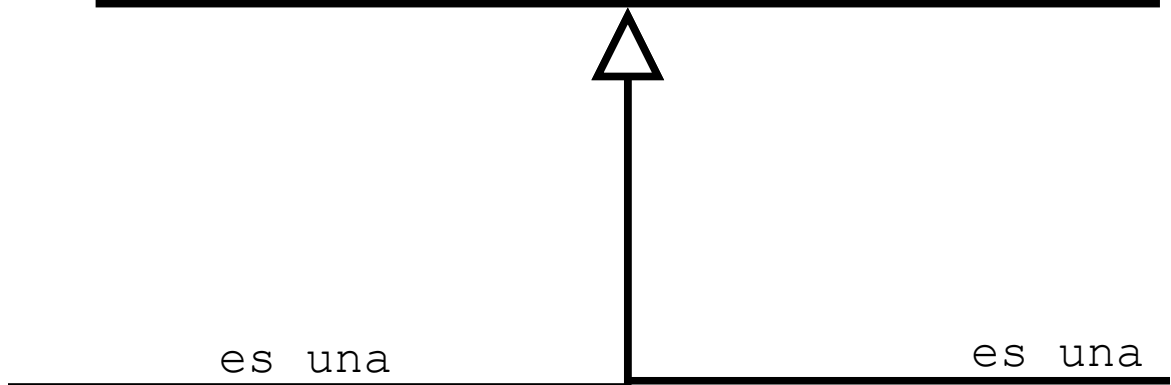
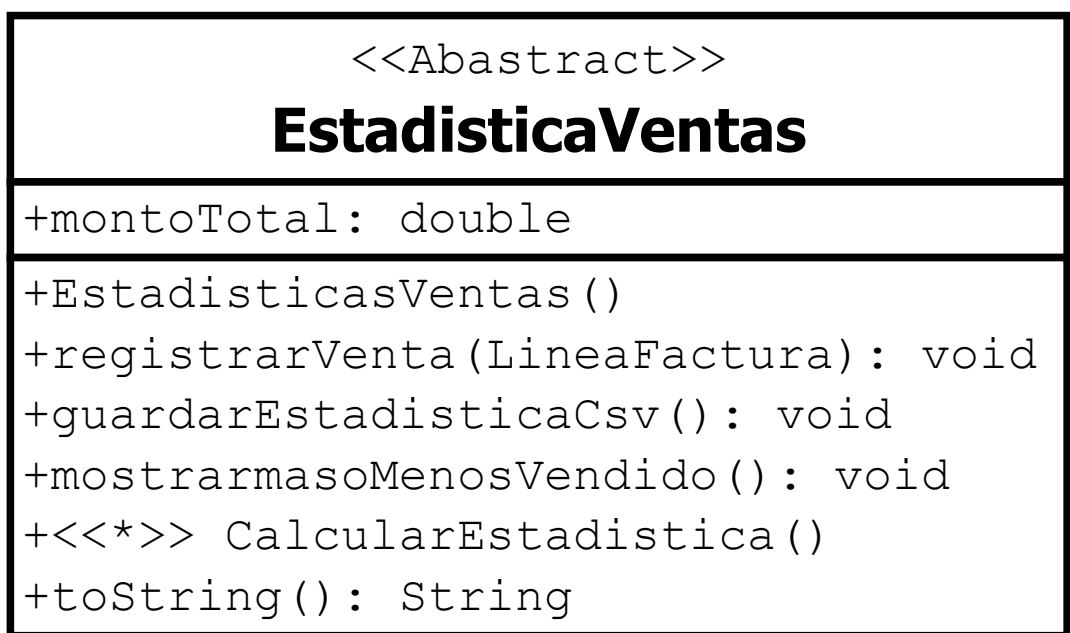


Contro

lador

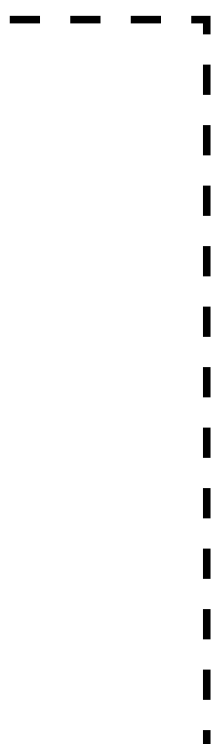




- - - - -

└──┘

.....



Entrada



de texto de los Pr



roductos



Vista

Main



EstadisticaPo

```
+categoria: String  
+cantidadCategoriaVend.  
+montoTotal: double
```

```
+EstadisticaPorCategor.  
+EstadisticaPorCategor.  
+calcularEstadistica()
```

orCategoria
ida: int
ia() ia(categoria:String) : void

E
+idProducto: < +nombreProduct +cantidadProdu +montoTotal: <
+EstadisticaPo +EstadisticaPo +calcularEstac

+idProducto: +nombre: Stri +cantidadStoc

estadísticaPorProducto

String

to: String

ctoVendida: int

double

orProducto()

orProducto(id:String,nombre:String)

distica(): void

tiene

1

<<Abstract>>

Producto

String

ing

sta int

Inventario

+productos: ArrayList<Producto>

+Inventario()

