

Universidad Tecnológica Nacional Facultad Regional Avellaneda			
Carrera:	Técnico Superior en Programación		
Materia:	LABORATORIO DE PROGRAMACIÓN II		
Código:		Fecha:	Noviembre 2020
División:	2°C	Docentes:	Davila, Federico Cerizza, Mauricio
Cuatrimestre:	2°	Modalidad:	VIRTUAL
Trabajo Práctico N°4		Alumno:	LEMOS, Lautaro L.
		Legajo:	110054

Funcionalidad

El programa funciona como control de producción, stock y ventas. Tiene tres sectores delimitados:

En un principio a la derecha hay una tabla de datos, con la cantidad de productos en producción y el stock de los mismos, está se actualiza una vez por segundo con la información que tiene la base de datos. También se simula en segundo plano la producción de la fábrica y se incrementa de forma aleatoria el stock de los productos.

En el primer recuadro "FABRICA", hay un par de campos editables para ingresar una descripción de un producto y el precio del mismo; con el botón "Agregar a Producción" se ingresa el producto y comienza la producción del mismo. Seleccionando un producto en la tabla y presionando "Quitar de Producción" se elimina el mismo de la base de datos y por lo tanto de la producción.

En el segundo recuadro "LOCAL", se selecciona un cliente, la cantidad de productos vendidos y seleccionando el producto en la tabla de la derecha, se habilita el botón "Ingresar Venta" (si hay stock suficiente), al presionar el botón, se muestra un mensaje con la información de la venta y queda registrada la misma en un archivo de log de ventas. Presionando "Ver Log de Ventas" se accede a un formulario que muestra todas las ventas que fueron registradas y la ganancia total y parcial de cada jornada.

Excepciones

ArchivosException

Excepción que indica en su mensaje: "Error en manejo de archivos" y transmite la `innerException`.

CuitInvalidoException

Excepción que indica en su mensaje: "La nacionalidad no se condice con el número de DNI".

Test Unitarios

TestCuitInvalidoException

Verifica si al ingresar un cuit inválido se lanza la excepción `CuitInvalidoException`.

TestFormatearTexto

Verifica si el método de extensión `FormatearTexto` funciona correctamente.

Tipos Genéricos

```
IArchivo<T>
```

La interface `IArchivo` utiliza el tipo genérico `T` para el tipo de dato a Leer y Guardar.

Interfaces

```
interface IArchivo<T>
```

`IArchivo` establece las firmas de los métodos para guardar y leer tanto un archivo de texto como una serialización `Xml`.

Archivos y serialización

```
public class Texto : IArchivo<string>
```

Establece los métodos para Guardar y Leer un archivo de texto.

```
public class Xml<V> : IArchivo<V>
```

Establece los métodos para Guardar y Leer un archivo `XML` vía serialización.

SQL y Bases de datos

```
public class ProductoDAO
```

Establece los métodos para acceder a la base de datos 'Productos' y guardar, leer, modificar y borrar registros.

Hilos

```
public class Fabrica  
private Thread hiloFabrica;
```

Simula la producción de la fábrica mediante un hilo y un tiempo random de velocidad de producción.

```
public partial class FormPrincipal  
private Thread hiloListado;
```

Lanza 'actualizarInfo' que:

Define un tiempo de demora entre actualizaciones de la lista y actualiza la misma.

Eventos

```
public class Fabrica  
public event encargadoStock EventoStock;
```

Cuando se lanza el evento `EventoStock` se debe agregar un producto al stock.

```
public partial class FormPrincipal  
fabrica.EventoStock += Fabrica_EventoStock;
```

Evento de fábrica, selecciona un producto random de la lista y le incrementa el stock en uno.

Métodos de extensión

`FormatearTexto`

Quita los espacios antes y después del texto, y deja sólo el primer carácter en mayúscula.