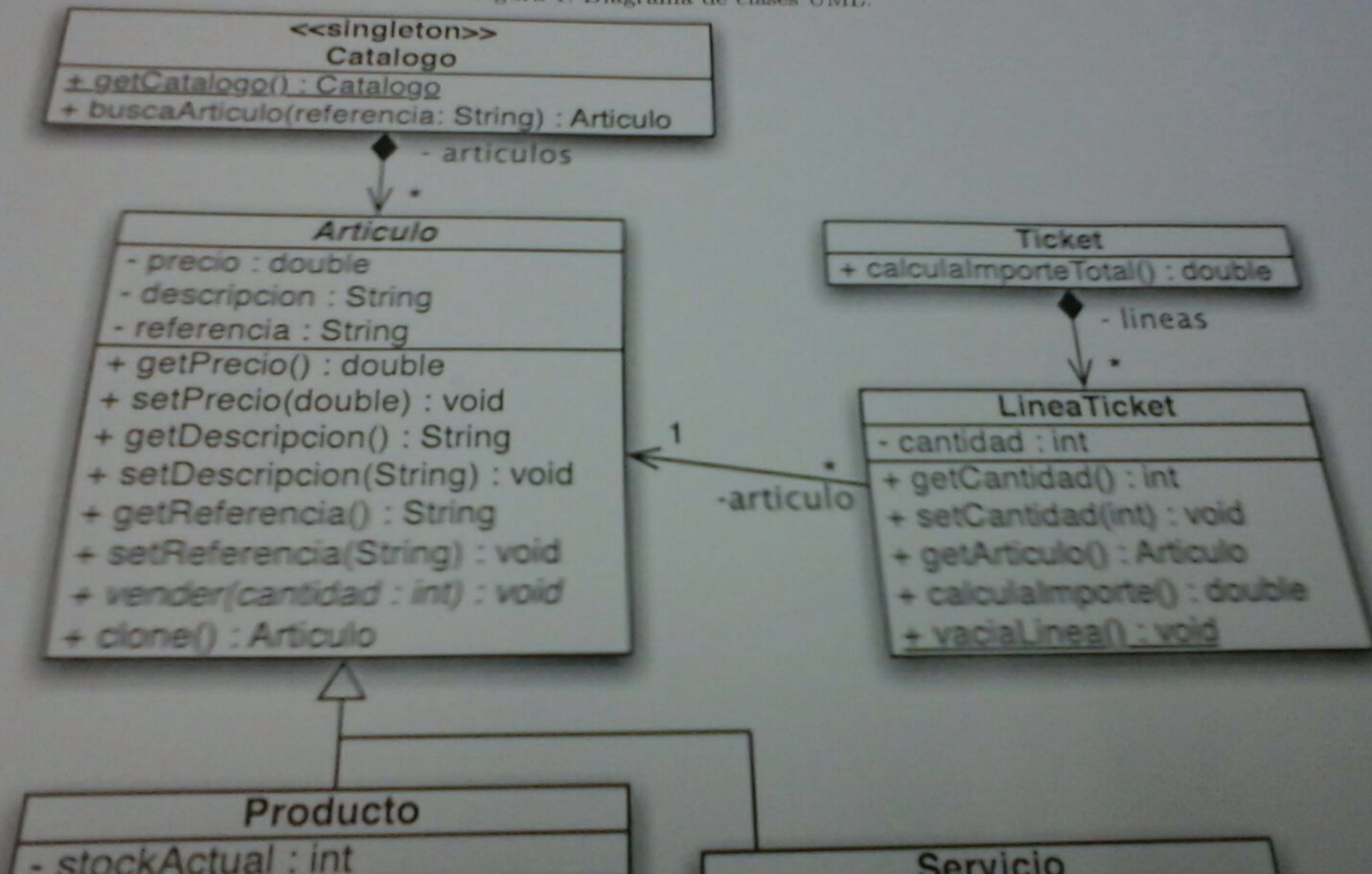
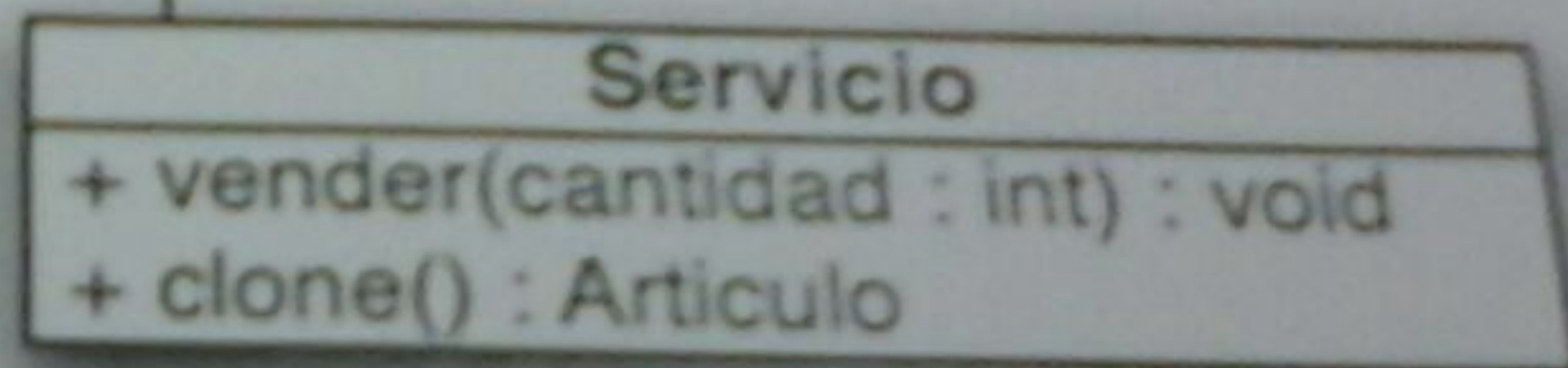
