a criação de listas dinamicamente.

Por exemplo, você pode começar com uma lista vazia e, em seguida, adicionar itens à lista usando uma série de chamadas ao método **append()**.

O método **remove()** exclui somente a primeira ocorrência do valor na lista.

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

▶ colar4

```
import this
bikes5.py x
1 motos = ['honda', 'yamaha', 'suzuki']
2 primeira_moto = motos.pop(0)
3 print(f"A primeira moto que eu possuí foi uma {primeira_moto.title()}")
```

```
A primeira moto que eu possuí foi uma Honda.
[Finished in 189ms]
```

Exercícios

- 1. Lista de convidados:** Se você pudesse convidar qualquer pessoa, viva ou morta, para o jantar, quem você convidaria? Faça uma lista que inclui pelo menos três pessoas que você gostaria de convidar para jantar. Em seguida, use a lista para imprimir uma mensagem para cada pessoa, convidando-a para jantar.
- 2. Alterando a Lista de Convidados:** Você acabou de ouvir que um de seus convidados não poderá comparecer ao jantar, assim você precisará enviar um novo convite. Você vai ter que pensar em alguém convidar.
 - Comece com o seu programa do exercício anterior. Adicione uma chamada `print()` no final do seu programa, informando o nome do hóspede que não pode comparecer.
 - Modifique a sua lista, substituindo o nome do hóspede que não pode comparecer com o nome da nova pessoa que você está convidando.
 - Imprima um segundo conjunto de mensagens de convite, um para cada pessoa que ainda está na sua lista.
- 3. Mais Convidados:** Você acabou de encontrar uma grande mesa de jantar, então agora há mais espaço disponível. Você tem lugares para mais três convidados para o jantar.
 - Comece com o seu programa do exercício anterior. Adicione uma chamada `print()` no fim do programa para informar às pessoas que você encontrou uma mesa maior de jantar.
 - Use `insert()` para adicionar um novo convidado no início da lista.
 - Use `insert()` para adicionar um novo convidado no meio da lista.
 - Use `append()` para adicionar um novo convidado no final da lista.
 - Imprimir um novo conjunto de mensagens de convite, um para cada pessoa em sua lista.
- 4. Encolhendo a Lista de Convidados:** Você acabou de descobrir que sua nova mesa de jantar não chegará tempo para o jantar, e você tem espaço para apenas duas pessoas.
 - Comece com o seu programa do exercício anterior. Adicione uma nova linha que imprime uma mensagem dizendo que você pode convidar apenas duas pessoas para o jantar.
 - Use **`pop()`** para remover pessoas da sua lista, um de cada vez, até que apenas dois nomes permanecem na sua lista. A cada vez que você retirar um nome da sua lista, imprima uma mensagem para a pessoa saber que você sente muito, mas não poderá confirmar o convite para jantar.
 - Imprima uma mensagem para cada uma das duas pessoas ainda em sua lista, confirmando-lhes que eles ainda estão convidados.
 - Usar **`del`** para remover os dois últimos nomes da sua lista, de modo a ter uma lista vazia. Imprimir a sua lista para certificar-se de que você realmente tem uma lista vazia no final do seu programa.

Organizando uma Lista

- Ordem inversa de posição dos itens da lista
 - Método **reverse()**
- As listas podem ser ordenadas de dois modos
 - Permanentemente
 - Método **lista.sort(reverse=[False]/True)**
 - Temporariamente
 - Função **sorted(lista,reverse=[False]/True)**

O método **reverse()** altera a ordem de uma lista permanentemente, mas você pode reverter para o original a qualquer momento chamando o método **reverse()** para a mesma lista novamente.

Comprimento de uma lista (qtde de elementos armazenados): função **len(lista)**

File Edit Selection Find View Goto Tools Project Preferences Help

```
FOLDERS
  ▶ colar4

import this  bikes5.py  carros2.py
1  carros = ['bmw', 'audi', 'toyota', 'subaru']
2  print("Lista original :", carros)
3  carros.reverse()
4  print("Lista invertida:", carros)
5  print("Qtde de carros na lista:", len(carros))
```

```
Lista original : ['bmw', 'audi', 'toyota', 'subaru']
Lista invertida: ['subaru', 'toyota', 'audi', 'bmw']
Qtde de carros na lista: 4
[Finished in 302ms]
```


Exercícios

- 1. Viajando pelo Mundo:** Pense em pelo menos cinco lugares do mundo você gostaria de visitar.
 - Armazenar as localizações em uma lista. Certifique-se de que a lista não está em ordem alfabética.
 - Imprimir a lista na ordem original. Não se preocupe sobre como imprimir a lista ordenadamente, basta imprimi-la na mesma ordem em que foi criada.
 - Use **sorted()** para imprimir uma lista, em ordem alfabética, sem modificar a lista original.
 - Mostrar que a lista ainda está em sua ordem original de sua impressão.
 - Use **sorted()** para imprimir a sua lista em ordem alfabética inversa, sem alterar a ordem da lista original.
 - Mostrar que a lista ainda está em sua ordem original, imprimindo-a novamente.
 - Usar **reverse()** para alterar a ordem da lista. Imprimir a lista para mostrar que a ordem foi alterada.
 - Usar **reverse()** para alterar a ordem da sua lista novamente. Imprimir a lista para mostrar que ela está de volta à sua ordem original.
 - Usar **sort()** para alterar a sua lista, armazenando-a em ordem alfabética. Imprimir a lista para mostrar que sua ordem foi alterada.
 - Usar **sort()** para alterar a sua lista, armazenando-a na ordem alfabética inversa. Imprimir a lista para mostrar que a ordem foi alterada.
- 2. Convidados pro jantar:** Trabalhando com um dos programas dos exercícios anteriores, use a função **len()** para imprimir uma mensagem indicando o número de pessoas que você está convidando para jantar.
- 3. Cada Função:** Pense em algo que você pode armazenar em uma lista. Por exemplo, você poderia fazer uma lista de montanhas, rios, países, cidades, idiomas, ou qualquer outra coisa que você gostaria. Escrever um programa que cria uma lista que contém esses itens e, em seguida, usa cada função apresentada neste capítulo pelo menos uma vez.

Erros de Índice ao Manipular uma Lista

- Erro comum para quem inicia no uso de listas

```
motos = ['honda', 'yamaha', 'suzuki']  
print(motos[3])
```

Traceback (most recent call last):

```
File "motos.py", line 2, in <module>  
    print(motos[3])
```

IndexError: list index out of range

Erro de índice: índice de lista fora da faixa → significa que o Python não consegue encontrar um item na posição (índice) fornecida ao programa

Lembre-se de que o último item da lista pode ser acessado com índice igual a **len(lista)-1** ou a **-1**

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

▶ colar4

```
import this  
1 motos = ['honda', 'yamaha', 'suzuki']  
2 print(f"A última moto que eu possuí foi uma {motos[-1].title()}")  
3 print(f"A última moto que eu possuí foi uma {motos[len(motos)-1].title()}")
```

```
A última moto que eu possuí foi uma Suzuki.  
A última moto que eu possuí foi uma Suzuki.  
[Finished in 165ms]
```

Resumo

- Definição de lista em Python
- Como trabalhar com itens de uma lista, individualmente
- Como criar, adicionar e remover itens de uma lista
- Como ordenar listas permanente e temporariamente
- Como encontrar o tamanho de uma lista
- Como evitar erros de índice ao trabalhar com listas

