

## Examen FINAL. (2 horas)

*En los ejercicios se valorarán aspectos como: Corrección. Facilidad de mantenimiento (claridad del código). Clean Code. Fiabilidad. Eficiencia. Principios SOLID. Portabilidad. Intuitivo para el usuario, el control de la entrada de datos. Comprime una carpeta con tu proyecto para entregarlo. La no utilización de programación orientada a objetos (polimorfismo, sobreescritura, herencia, interfaces, constructores...) conlleva la anulación del ejercicio.*

### Ejercicio 1 (7 puntos)

Queremos almacenar datos de nuestras figuras coleccionables favoritas. Las figuras pueden ser de varios tipos:

- **Figuras de Acción.** Con un nombre, año de lanzamiento, una lista de accesorios, un precio base y un campo para indicar su estado: precintada, en su embalaje original, sin embalaje original.
- **Estatuas.** Con un nombre, año de lanzamiento, franquicia, un precio base y un campo para indicar su estado: precintada, en su embalaje original, sin embalaje original.
- **Funko Pops.** Con un nombre, año de lanzamiento, si es o no edición limitada, un precio base y un campo para indicar su estado: precintada, en su embalaje original, sin embalaje original.

Existe un método para calcular el precio real de las figuras.

El precio real de las figuras de acción y de las estatuas depende del estado según la siguiente fórmula:

- Si está precintada:  $\text{PRECIO REAL} = \text{PRECIO BASE} * 2$
- Si está en su embalaje original:  $\text{PRECIO REAL} = \text{PRECIO BASE}$
- Si no tiene su embalaje original:  $\text{PRECIO REAL} = \text{PRECIO BASE} / 2$

El precio de los Funko Pops depende del año de lanzamiento según la siguiente fórmula:

$$\text{PRECIO REAL} = \text{PRECIO BASE} + ((\text{AÑO ACTUAL} - \text{AÑO LANZAMIENTO}) * 10)$$

En cuanto a los **accesorios**, tendremos la siguiente información: nombre, material y descripción.

1. Crea las clases **Figura** (abstracta) y **Accesorio** con los tipos de datos correctos, getters, setters y constructor. **(0,5 puntos)**.
2. Crea las subclases de la clase **Figura** con los tipos de datos correctos, getters, setters y constructor. **(1 punto)**

3. Crea un método abstracto en Figura para calcular el precio real y sobreescríbelo en las subclases de Figura para calcular el precio real correctamente según las fórmulas. **(1 punto)**
4. Crea un programa principal que contendrá un listado con las figuras de la clase Figura y las siguientes opciones cada una en un método estático independiente:
  - **Añadir figura.** Pide por consola el tipo de figura y lo añade al listado. **(1 punto)**
  - **Añadir nuevo accesorio** (a una figura de acción). Pedirá el nombre de una figura y los datos de un accesorio y lo añadirá. **(0,5 puntos)**
  - **Mostrar listado.** Mostrará el nombre de cada figura, su precio base y su precio real. **(1 punto)**
  - **Mostrar Funko Pops de edición limitada.** Mostrará el nombre de cada Funko Pop, que pertenezca a una edición limitada. **(1 punto)**
5. **Tener realizado TODO el ejercicio y además: claridad del código, tener Clean Code, seguir los principios SOLID, seguir el estándar de Java, ser intuitivo para el usuario, controlar la entrada de datos, NO usar variables globales, NO tener código duplicado, nombres de métodos, clases y variables descriptivos. (1 punto)**

## Ejercicio 2 (3 puntos)

Haz una aplicación para leer y escribir un listado de Funko Pop desde un fichero. La clase de Funko Pop es la misma que en el ejercicio anterior.

El formato del fichero será el siguiente:

NOMBRE#AÑO#EDICIÓN ESPECIAL#PRECIO BASE#EMBALAJE ORIGINAL

Por ejemplo un posible fichero se vería así:

Stitch Hula#2023#Sí#25.99#EMBALAJE ORIGINAL

Naruto Shippuden#2022#NO#25.99#SIN EMBALAJE ORIGINAL

Iron Man Classic#2022#NO#18.99#EMBALAJE ORIGINAL

Stitch Space Suit#2023#NO#22.99#EMBALAJE ORIGINAL

## La aplicación debe:

- **Leer el fichero** al iniciar y poner todos los Funko Pop en un listado **(1 punto)**.
- Permitir al usuario añadir un Funko Pop al listado y **actualizar el fichero (0,75 puntos)**.
- Permitir al usuario borrar un Funko Pop del listado y **actualizar el fichero (0,75 puntos)**.
- Tener una opción para mostrar el número total de Funko Pop, mostrar el año del Funko Pop más nuevo, mostrar el año del Funko Pop más antiguo y mostrar la media de precio base de los Funko Pop. **(0,5 puntos)**.