+Inventario(productos:ArrayList<Product

+agregarProducto(): void

+buscarProducto(): Producto

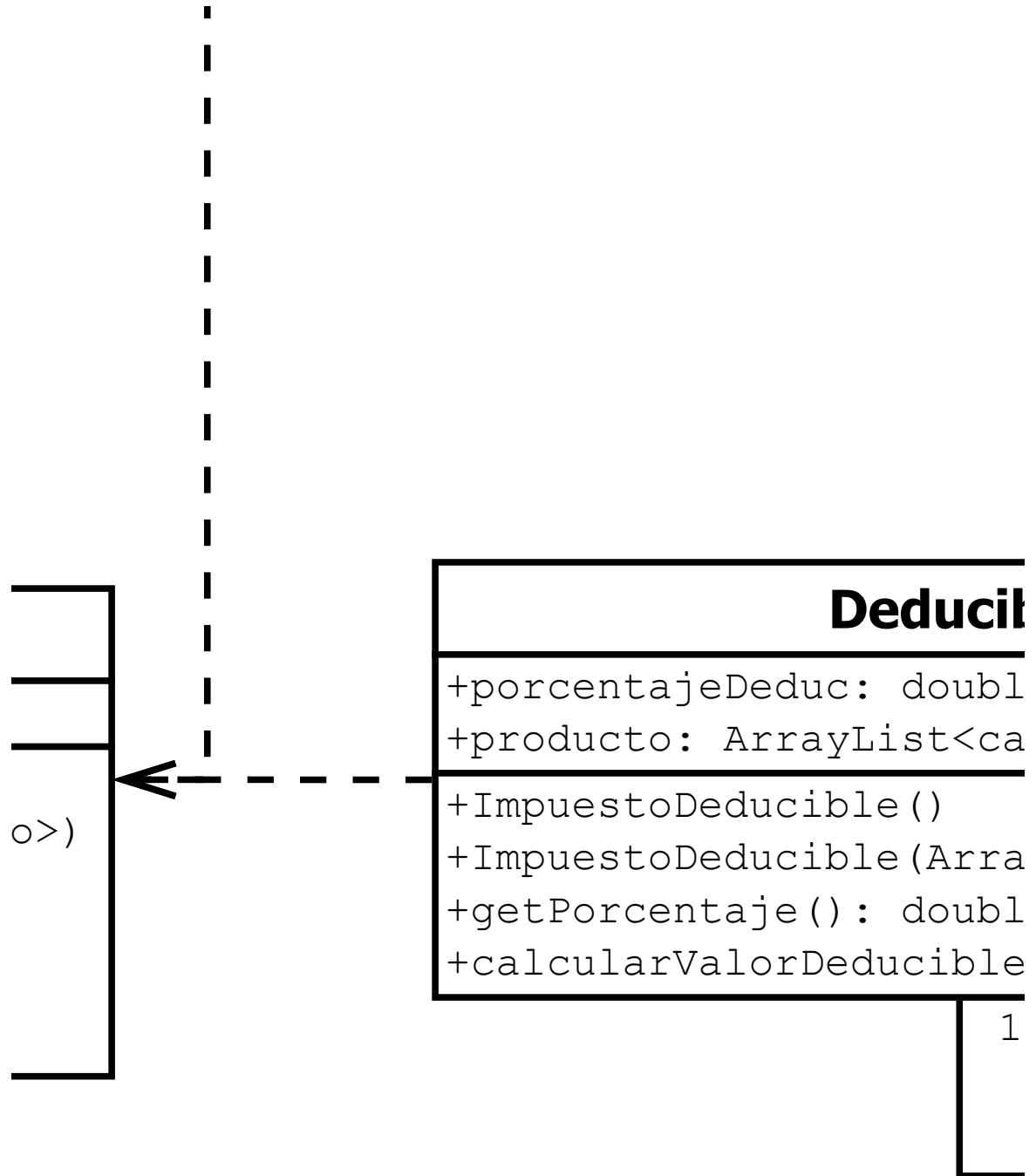
+ListarInventario(): void

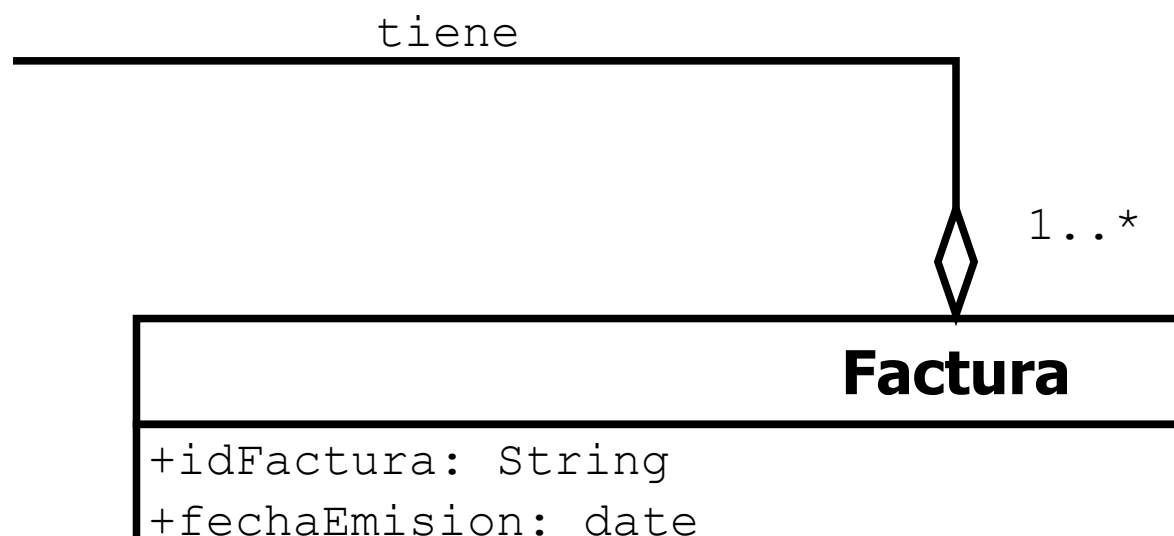
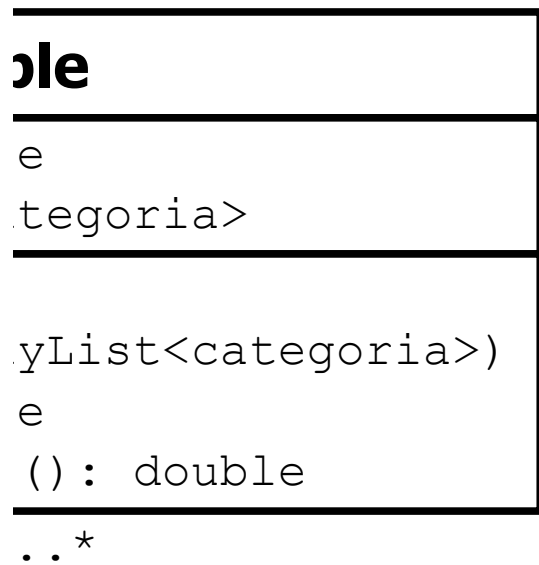
+toString(): String

