Contro







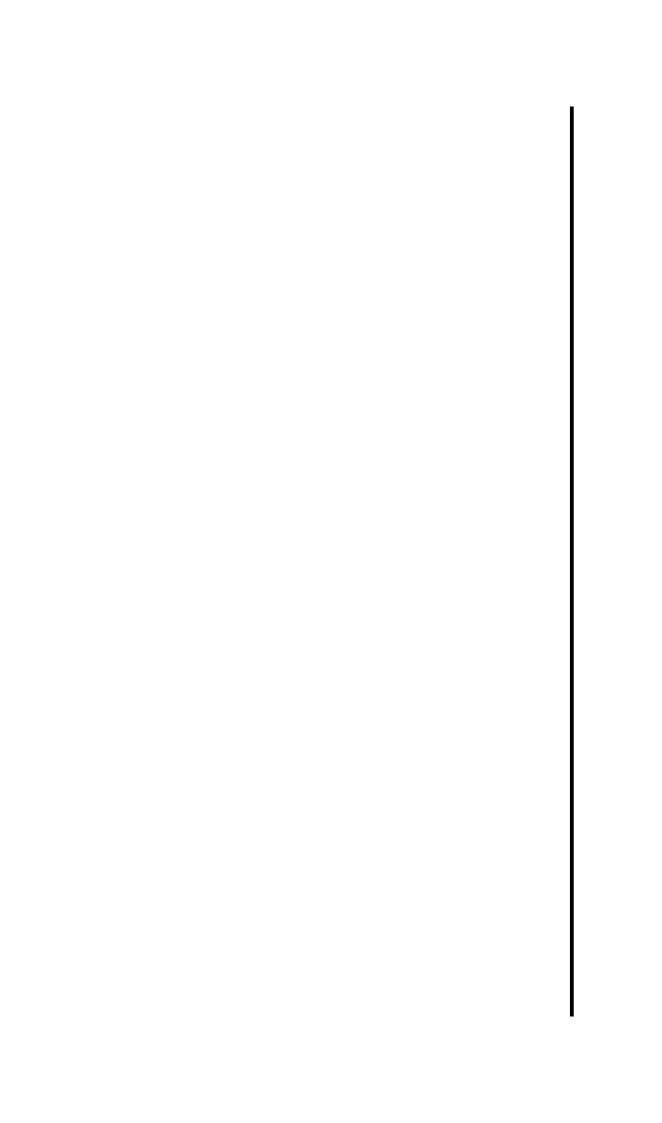












#### <<Abastract>>

#### **EstadisticaVentas**

```
+montoTotal: double
```

```
+EstadisticasVentas()
```

+registrarVenta(LineaFactura): void

+guardarEstadisticaCsv(): void

+mostrarmasoMenosVendido(): void

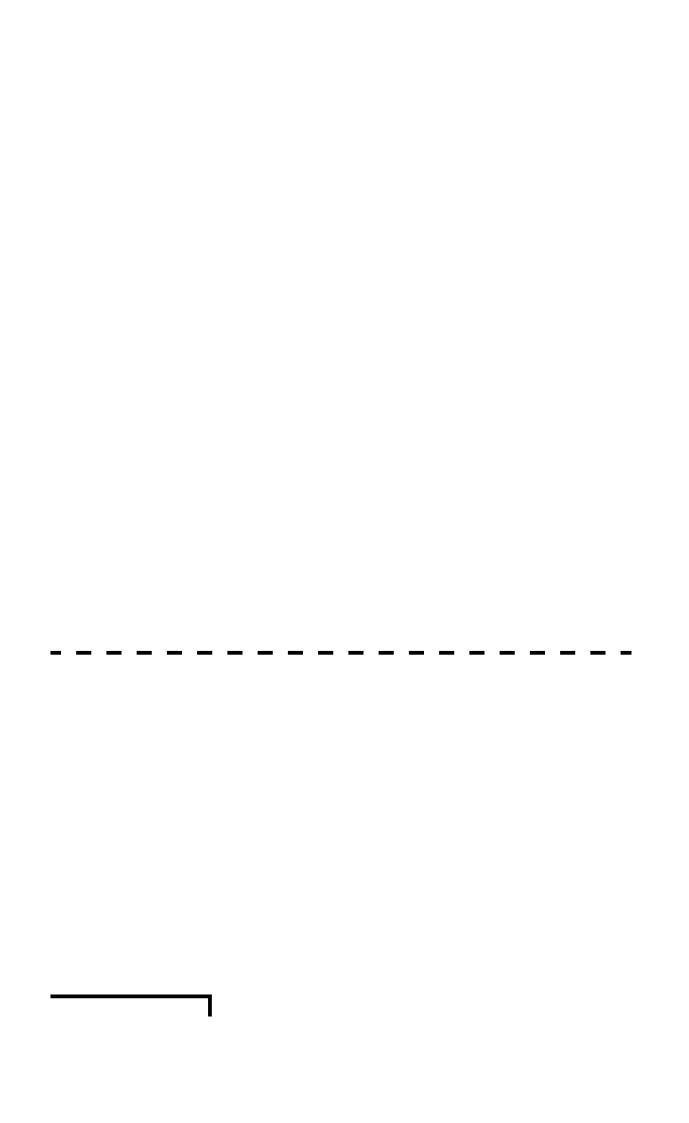
+<<\*>> CalcularEstadistica()

