

EJERCICIO 1.

Realiza un programa para gestionar las ventas diarias de una tienda Virtual. Nuestra tienda no permite comprar varios productos a la vez, por lo que cada venta está asociada a un único código de producto (aunque de éste se hayan podido adquirir varias unidades).

Para ello crearemos una *clase RegistroVentasDiaria* con los siguientes atributos:

- Fecha: De tipo `LocalDate`.
- ListaVentas: array con las cantidades abonadas por los consumidores para cada una de las ventas. Por ejemplo, si ese día se han realizado 5 ventas: 500.25, 100.50, 17.75, 250.50, 500.50, 10.60
- ListaCodProducto: array con los códigos de productos para cada una de las ventas de listaVentas. Por ejemplo, para las 5 ventas anteriores: 4, 2 ,3, 3, 4, 4.

Cod 4	Cod. 2	Cod. 3	Cod. 3	Cod 4	Cod 4
500.25	100.50	17.75	250.50	500.50	10.60

- Cada venta de la lista de ventas tendrá su correspondiente código de producto en la lista de códigos.

Los métodos de la clase son:

- Constructor que inicializa los atributos de la clase: día, mes, año, lista de ventas y lista de códigos:
public RegistroDiarioVentas(LocalDate fecha, double[] listaVentas, int[] listaCodigos)
- getters y setters para fecha
- imprimirRegistroVentas, que imprime por consola el objeto Venta con sus atributos.
- double calcularMaximo(): Devuelve la venta con el valor máximo.
- int calcularProductoEstrella(): devuelve el código de producto estrella de la tienda que será aquel más comprado, es decir, aquel cuyos código se repite más veces en la *listaCodProducto*.

Realiza un programa principal que:

- Crea un array de ventas y uno de códigos de producto. No hace falta que lo leas de teclado, puedes darle los valores que desees.
- Crea un objeto Registro de Ventas Diarias e inicializa éste con la fecha de hoy, el array de ventas y el de código de producto creados en el punto anterior.
- Imprime los datos del registro diario de ventas.
- Modifica el año utilizando el método correspondiente.
- Imprime los datos del registro diario de ventas.
- Calcula el máximo de las ventas e imprime su valor
- Calcula el producto Estrella e imprime su valor.

EJERCICIO 2.

- Crear la clase Venta estará forma de la siguiente manera:
 - Atributos
 - codProducto: código del producto de la venta.
 - abonado: con la cantidad abonada por el consumidor para esa venta.
 - Métodos
 - constructor para los dos atributos, getter, setter, toString

- Modificar la clase *RegistrosVentasDiaria* para que esté estructurada de la siguiente manera:
 - Atributos
 - Fecha: como en el caso anterior `LocalDate`
 - listaProductos: que será un array de objetos de la clase `Venta`. De esta forma, nos evitamos tener las listas de códigos y ventas por separado.
 - Métodos
 - El constructor, que tendrá como parámetro de entrada el día, el mes, el año y la capacidad máxima de la lista de ventas.
 - `toString` o `imprimirRegistroVentasDiaria` para imprimir los atributos del objeto.
- boolean `agregarVenta(Venta venta)`: Introduce una venta en el array de ventas.
- int `buscarVenta(int codigo)`: Busca la primera venta con el código introducido como parámetro y devuelve su posición.
- Modifica la clase principal para el caso.