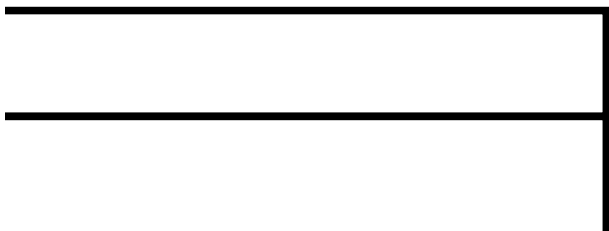


1..*

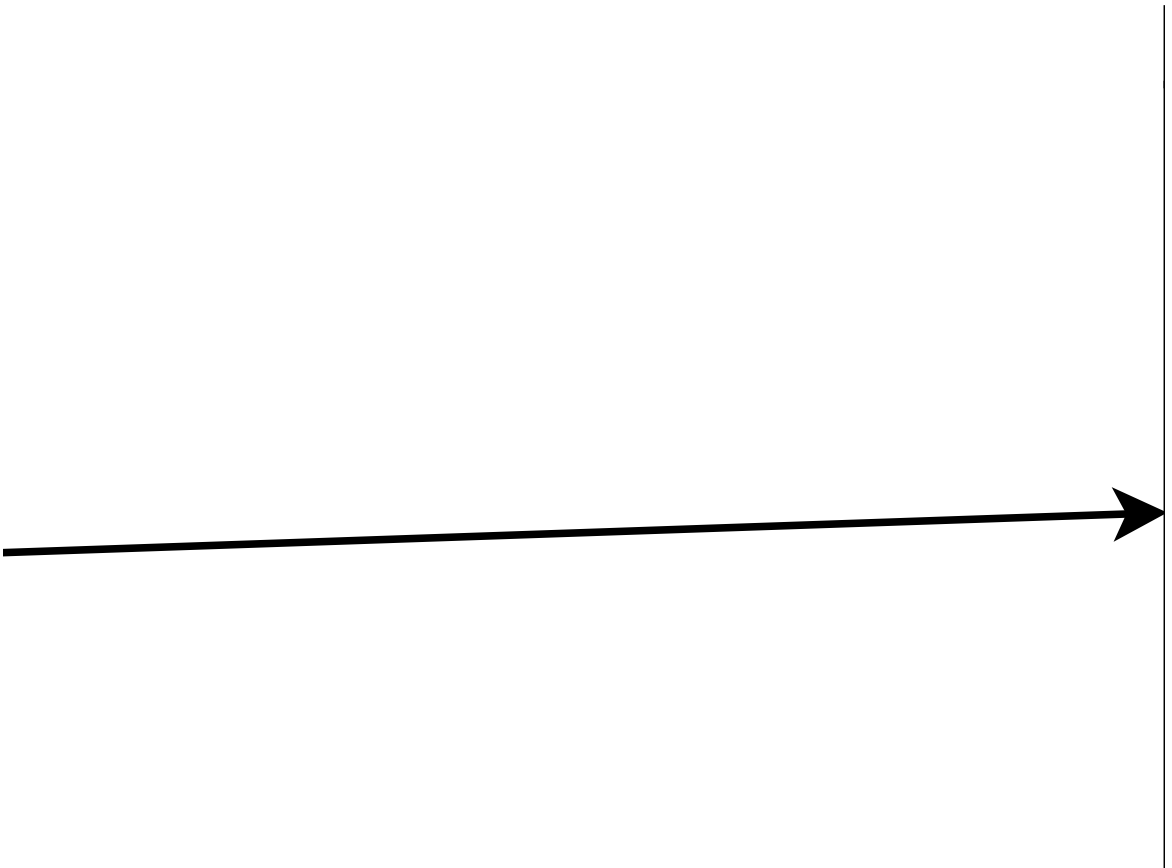










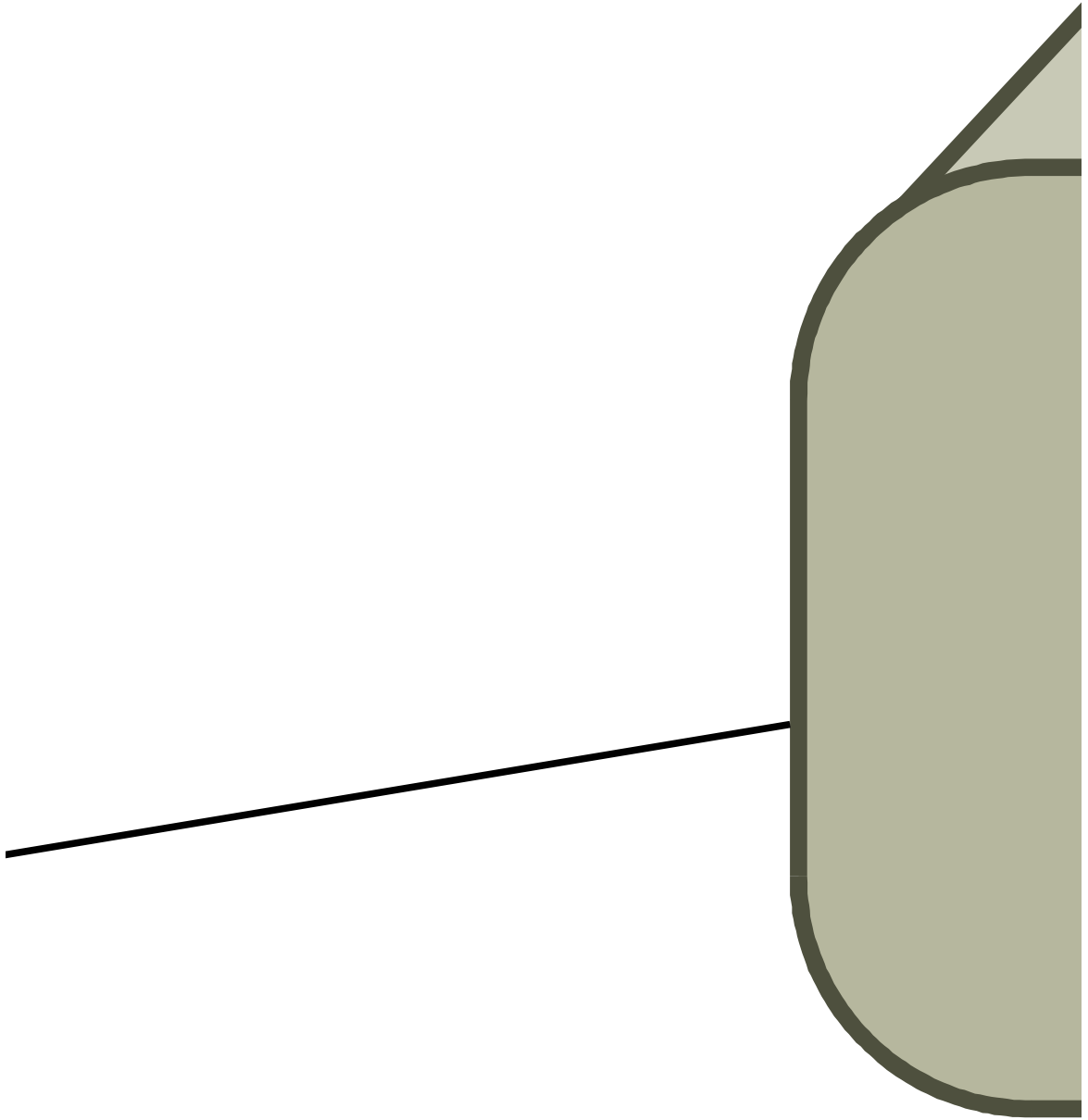


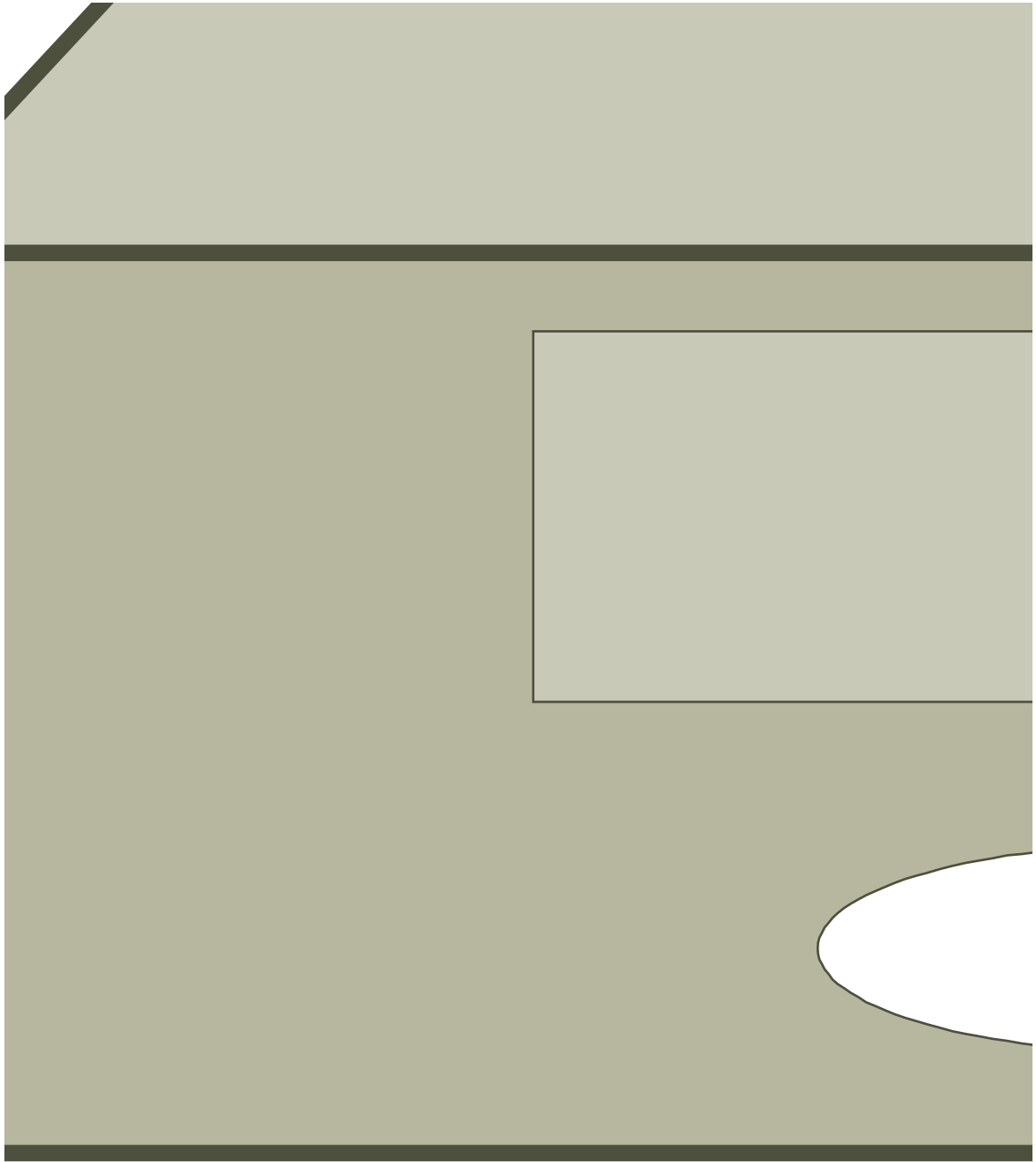
Modelo

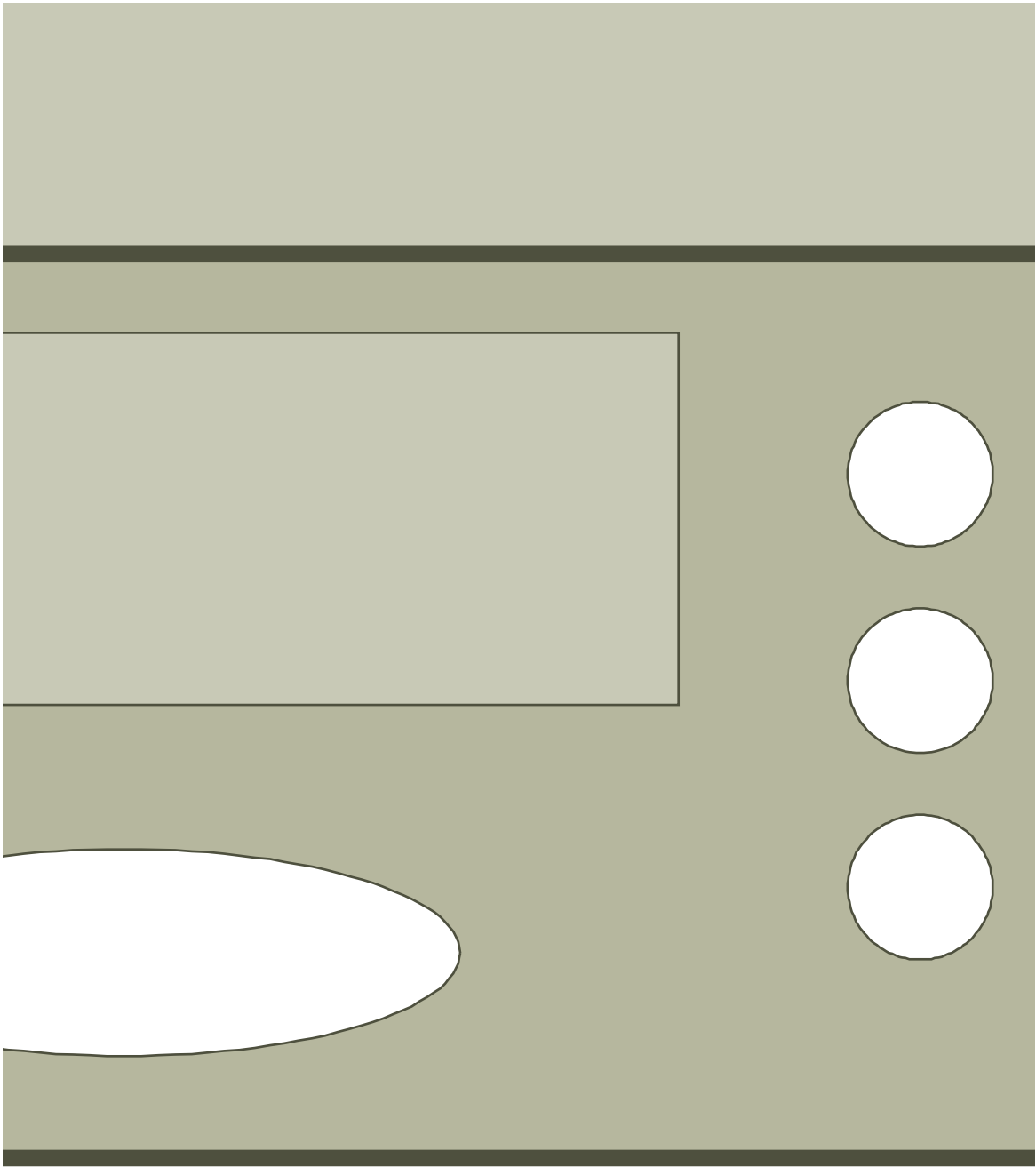
LeerDatosArchiho

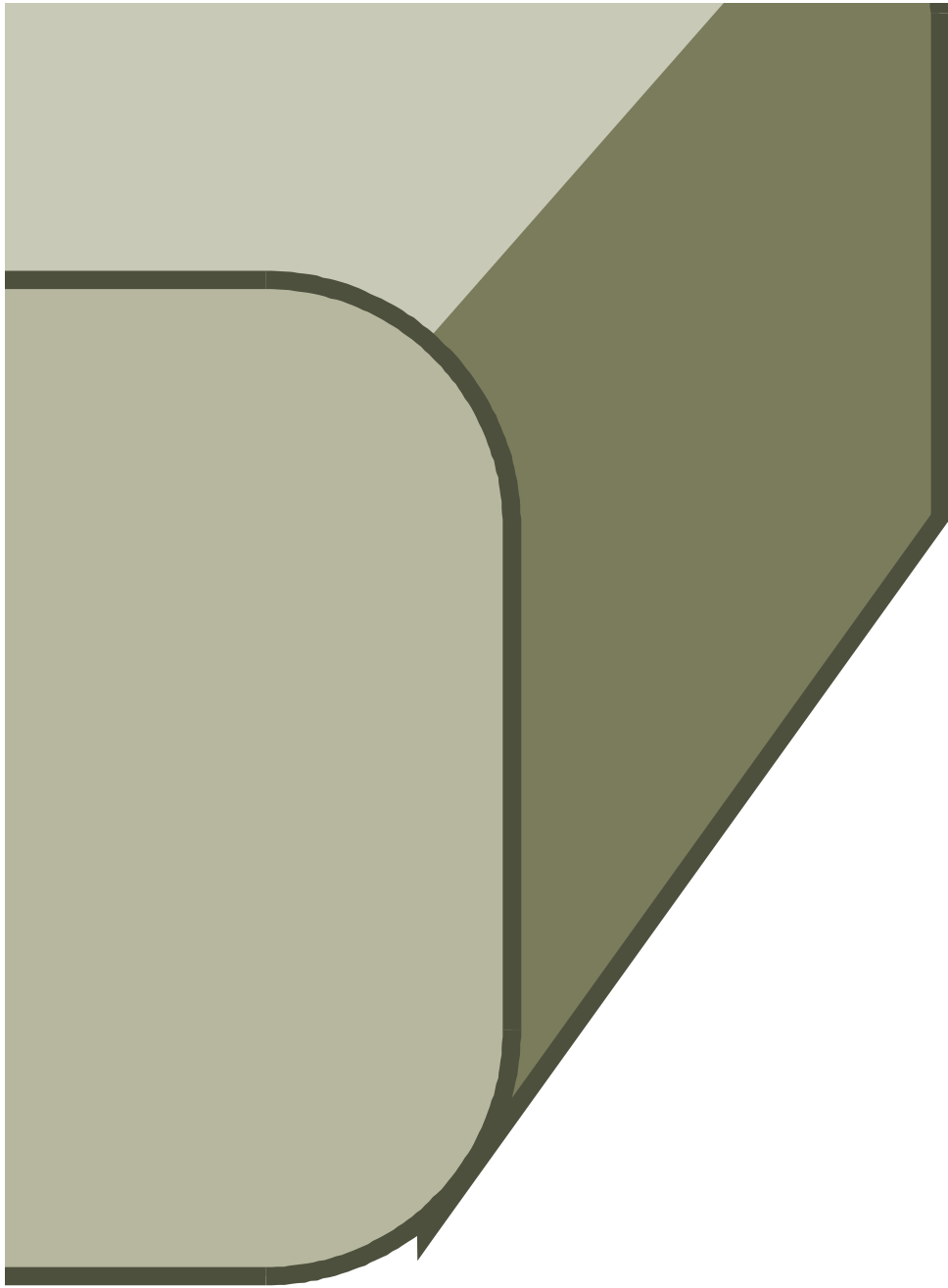
GuardarDatosArchivo

GuardarBaseDatos









L



	+pre
	+Pro
	+Pro
	+de

```

+cantidadStock
+categoria: String
+fechaCaducidad: Date

+producto()
+producto(nombre: String, cantidadStock: Int, categoria: String, fechaCaducidad: Date)
