

Python - Métodos de Matriz

Método array.reverse()

Assim como os tipos de sequência, a classe array também suporta o método reverse() que reorganiza os elementos na ordem inversa.

Sintaxe

```
array.reverse()
```

Parâmetros

Este método não tem parâmetros

Exemplo

```
import array as arr
a = arr.array('i', [1, 2, 3, 4, 5])
a.reverse()
print (a)
```

Ele produzirá a seguinte **saída** -

```
array('i', [5, 4, 3, 2, 1])
```

A classe array também define os seguintes métodos úteis.

Método array.count()

O método count() retorna o número de vezes que um determinado elemento ocorre no array.

Sintaxe

```
array.count(v)
```

Parâmetros

- **v** - O valor cujas ocorrências devem ser contadas

Valor de retorno

O método `count()` retorna um número inteiro correspondente ao número de vezes que `v` aparece no array.

Exemplo

```
import array as arr
a = arr.array('i', [1, 2, 3, 2, 5, 6, 2, 9])
c = a.count(2)
print ("Count of 2:", c)
```

Ele produzirá a seguinte **saída** -

```
Count of 2: 3
```

método `array.index()`

O método `index()` na classe `array` encontra a posição da primeira ocorrência de um determinado elemento no array.

Sintaxe

```
array.index(v)
```

Parâmetros

- **v** - o valor para o qual o índice deve ser encontrado

Exemplo

```
a = arr.array('i', [1, 2, 3, 2, 5, 6, 2, 9])
c = a.index(2)
print ("index of 2:", c)
```

Ele produzirá a seguinte **saída** -

Método array.fromlist()

O método fromlist() anexa itens de uma lista Python ao objeto array.

Sintaxe

```
array.fromlist(l)
```

Parâmetros

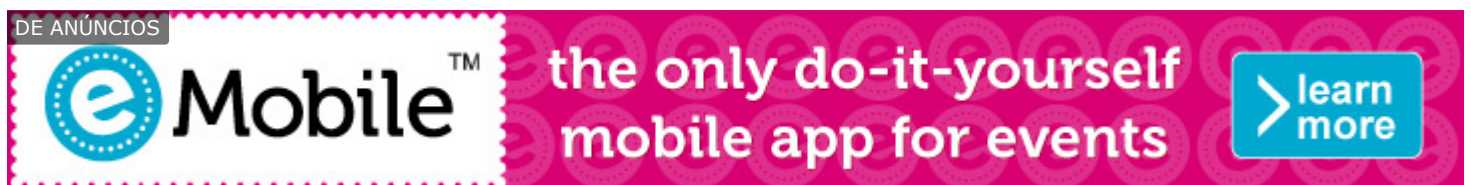
- **i** - A lista cujos itens são anexados ao array. Todos os itens da lista devem ser do mesmo tipo de arr.

Exemplo

```
import array as arr
a = arr.array('i', [1, 2, 3, 4, 5])
lst = [6, 7, 8, 9, 10]
c = a.fromlist(lst)
print (a)
```

Ele produzirá a seguinte **saída** -

```
array('i', [1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
```



Método array.tofile()

O método tofile() na classe array grava todos os itens (como valores de máquina) no array no objeto de arquivo f.

Sintaxe

```
array.tofile(f)
```

Parâmetros

- **f** - o objeto de arquivo obtido com a função `open()`. O arquivo a ser aberto no modo `wb`.

Exemplo

```
import array as arr
f = open('list.txt', 'wb')
arr.array("i", [10, 20, 30, 40, 50]).tofile(f)
f.close()
```

Saída

Após executar o código acima, um arquivo denominado "list.txt" será criado no diretório atual.

DE ANÚNCIOS

Método `array.fromfile()`

O método `fromfile()` lê um arquivo binário e anexa um número especificado de itens ao objeto `array`.

Sintaxe

```
array.fromfile(f, n)
```

Parâmetros

- **f** - O objeto de arquivo referente a um arquivo de disco aberto no modo `rb`
- **n** - número de itens a serem anexados

Exemplo

```
import array as arr
a = arr.array('i', [1, 2, 3, 4, 5])
f = open("list.txt", "rb")
```

```
a.fromfile(f, 5)
print (a)
```

Ele produzirá a seguinte **saída** -

```
array('i', [1, 2, 3, 4, 5, 10, 20, 30, 40, 50])
```