Python iterables

python

- https://kinsta.com/es/blog/iterables-de-python/
- https://ellibrodepython.com/iterator-python

iterables

```
even_list = [2, 4, 6, 8, 10]
my_iterator = iter(even_list)
print(next(my_iterator)) # Prints 2
print(next(my_iterator)) # Prints 4
print(next(my_iterator)) # Prints 6
```

funciones sobre iterables

- sum — Devuelve la suma de un iterable dado, siempre que la colección sea de tipos numéricos (enteros, valores de coma flotante y números complejos)
- any — Devuelve true si alguno de los elementos del iterable es verdadero. En caso contrario, devuelve false.
- all — Devuelve true si todos los elementos del iterable son verdaderos. En caso contrario, devuelve false.
- max — Devuelve el valor máximo de una colección iterable dada
- min — Devuelve el valor mínimo de una colección iterable dada
- len — Devuelve la longitud de un iterable dado
- append — Añade un valor al final de una lista iterable
- inssert - Añade un valor en la posición especificada

comprensiones

```
my_numbers = [11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
even_number_list = [num for num in my_numbers if num%2 == 0]
print(even_number_list)
```

zip

```
fruits = ["apple", "orange", "banana"]
rating = [1, 2, 3]

fruits_rating = zip(rating, fruits)
print(list(fruits_rating)) # [(1, 'apple'), (2, 'orange'), (3, 'banana')]
```

filter

Devuelve los elementos que dan cierto en la función asociada

```
def is_even(n):
    return n%2 == 0
```

```
nums_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
even_list = filter(is_even, nums_list)
print(list(even_list)) # [2, 4, 6, 8, 10]
```

map

```
my_list = [2, 4, 6, 8, 10]
square_numbers = map(lambda x: x ** 2, my_list)
print(list(square_numbers)) # [4, 16, 36, 64, 100]
```

sorted

```
# set
py_set = {'e', 'a', 'u', 'o', 'i'}
print(sorted(py_set, reverse=True)) # ['u', 'o', 'i', 'e', 'a']

# dictionary
py_dict = {'e': 1, 'a': 2, 'u': 3, 'o': 4, 'i': 5}
print(sorted(py_dict, reverse=True)) # ['u', 'o', 'i', 'e', 'a']

# frozen set
frozen_set = frozenset(('e', 'a', 'u', 'o', 'i'))
print(sorted(frozen_set, reverse=True)) # ['u', 'o', 'i', 'e', 'a']

# string
string = "kinsta"
print(sorted(string)) # ['a', 'i', 'k', 'n', 's', 't']

# list
py_list = ['a', 'e', 'i', 'o', 'u']
print(py_list) # ['a', 'e', 'i', 'o', 'u']
```

generadores

```
def even_generator(n):
    counter = 0
    while counter <= n:
        if counter % 2 == 0:
            yield counter
        counter += 1

for num in even_generator(20):
    print(num)
```

```
cube = (num ** 3 for num in range(1, 6))
for c in cube:
    print(c)
```

diccionario

```
diccionario = {10:'diez', 20:'veinte', 30:'treinta'}

for clave in diccionario.keys():
  print(clave) # 10,20,30

for valor in diccionario.values():
  print(valor) # diez,veinte,treinta

for item in diccionario.items():
  print(item[0], '->', item[1]) # 10 -> diez, 10 -> diez, 20 -> veinte, 30 ->
treinta

for clave, valor in diccionario.items():
  print(clave, '->', valor) # 10 -> diez, 10 -> diez, 20 -> veinte, 30 -> treinta
```

lista

- Enumerate():

```
fruits_list = ["Apple", "Mango", "Peach", "Orange", "Banana"]

for index, fruit in enumerate(fruits_list):
    print(fruit, index)

for index, fruit in enumerate(fruits_list, start = 2):
    print(fruit, index) # index empieza en 2
```

-
```
for elemento in lista
    print(elemento)

# recorrer por índices
for x in range(0,len(lista)):
    print (lista[x])
```

From:
https://matebcn.eu/ - **miguel angel torres egea**

Permanent link:
https://matebcn.eu/development:python:iterables

Last update: **29/09/2025 10:23**