

Documentación Técnica del Proyecto Java

Generated by Doxygen 1.9.8

1 Class Index	1
1.1 Class List	1
2 Class Documentation	3
2.1 Main Class Reference	3
2.2 Producto Class Reference	3
2.2.1 Detailed Description	3
2.2.2 Constructor & Destructor Documentation	4
2.2.2.1 Producto()	4
2.2.3 Member Function Documentation	4
2.2.3.1 actualizarDescripcion()	4
2.2.3.2 agregarStock()	4
2.2.3.3 aplicarDescuento()	5
2.2.3.4 comprarProducto()	5
2.2.3.5 mostrarProducto()	5
2.2.3.6 venderProducto()	6
Index	7

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Main	3
Producto	3

Chapter 2

Class Documentation

2.1 Main Class Reference

Static Public Member Functions

- static void **main** (String[] args)

The documentation for this class was generated from the following file:

- src/Main.java

2.2 Producto Class Reference

Public Member Functions

- [Producto](#) (String nombre, Integer unidades, String descripcion, Double precio)
- void [mostrarProducto](#) ()
- Double [comprarProducto](#) (Integer unidades)
- void [aplicarDescuento](#) (Integer descuento)
- void [agregarStock](#) (Integer unidades)
- void [venderProducto](#) (Integer unidades)
- void [actualizarDescripcion](#) (String nuevaDescripcion)

Public Attributes

- String **nombre**
- Integer **unidades**
- String **descripcion**
- Double **precio**

2.2.1 Detailed Description

Representa un producto dentro de un inventario, incluyendo sus datos básicos como nombre, unidades disponibles, descripción y precio. Permite realizar operaciones comunes como aplicar descuentos, calcular el costo total de una compra y gestionar el stock.

2.2.2 Constructor & Destructor Documentation

2.2.2.1 Producto()

```
Producto.Producto (
    String nombre,
    Integer unidades,
    String descripcion,
    Double precio ) [inline]
```

Crea un nuevo producto con los datos proporcionados.

Parameters

<i>nombre</i>	nombre del producto
<i>unidades</i>	cantidad inicial disponible
<i>descripcion</i>	breve descripción del producto
<i>precio</i>	precio unitario del producto

Exceptions

<i>IllegalArgumentException</i>	si nombre es vacío, unidades o precio son negativos
---------------------------------	-----------------------------------------------------

2.2.3 Member Function Documentation

2.2.3.1 actualizarDescripcion()

```
void Producto.actualizarDescripcion (
    String nuevaDescripcion ) [inline]
```

Actualiza la descripción del producto.

Parameters

<i>nuevaDescripcion</i>	nuevo texto descriptivo para el producto
-------------------------	------------------------------------------

Exceptions

<i>IllegalArgumentException</i>	si la nueva descripción es vacía o nula
---------------------------------	-----------------------------------------

2.2.3.2 agregarStock()

```
void Producto.agregarStock (
    Integer unidades ) [inline]
```

Aumenta el stock del producto sumando las unidades indicadas.

Parameters

<i>unidades</i>	cantidad de unidades a agregar
-----------------	--------------------------------

Exceptions

<i>IllegalArgumentException</i>	si la cantidad es negativa o cero
---------------------------------	-----------------------------------

2.2.3.3 aplicarDescuento()

```
void Producto.aplicarDescuento (
    Integer descuento ) [inline]
```

Aplica un descuento porcentual al precio actual del producto.

Parameters

<i>descuento</i>	porcentaje de descuento a aplicar (0 a 100)
------------------	---------------------------------------------

Exceptions

<i>IllegalArgumentException</i>	si el descuento es negativo o mayor a 100
---------------------------------	-------------------------------------------

2.2.3.4 comprarProducto()

```
Double Producto.comprarProducto (
    Integer unidades ) [inline]
```

Calcula el precio total al comprar una cantidad específica de unidades.

Parameters

<i>unidades</i>	cantidad de unidades que se desea comprar
-----------------	-------------------------------------------

Returns

el valor total de la compra

Exceptions

<i>IllegalArgumentException</i>	si la cantidad de unidades es negativa o cero
---------------------------------	-----------------------------------------------

2.2.3.5 mostrarProducto()

```
void Producto.mostrarProducto ( ) [inline]
```

Muestra la información completa del producto en consola.

2.2.3.6 venderProducto()

```
void Producto.venderProducto (
    Integer unidades ) [inline]
```

Resta unidades del stock al vender el producto.

Parameters

<i>unidades</i>	cantidad de unidades a vender
-----------------	-------------------------------

Exceptions

<i>IllegalArgumentException</i>	si la cantidad es negativa o cero
<i>IllegalStateException</i>	si la cantidad a vender excede el stock disponible

The documentation for this class was generated from the following file:

- src/Producto.java

Index

actualizarDescripcion

Producto, [4](#)

agregarStock

Producto, [4](#)

aplicarDescuento

Producto, [5](#)

comprarProducto

Producto, [5](#)

Main, [3](#)

mostrarProducto

Producto, [5](#)

Producto, [3](#)

actualizarDescripcion, [4](#)

agregarStock, [4](#)

aplicarDescuento, [5](#)

comprarProducto, [5](#)

mostrarProducto, [5](#)

Producto, [4](#)

venderProducto, [6](#)

venderProducto

Producto, [6](#)