

TP4. Proyecto de Heladeria

Alumno: Francisco Allende

División: 2A

Docente: Maximiliano Neiner

Mi TP3 tuvo correcciones que fueron aplicadas y estan expresadas en su respectivo PDF, en la carpeta "Recuperatorio_TP3" de mi repositorio en github.

El proyecto propone ser un desktop app que permita el manejo del stock de la heladería Froddo y también sus ventas. Esta versión final de mi proyecto de heladería aplica todos los conceptos vistos durante la totalidad de la cursada.

- Tanto las tablas de Stock como las de TomarPedido poseen un CRUD completo que permite agregar nuevos productos - en la tabla de stock - o pedidos - en la tabla de pedidos -, modificarlos, eliminarlos y hacer un update de la tabla.
- La query del SQL esta en un archivo.txt en este mismo repo, en la misma carpeta donde esta este PDF y la solución del proyecto
- El datagrid es manipulable con la flechita en su lateral derecho que viene por defecto, utilizada en modificar y remover, facilitando su uso ya que no se pide más el id de cada producto
- En ambas tablas no hay limite alguno de productos o pedidos nuevos
- El CRUD es a prueba de inyecciones SQL ya que usa valores parametrizados
- La BBDD de SQL es la fuente de datos principal del programa, la serialización solo se utiliza para exportar
- Si bien ya estaban presentes en el recuperatorio del TP3, recuerdo que facilitando el proceso de compra de en tomar pedido (y de añadir y modificar también en stock) se creo un form auxiliar que captura los datos en textboxes, sin botones por cada paso, haciendo mucho más fácil y agil este proceso
- Se utilizan hilos, async await y tasks en el método "OrdenarYCargar" tanto del formulario de Stock como del de Tomar Pedido, simulando el tiempo de búsqueda de los datos con Thread.Sleep.

- En este mismo método, se ordenan las listas que serán exportadas en formato json con un Sort de tipo lambda, el cual varía el criterio según si exporta la lista de stock o de ventas
 - Se aplican eventos propios con su debida notificacion en los formularios AuxiliarFormPedido y el auxiliar de stock
 - Se utilizan delegados en stock y tomar pedido
 - Se creo una clase extendida de tipo string que extiende sus posibilidades, agilizando validaciones
 - Se añadieron nuevos tests, los cuales todos pasan correctamente.
 - Se mejoró el diseño del formulario
-