

Python: Estruturas de Dados

Prof. Dr. Delcino Picinin Júnior
Prof. Dr. Luiz Antonio Schalata Pacheco
Prof Ms. Antônio Miguel Faustini Zarth

Garopaba, 2022



Sumário

- 1 Listas
- 2 Tuplas
- 3 Dicionários

Listas

Uma lista no Python tem os itens lado a lado, separados por vírgula.

A lista deve estar entre colchetes.

Uma vez que você tenha criado a lista, você pode adicionar, remover ou procurar itens nela (lista é um tipo de dados mutável, isto é, um tipo de dados que pode ser alterado).

Listas

Escreve a Lista

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def leLista():
5      global frutas
6      frutas=["maça", "banana", "mamão", "uva"]
7      escreveLista()
8
9  def escreveLista():
10     global frutas
11     print("Lista de Frutas {}".format(frutas))
12
13 if __name__ == '__main__':
14     leLista()
```

Listas

Escreve Tamanho Lista

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def leLista():
5      global frutas
6      frutas=["maça", "banana", "mamão", "uva"]
7      escreveTamanhoLista()
8
9  def escreveTamanhoLista():
10     global frutas
11     print("Eu tenho {} frutas".format(len(frutas)))
12
13 if __name__ == '__main__':
14     leLista()
```

Listas

Escreve Elementos Lista

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def leLista():
5      global frutas
6      frutas=["maça", "banana", "mamão", "uva"]
7      escreveSeparadoLista()
8
9  def escreveSeparadoLista():
10     global frutas
11     for uma in frutas:
12         print("Fruta {}".format(uma))
13
14  if __name__ == '__main__':
15     leLista()
```

Listas

append(x) Adiciona um item ao fim da lista

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def leLista():
4      global frutas
5      frutas=["maça", "banana", "mamão", "uva"]
6      nome=input("Digite o nome de uma fruta")
7      frutas.append(nome)
8      escreveLista()
9
10 def escreveLista():
11     global frutas
12     print("Lista de Frutas {}".format(frutas))
13
14 if __name__ == '__main__':
15     leLista()
```

Listas

extend(L) Concatena duas listas

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def leLista():
4      global frutas
5      frutas=["maça", "banana", "mamão", "uva"]
6      outrasFrutas=["limão", "melancia", "manga"]
7      frutas.extend(outrasFrutas)
8      escreveLista()
9
10 def escreveLista():
11     global frutas
12     print("Lista de Frutas {}".format(frutas))
13
14 if __name__ == '__main__':
15     leLista()
```


Listas

insert(i, x) Insere um item em uma determinada posição.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def leLista():
4      global frutas
5      frutas=["maça", "banana", "mamão", "uva"]
6      nome=input("Digite o nome de uma fruta")
7      frutas.insert(0,nome) # inserida antes da maça
8      escreveLista()
9
10 def escreveLista():
11     global frutas
12     print("Lista de Frutas {}".format(frutas))
13
14 if __name__ == '__main__':
15     leLista()
```

Listas

remove(x) apaga um item da lista.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def leLista():
4      global frutas
5      frutas=["maça","banana","mamão","uva"]
6      nome=input("Digite o nome para apagar")
7      frutas.remove(nome)
8      escreveLista()
9
10 def escreveLista():
11     global frutas
12     print("Lista de Frutas {}".format(frutas))
13
14 if __name__ == '__main__':
15     leLista()
```

Listas

del lista[x] apaga o item da posição x.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def leLista():
4      global frutas
5      frutas=["maça", "banana", "mamão", "uva"]
6      posicao=int(input("Digite a posição para apagar"))
7      del frutas[posicao]
8      escreveLista()
9
10 def escreveLista():
11     global frutas
12     print("Lista de Frutas {}".format(frutas))
13
14 if __name__ == '__main__':
15     leLista()
```

Listas

index(x) posição onde está na lista, primeira é 0.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def leLista():
4      global frutas
5      frutas=["maça", "banana", "mamão", "uva"]
6      escreveLista()
7      nome=input("Digite o nome para apagar")
8      print("Posição {} ".format(frutas.index(nome)))
9
10 def escreveLista():
11     global frutas
12     print("Lista de Frutas {}".format(frutas))
13
14 if __name__ == '__main__':
15     leLista()
```

Listas

pop(i) Remove o item de uma dada posição da lista e o retorna.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaLista():
5      frutas=["maça","banana","mamão","uva"]
6      print("Lista de Frutas {}".format(frutas))
7      posicao=int(input("Digite posição para apagar"))
8      apagado=frutas.pop(posicao)
9      print("Apagou {} ".format(apagado))
10     print("Lista de Frutas {}".format(frutas))
11
12 if __name__ == '__main__':
13     ManipulaLista()
```

Listas

count(x) Retorna o número de vezes que x aparece na lista.

```
1 #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaLista():
5      frutas=["maça","banana","maça","uva"]
6      print("Lista de Frutas {}".format(frutas))
7      nome=input("Digite fruta para contar")
8      total=frutas.count(nome)
9      print("Apareceu {} vezes".format(total))
10
11 if __name__ == '__main__':
12     ManipulaLista()
```

