```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;

namespace knapsackplanner
{
    public partial class Form1 : Form
    {

        private BindingList<Item> _items;
        private BindingList<Container> _containers;
        private enum ItemSelected { Item, Container, ContainedItem};
        private ItemSelected _currentSelection;
        public Form1()
        {
            InitializeComponent();
            _items = new BindingList<Item>();
            _containers = new BindingList<Container>();
            _dataGridViewItems.DataSource = _items;
            _dataGridViewContainers.DataSource = _containers;

            StreamReader reader;
            string[] lineRead = new string[5];


            try
            {
                //open file
                reader = File.OpenText("knapsack.csv");



                //run through all lines
                while (!reader.EndOfStream)
                {
                    //read next line
                    lineRead = reader.ReadLine().Split(',');
                    if (lineRead[0] == "ITEM")
                    {
                        _items.Add(new Item(lineRead[1], double.Parse(lineRead[2]), double.Parse(lineRead[3]), decimal.Parse(lineRead[4])));
                    }
                    else if(lineRead[0] == "CONTAINER")
                    {
                        _containers.Add(new Container(lineRead[1], double.Parse(lineRead[2]), double.Parse(lineRead[3])));
                    }
                }
                reader.Close();
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            _dataGridViewStoredItems.DataSource = _containers[0];
        }

        internal Container Container
        {
            get => default;
            set
            {
            }
        }

        private void _buttonMoveTo_Click(object sender, EventArgs e)
        {
            DataGridViewCell containerCell =
              _dataGridViewContainers.CurrentCell;
            if (containerCell == null)
            {
                MessageBox.Show("No Container available!");
                return;
            }
            DataGridViewCell itemCell = _dataGridViewItems.CurrentCell;
            if (itemCell == null)
            {
                MessageBox.Show("No Item available!");
                return;
            }
            int containerIndex = containerCell.RowIndex;
            int itemIndex = itemCell.RowIndex;
            Item item = _items[itemIndex];
            Container container = _containers[containerIndex];
            double totalWeight = 0;
            double totalVolume = 0;
            foreach(Item item1 in container.ItemsContained)
            {
                totalWeight += item1.Weight;
                totalVolume += item1.Volume;
            }
            if ((totalWeight+item.Weight)>container.MaxWeight || (totalVolume
              + item.Volume) > container.MaxVolume)
            {
                MessageBox.Show("Exceeds max weight/volume!");
                return;
            }

            container.ItemsContained.Add(item);
            _items.Remove(item);
            container.Update();
            _dataGridViewContainers.DataSource = null;
            _dataGridViewContainers.DataSource = _containers;
        }

        private void _buttonMoveFrom_Click(object sender, EventArgs e)
        {
            DataGridViewCell containerCell = _dataGridViewContainers.CurrentCell;
            if (containerCell == null)
            {
                MessageBox.Show("No Container available!");
                return;
            }
            DataGridViewCell itemCell = _dataGridViewStoredItems.CurrentCell;
            if (itemCell == null)
            {
                MessageBox.Show("No Item available!");
                return;
            }
            int containerIndex = containerCell.RowIndex;
            int itemIndex = itemCell.RowIndex;
            Container container = _containers[containerIndex];
            Item item = container.ItemsContained[itemIndex];
            container.ItemsContained.Remove(item);
            _items.Add(item);
            container.Update();
            _dataGridViewContainers.DataSource = null;
            _dataGridViewContainers.DataSource = _containers;
        }



        private void _dataGridViewContainers_SelectionChanged(object sender, EventArgs e)
        {
            DataGridViewCell cell = _dataGridViewContainers.CurrentCell;
            if (cell != null)
            {
                _currentSelection = ItemSelected.Container;
                int containerIndex = cell.RowIndex;
                Container container = _containers[containerIndex];
                container.Bind(_dataGridViewStoredItems);

                _textBoxName.Text = container.Name;
                _numericUpDownWeight.Value = (decimal)container.MaxWeight;
                _numericUpDownVolume.Value = (decimal)container.MaxVolume;
                _numericUpDownValue.Value = 0;

            }
        }

        private void _dataGridViewItems_SelectionChanged(object sender, EventArgs e)
        {
            DataGridViewCell cell = _dataGridViewItems.CurrentCell;
            if (cell != null)
            {
                _currentSelection = ItemSelected.Item;
                int itemIndex = cell.RowIndex;
                Item item = _items[itemIndex];

                _textBoxName.Text = item.Name;
                _numericUpDownWeight.Value = (decimal)item.Weight;
                _numericUpDownVolume.Value = (decimal)item.Volume;
                _numericUpDownValue.Value = item.Value;

            }
        }

        private void _dataGridViewStoredItems_SelectionChanged(object sender, EventArgs e)
        {