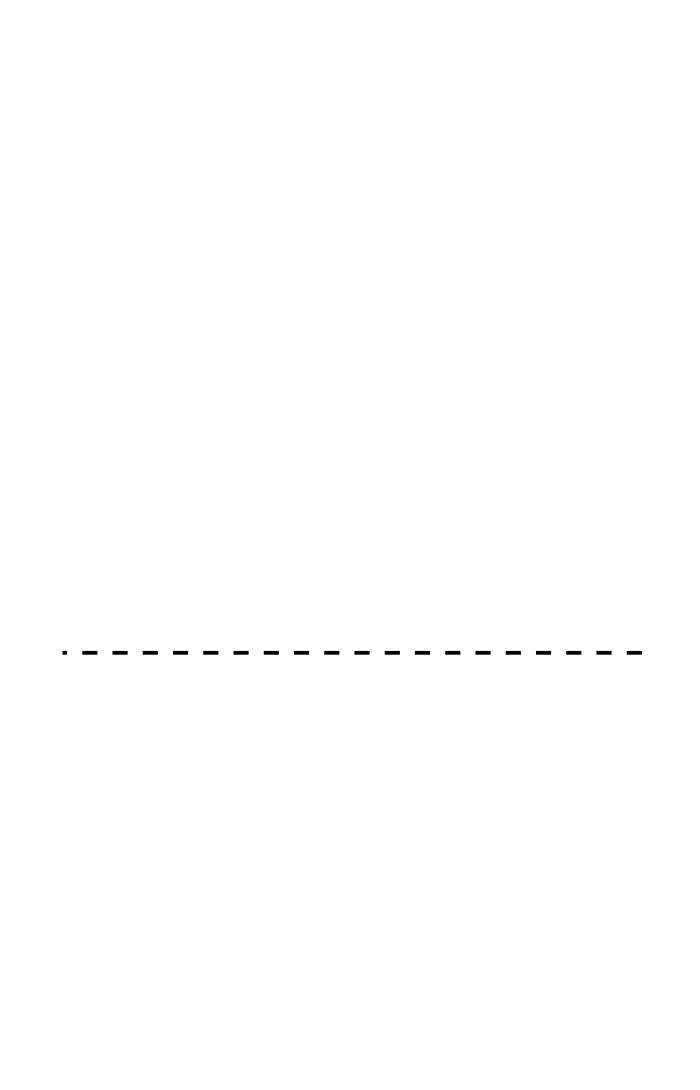
+toString(): String

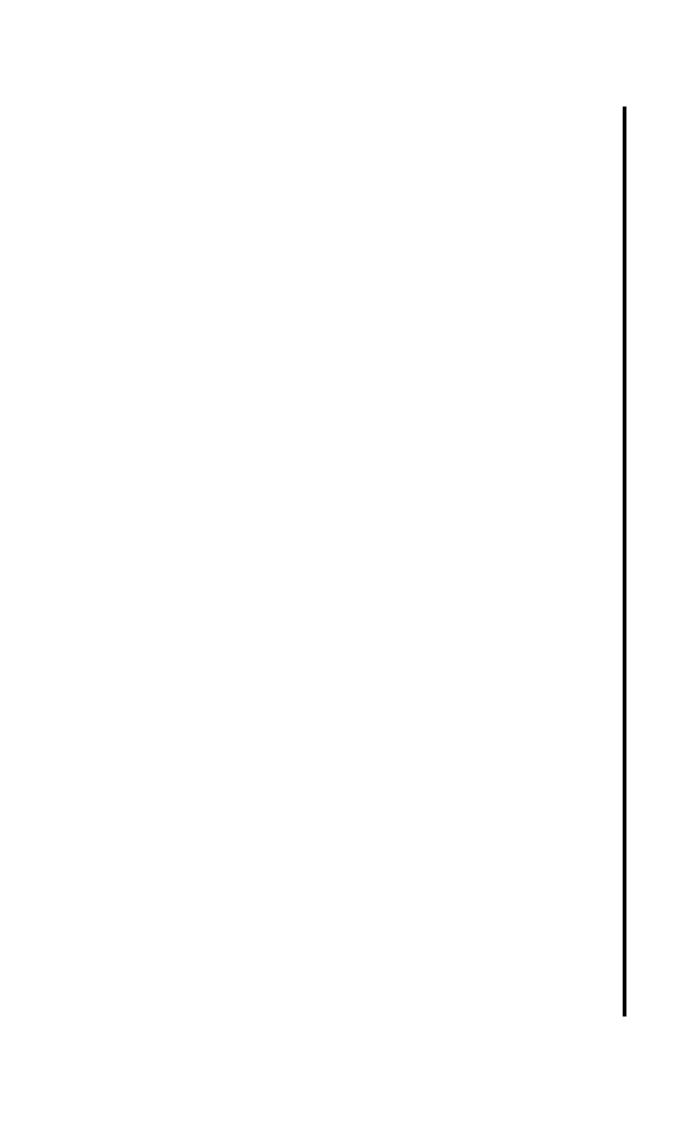


es una

es una



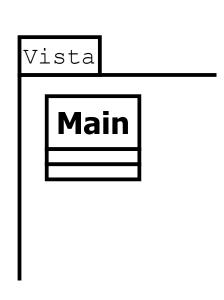


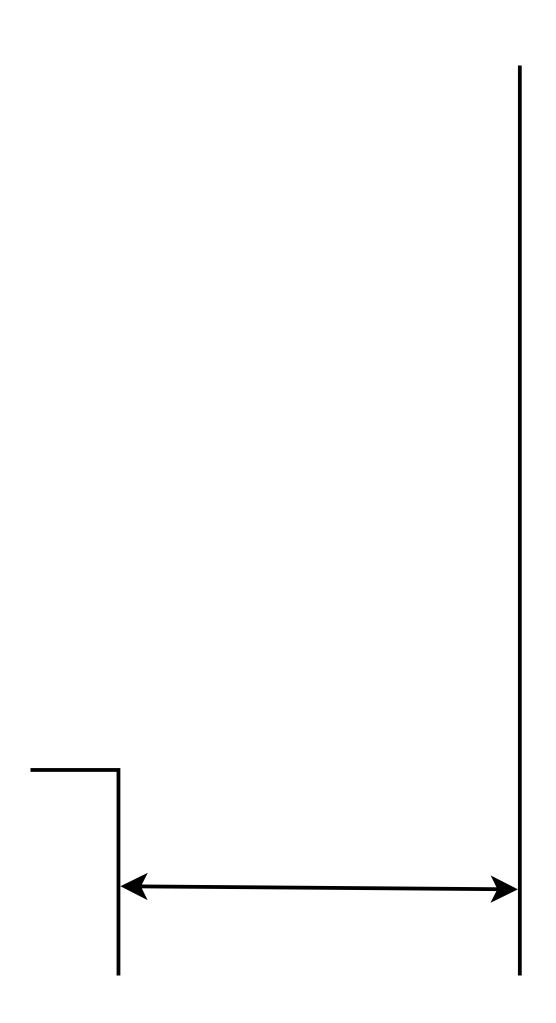


## Entrada

# de texto de los Pr

## oductos





# **EstadisticaPc**

+categoria: String

+cantidadCategoriaVend

+montoTotal: double

+EstadisticaPorCategor.

+EstadisticaPorCategor

+calcularEstadistica()

# **orCategoria**

ida: int

ia()

ia(categoria:String)

: void

#### E

+idProducto: {
+nombreProduct
+cantidadProduct
+montoTotal: (