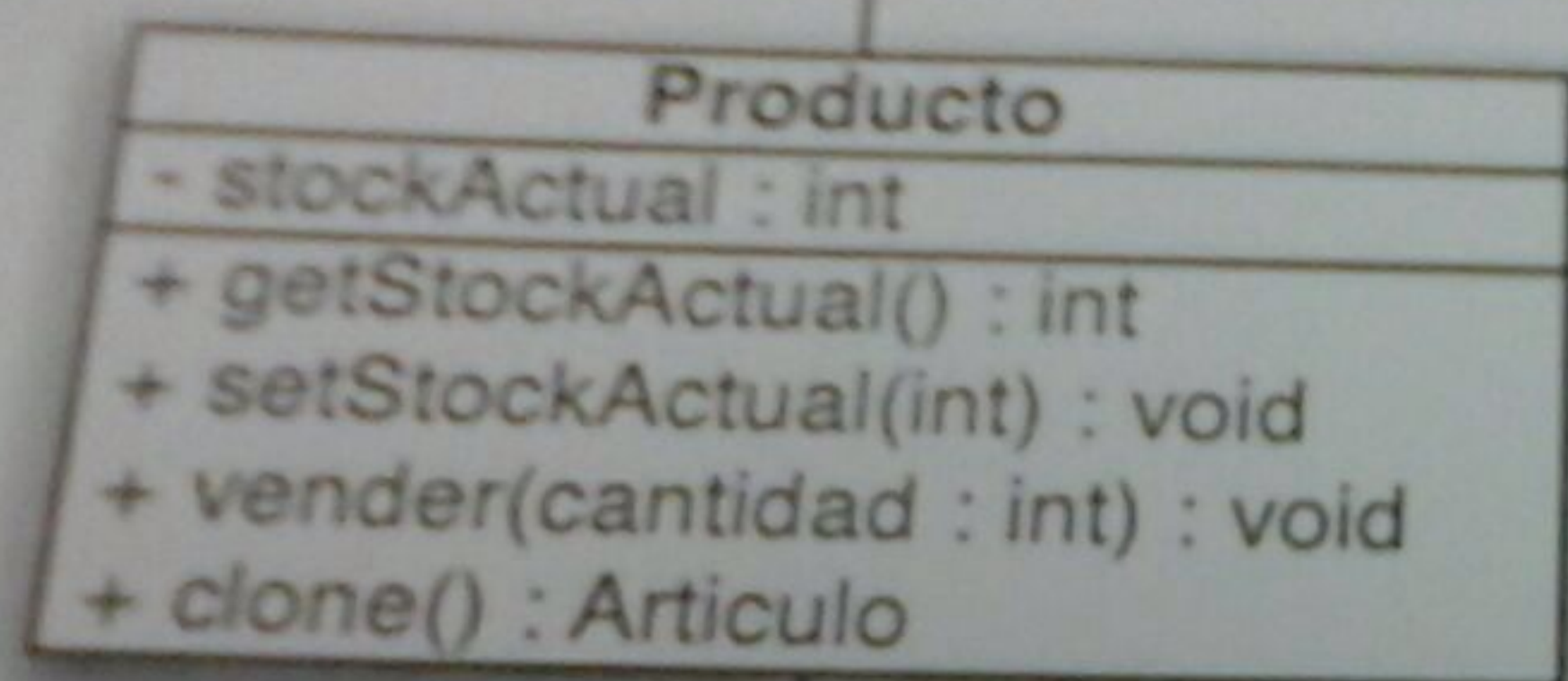


