

Ejemplo groupby

```
from itertools import groupby
```

```
# Primero vemos qué viene en cada par de elementos de groupby  
[(k, list(g)) for k, g in groupby('AAAABBBCCDAABBB')]
```

```
[('A', ['A', 'A', 'A', 'A', 'A']),  
 ('B', ['B', 'B', 'B', 'B']),  
 ('C', ['C', 'C']),  
 ('D', ['D']),  
 ('A', ['A', 'A', 'A']),  
 ('B', ['B', 'B', 'B', 'B'])]
```

Por cada letra, se crea un grupo que tiene como clave la letra misma, y como valor una lista con la cantidad de ocurrencias de la letra.

```
# Podemos ocupar este output para generar un diccionario  
{k: len(list(g)) for k, g in groupby('AAAABBBCCD')}
```

```
{'A': 4, 'B': 3, 'C': 2, 'D': 1}
```

Supongamos que tenemos una lista de palabras, que queremos agrupar según su largo.

Vimos que como primer argumento de `groupby` ingresamos una estructura iterable, en el caso anterior, un `string`, donde en cada iteración cada elemento corresponde a una letra. Pero también puede ser sobre una lista, donde cada grupo será cada elemento de la lista.

Esto por un lado, ya nos permite agrupar en base a una lista de palabras. Por otro lado, `groupby` recibe un segundo parámetro opcional, que corresponde cuál será la clave o `key` para la estructura retornada. Si no se especifica este parámetro, toma por defecto el valor mismo de la estructura agrupada. En caso de especificarse, debe corresponder a una función que se aplique al grupo y que retorne un valor.

Teniendo ya esta información, podemos crear el diccionario.

```
# Agrupar palabras por su largo
words = ["hola", "a", "todos", "y", "cada", "uno"]
# Luego agrupamos
{k: list(v) for k, v in groupby(words, key=len)}
```

```
{4: ['cada'], 1: ['y'], 5: ['todos'], 3: ['uno']}
```

Como se puede ver, el output no es el correcto. Esto sucede porque los elementos no están ordenados según el criterio por el que los queremos agrupar. Por lo tanto, primero ordenaremos las palabras según su largo y luego agruparemos.

```
# Primero ordenamos
words.sort(key=lambda x: len(x))
# Luego agrupamos
{k: list(v) for k, v in groupby(words, key=len)}
```

```
{1: ['a', 'y'], 3: ['uno'], 4: ['hola', 'cada'], 5: ['todos']}
```