

Python - listas de junção

Em Python, List é classificado como um objeto do tipo sequência. É uma coleção de itens, que podem ser de diferentes tipos de dados, com cada item tendo um índice posicional começando com 0. Você pode usar diferentes maneiras de unir duas listas Python.

Todos os objetos do tipo sequência suportam operador de concatenação, com o qual duas listas podem ser unidas.

```
L1 = [10,20,30,40]
L2 = ['one', 'two', 'three', 'four']
L3 = L1+L2
print ("Joined list:", L3)
```

Ele produzirá a seguinte **saída** -

```
Joined list: [10, 20, 30, 40, 'one', 'two', 'three', 'four']
```

Você também pode usar o operador de concatenação aumentada com o símbolo "+=" para anexar L2 a L1

```
L1 = [10,20,30,40]
L2 = ['one', 'two', 'three', 'four']
L1+=L2
print ("Joined list:", L1)
```

O mesmo resultado pode ser obtido usando o método extend(). Aqui, precisamos estender L1 para adicionar elementos de L2 nele.

```
L1 = [10,20,30,40]
L2 = ['one', 'two', 'three', 'four']
L1.extend(L2)
print ("Joined list:", L1)
```

Para adicionar itens de uma lista a outra, uma solução iterativa clássica também funciona. Percorra os itens da segunda lista com um loop for e anexe cada item da primeira.

```
L1 = [10,20,30,40]
L2 = ['one', 'two', 'three', 'four']

for x in L2:
    L1.append(x)
```

```
print ("Joined list:", L1)
```

Uma abordagem um pouco complexa para mesclar duas listas é usar a compreensão de lista, como mostra o código a seguir -

```
L1 = [10,20,30,40]  
L2 = ['one', 'two', 'three', 'four']  
L3 = [y for x in [L1, L2] for y in x]  
print ("Joined list:", L3)
```