+setIdProducto(id: String)
+setCantidad(cantidadStock: Int)
+setCategoria(categoria: String)
+getIdProducto(): String
+getCantidad(): Int
+getCategoria(): String
+disminuirStock(cantidad: Int)
+aumentarStock(cantidad: Int)
+<<*>> determinarPrecio(): Double
+toString(): String

```

es un

ProductoNormal

```

precioNormal: double

```

```

productoNormal()

```

```

productoNormal(id: String, nombre: String,
               cantidadStock: Int, categoria: String,
               fechaCaducidad: Date, precioNormal: Double)

```

```

terminarPrecio(): void

```


Stock: int

String

date: Date

nombre:String, cantidadStock:int,
categoria:String)

setNombre(nomProducto:String): void

setCantidad(cantidad:int): void

setCategoria(categoria:String): void

getNombre(): string

getCantidad(): int

getCategoria(): String

decrementarStock(cantidad:int): boolean

incrementarStock(): void

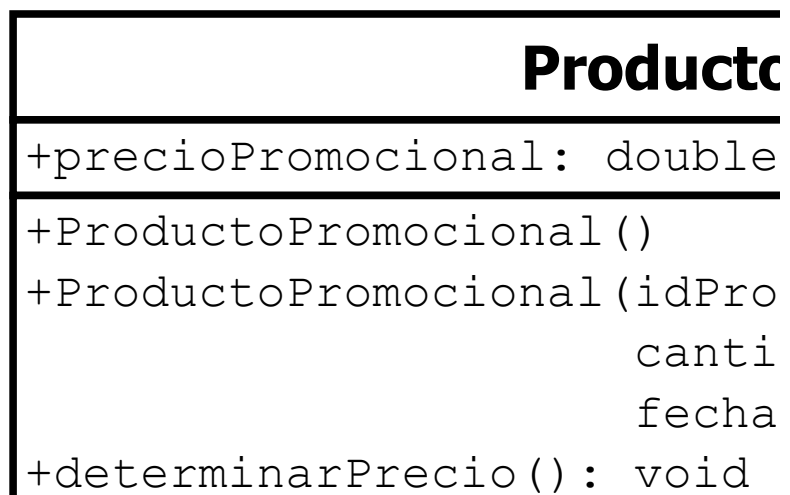
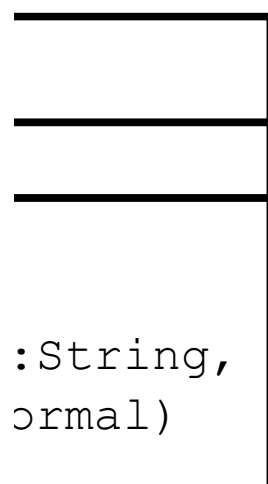
determinarPrecio(): void

