|  |  |
| --- | --- |
| Una imagen de una carretera de curvas con árboles  PATRONES DE DISEÑO  Implementación de los patrones de diseño. | Descripción breve  Este documento contiene las especificaciones sobre la implementación de los patrones de diseño solicitados para la elaboración del proyecto integrador para la asignatura de diseño de software.   * Eusebio Ajax Santos. * Emmanuel Chablé Collí. * Nicolás Canul Ibarra. |

Contenido

[Patrón de diseño: Decorator. 3](#_Toc42531457)

[Patrón de diseño: Iterator. 4](#_Toc42531458)

[Patrón de diseño: Factory. 5](#_Toc42531459)

[Patrón de diseño: Singleton. 6](#_Toc42531460)

[Patrón de diseño: State. 7](#_Toc42531461)

[Patrón de diseño: Observer. 8](#_Toc42531462)

# Patrón de diseño: Decorator.

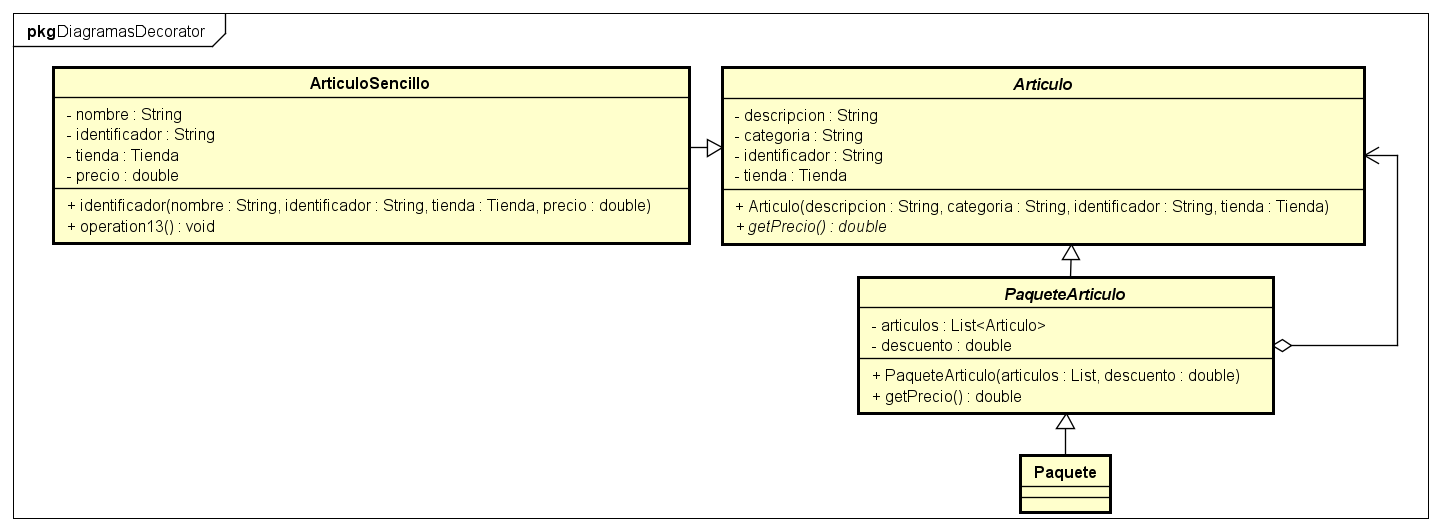
Diagrama de clases:

Ilustración 1. Diagrama de clases del patrón de diseño **Decorator**.

