La función map()

En el más simple de todos los casos posibles, la función map(), toma dos argumentos:

- Una función.
- Una lista.

La descripción anterior está extremadamente simplificada, ya que:

- El segundo argumento map() puede ser cualquier entidad que se pueda iterar (por ejemplo, una tupla o un generador).
- map() puede aceptar más de dos argumentos.

La función map() aplica la función pasada por su primer argumento a todos los elementos de su segundo argumento y devuelve un iterador que entrega todos los resultados de funciones subsequentes.

```python
#map(function, list)

#Se genera una lista de 0 a 4 [0,1,2,3,4]
list_1 = [x for x in range(5)]

#Se aplica una función lambda 2^x(x es cada
elemento de la lista anterior), el resultado se
convierte en lista y se almacena en otra variable
[1, 2, 4, 8, 16]
list_2 = list(map(lambda x: 2 ** x, list_1))
print(list_2)

#Ahora a cada elemento de la nueva lista se lo
eleva al cuadrado y se imprime en consola 1 4 16
64 256
for x in map(lambda x: x * x, list_2):
      print(x, end=' ')
print()
```

- Se construye la list_1 con valores del 0 al 4.

- Después, se utiliza map junto con la primer lambda para crear una nueva lista en la que todos los elementos han sido evaluados como 2 elevado a la potencia tomada del elemento correspondiente de list_1.

- list_2 se imprime.

- En el siguiente paso, se usa nuevamente la función map() para hacer uso del generador que devuelve, e imprimir directamente todos los valores que entrega; como puedes ver, hemos usado la segunda lambda aquí.

- Solo eleva al cuadrado cada elemento de list_2.