

Gestión de Artículos

Crea un proyecto llamado *GestionDeArticulos*.

El programa simulará una gestión de artículos simple en una pequeña tienda de comestibles. Se darán de alta productos, se pondrán productos en oferta y se eliminarán los productos caducados.

Se utilizarán las siguientes clases:

Clase Artículo

Atributos de la clase: código (String), descripción (String), precio (double), fechaCaducidad (Fecha), productoEstrella (boolean). Además contiene un atributo static porcentajeDescuento (double). El valor inicial del porcentaje de descuento es 30. Todos los atributos son privados.

Escribe en la clase los métodos que consideres necesarios: Constructor o constructores adecuados, Setters y Getters, toString(), etc.

Clase Fecha

Los atributos de la clase son día, mes y año de tipo int. Todos los atributos son privados.

Escribe en la clase todos los métodos que consideres necesarios.

Clase Menu

El programa utilizará la siguiente clase Menu:

```
public class Menu {

    private int opcion;

    public void mostrar() {
        System.out.println(" 1. Nuevo artículo");
        System.out.println(" 2. Mostrar artículos");
        System.out.println(" 3. Poner en oferta");
        System.out.println(" 4. Eliminar artículos");
        System.out.println(" 5. Producto estrella");
        System.out.println(" 6. Eliminar productos estrella");
        System.out.println(" 7. Modificar descuento");
        System.out.println(" 0. FIN");
    }

    public int leer() {
        Scanner sc = new Scanner(System.in);
        do {
            System.out.print("Introduzca opción: ");
            opcion = sc.nextInt();
        } while (opcion < 0 || opcion > 7);
        sc.nextLine(); //limpiar el intro
        return opcion;
    }
}
```

Clase Principal

Se proporciona el inicio de la clase principal:

```
public class GestionDeArticulos {  
    static Articulo[] tienda = new Articulo[100];  
    static int indice;  
    static ArrayList<Articulo> ofertas = new ArrayList();  
    static Scanner sc = new Scanner(System.in);  
  
    public static void main(String[] args) {  
        . . . . .  
    }  
}
```

El array *tienda* contendrá todos los Artículos de la tienda. El número máximo de artículos que puede almacenar esta tienda es 100.

El ArrayList *ofertas* contendrá los artículos que estén en oferta. Inicialmente estará vacío.

Métodos que se deben ejecutar en la clase principal correspondientes a cada opción del menú:

Opción 1: método `nuevoArticulo()`. Añade un nuevo artículo a la tienda.

Se piden por teclado todos los datos del artículo y se añade al array *tienda*. El dato *productoEstrella* no se introduce por teclado. Inicialmente este valor será *false*.

Se debe comprobar que el código del artículo es válido. Los códigos deben tener como mínimo 3 caracteres. Los dos primeros deben ser letras y el resto deben ser dígitos. Si el código no es válido se vuelve a pedir.

En la tienda no puede haber artículos repetidos. Si se intenta añadir un artículo existente se mostrará un mensaje y se volverá al menú principal. Consideramos que dos artículos son iguales si tienen el mismo código.

Tampoco se pueden añadir artículos caducados, o sea, artículos cuya fecha de caducidad sea menor que la fecha actual. Los productos que caduquen hoy sí se admiten. Si la fecha de caducidad no es válida se vuelve a pedir hasta que sea correcta.

Opción 2: método `mostrarArticulos()` Muestra todos los artículos de la tienda ordenados alfabéticamente por descripción. Si el artículo es un producto estrella se muestra PRODUCTO ESTRELLA cuando se muestre el artículo en pantalla. A continuación muestra los artículos en oferta ordenados por precio de menor a mayor. Si dos artículos en oferta tienen el mismo precio, se muestran ordenados por descripción.

Si algunos de los arrays está vacío se muestra un mensaje indicándolo.

Opción 3: método `ponerEnOferta()`. Este método añade al ArrayList *ofertas* los artículos que caducan este mes SIN ELIMINARLOS del array *tienda*.

Esta operación se realiza de la siguiente forma:

- Si un artículo de la tienda caduca este mes se modifica su precio reduciéndolo en el porcentaje indicado en *porcentajeDescuento* y a continuación se añade el artículo al ArrayList de ofertas. Es posible que ese artículo ya se encuentre en oferta (por ejemplo, porque ya hemos ejecutado esta opción de menú con los mismos artículos). En ese caso no se debe hacer nada, ni reducir de nuevo el precio ni añadirlo de nuevo a ofertas.

- Si un artículo de la tienda caduca este mes pero se trata de un producto estrella, en ese caso primero se modifica su precio dejando el precio que tenía originalmente y se cambia el atributo *productoEstrella* a *false*. A continuación se procede como en el punto anterior.

Opción 4: método `eliminarArtículos()`. Este método es el encargado de eliminar artículos caducados o a punto de caducar. Se eliminarán todos los artículos que tengan una fecha de caducidad menor o igual a la actual. Los artículos se eliminan de ambos arrays.

Opción 5: método `obtenerProductoEstrella()`. Se realizará lo siguiente:

Se selecciona un artículo al azar de entre todos los de la tienda, se modifica su precio reduciéndolo a la mitad y se asigna al atributo *productoEstrella* el valor *true*.

Se debe comprobar que este artículo no sea ya un producto estrella o esté ya en oferta. En cualquiera de estos dos casos se selecciona otro artículo.

Debe haber como mínimo dos artículos en la tienda para poder hacerlo. Si no los hay se muestra un mensaje y se vuelve al menú.

También se debe comprobar que no todos los artículos de la tienda estén ya en oferta o sean artículos estrella. En ese caso se muestra un mensaje indicándolo y se vuelve al menú principal.

Opción 6: método `eliminarProductosEstrella()`. Se realizará lo siguiente:

Se modifica el precio de todos los artículos estrella dejando su precio original y cambiando el valor del atributo *productoEstrella* a *false*.

Opción 7: método `modificarDescuento()`. Modifica el valor del porcentaje de descuento (atributo *static*).

Se muestra el valor actual del porcentaje de descuento, se introduce por teclado el nuevo valor y se modifica.