Script:

Crear base de datos [TP4-Informe] y ejecutar:

USE [TP4-Informe]

GO

/\*\*\*\*\*\* Object: Table [dbo].[INFORME] Script Date: 6/12/2020 18:54:52 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[INFORME](

[nombre] [varchar](50) NOT NULL,

[apellido] [varchar](50) NOT NULL,

[articulo] [varchar](50) NOT NULL,

[formaDePago] [varchar](50) NOT NULL,

[precioAPagar] [float] NOT NULL,

[fecha] [date] NOT NULL

) ON [PRIMARY]

GO

Guia TP°4 Meli

La idea fue crear un software de venta de una farmacia el cual almacena en un objeto de tipo Farmacia un farmaceutico y una lista de clientes. Estos clientes estaban pensados para poder realizar una sola compra y con productos limitados, sean estos analgésico, crema y antibiotico. La clase cliente almacena el tipo de producto que cada uno compra y también su precio final a pagar ya que según la forma de pago se realiza un tipo de descuento. Asi en la base de datos se guardan los clientes con producto, nombre, apellido, forma de pago, producto, y precio final a pagar.

Guia de temas a exponer en el tp:

1. Excepciones: Desarrolle un proyecto de excepciones en el cual cree tres clases que heredaban de exception. La primera es DatoInvalidoException que es lanzada cuando al crear un cliente uno de sus datos no es valido, la segunda es la FormVentaException que es lanzada cuando uno de los campos de tipo de producto o de forma de pago del formulario de ventas esta mal completado y la tercera es TicketException que es lanzada cuando no puede crearse un ticket.
2. Test Unitarios: Fueron realizados dos test unitarios, uno para comprobar el correcto uso de la excepción DatoInvalidoException y otro para probar el método de extensión.
3. Tipos Genericos: Fue utilizado en una interfaz de tipo genérica string que obligaba a usar el método Escribirlo (similar a ToString) que retornaba un string, fue utilizado en la clase cliente.
4. Interfaces: Fue creada la interfaz genérica IToString y aplicada en la clase cliente para escribir cada cliente.
5. Archivos y Serializacion: En el formulario de venta luego de efectuar la primera venta y que sea comprobada como correcta se habilita el botón para generar un ticket que genera un archivo txt con el apellido del cliente en el escritorio con todos los datos de la venta.
6. Sql y Bases de Datos: Se utilizo para persisitir los datos de los clientes que realizaron las compras con los botones de leer y escribir en el InformeForm que muestra todas las ventas realizadas hasta el momento.
7. Hilos: Fueron utilizados en el VentaForm cada vez que tocamos el botón para confirmar la venta para en segundo plano cargar ese cliente en la base de datos
8. Eventos: Fue usado para agregar en la lista de clientes que contiene InformeForm al cliente agregado cada vez que se presiona el botón de confirmar la venta mediante una suscripción dentro de VentaForm.
9. Metodos de Extension: Se creo un método de extensión a modo demostrativo de la clase cliente en la clase Extensión del proyecto Entidades, el cual devuelve la ultima letra del nombre del cliente.