

In [4]:

```
1 class Underscore:
2     def map(self, iterable, callback):
3         for i in range(len(iterable)):
4             iterable[i] = callback(iterable[i])
5         return iterable
6
7     def find(self, iterable, callback):
8         for i in range(len(iterable)):
9             if callback(iterable[i]):
10                return iterable[i]
11
12    def filter(self, iterable, callback):
13        new = []
14        for i in range(len(iterable)):
15            if callback(iterable[i]):
16                new.append(iterable[i])
17        return new
18
19    def reject(self, iterable, callback):
20        new = []
21        for i in range(len(iterable)):
22            if callback(iterable[i])!=False:
23                new.append(iterable[i])
24        return new
25
26
27 # has creado una libreria con 4 métodos
28 # se crea la instancia de la clase
29 _ = Underscore() # sí, estamos configurando una instancia a una variable que es un guión bajo
30 # debe retornar [2, 4, 6] después que termines de implementar el código de arriba
31 evens = _.filter([1, 2, 3, 4, 5, 6], lambda x: x % 2 == 0)
32 print(evens)
33
34 # debe retornar [2,4,6]
35 w = _.map([1,2,3], lambda x: x*2)
36 print(w)
37
38 # debe retornar el primer valor que es mayor que 4
39 #Examina cada valor de la lista , devolviendo el primero que pasa una prueba de verdad
40 x = _.find([1,2,3,4,5,6], lambda x: x>4)
41 print(x)
42
```

```
43 # debe retornar [2,4,6]
44 #Examina cada valor de la lista , devolviendo una matriz de todos los valores que pasan una prueba de verda
45 y = _.filter([1,2,3,4,5,6], lambda x: x%2==0)
46 print(y)
47
48 # debe retornar [1,3,5]
49 #Devuelve los valores de la lista sin los elementos que pasa la prueba de verdad
50 z = _.reject([1,2,3,4,5,6], lambda x: x%2==0)
51 print(z)
```

[2, 4, 6]

[2, 4, 6]

5

[2, 4, 6]

[1, 3, 5]

In []:

1