+EstadisticaP(
+EstadisticaP(
+calcularEsta(

+idProducto:
+nombre: Stri

## **stadisticaPorProducto**

String

to: String

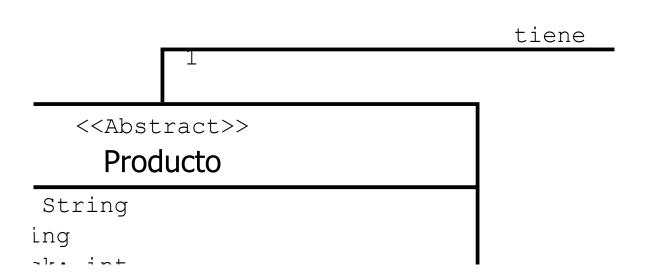
actoVendida: int

double

>rProducto()

orProducto(id:String, nombre:String)

distica(): void



### **Inventario**

```
+productos: ArrayList<Producto>
```

+Inventario()

+Inventario(productos:ArrayList<Product

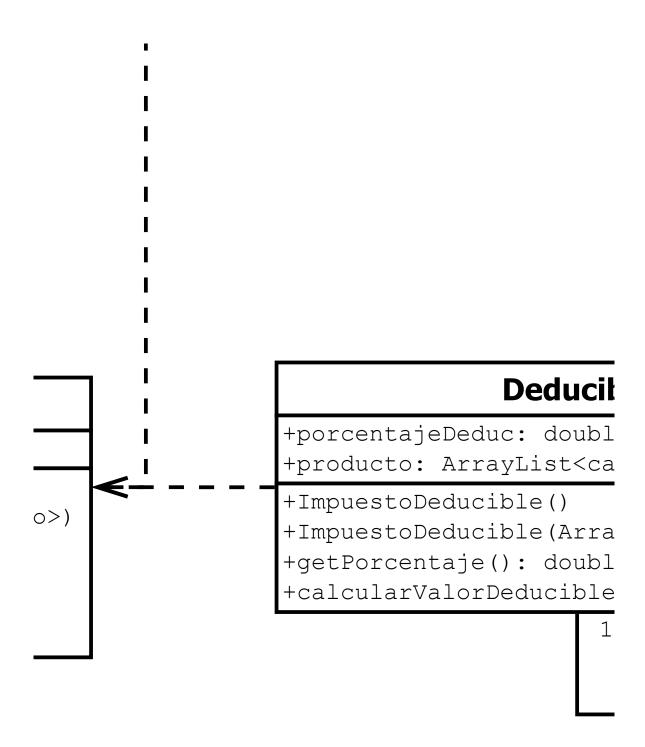
+agregarProducto(): void

+buscarProducto(): Producto

+ListarInventario(): void

+toString(): String





# ole

е

tegoria>

yList<categoria>)