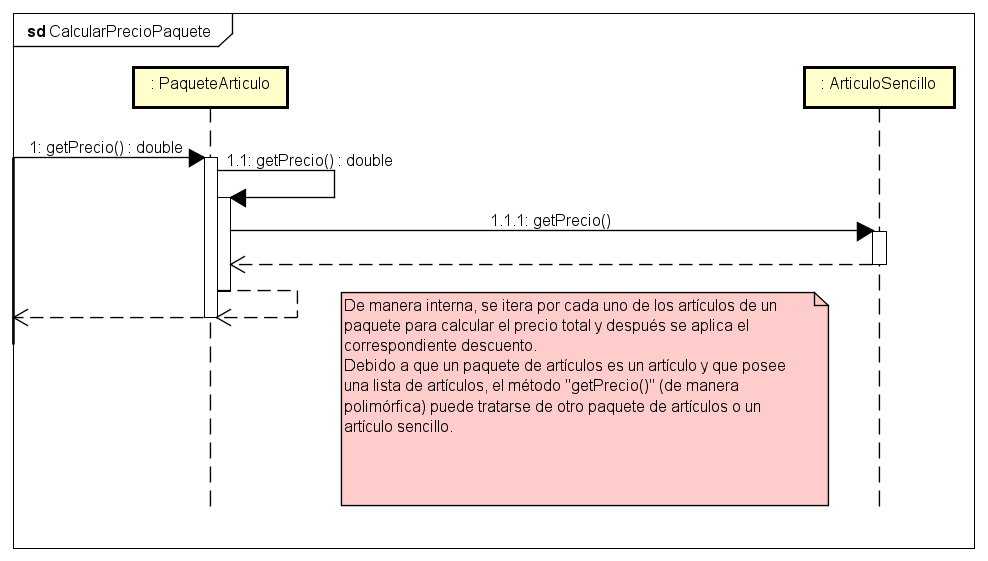
Diagrama de secuencia:

Ilustración 2. Diagrama de secuencia para el cálculo del precio de un paquete de artículos.

Observaciones:

Para esta situación, el uso del patrón **Decorator** nos es muy útil para crear paquetes de artículos e incluso paquetes de paquetes, ya que cada paquete es un artículo. De igual forma, el cálculo del precio para un paquete se puede ejecutar de una manera muy sencilla siguiendo el diseño que este patrón nos ofrece.

# Patrón de diseño: Iterator.

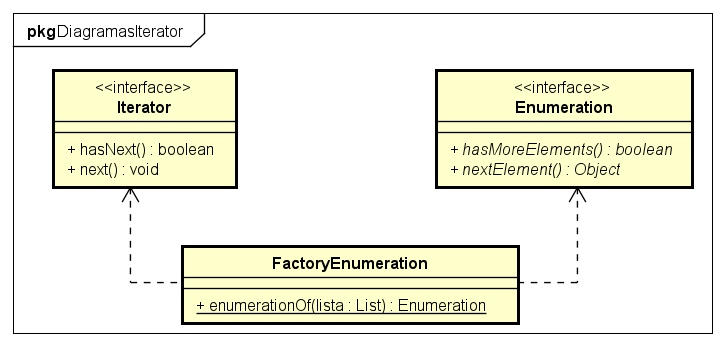
Diagrama de clases:

Ilustración 3. Diagrama de clases para el patrón de diseño **Iterator**.

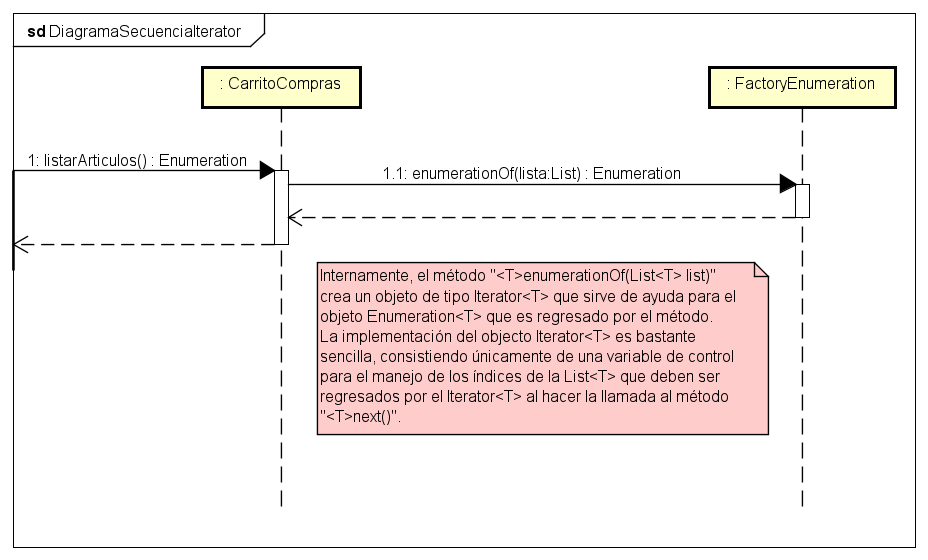
Diagrama de secuencia:

Ilustración 4. Diagrama de secuencia para la creación de objeto de tipo **Enumeration**<T> a partir de una List<T>.

