

Python - Remover itens da lista

Os métodos da classe de lista **remove()** e **pop()** podem remover um item de uma lista. A diferença entre eles é que `remove()` remove o objeto dado como argumento, enquanto `pop()` remove um item no índice fornecido.

Usando o método `remove()`

O exemplo a seguir mostra como você pode usar o método `remove()` para remover itens da lista -

```
list1 = ["Rohan", "Physics", 21, 69.75]
print ("Original list: ", list1)

list1.remove("Physics")
print ("List after removing: ", list1)
```

Ele produzirá a seguinte **saída** -

```
Original list: ['Rohan', 'Physics', 21, 69.75]
List after removing: ['Rohan', 21, 69.75]
```

Usando o método `pop()`

O exemplo a seguir mostra como você pode usar o método `pop()` para remover itens da lista -

```
list2 = [25.50, True, -55, 1+2j]
print ("Original list: ", list2)
list2.pop(2)
print ("List after popping: ", list2)
```

Ele produzirá a seguinte **saída** -

```
Original list: [25.5, True, -55, (1+2j)]
List after popping: [25.5, True, (1+2j)]
```

Usando a palavra-chave “del”

Python possui a palavra-chave “del” que exclui qualquer objeto Python da memória.



Exemplo

Podemos usar “del” para excluir um item de uma lista. Dê uma olhada no exemplo a seguir -

```
list1 = ["a", "b", "c", "d"]
print ("Original list: ", list1)
del list1[2]
print ("List after deleting: ", list1)
```

Ele produzirá a seguinte **saída** -

```
Original list: ['a', 'b', 'c', 'd']
List after deleting: ['a', 'b', 'd']
```

Exemplo

Você pode excluir uma série de itens consecutivos de uma lista com o operador de fatiamento. Dê uma olhada no exemplo a seguir -

```
list2 = [25.50, True, -55, 1+2j]
print ("List before deleting: ", list2)
del list2[0:2]
print ("List after deleting: ", list2)
```

Ele produzirá a seguinte **saída** -

```
List before deleting: [25.5, True, -55, (1+2j)]
List after deleting: [-55, (1+2j)]
```