е

(): double

\*

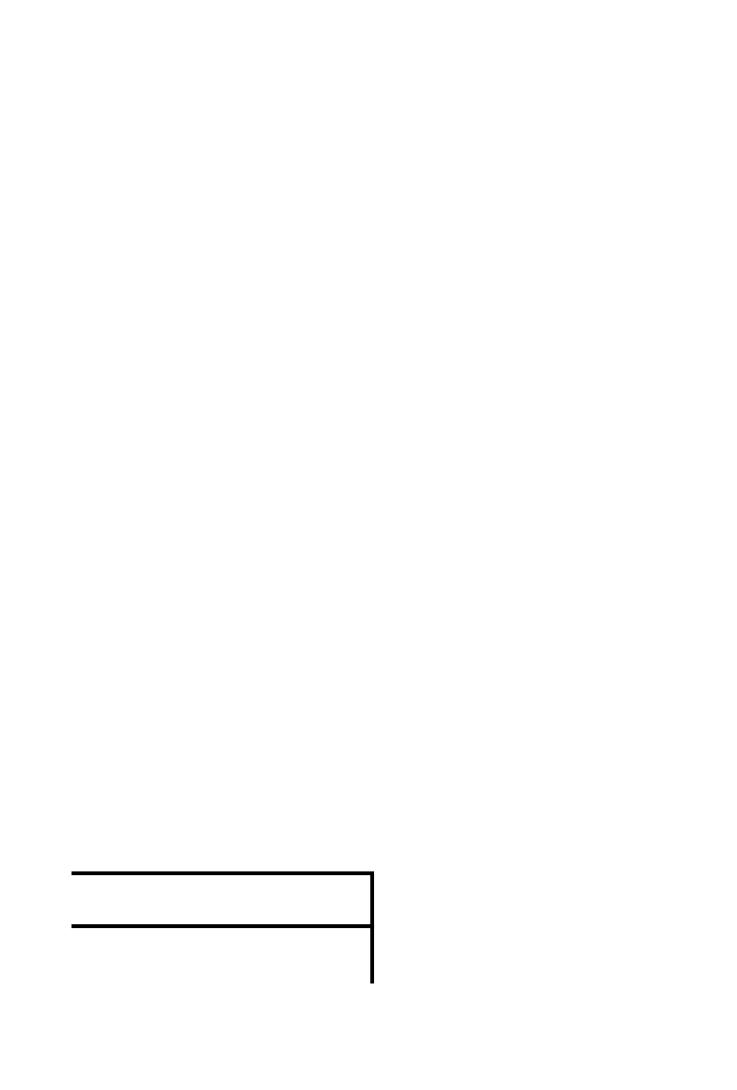
#### tiene

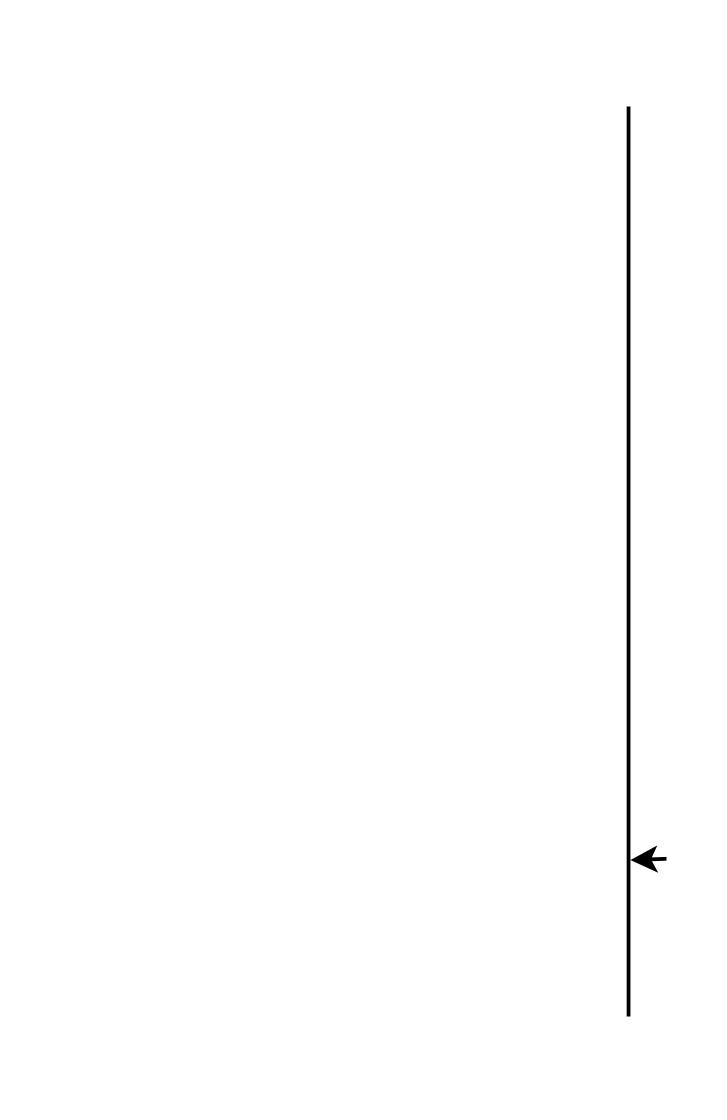
1..\*

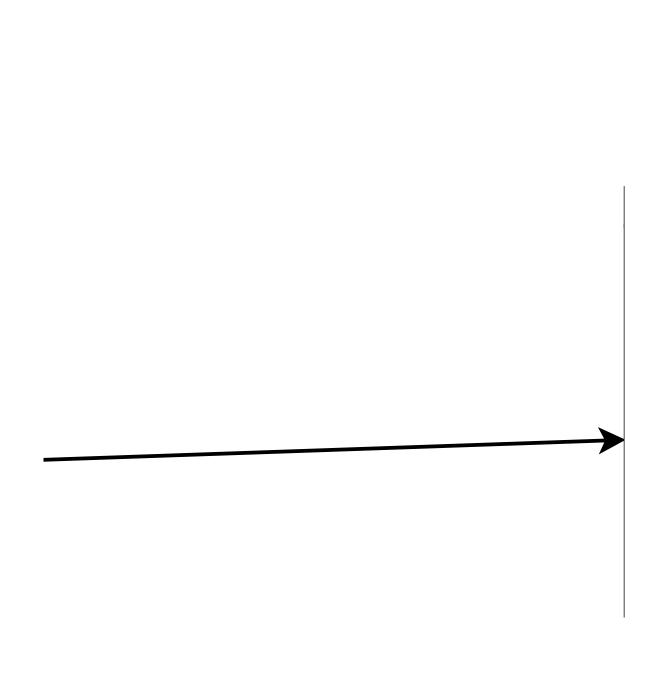
# **Factura**

+idFactura: String

+fechaEmision: date





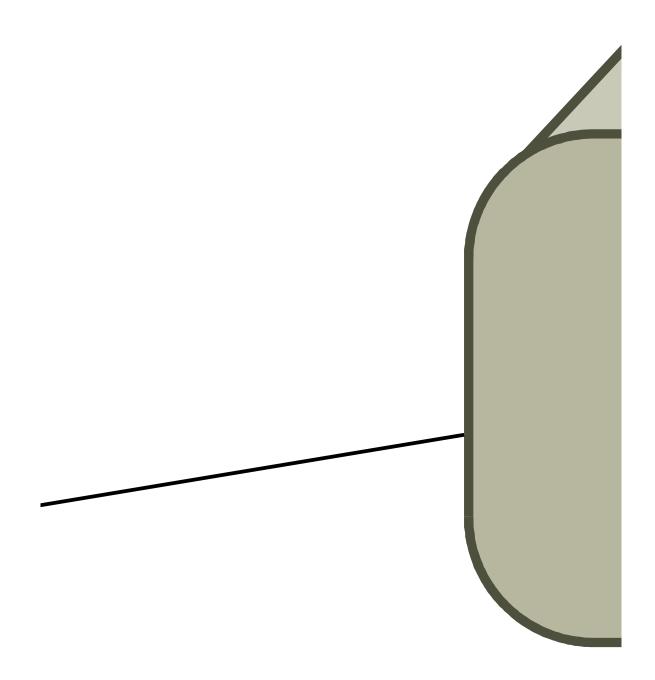


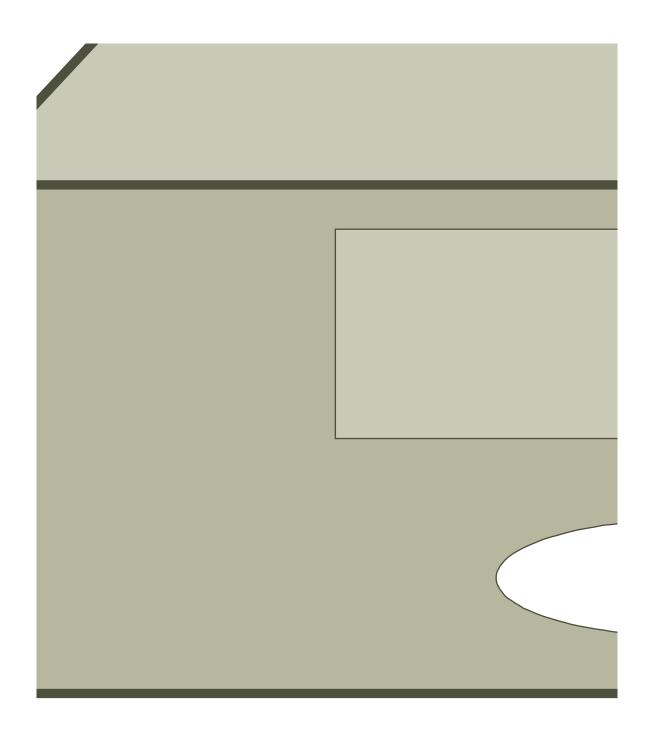
Modelo

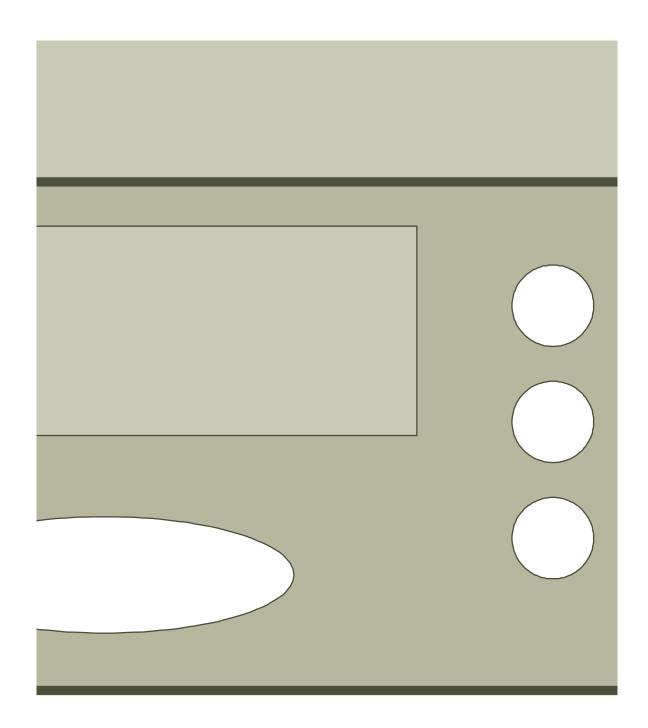
LeerDatosArcihvo

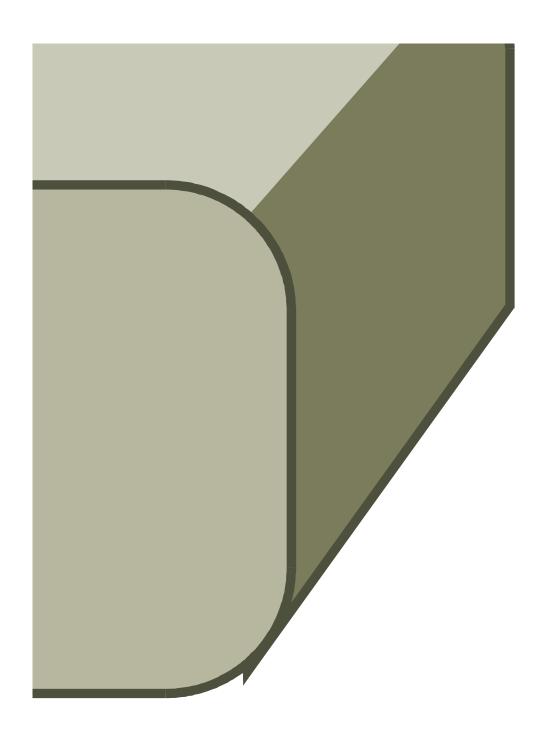
**GuardarDatosArchivo** 

**GuardarBaseDatos** 









+pre

+Pr

+de

```
+CallLluau5LO(
+categoria: S
+fechaCaducio
+producto()
+producto(non
           cat
+setIdProduct
+setCantidad
+setCategoria
+getIdProduct
+getCantidad
+getCategoria
+disminuirSto
+aumentarStoc
+<<*>> detern
+toString():
```

# ProductoNormal

ecioNormal: double

oductoNormal()

oductoNormal(id:String, nombre:String,

cantidadStock:int,categoria

fechaCaducidad:Date,precioNo

terminarPrecio(): void

```
JK: IIIL
String
dad: Date
nbre:String,cantidadStock:int,
                                  1..*
tegoria:String)
to (nomProducto:String): void
(cantidad:int): void
a(categoria:String): void
to(): string
(): int
a(): String
>ck(cantidad:int): boolean
ck(): void
ninarPrecio(): void
String
                        es un
                                 Productc
               +precioPromocional:
                                    double
               +ProductoPromocional()
               +ProductoPromocional(idPro
                                     canti
:String,
                                     fecha
ormal)
               +determinarPrecio(): void
```

+Producto: Produc

+cantidad: int

+precioUnitario:

+subtotalLinea: d

+LineaFactura()