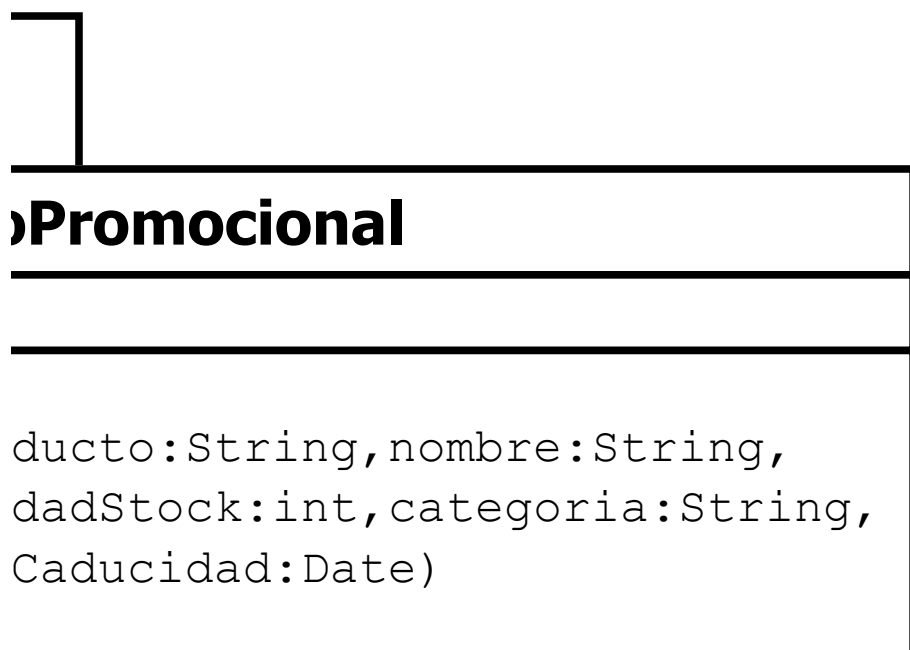
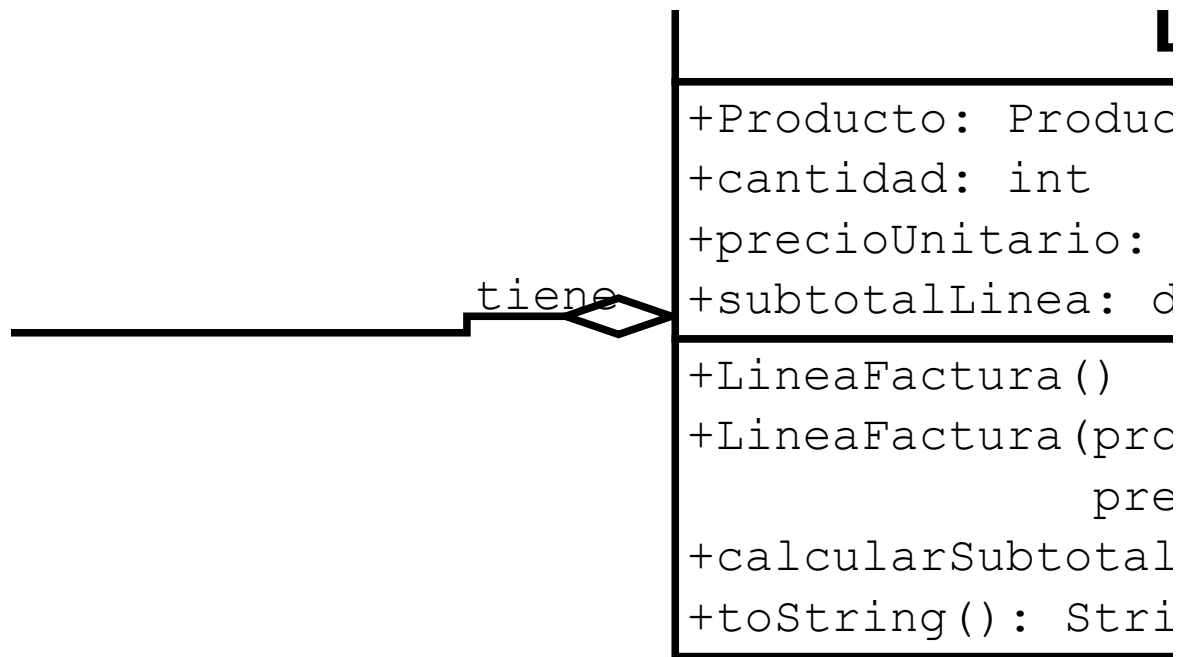
String

1..*



es un





LineaFactura

id: int

precio: double

cantidad: double

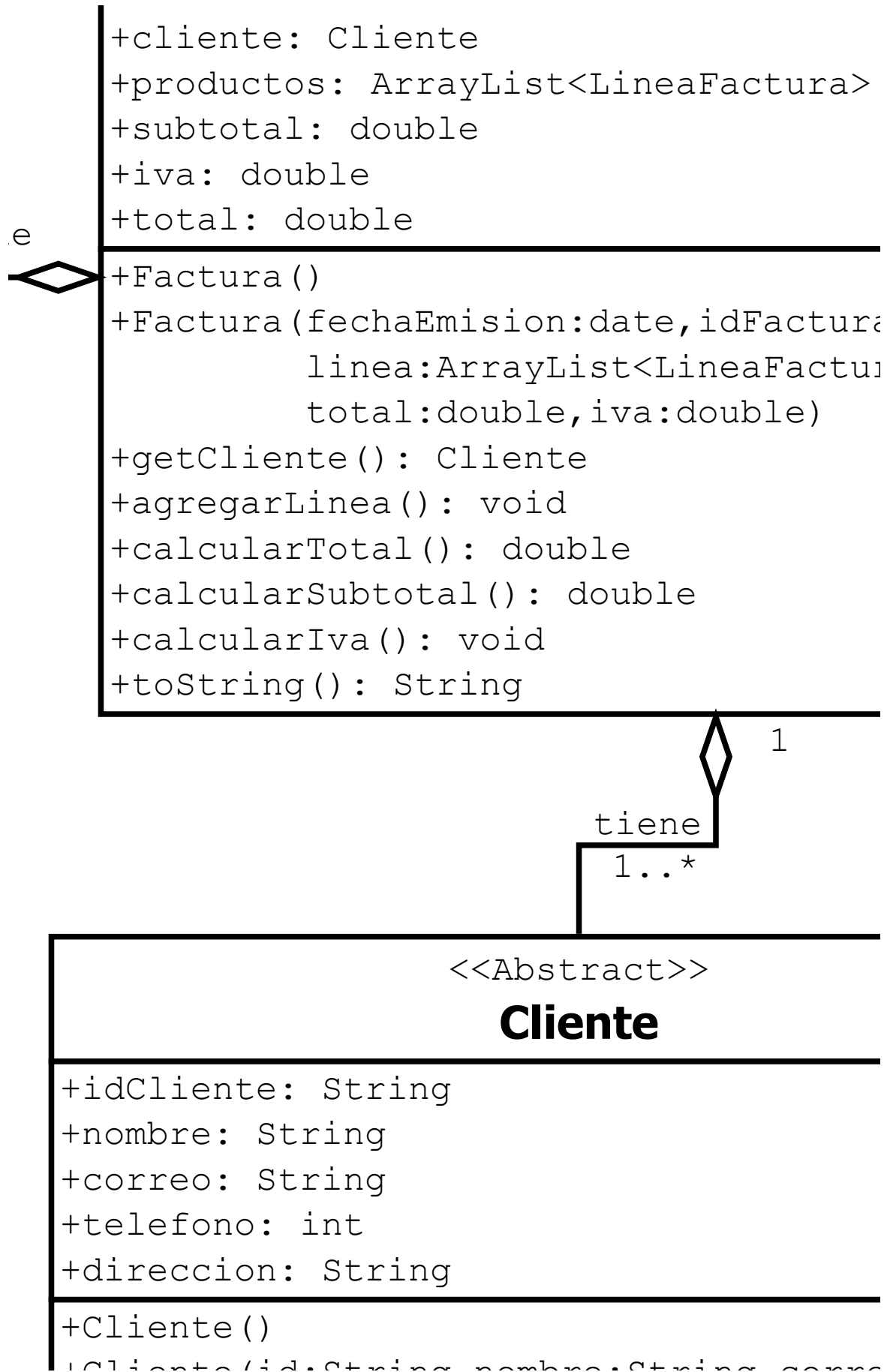
producto: Producto, cantidad: int,
precioUnitario: double)

