

Funcionalidad TP4:

Para realizar el TP4 modifique la parte de consumiciones de mi aplicación, guardando las compras realizadas en una base de datos (generada por el script que adjunte, con algunas compras hardcodeadas) y realizando un CRUD para el manejo de estas compras.

Temas aplicados:

Todos los temas del tp3 siguen estando aplicados

Base de datos: Cree una base de datos llamada TP4_DB que contiene una tabla llamada Tabla_Compras, en la cual maneje todas las compras de consumiciones realizadas en mi aplicación. La conexión con la base de datos la realice mediante una clase llamada ConexionBD, en donde agregue métodos para recuperar la lista de compras, agregar una compra a la tabla, modificar y borrar compras existentes.

Delegados y expresiones lambda: Este tema lo utilicé para permitir que el usuario pueda ordenar la lista de clientes (que se abre cuando se selecciona a un cliente para reservar una butaca o para asignarle una compra) por criterio de Id de mayor a menor o de Nombre de la A a la Z. Esto se encuentra en el form TablaClientes.

Programación multihilo: Esto se ve aplicado en la función de ActualizarDatos del Form ListaCompras, la cual llama al método ObtenerListaCompras de la clase ConexionBD para cargar el listbox. Realice esta función asíncrona y utilice un task en ObtenerListaCompras para que el form no se pare mientras se cargan los datos, sino que sea utilizable y que el proceso de cargar los datos sea asíncrono. (Comente un Thread.Sleep en ObtenerListaCompras para validar que esto funcionaba y simular una gran cantidad de datos cargados). Mientras la lista no sea recuperada, el listbox mostrara un "Cargando..."

Eventos: Cree dos eventos propios en las clases ListaClientes y ListaCompras que cambian el tema del form y lo ponen en "modo nocturno" con colores mas apagados. Para llamar a este evento, lo sume al evento onclick del botón btn_ModoNocturno mediante el delegado EventHandler. (`this.btn_ModoNocturno.Click += new EventHandler(CambioTema)`)

Metodos de extensión:

NombreEsValido: Extension de la clase string que retorna true si el string es de menos de 30 caracteres y no esta vacio o null, utilizado para validar que se ingrese un nombre que pueda ser ingresado en la base de datos.

CompararNumeros: Extension de la clase double, compara el numero invocador con el numero pasado por parámetro para poder ordenarlos. Es utilizado en el ordenamiento de los clientes por Id.