+LineaFactura(prc

pre

+calcularSubtotal

+toString(): Stri

# **Promocional**

ducto:String, nombre:String,

dadStock:int, categoria:String,

Caducidad:Date)

```
ineaFactura

to

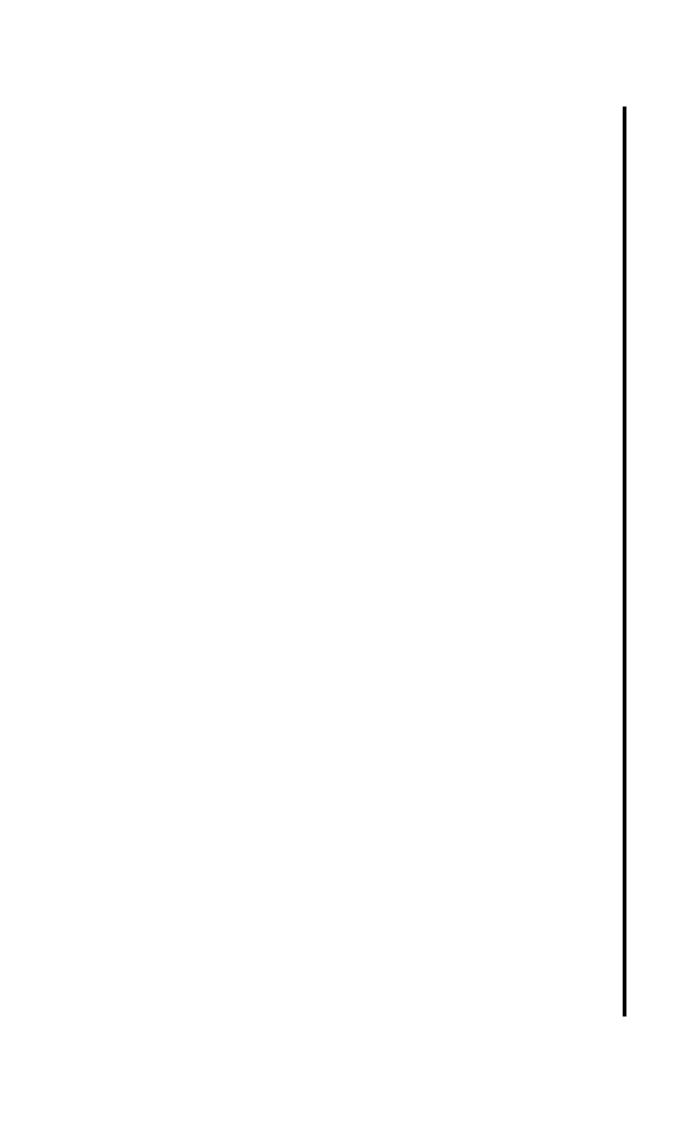
double
louble

ducto:Producto, cantidad:int,
cioUnitario:double)
.(): void
.ng
```

```
+cliente: Cliente
  +productos: ArrayList<LineaFactura>
  +subtotal: double
  +iva: double
  +total: double
  +Factura()
  +Factura (fechaEmision:date,idFactura
           linea:ArrayList<LineaFactus</pre>
           total:double, iva:double)
  +getCliente(): Cliente
  +agregarLinea(): void
  +calcularTotal(): double
  +calcularSubtotal(): double
  +calcularIva(): void
  +toString(): String
                                  1
                          tiene
                   <<Abstract>>
                     Cliente
+idCliente: String
+nombre: String
+correo: String
+telefono: int
+direccion: String
+Cliente()
```