get(): void

toString(): String

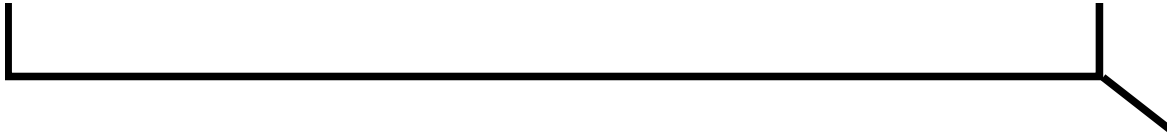
1..*

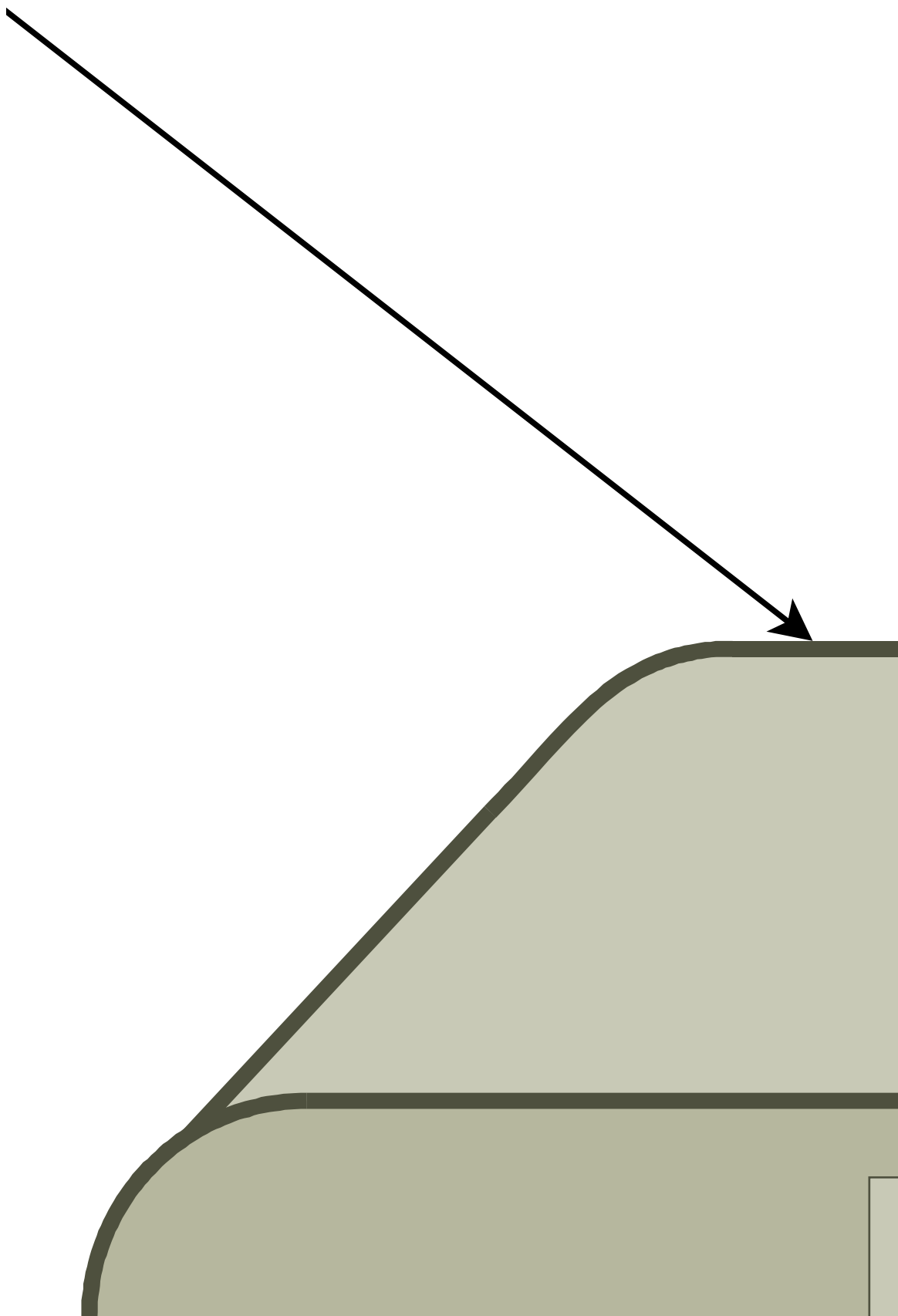
tien



```
a:Random(String),  
ra>,subtotal:double,
```

```
co:String
```