Figura 1: Diagrama de clases UML.

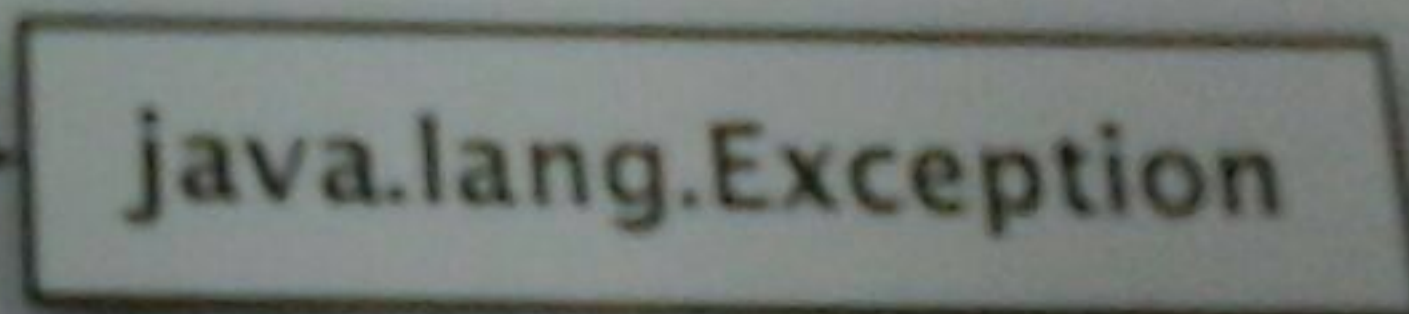
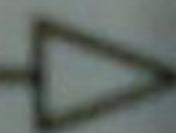
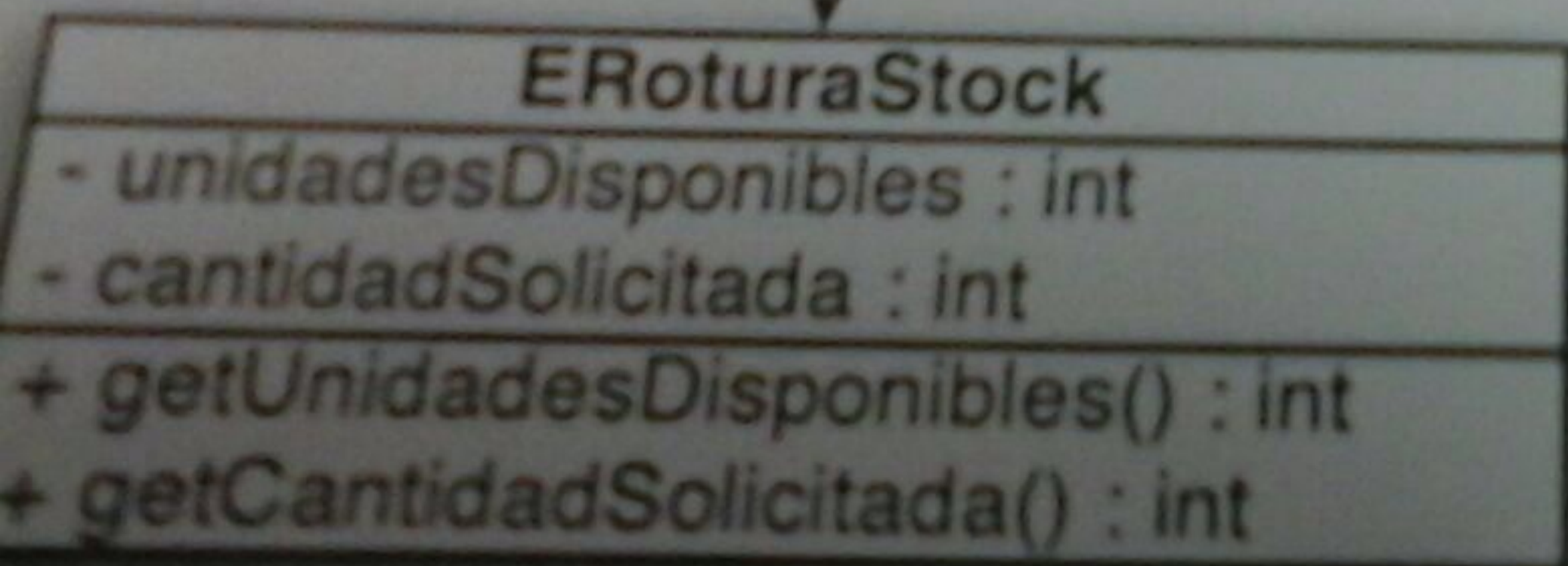