Explicación:

Los objectos **Enumeration**<T> son utilizados para recorrer una colección de elementos en algún orden no especificado. Para nuestro caso, estos objectos se usan para iterar sobre algún conjunto de datos (clientes o artículos) y posteriormente ir mostrándolos en pantalla (tablas o listas).

# Patrón de diseño: Abstract factory.

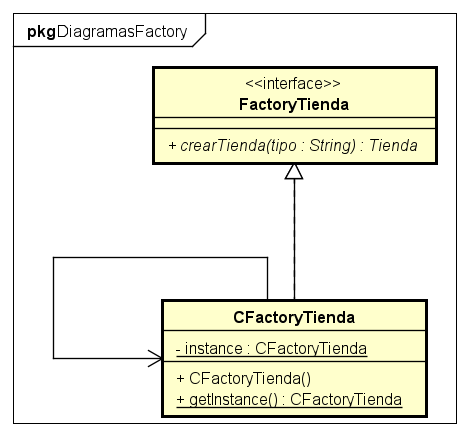
Diagrama de clases:

Diagrama de secuencia:

Explicación:

# Patrón de diseño: Singleton.

Diagrama de clases:

Diagrama de secuencia:

Explicación:

# Patrón de diseño: State.

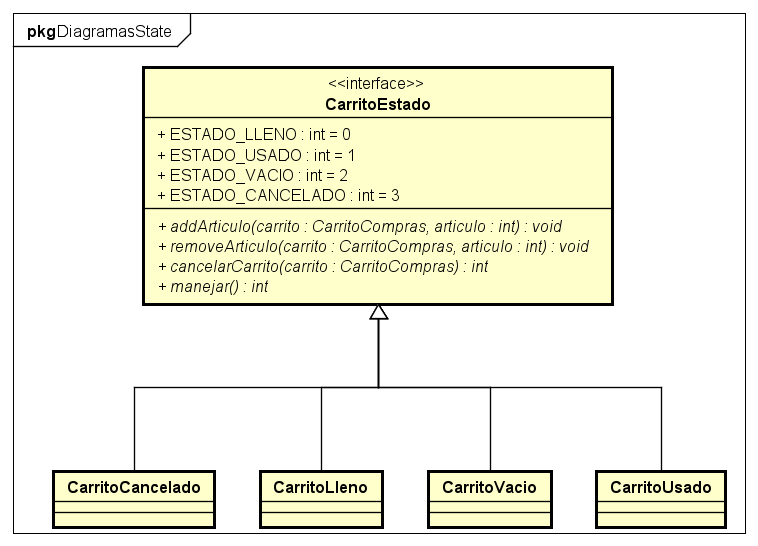
Diagrama de clases:

Diagrama de secuencia:

Explicación:

# Patrón de diseño: Observer.

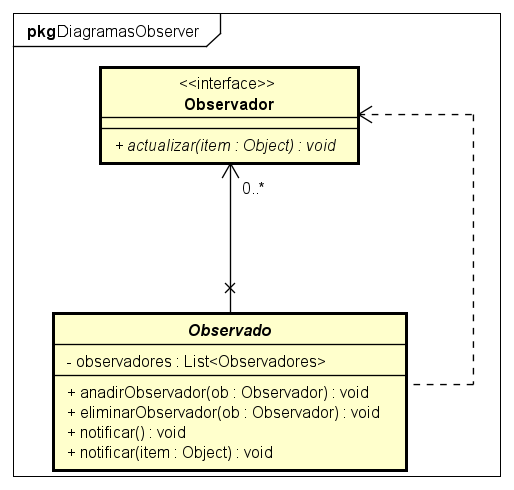
Diagrama de clases:

Diagrama de secuencia:

Explicación: