

T.P. N° 4
Alumno: Vazquez, Ezequiel Omar
Div.: 2A

La aplicación es una base de datos de una librería, donde las clases **Libro**, **Revista** y **Enciclopedia**, derivadas de la clase padre **Texto**, son añadidas a una clase generica **Carrito<>**, esta las almacena en una lista y las administra.

La aplicación permite realizar una venta, imprimiendo el ticket detallando la transacción, y actualizando los items salientes en la base de datos. Simultáneamente permite a la administración manipular los items en la base de datos (alta,baja y modificación).

Temas aplicados:

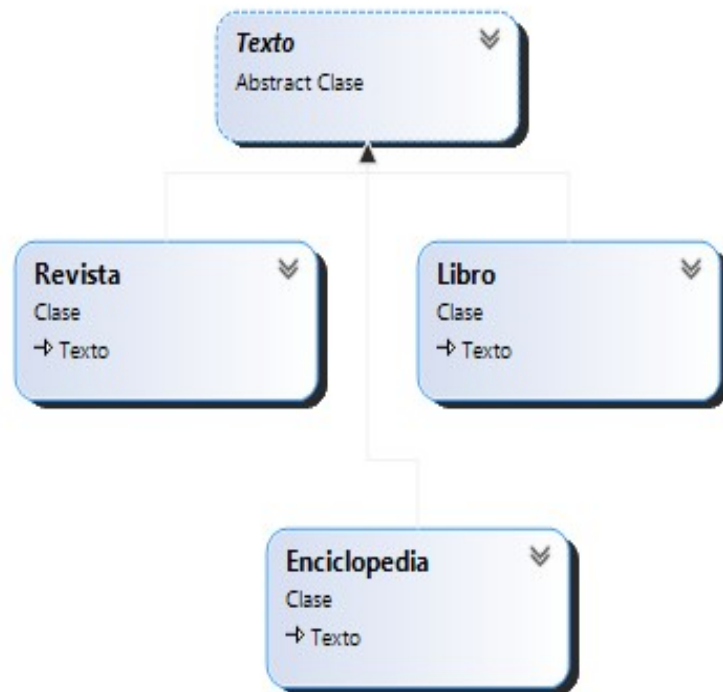
1. Excepciones -> **manejadorSQL.manejadorSql** - **Form_Libreria.FormPrincipal**
El manejo de excepciones de la app sucede durante el manejo de la base de datos, y en la interaccion con el usuario en el form principal.
2. Test unitarios -> **UnitTest.UnitTest1**
Testea las funciones Agregar/modificar/eliminar de **manejadorSQL.manejadorSql**.
3. Tipos Genericos -> **Entidades.Carrito<>**
Lista generica encargada de recibir al tipo **Texto** y administrarlo.
4. Interfaces -> **Entidades.IDescuentoMedioDePago**
Se implementa la interfaz en **Entidades.Carrito<>** con el fin de aplicarle un descuento al precio parcial, correspondiente al medio de pago.
5. Archivos y Serializacion -> **Form_Libreria.FormPrincipal**
Al presionar el boton Comprar se imprime el ticket detallando la transacción
6. Sql -> **manejadorSQL.manejadorSql**
Todas las interacciones con la base de datos, y las llamadas sql se encuentran en esta clase.
7. Hilos -> **Form_Libreria.FormPrincipal**
Al presionar el boton Solo Empleados se corre el hilo que abrirá el form ABM, a modo de poder ejecutar cambios sin interrumpir el proceso de venta.
8. Eventos -> **Form_Libreria.FormPrincipal**
Evento que trae datos de la base, y actualiza la vista de las listas. Es invocado siempre que se modifica el stock de items.

9. Metodos de Extension -> Entidades.ExtencionCarrito

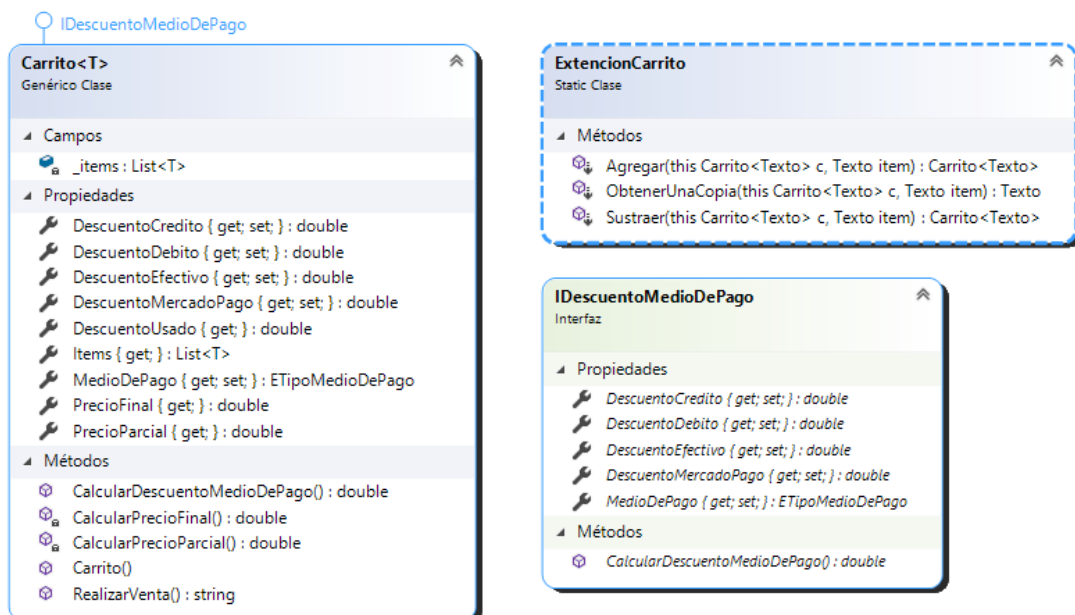
Clase de extension con metodos Agregar(),Sustraer(),ObtenerUnaCopia(). Extiende la clase Carrito<> para ejecutar funciones especificas de la clase Texto.

