

# Funciones anónimas

**Ejercicio: ordenar una lista de tuplas por el primer elemento de la tupla mediante una función *lambda* (uso de *sorted*)**

- *sorted( )* se usa para ordenar los elementos de un iterable y devuelve una nueva lista con los elementos ordenados. La lista original no se modifica.

**Sintaxis:**

- *sorted(iterable, key=Funcion, reverse=Boolean)*
  - Si *reverse* es *False*, se ordenan en orden ascendente.
  - Si *reverse* es *True*, se ordena en orden descendente.
  - Si *reverse* no se define, toma el valor de *False* por default
  - Especificar una función es opcional. Si no se especifica, toma el valor de *None*.

## 1. Definición de la Lista de Tuplas:

- Se define una lista de tuplas, donde cada tupla contiene un número y su representación en palabras

## 2. Definición de la Función Lambda:

- La función lambda para extraer el segundo elemento de cada tupla se define como sigue: Aquí, *lambda* define una función anónima que toma una tupla *x* y retorna su segundo elemento (*x*[1]).

## 3. Aplicación de *sorted*:

- Se usa *sorted* para ordenar la lista *datos* según el segundo elemento de cada tupla: *sorted* toma la lista *datos* y usa la función *funcion\_primer\_elemento* como la clave (*key*) para determinar el orden. Ordena las tuplas de acuerdo al segundo elemento en orden alfabético.

## 4. Impresión del Resultado:

- Finalmente, se imprime la lista ordenada: Esto muestra la lista de tuplas ordenadas por el segundo elemento.

```
datos = [(3, "tres"), (1, "uno"), (2, "dos")]

funcion_primer_elemento = lambda x : x[1]

datos_ordenados = sorted(datos, key=funcion_primer_elemento)

print(datos_ordenados)
```

```
[(2, 'dos'), (3, 'tres'), (1, 'uno')]
```