

# Ejercicios « Módulos y paquetes

## Ejercicio 1

## Ejercicio 2

Crea un script llamado **generador.py** que cumpla las siguientes necesidades:

- Debe incluir una función llamada **leer\_numero()**. Esta función tomará 3 valores: **ini**, **fin** y **mensaje**. El objetivo es leer por pantalla un número  $\geq$  que ini y  $\leq$  que fin. Además a la hora de hacer la lectura se mostrará en el input el **mensaje** enviado a la función. Finalmente se devolverá el valor. Esta función tiene que devolver un número, y tiene que repetirse hasta que el usuario no lo escriba bien (lo que incluye cualquier valor que no sea un número del ini al fin).
- Una vez la tengas creada, deberás crear una nueva función llamada **generador**, no recibirá ningún parámetro. Dentro leerás dos números con la función **leer\_numero()**:
- El primer número será llamado **numeros**, deberá ser entre 1 y 20, ambos incluidos, y se mostrará el mensaje **¿Cuántos números quieres generar? [120]:**
- El segundo número será llamado **modo** y requerirá un número entre 1 y 3, ambos incluidos. El mensaje que mostrará será: **¿Cómo quieres redondear los números? [1]Al alza [2]A la baja [3]Normal:**.
- Una vez sepas los números a generar y cómo redondearlos, tendrás que realizar lo siguiente:
- Generarás una lista de **números aleatorios decimales** entre 0 y 100 con tantos números como el usuario haya indicado.
- A cada uno de esos números deberás redondearlos en función de lo que el usuario ha especificado en el modo.
- Para cada número muestra durante el redondeo, el número normal y después del redondeo.
- Finalmente devolverás la lista de números redondeados.

El objetivo de este ejercicio es practicar la reutilización de código y los módulos **random** y **math**.

### Recordatorio

El redondeo tradicional *round()* no requiere importar ningún módulo, es una función por defecto.

## Solución

```
import random
import math

# Completa el código aquí

def leer_numero(ini, fin, mensaje):
    while True:
        try:
            valor = int( input(mensaje) )
        except:
            pass
        else:
            if valor >= ini and valor <= fin:
                break
    return valor

def generador():
    numeros = leer_numero(1,20,"¿Cuántos números quieres generar? [1-20]: ")
    modo = leer_numero(1,3,"¿Cómo quieres redondear los números? " \
                       "[1]Al alza [2]A la baja [3]Normal: ")

    lista = []

    for i in range(numeros):
        numero = random.uniform(0,101)
        if modo == 1:
            print("{} => {}".format(numero, math.ceil(numero)) )
            numero = math.ceil(numero)
        elif modo == 2:
            print("{} => {}".format(numero, math.floor(numero)) )
            numero = math.floor(numero)
        elif modo == 3:
            print("{} => {}".format(numero, round(numero)) )
            numero = round(numero)
        lista.append(numero)

    return lista

generador()
```

```
PS C:\Users\USUARIO\Documents\Python V> & C:/Users/USUARIO/AppData/Local/Programs/Python/Python39-64/Python.exe -i C:\Users\USUARIO\Documents\Python V\Programa.py"
```

```
¿Cuántos números quieres generar? [1-20]: 11
```

```
¿Cómo quieres redondear los números? [1]Al alza [2]A la baja [3]Normal: 3
```

```
73.09809050339102 => 73
```

```
46.57395151780605 => 47
```

```
20.20437333470669 => 20
```

```
31.29690286394894 => 31
```

```
45.34965020138655 => 45
```

```
59.68744069761809 => 60
```

```
100.08856424221803 => 100
```

```
83.04642627259507 => 83
```

```
78.20523081362006 => 78
```

```
81.3465698846896 => 81
```

```
29.60649045389715 => 30
```

```
PS C:\Users\USUARIO\Documents\Python V> []
```