



TRABAJO PRÁCTICO N°5

Asignatura: Programación Orientada a Objetos

Importante: recordar que utilizaremos un *guión bajo inicial* en cada nombre de atributo como convención para indicar que deben tratarse como atributo privado, mientras que todos los métodos serán considerados públicos.

Ejercicio 1

Implementar una clase **A**, que contenga un constructor para asignar el valor de un atributo `_a1`, y un método **calcularCuadrado** que calcule y retorne el cuadrado de `_a1`. Luego incorporar los siguientes requisitos:

- Crear una clase **B** por herencia de **A**, que contenga un constructor para asignar el valor de `_a1` y un segundo parámetro `_b1`.
- Sobreescribir el método **calcularCuadrado**, de forma que calcule y retorne el cuadrado de `_b1`.
- Crear una clase **C** por herencia de **B**, que contenga un constructor para asignar valores de `_a1` y `_b1`.
- Agregar a **C** un método **sumar**, que calcule y retorne la suma de `_a1` y `_b1`.

Nota: con esta jerarquía, la clase A es la clase padre de B, mientras que C es clase hija de B.

Ejercicio 2

Crear una **clase abstracta A**, que especifique dos **métodos abstractos incrementarValor(cantidad)** y **reducirValor(cantidad)**. Luego utilizar a **A** como padre para crear una clase **B** y completar la implementación. **B** debe tener un atributo `_valor`, que se pueda incrementar o reducir en una cantidad determinada utilizando los métodos indicados.

Ejercicio 3

Crear una clase **Precio**, que modele los diferentes valores de venta de un producto teniendo en cuenta su precio de lista, recargo por pago en 3 cuotas y descuento por pago en efectivo. La clase debe contener un **constructor sobrecargado** para poder instanciar objetos con las siguientes opciones:

- p1 = Precio(1000)** # asigna \$1000 al precio de lista, no hay descuentos y no se permite pago en cuotas
- p2 = Precio(1000, 10)** # asigna \$1000 al precio de lista, un descuento del 10% por pago efectivo, no se permite pago en cuotas,
- p3 = Precio(1000, (10, 25))** # asigna \$1000 al precio de lista, un descuento del 10% por pago efectivo, y opciones de pago en 3 cuotas con 25% de recargo.

Agregar un método **mostrarOpcionesPago**, que imprima un mensaje con el siguiente formato de acuerdo a la información del objeto creado:



p1.mostrarOpcionesPago() # salida esperada: 'El precio de lista es \$1000 - No hay descuento por pago efectivo - No hay opción de pago en cuotas

p2.mostrarOpcionesPago() # salida esperada: 'El precio de lista es \$1000 - Descuento de 10% por pago efectivo - No hay opción de pago en cuotas

p3.mostrarOpcionesPago() # salida esperada: 'El precio de lista es \$1000 - Descuento de 10% por pago efectivo - Recargo de 25% por pago en 3 cuotas.

Ejercicio 4

Crear una clase **Numeros**, que pueda ser instanciada a partir de una lista de valores numéricos reales. Luego incorporar los **métodos mágicos** necesarios para poder iterar un objeto extrayendo únicamente los valores positivos.

Ejemplo:

Si se crea el objeto

```
num1 = Numeros([3, 6, -2, 0.5, -103, -71.3, 12.34])
```

luego debe poder ejecutarse el código:

```
for positivo in num1:  
    print(positivo)
```

Salida esperada:

```
3  
6  
0.5  
12.34
```