

## Trabajo practica N°4

Desarrollo: Sistema de ventas.

Alumno: Wilson Huallpa.

Curso: 2D

El programa realiza carga, elimina y modificación de un producto de la base de datos mediante el formulario productos. Por otra parte también cuenta con un proyecto consola que realiza varias funcionalidades de ventas y agrega productos.

### Excepciones.

El proyecto Manejo de excepciones.

- `ArchivoException`: se lanza en caso de error en la lectura/escriptura de archivos.
- `BaseDeDatosException`: se lanza en caso de error en las operaciones de con la base de datos en la clase `BaseDeDatos`.
- `StringException`: se lanza en caso de que el string es null / menor ala cantidad de 2 letras.

### Test unitarios.

El proyecto `TestUnitario` verifica si se imprime correctamente el ticke de una venta. Y también verifica si no es null la creación un carrito de compra ramdon.

### Tipos Genericos

Se utiliza en la interfaz `IArchivos`. Esta de determinado que las clases que apliquen, creen métodos genéricos de lectura y escritura de archivos.

### Interfaces

Se creo la interfaz `IArchivos` en el proyecto `archivos`. Es implementada por las clases `Texto` y `Xml` perteneciente al mismo proyecto.

### Archivo y Serialización

Dentro del proyecto entidades, en la clase `venta` se encuentran 2 metodos.

- `public static bool PrintTicket(Venta venta)`  
Genera una diseño de ticket y lo guarda un archivo de texto
- `public static string Leer(string path)`  
lee el ticket guardado en formato texto.

En la clase `inventario`, se encuentra 2 metodos.

- `public static bool GuardarVentas(List<Venta> ventas)`

Guarda el listado de ventas en un archivo xml.

- `public static void LeerVentas()`  
lee el listado de ventas guardado en un archivo xml

## Base de datos

En el proyecto entidades en la clase `BaseDeDatos` se encarga de administrar la base de datos. Cada vez que se agrega / elimina / modifica un producto es guardado automáticamente en la base de datos.

```
USE [master]
GO
/***** Object: Database [Inventario]    Script Date: 30/11/2020 6:53:02 *****/
CREATE DATABASE [Inventario]
    CONTAINMENT = NONE
    ON PRIMARY
    ( NAME = N'Productos', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL15.SQLEXPRESS01\MSSQL\DATA\Productos.mdf' , SIZE = 8192KB , MAXSIZE =
UNLIMITED, FILEGROWTH = 65536KB )
    LOG ON
    ( NAME = N'Productos_log', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL15.SQLEXPRESS01\MSSQL\DATA\Productos_log.ldf' , SIZE = 8192KB ,
MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
    WITH CATALOG_COLLATION = DATABASE_DEFAULT
GO
ALTER DATABASE [Inventario] SET COMPATIBILITY_LEVEL = 150
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [Inventario].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [Inventario] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [Inventario] SET ANSI_NULLS OFF
GO
ALTER DATABASE [Inventario] SET ANSI_PADDING OFF
GO
ALTER DATABASE [Inventario] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [Inventario] SET ARITHABORT OFF
GO
ALTER DATABASE [Inventario] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [Inventario] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [Inventario] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [Inventario] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [Inventario] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [Inventario] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [Inventario] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [Inventario] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [Inventario] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [Inventario] SET DISABLE_BROKER
```

```

GO
ALTER DATABASE [Inventario] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [Inventario] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [Inventario] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [Inventario] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [Inventario] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [Inventario] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [Inventario] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [Inventario] SET RECOVERY SIMPLE
GO
ALTER DATABASE [Inventario] SET MULTI_USER
GO
ALTER DATABASE [Inventario] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [Inventario] SET DB_CHAINING OFF
GO
ALTER DATABASE [Inventario] SET FILESTREAM( NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [Inventario] SET TARGET_RECOVERY_TIME = 60 SECONDS
GO
ALTER DATABASE [Inventario] SET DELAYED_DURABILITY = DISABLED
GO
ALTER DATABASE [Inventario] SET ACCELERATED_DATABASE_RECOVERY = OFF
GO
ALTER DATABASE [Inventario] SET QUERY_STORE = OFF
GO
USE [Inventario]
GO
/***** Object: Table [dbo].[Productos]    Script Date: 30/11/2020 6:53:03
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Productos](
    [Codigo] [numeric](18, 0) NOT NULL,
    [Descripcion] [nvarchar](50) NOT NULL,
    [Tipo] [nvarchar](50) NOT NULL,
    [Stock] [numeric](18, 0) NOT NULL,
    [Precio] [float] NOT NULL
) ON [PRIMARY]
GO
USE [master]
GO
ALTER DATABASE [Inventario] SET READ_WRITE
GO

```

## Hilos

Los métodos `cajero_01` y `cajero_02` de la clase `inventario`. Genera un hardcodeo de ventas realizada a través de hilos diferente. Simulan ventas realizadas trabajando de forma simultane.

## Eventos

Se declara un delegado `public delegate void MiInventario()`. Se instancia un evento estatico de dicho tipo de delegado en la clase `inventario`. Esta misma se dispara al comienzo de programa cargando todos los datos de productos desde una tabla de mi base de datos. También leerá ventas realizada anteriormente desde documento `ventas.xml`.

## Métodos de extensión

En el proyecto `Entidades` se creó la clase `MetodoExtension` se lo cuales contienen:

- `public static string MostrarListaProducto(this List<Producto> productos){}`

Recorre una lista de producto y muestra todo sus elemento con su respectivo datos en diseño de tabla. Devuelve un string con todos los datos cargados en el `stringbuilder`.

- `public static string DiseñoTicke(this Venta v){}`  
crea un diseño de ticke con todos los datos de una venta realizada de la cual incluye nombre del comercio/ hora y fecha,.