



#### → Formulario:

1. Al hacer clic en lista de producto se agrega en la lista de venta.
2. En caso de falla de conexión con la BD existe un respaldo local en formato XML que permite continuar con las operaciones de venta.
3. Si desea simular el uso del sistema , esta opción permite ejecutar una simulación de un cajero ateniendo un cliente.
4. Esta opción Completa la operación de venta. Genera un respaldo de la venta en la BD e Imprime un recibo que se guarda en formato Txt en el directorio del proyecto (~\Impresora\).

#### → TestConsole:

El código en el program es el siguiente.

```
static void Main(string[] args)
{
    Deposito deposito = new Deposito();
    Dbo<Venta>.GetMenu();
    Console.WriteLine("Primer Cliente entra en caja ");
    Cliente c1 = new Cliente();
    if (c1.viewCarrito().Count == 0)
    {
        Random random = new Random();
```

```

        for (int i = 0; i < 5; i++)
        {
            foreach (Producto item in Deposito.viewProductos())
            {
                if (item.Codigo == random.Next(0, 5))
                {
                    c1 += item;
                }
            }
        }
    }

    Cliente axuCliente = new Cliente();

    foreach (Producto item in c1.viewCarrito())
    {
        Deposito.AgregarVenta(item);
    }
    Console.WriteLine("Articulos de esta Venta");
    Console.WriteLine(Deposito.Articulos());
    Console.WriteLine("Monto Total a Pagar :"+Deposito.CalculoImporte().ToString());
    Deposito.GenerarVenta(new Venta(Deposito.Articulos(), Deposito.CalculoImporte()));
    Console.WriteLine("recibo impreso");
    Console.WriteLine("<----->");
    Console.ReadKey();
    try
    {
        Console.WriteLine("segundo Cliente entra en caja ");
        Cliente c2 = new Cliente();
        Cliente.CarritoVacio(c2.viewCarrito());
    }
    catch (CarritoVacioException e)
    {
        Console.WriteLine(e.Message);
    }
    Console.ReadKey();
}

```

Salida por consola:

- Simula atención a dos cliente. El primero con una lista de compra generada aleatoriamente.
- El segundo con un carrito vacío. Este lanza la exception CarritoVacioException de manera que no se guarde una compra vacía en la Bd.

```

Primer Cliente entra en caja
Articulos de esta Venta
2  producto2
3  producto3
4  producto4
3  producto3

Monto Total a Pagar :47.96
recibo impreso
<----->
segundo Cliente entra en caja
Carrito Vacio , Vuelva pronto

```

- **Clase 15 – Excepciones:**

Se programaron 2 excepciones particulares, también se hace manejo de excepciones en varias partes del código (FrmPrincipal-DbO-TestConsola)

- **ArchivoException:** Excepción de tipo archivo, muestra un mensaje de error y el origen.
- **CarritoVacioException:** Excepción de tipo Lista de compra vacía, muestra un mensaje 'Carrito Vacio , Vuelva pronto'.

- **Clase 16 - Test Unitarios:**

Se generaron 4 test unitarios:

- **TestValorNumerico:** Testea que el tipo de valor sea una instancia de tipo float en el objeto producto.
- **TestCarrito:** Testea que la lista de productos en el objeto cliente no sea null.
- **TestProductos:** Testea que la lista de productos en la clase Deposito no sea null.

- **Clase 17 - Tipos Genéricos:**

Se generó una clase Interfaz IArchivo<T> la cual implementa tipo genérico

- **void Guardar( T datos):** Método de interfaz recibe Tipo Genérico en su firma.

- **Clase 18 – Interfaces:**

Se generó una clase Interfaz IArchivo<T> la cual implementa tipo genérico y se implementó en las clases (Texto, Xml y DbO)

- **void Guardar( T datos):** Método de interfaz recibe Tipo Genérico en su firma.
- **public void Guardar( string datos):** Implementación de la Interfaz IArchivo, método Guardar. Guarda en un archivo tipo Txt
- 

- **Clase 19 - Archivos y serialización**

En la clase Texto y Xml

- `public void Guardar( string datos):`Implementacion de la Ingerfas IArchivo, metodo Guardar. Guarda en un archivo tipo Txt
- `public T Leer(string archivo, T datos):` En caso de una falla de conexion con la BD. lee en un archivo tipo Xml con la lista de productos.

