

Programación Orientada a Objetos

Curso 2023/2024

Sesión 5

Ejercicios

Queremos desarrollar el software de gestión de una máquina expendedora que ofrece un solo producto. Antes de presentar el tipo de datos que representa a la máquina, comenzamos con el concepto Dinero que utilizamos para llevar la cuenta de las monedas que contiene la máquina y que introduce el usuario de la máquina. Los tipos de datos de este ejercicio serán programados en el paquete `maquinas`.

El tipo de datos **Dinero** representa la cantidad de monedas en el sistema monetario Euro: 5 céntimos, 10 céntimos, 20 céntimos, 50 céntimos, 1 euro y 2 euros. Se dejan fuera las monedas de uno y dos céntimos. Por tanto, la clase que implementa el tipo de datos tiene seis propiedades que llevan la cuenta del número de monedas de cada tipo. El constructor de la clase recibe como parámetros el número de monedas de cada tipo.

Junto al conteo de monedas, la clase mantiene una séptima propiedad con el valor de todas las monedas en euros (1,2, 0,5, etc.).

La funcionalidad de la clase Dinero es una operación que calcula la suma con otra instancia de la clase. El resultado de la operación es una nueva instancia de la clase Dinero con el resultado de la suma.

Implementa la clase Dinero para que defina **objetos inmutables**. Por tanto, la funcionalidad de la clase no debe alterar los atributos, sino que debe retornar un nuevo objeto con el resultado.

Añade a la clase Dinero seis *constantes* públicas que representen el dinero constituido por una sola moneda. Es decir, tendremos una constante que representa una instancia de la clase Dinero que contiene una única moneda de cinco céntimos, otra constante para una moneda de diez céntimos, etc. Introduce una última constante con nombre NADA que no contiene ninguna moneda.

Una **Máquina expendedora** tiene las siguientes propiedades:

- producto: cadena de texto. Identifica el producto que sirve la máquina.
- precio del producto: número real. Representa el dinero en euros.
- stock: número entero. Representa el número de unidades del producto que mantiene la máquina.
- dinero acumulado: Dinero. Cantidad de dinero que contiene la máquina.
- dinero transacción: Dinero. Representa el dinero que ha introducido el usuario que quiere adquirir un producto.
- vacía: booleano. Es una *propiedad calculada* que toma el valor verdadero si el stock es cero.

La clase ofrece un constructor que recibe como parámetros las tres primeras propiedades. Los atributos con las propiedades que representan dinero se inicializan con la constante NADA.

Las tres primeras propiedades (producto, precio y stock) se pueden modificar.

La funcionalidad de la máquina es la siguiente:

- Insertar dinero. Recibe como parámetro una cantidad de dinero. Añade este dinero a la transacción.
- Devolver dinero. El objetivo de la operación es retornar el dinero al usuario antes de adquirir el producto. La acción de esta operación es dejar a cero (NADA) el dinero de la transacción.
- Comprar. Si la máquina no está vacía y el valor del dinero de la transacción es igual o superior al precio del producto: añade el dinero de la transacción al dinero acumulado en la máquina, decrementa en una unidad el stock y deja el dinero de la transacción en nada. El método retorna un valor verdadero en este caso. En caso contrario, esto es, que no se cumplan las condiciones, retorna un valor falso. Por último, nótese que, si el usuario introduce más dinero que el valor del producto, el dinero en exceso no se retorna.

Por último, escribe un **programa** que haga una prueba básica de la funcionalidad:

- Crea una máquina que sirva el producto "Agua", con un stock inicial de 10 y un precio de 60 céntimos.
- El usuario introduce una moneda de un euro (utiliza la constante).
- El usuario pide la devolución del dinero.
- El usuario introduce una moneda de 50 céntimos.
- Solicita a la máquina comprar el producto. Muestra el resultado por la consola. Se espera un resultado falso porque no tiene dinero suficiente en la transacción.
- El usuario introduce 20 céntimos.
- Solicita de nuevo la compra. En esta ocasión debe ser aceptada.
- Muestra el stock de la máquina, el dinero acumulado y el dinero de la transacción.