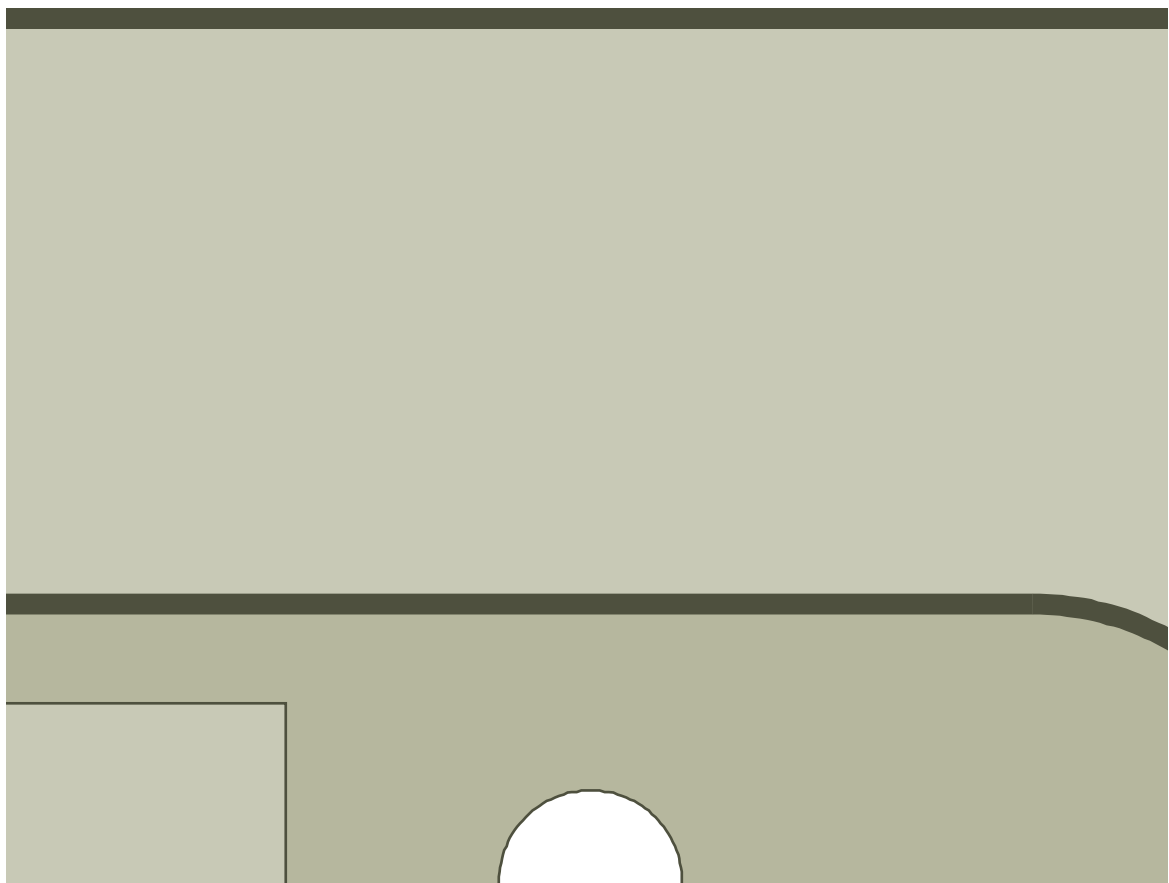


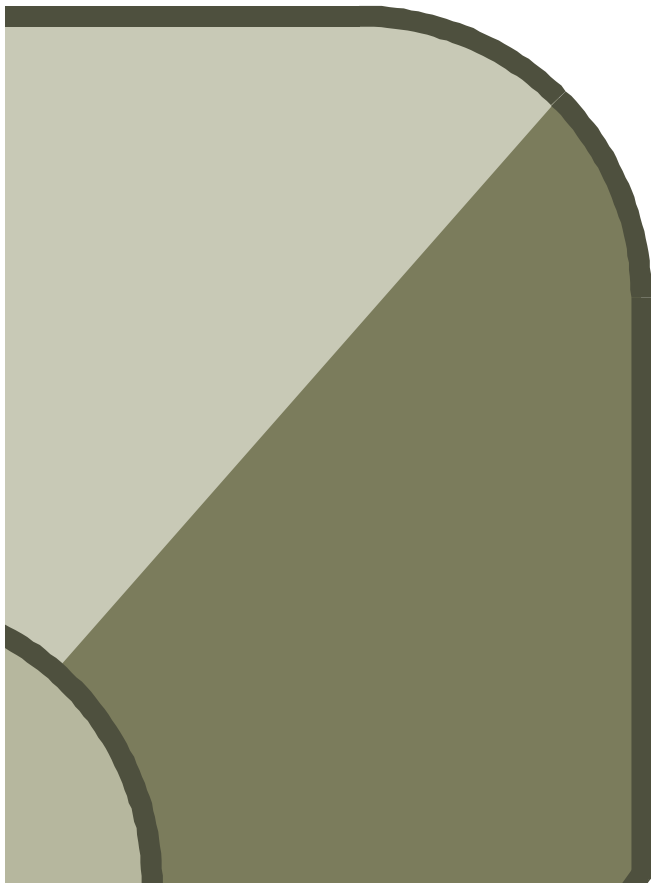


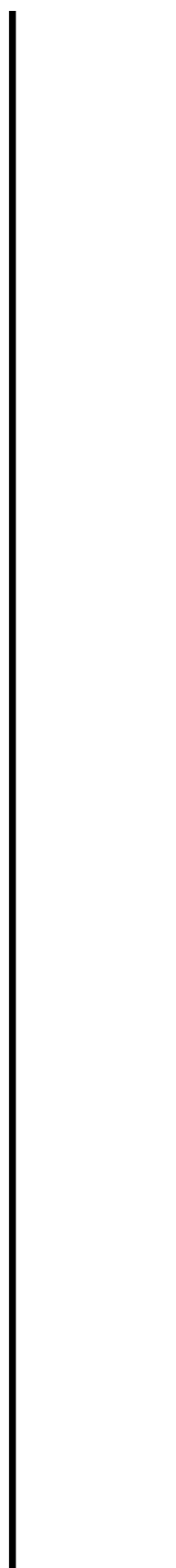
Serealizacion de l



as factudas







+to:

Stirng(): String

+toString(): String





```
classDiagram
    class ClienteNormal {
        +puntosCuponDescuento: int
        +ClienteNormal()
        +ClienteNormal(id:String, nombre: String, numeroTelefono: int, puntosDescuento: int)
        +determinarDescuento(): void
    }
```

ClienteNormal

+puntosCuponDescuento: int

+ClienteNormal()

+ClienteNormal(id:String,nombre:
numeroTelefono:int,
puntosDescuento:int)

+determinarDescuento(): void

```
+cliente(id:String,nombre:String,correo:
        numeroTelefono:int,direccion:
+<<*>> determinarDescuento(): void
+toString(): String
```



es un

nal

```
:String,correo:String,
nt,direccion:String,
int)
```

```
+punt
```

```
+cli
```

```
+Cli
```

```
+det
```

```
eo:String,  
String)
```

es un

ClienteNatural

```
tosCompra: int
```

```
enteNatural()
```

```
enteNatural(id:String,nombre:String,  
            correo:String,numeroTelefono  
            direccion:String)
```

```
erminarDescuento(): void
```

`:int,`