- **Clase 21 – SQL**

Se utilizan Select y Insert en la clase Dbo

- `public void Guardar( T datos):`Guarda los datos utilizando comando SQL Insert into Ventas(Articulos,ImporteTotal) values(@descripcion,@preciopublic static void GetMenu() Metodo el cual conecta con la base de datos ejecuta el SELECT \* from Productos

- **Clase 22 - Bases de datos**

**Clase Dbo contiene un constructor `static Dbo()` el cual inicializa el objeto tipo SqlConnection**

- `public void Guardar( T datos):`Guarda los datos utilizando comando SQL Insert into Ventas(Articulos,ImporteTotal) values(@descripcion,@precio
- `public static void GetMenu()` Metodo el cual conecta con la base de datos ejecuta el SELECT \* from Productos

- **Clase 23 – Hilos**

**Se utiliza un hilo para ejecutar la simulación de cliente**

`private void ComienzaHiloSim():` comienza un hilo de proceso aparte del el formulario.

- **Clase 24 – Eventos**

**Se utiliza un evento para concretar la simulación de caso uso.**

`private void Simulacion():`Utiliza el evento simCobrar +=Cobrar; para ejectuar el metodo Cobrar y completar la transaccion simulada.

- **Clase 25 - Métodos de extensión**

En clase ExtensionDeposito

- `public static float ExtensionCalculoImporte(this Deposito deposito):`extiende la funcionalidad de la clase Deposito, Redondea a solo dos numeros el punto floante. (los ultimos dos decimales)

→ Diagrama de Base de datos

Ventas	
IdVenta	
Articulos	
ImporteTotal	

Productos	
ID	
NombreProducto	
Precio	

→ Script de BD presente en la carpeta base de proyecto.

(Nombre de archivo:scriptMLtp4.sql)

```
USE [master]
GO
/***** Object: Database [Tp4MiguelLandaeta2d]  Script Date: 11/29/2020 4:20:38 AM *****/
CREATE DATABASE [Tp4MiguelLandaeta2d]
    CONTAINMENT = NONE
    ON PRIMARY
    ( NAME = N'Tp4MiguelLandaeta2d', FILENAME = N'D:\SQL_SERVER\
MSSQL15.MSSQLSERVER01\MSSQL\DATA\Tp4MiguelLandaeta2d.mdf' , SIZE = 8192KB ,
    MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )
    LOG ON
    ( NAME = N'Tp4MiguelLandaeta2d_log', FILENAME = N'D:\SQL_SERVER\
MSSQL15.MSSQLSERVER01\MSSQL\DATA\Tp4MiguelLandaeta2d_log.ldf' , SIZE = 8192KB ,
    MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
    WITH CATALOG_COLLATION = DATABASE_DEFAULT
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET COMPATIBILITY_LEVEL = 150
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [Tp4MiguelLandaeta2d].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET ANSI_NULLS OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET ANSI_PADDING OFF
GO
```

```
ALTER DATABASE [Tp4MiguelLandaeta2d] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET ARITHABORT OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET DISABLE_BROKER
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET RECOVERY FULL
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET MULTI_USER
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET DB_CHAINING OFF
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET FILESTREAM( NON_TRANSACTED_ACCESS
= OFF )
GO
```

```
ALTER DATABASE [Tp4MiguelLandaeta2d] SET TARGET_RECOVERY_TIME = 60 SECONDS
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET DELAYED_DURABILITY = DISABLED
GO
EXEC sys.sp_db_vardecimal_storage_format N'Tp4MiguelLandaeta2d', N'ON'
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET QUERY_STORE = OFF
GO
USE [Tp4MiguelLandaeta2d]
GO
/***** Object: Table [dbo].[Productos]  Script Date: 11/29/2020 4:20:38 AM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Productos](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [NombreProducto] [nvarchar](50) NOT NULL,
    [Precio] [float] NOT NULL
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Ventas]  Script Date: 11/29/2020 4:20:38 AM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Ventas](
    [IdVenta] [int] IDENTITY(1,1) NOT NULL,
    [Articulos] [nvarchar](max) NOT NULL,
    [ImporteTotal] [float] NOT NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
USE [master]
GO
ALTER DATABASE [Tp4MiguelLandaeta2d] SET READ_WRITE
GO
```