

Live de Python #30

Collections #3 - ChainMap e UserDict

Roteiro

- `itertools.chain`
- Avaliação de curto circuito
- `ChainMap`
- Customizando um dict
- `UserDict`



itertools.chain

Produz um iterador de junção de iteradores. 'chain' itera em todos os valores de uma sequência e ao final dessa sequência se inicia a próxima.

```
In [1]: from itertools import chain  
  
In [2]: a = [1, 2, 3, 4]  
  
In [3]: b = [5, 6, 7, 8]  
  
In [4]: for val in chain(a, b):  
...:     print(val)  
...:
```

itertools.chain

```
In [5]: from itertools import chain
```

```
In [6]: list(chain([1,2,3,4], [5,6,7,8], [9, 10]))
```

```
Out[6]: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```



itertools.chain

```
In [5]: from itertools import chain  
  
In [6]: list(chain([1,2,3,4], [5,6,7,8], [9, 10]))  
Out[6]: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```



itertools.chain

```
In [1]: def chain(*iterables):  
...:     for iterable in iterables:  
...:         for value in iterable:  
...:             yield value  
...:  
  
In [2]: list(chain([1,2,3,4], [5,6,7,8]))  
Out[2]: [1, 2, 3, 4, 5, 6, 7, 8]
```

Avaliação de curto circuito

Se o primeiro satisfaz, paramos por aqui.

```
In [1]: 7 or 8  
Out[1]: 7  
  
In [2]: 7 and 8  
Out[2]: 8
```

Avaliação de curto circuito

Se o primeiro satisfaz, paramos por aqui.

```
In [7]: def _in(x, iterable):  
...:     for val in iterable:  
...:         if val == x:  
...:             return True  
...:     return False  
...:
```

```
In [8]: _in(1, [1,1,1,1])
```

```
Out[8]: True
```