**Map, Reduce, & Filter**

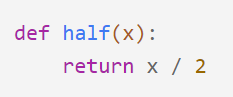
**Map**

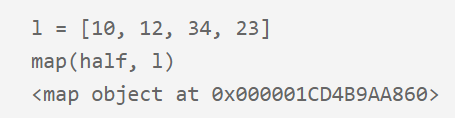
**map(**función,iterable**)**:

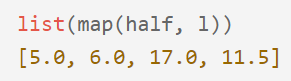
→ map aplica **una función a todos los elementos del iterable** (en este caso una lista) → por lo tanto, ***nos ahorramos el iterar*** (for loop)

→ el resultado del map es un ITERADOR (=un objeto de python “map”)

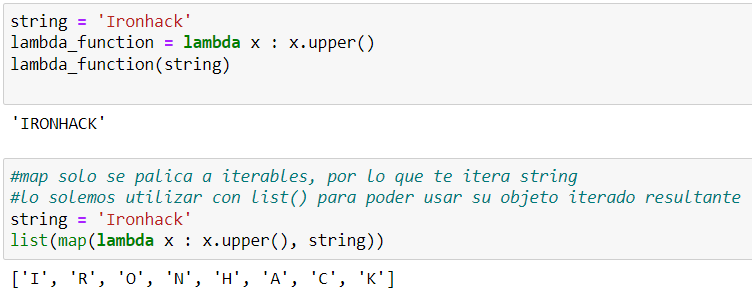
→ si listamos ese map nos da el resultado en una lista



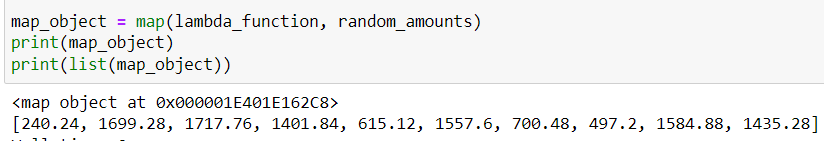
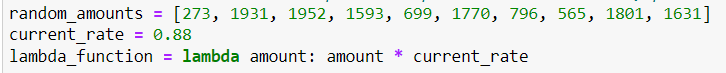




---> se suele usar mucho **con lambda functions**



*OTRO EJEMPLO:*



*→ NOS ESTAMOS AHORRANDO HACER UN FOR LOOP*

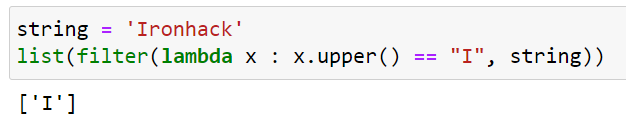
**Filter**

**filter(**función,iterable**)**:

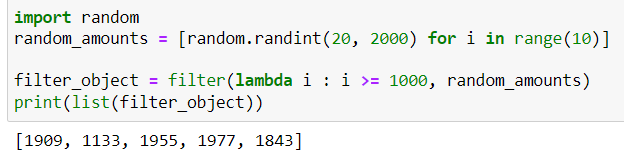
→ hace **la misma función que el map()** pero **añadiendo una condición (if)** a realizar sobre el iterable

→ el resultado del FILTER es un ITERADOR (=un objeto de python “filter”)

→ si listamos ese filter nos da el resultado en una lista



*OTRO EJEMPLO*



**Reduce**

**reduce(**función,iterable**)**:

→ AGREGA todos los elementos de un iterable y les aplica una operación simple(expresada en la función, ej.: sumar, restar, multiplicar, etc).

→ no da un objeto, sino su valor

*1º siempre hay que hacer un import*