Namespace Entidades

Gerarquía de clases:



ListaGenerica Carrito, Extensión e Interfaz



Formulario FormPrincipal:

FormPrincipal
Clase
→ Form

Campos

- buttonComprar : Button
- buttonLimpiar : Button
- buttonMas : Button
- buttonMenos : Button
- buttonSoloEmpleados : Button
- carrito : Carrito<Texto>
- comboBox1 : ComboBox
- components : IContainer
- dataGridViewCarrito : DataGridView
- dataGridViewLibreria : DataGridView
- hilo : Thread
- labelDtoMedioPago : Label
- labelParcial : Label
- labelParcialDtoValor : Label
- labelParcialValor : Label
- labelPtjeDes : Label
- labelTotal : Label
- labelTotalValor : Label
- libreria : Carrito<Texto>
- sql : manejadorSql

Propiedades

- SeleccionActualLibreria { get; } : Texto

Métodos

- ABMForm() : void
- ActualizarLibreria() : void
- buttonComprar_Click(object sender, EventArgs e) : void
- buttonLimpiar_Click(object sender, EventArgs e) : void
- buttonMas_Click(object sender, EventArgs e) : void
- buttonMenos_Click(object sender, EventArgs e) : void
- buttonSoloEmpleados_Click(object sender, EventArgs e) : void
- comboBox1_SelectedIndexChanged(object sender, EventArgs e) : void
- ConfigurarDataGridViewCarrito() : void
- ConfigurarDataGridViewLibreria() : void
- dataGridViewCarrito_CellContentClick(object sender, DataGridViewCellEventArgs e) : void
- dataGridViewLibreria_CellClick(object sender, DataGridViewCellEventArgs e) : void
- Dispose(bool disposing) : void
- FormPrincipal()
- ImprimirTicket() : void
- InitializeComponent() : void

Eventos

- ActualizarLista : DelegadoActualizar

Formularios FormABM, FormABMValues y DelegadoActualizar:

FormABM
Clase
→ Form

Campos

- buttonAgregar : Button
- buttonEliminar : Button
- buttonModificar : Button
- components : IContainer
- frmPrinc : FormPrincipal
- sql : manejadorSql

Métodos

- buttonAgregar_Click(object sender, EventArgs e) : void
- buttonEliminar_Click(object sender, EventArgs e) : void
- buttonModificar_Click(object sender, EventArgs e) : void
- Dispose(bool disposing) : void
- FormABM(FormPrincipal frmPrinc)
- InitializeComponent() : void

DelegadoActualizar
Delegado

FormABMValues
Clase
→ Form

Campos

- buttonAceptar : Button
- buttonCancelar : Button
- comboBoxGenero : ComboBox
- comboBoxTipo : ComboBox
- components : IContainer
- labelCodigo : Label
- labelCodigoValor : Label
- labelGenero : Label
- labelPrecio : Label
- labelStock : Label
- labelTipo : Label
- labelTitulo : Label
- textBoxPrecio : TextBox
- textBoxStock : TextBox
- textBoxTitulo : TextBox
- tx : Texto

Propiedades

- Texto { get; } : Texto

Métodos

- buttonAceptar_Click(object sender, EventArgs e) : void
- buttonCancelar_Click(object sender, EventArgs e) : void
- comboBoxTipo_SelectedIndexChanged(object sender, EventArgs e) : void
- Dispose(bool disposing) : void
- FormABMValues()
- FormABMValues(Texto t)
- InitializeComponent() : void

manejadorSql:

manejadorSql
Clase

▲ Campos

comando : SqlCommand

conexion : SqlConnection

lector : SqlDataReader

libreria : Carrito<Texto>

▲ Propiedades

Libreria { get; } : Carrito<Texto>

▲ Métodos

AgregarBD(Texto tx) : bool

AgregarTexto(Texto t) : bool

EliminarTexto(Texto t) : bool

manejadorSql()

ModificarTexto(Texto t) : bool

ObtenerDatos() : int

SustraerBD(Texto tx) : bool

UnitTest:

UnitTest1
Clase

▲ Métodos

Test_AgregarTexto() : void

Test_EliminarTexto() : void

Test_ModificarTexto() : void