+ vender(cantidad : int) : void
+ clone() : Artículo

+ getArticulo() : Artículo
+ calculaImporte() : double
+ vaciaLinea() : void



<<uses>>



1. Todas las clases que representan excepciones en Java tienen a la clase `Object` como una de sus superclases.
2. La clase `ERoturaStock` es una clase de tipo excepción y en el UML tiene una relación de dependencia con `Producto` porque usa información proporcionada por ésta última.
3. Viendo el diagrama, podemos decir que diferentes objetos de tipo `Ticket` pueden compartir objetos de tipo `LineaTicket` a través de la relación del UML.
4. El método invocado por un objeto en respuesta a un mensaje viene dado entre otras cosas por la clase del objeto receptor en tiempo de ejecución si el método invocado tiene enlace dinámico.
5. Si en una clase no se declara implícita o explícitamente un constructor por defecto, no se pueden crear instancias de esa clase.
6. Si en una clase no se declara implícita o explícitamente ningún constructor, no se pueden crear instancias de esa clase.
7. En java un objeto de clase `Class` es creado en tiempo de ejecución cada vez que se crea un nuevo objeto en el programa.
8. La inversión de control implica que al usar un framework de java el método `main()`, es decir el punto de entrada de la aplicación, formara siempre parte del código del framework.

public void vender(int cantidad) { // cuerpo vacio }

9 En Java, si un método no abstracto f() definido en una superclase es sobrescrito
invocar el método sobrescrito desde la subclase mediante la instrucción super.f

10 Dado el diagrama UML, el siguiente código de Java contiene un error de compilación.
De los métodos invocados declara lanzar la excepción ENotImplemented()

```
Articulo a= new Servicio();  
a.setDescripcion("Dinner");  
a.vender(10);
```

11 Dado el diagrama UML, el siguiente código de Java contiene un error de compilación.
De los métodos invocados declara lanzar la excepción ENotImplemented()

```
Articulo a= new Producto();  
a.setDescripcion("Table");  
a.setPrecioActual(20);  
a.vender(10);
```

Todas las clases que representan excepciones en Java tienen a la clase