Listas

sort() Organiza os itens da lista.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaLista():
5      frutas=["maça","banana","mamão","uva"]
6      print("Lista de Frutas {}".format(frutas))
7      frutas.sort()
8      print("Lista de Frutas {}".format(frutas))
9
10 if __name__ == '__main__':
11     ManipulaLista()
```

Listas

reverse() Reorganiza os elementos da lista em ordem decrescente.

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaLista():
5      frutas=["maça","banana","mamão","uva"]
6      print("Lista de Frutas {}".format(frutas))
7      frutas.reverse()
8      print("Lista de Frutas {}".format(frutas))
9
10 if __name__ == '__main__':
11     ManipulaLista()
```


Listas

[:] criando sub-listas

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaLista():
5      frutas=["maça","banana","mamão","uva"]
6      print("Primeira Fruta {}".format(frutas[0]))
7      lista1=frutas[1:] #cria nova lista sem maçã
8      print("Lista {}".format(lista1))
9      lista2 = frutas[:-1] # cria nova lista sem uva
10     print("Lista {}".format(lista2))
11     lista3=frutas[1:-2] # sem maçã uva e mamão
12     print("Lista {}".format(lista3))
13
14 if __name__ == '__main__':
15     ManipulaLista()
```

Tuplas

Tupla: **(posição 0, posição 1, posição n)**

Tuplas são semelhantes a listas, porém elas são imutáveis.

Então você não pode modificar as tuplas, só substituir a tupla inteira por uma nova.

Elas são definidas com seus elementos separados por vírgula dentro de parêntese.

Tupla

Tupla - juntar nome e idade

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def ler():
4      nome = input("Digite seu nome")
5      idade = int(input("Digite sua idade"))
6      pessoa = (nome, idade) # tupla aqui
7      return pessoa
8
9  def ManipulaPessoa():
10     pes = ler()
11     print("Seu Nome   : {}".format(pes[0]))
12     print("Sua Idade  : {}".format(pes[1]))
13
14  if __name__ == '__main__':
15     ManipulaPessoa()
```

Tupla

Tupla - recria tupla para somar idade

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaPessoa():
5      nome = input("Digite seu nome")
6      idade = int(input("Digite sua idade"))
7      pes = (nome, idade)
8      print("Seu Nome : {}".format(pes[0]))
9      print("Sua Idade : {}".format(pes[1]))
10     pes=(pes[0],pes[1]+1) #para somar 1 na idade
11     print("Seu Nome : {}".format(pes[0]))
12     print("Sua Idade : {}".format(pes[1]))
13
14 if __name__ == '__main__':
15     ManipulaPessoa()
```

Tupla

Tupla para Lista, Lista para Tupla

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3
4  def ManipulaCoisas():
5      coisasTupla = ("cadeira", "mesa", "cama")
6      print("Coisas Tupla : {}".format(coisasTupla))
7      coisasLista = ["bolo", "chocolate", "pão"]
8      print("Coisas Lista : {}".format(coisasLista))
9      novaLista = list(coisasTupla) #transforma em lista
10     print("Nova Lista : {}".format(novaLista))
11     novaTupla = tuple(coisasLista) #transforma em tupla
12     print("Nova Tupla : {}".format(novaTupla))
13
14     if __name__ == '__main__':
15         ManipulaCoisas()
```

Dicionários

Um dicionário é como um livro de endereços onde você pode encontrar o endereço de uma pessoa apenas sabendo seu nome, ou seja, nós associamos chaves (nome) com valores (endereço).

Observe que a chave deve ser única, pois caso existam duas chaves iguais (duas pessoas com o mesmo nome) você pode encontrar informações incorretas.

Você só pode usar objetos imutáveis (como strings) como chaves de um dicionário, mas para os valores você pode usar objetos mutáveis ou imutáveis. Isto basicamente significa dizer que você só deve usar objetos simples como chaves.

Pares de chaves e valores são especificados em um dicionário usando a notação

d = {chave1:valor1, chave2:valor2}

Dicionários

Lê duas pessoas e adiciona no dicionário

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def LePessoa():
4      global dicPes
5      nome = input("Digite seu nome")
6      sexo= input("Digite sexo m ou f")
7      idade = int(input("Digite sua idade"))
8      dicPes[nome] = (sexo,idade) #tupla no dicionario
9
10 if __name__ == '__main__':
11     dicPes={}
12     LePessoa()
13     LePessoa()
14     print("Dicionario: {}".format(dicPes))
15     #{'jose': ('m', 20), 'maria': ('f', 44)}
```

Dicionários

Observe que os pares chave/valor são separados por vírgula.

Dentro de cada par a chave é separada do valor por dois pontos e todo o dicionário deve estar envolvido com o sinal de chaves { }.

Dicionários

Alterar tupla dentro do dicionário

```
1  #!/usr/bin/env python3
2  # coding: utf-8
3  def manipulaPessoa():
4      global dicPes
5      dicPes={'jose': ('m', 20),'pedro': ('m', 25)}
6      dicPes['silvia'] = ('f',46)
7      print('Pessoas {}'.format(dicPes))
8      ps=dicPes['pedro']
9      ps=(ps[0],ps[1]+1)
10     dicPes['pedro']=ps
11     print('Pessoas {}'.format(dicPes))
12
13 if __name__ == '__main__':
14     dicPes={}
15     manipulaPessoa()
```

Python: Estruturas de Dados

Prof. Dr. Delcino Picinin Júnior
Prof. Dr. Luiz Antonio Schalata Pacheco
Prof Ms. Antônio Miguel Faustini Zarth

Garopaba, 2022