            DataGridViewCell cell = _dataGridViewStoredItems.CurrentCell;
            if (cell != null)
            {
                _currentSelection = ItemSelected.ContainedItem;
                DataGridViewCell containerCell = _dataGridViewContainers.CurrentCell;
                DataGridViewCell itemCell = _dataGridViewStoredItems.CurrentCell;
                int containerIndex = containerCell.RowIndex;
                Container container = _containers[containerIndex];
                int itemIndex = itemCell.RowIndex;
                if (container.ItemsContained.Count > 0)
                {
                Item item = container.ItemsContained[itemIndex];
                    _textBoxName.Text = item.Name;
                    _numericUpDownWeight.Value = (decimal)item.Weight;
                    _numericUpDownVolume.Value = (decimal)item.Volume;
                    _numericUpDownValue.Value = item.Value;
                }



            }
        }

        private void buttonEmpty_Click(object sender, EventArgs e)
        {
            DataGridViewCell containerCell = _dataGridViewContainers.CurrentCell;
            if (containerCell == null)
            {
                MessageBox.Show("No Container available!");
                return;
            }
            int containerIndex = containerCell.RowIndex;
            Container container = _containers[containerIndex];
            for(int i = container.ItemsContained.Count - 1; i>=0; i--)
            {

                _items.Add(container.ItemsContained[i]);
                container.ItemsContained.Remove(container.ItemsContained[i]);


            }
            container.Update();
            _dataGridViewContainers.DataSource = null;
            _dataGridViewContainers.DataSource = _containers;
        }

        private void _buttonAddItem_Click(object sender, EventArgs e)
        {
            _items.Add(new Item(_textBoxName.Text, (double)_numericUpDownWeight.Value, (double)_numericUpDownVolume.Value, _numericUpDownValue.Value));
        }

        private void _buttonDelete_Click(object sender, EventArgs e)
        {
            DataGridViewCell itemCell = _dataGridViewItems.CurrentCell;
            DataGridViewCell storedItemCell = _dataGridViewStoredItems.CurrentCell;
            DataGridViewCell containerCell = _dataGridViewContainers.CurrentCell;

            switch (_currentSelection)
            {

                case ItemSelected.Item:
                    if (itemCell != null)
                    {

                        int itemIndex = itemCell.RowIndex;
                        Item item = _items[itemIndex];
                        _items.Remove(item);
                    }
                    break;

                case ItemSelected.ContainedItem:
                    if (storedItemCell != null)
                    {
                        int containerIndex = containerCell.RowIndex;
                        Container container = _containers[containerIndex];
                        int itemIndex = storedItemCell.RowIndex;
                        if (container.ItemsContained.Count > 0)
                        {
                            Item item = container.ItemsContained[itemIndex];

                            container.ItemsContained.Remove(item);
                            container.Update();
                            _dataGridViewContainers.DataSource = null;
                            _dataGridViewContainers.DataSource = _containers;
                        }

                    }
                    break;

                case ItemSelected.Container:
                    if (containerCell != null)
                    {
                        int containerIndex = containerCell.RowIndex;
                        Container container = _containers[containerIndex];
                        _containers.Remove(container);
                    }
                    break;

                default:
                    break;
            }
        }

        private void _buttonAddContainer_Click(object sender, EventArgs e)
        {
            _containers.Add(new Container(_textBoxName.Text, (double)_numericUpDownWeight.Value, (double)_numericUpDownVolume.Value));
        }

        private void _buttonEdit_Click(object sender, EventArgs e)
        {
            DataGridViewCell itemCell = _dataGridViewItems.CurrentCell;
            DataGridViewCell storedItemCell = _dataGridViewStoredItems.CurrentCell;
            DataGridViewCell containerCell = _dataGridViewContainers.CurrentCell;

            switch (_currentSelection)
            {

                case ItemSelected.Item:
                    if (itemCell != null)
                    {

                        int itemIndex = itemCell.RowIndex;
                        Item item = new Item(_textBoxName.Text, (double)_numericUpDownWeight.Value, (double)_numericUpDownVolume.Value, _numericUpDownValue.Value);
                        _items[itemIndex] = item;
                    }
                    break;

                case ItemSelected.ContainedItem:
                    MessageBox.Show("Items cannot be edited while in a container");
                    break;

                case ItemSelected.Container:
                    if (containerCell != null)
                    {
                        int containerIndex = containerCell.RowIndex;
                        Container container = new Container(_textBoxName.Text, (double)_numericUpDownWeight.Value, (double)_numericUpDownVolume.Value);
                        _containers[containerIndex] = container;
                    }
                    break;

                default:
                    break;
            }
        }
    }
}

```