



# FUNCIONES ANÓNIMAS – LAMBDA – DENTRO DE OTRA FUNCIÓN

- ▶ El poder de lambda se muestra mejor cuando lo usamos como función anónima dentro de otra función.
- ▶ Tenemos en el siguiente caso una definición de función que toma un argumento y ese número se multiplica por un número desconocido que asignamos posteriormente.

```
def multiplier(n):  
    return lambda a : a * n  
  
mydoubler = multiplier(2) # lambda a: a * 2  
mytripler = multiplier(3) # lambda a: a * 3  
  
print(mydoubler(11))      # a= 11 => lambda a: a * 2 = 22  
print(mytripler(11))      # a= 11 => lambda a: a * 3 = 33
```



# MAP

- ▶ Nos permite aplicar una función sobre cada uno de los elementos de una colección.
- ▶ Hacemos uso de esta función siempre que tengamos que transformar el valor de un elemento en otro.

```
map(función a aplicar, objeto iterable)
```



- ▶ La función a aplicar retorna un nuevo valor => nueva colección.