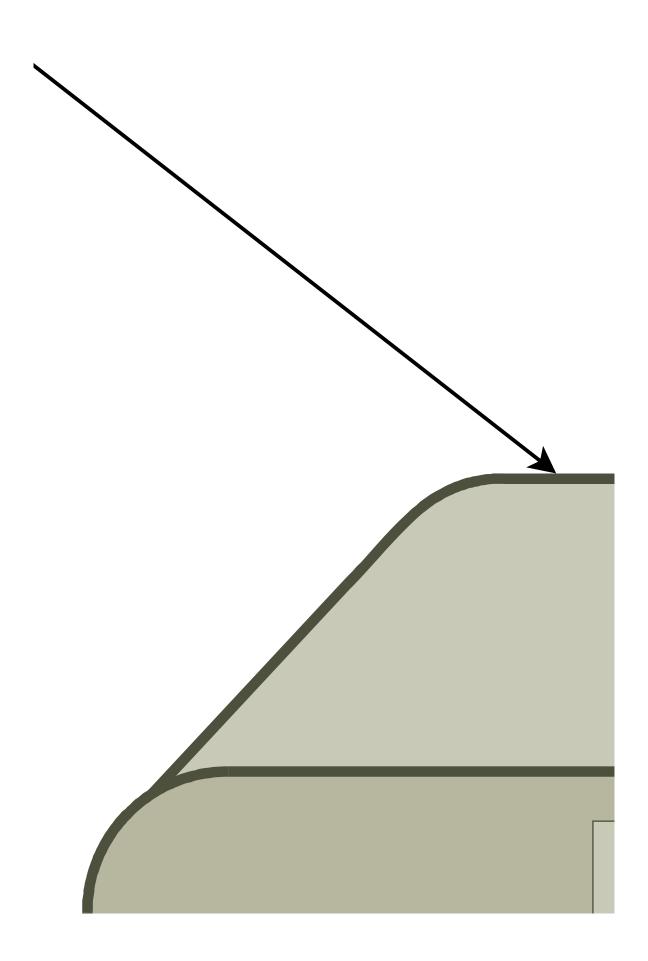
a:Random(String),
ra>, subtotal:double,

`~ · C + ~ · · ~ ~





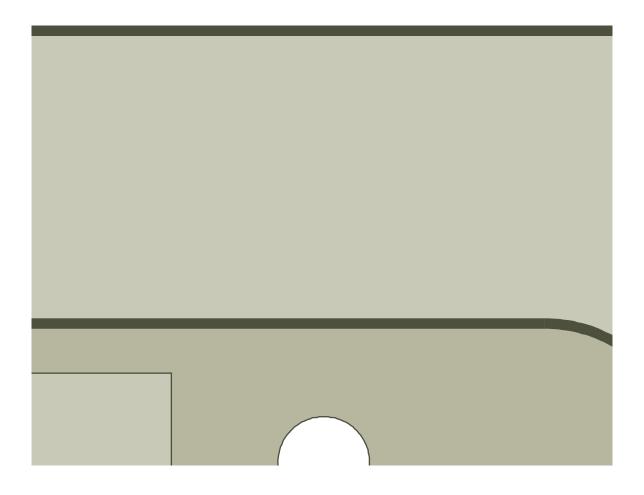


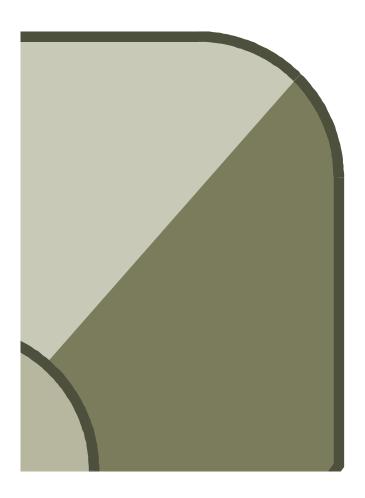


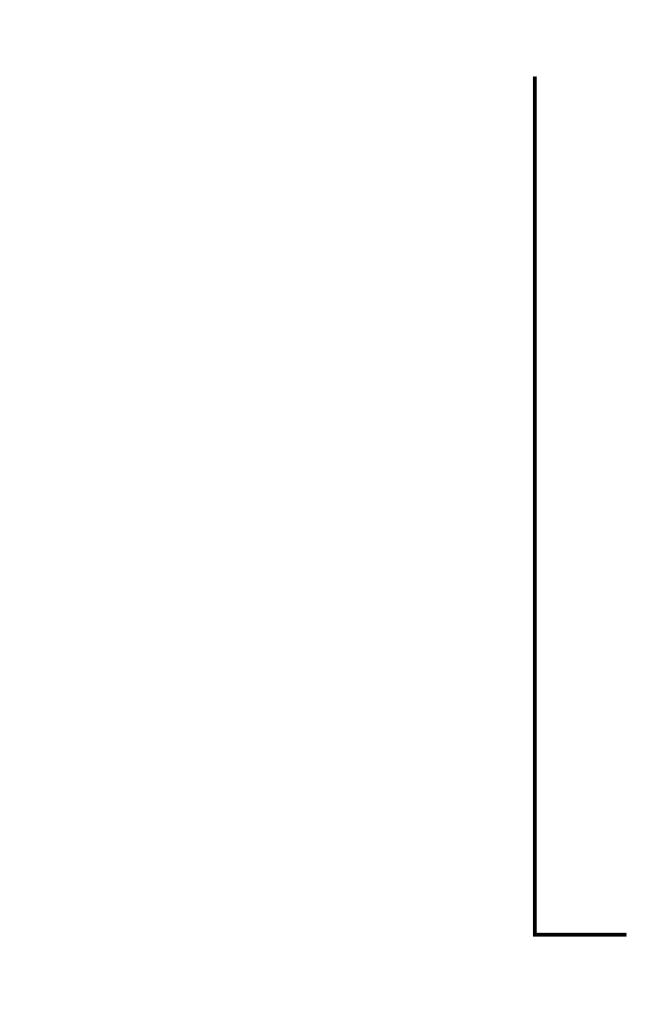
# Serealizacion de I



# as factudas











-----

Stirng(): String

		<pre>+toStirng():</pre>	String
--	--	-------------------------	--------

## ClienteNorı

+puntosCuponDescuento: int

+ClienteNormal()

+ClienteNormal(id:String, nombre

numeroTelefono:i

puntosDescuento:

+determinarDescuento(): void

numeroTelefono:int,direccion:
+<<\*>> determinarDescuento(): void
+toString(): String

es un

## mal

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

+pun

+cli

+Cli

+det

```
String)
```

es un

## **ClienteNatural**

tosCompra: int

enteNatural()

enteNatural(id:String, nombre:String,

correo:String,numeroTelefono

direccion:String)

erminarDescuento(): void

:int,			

