

CSIS-1175 – Winter 2020 - Assignment# 2

Due: Sun, Apr 5, 11:59 PM - Total Points: 50

NOTE: Assignment 2 is worth 12.5% of the final course grade

Submission guidelines: Please perform the following tasks. **Add comments** to your code explaining the variables and logic used. Not adding comments to each program and file you are submitting explaining the logic will result in loss of points for each question.

Create the C# project with your FirstInitialLastInitial_Assign2 - **There should be NO SPACES in your PROJECT NAME.** You must use upper and lower cases for the project name. (e.g., If your name is **John Smith**, then your project must be called **JS_Assign2**).

At the beginning of your source code, you must again enter your section #, your name and student id as a comment.

Zip up your entire program from the top folder and submit it via Blackboard before the due date. **Do not rename the folder. You will be responsible for accurately compressing the entire project folder (along with solution files) and submitting it on blackboard.** Late submission will not be accepted. The name of your zip file must be the same as your project name. If you do not follow the above instructions, including the file name and subject line, you will get marks deducted.

Note: Sharing your code, and looking at another student's code will constitute to cheating. In case of cheating both the student that copies and the student that shared or allowed the copying would get Zero and will reported to administration.

Task: In this assignment, you are required to implement the local grocery application in C#. Grocery Data (see given csv file, localgrocery.csv) contains ItemName, ItemCode, UnitPrice, StartingQty (starting quantity count for each item), QtyMinRestck (minimum quantity needed for restocking the item), QtySold, QtyRStcked.

Class Definition: You must use a Grocery class to maintain a grocery item with Properties: ItemName, ItemCode, StartingQty, QtyMinForRestock, QtySold, QtyRestocked. You will use two additional computed read-only properties: i. QuantityOnHand, which is computed as the Starting Quantity – Quantity Sold + Restocked Quantity, and ii. Total Sales for each item based on QtySold multiplied by UnitPrice. These two additional properties of TotalSales and QtyHand will be read-only properties, and cannot be directly set at any point. They will be updated to reflect the correct amount whenever you try to get these properties.

You will use a list of Grocery Items class objects to maintain all grocery items in the list (input file). Feel free to create additional fields or class methods as needed.

Form Design: Design the App with the following design features (see figure for sample). Your form should contain:

1. A grocery listbox that will display ItemCode, ItemName, StartingQty, QtyMinRestck, QtySold, QtyRStcked, UnitPrice, QtyHand and Sales. Set the listbox font to Consolas (or any other monospaced font).
2. Below the list box, you will have a set of controls in three group boxes for loading data, updating data, saving data. On the left, you have the Load Grocery Button in the Load Data GroupBox. The Update Data group box contains one set (of label, TextBox and Button) for incrementing Qty Sold, and another set (of label, TextBox and Button) for incrementing Qty Restocked. Finally, there is a Delete Selected Item Button. The Save Data group box contains three buttons one each to Save Grocery Data, Save Sales Report and Save Restock Needs Report.
3. Below all these controls, add a status label that will display status information as various user events are handled. You can set the AutoSize property of the label to false (so that the label can be resized). You can set the BackColor property (background color) to any color at the Web tab. You can set the TextAlign Property to Center. You can also set the border style to Fixed 3D (to give it an embossed look).
4. Any other form layout or arrangement is fine as long as all this information is there in a clear format.

Event Handlers:

1. **Load Grocery Data:** When the Load Grocery button is clicked, grocery records will be read from a text file (localgrocery.csv) from the default folder, and the lines read from the file is tokenized into field values, parsed (as needed) and then loaded onto the listbox and the list of class objects. In addition to all the field values in the input file, the listbox should, also display columns for QtyHand and Sales. Do some background research to see how String.Format() or PadRight() method can be used to format the listbox contents such that all values in a column are aligned. Make sure your listbox has **monospaced font**. Finally, update the status label to indicate how many grocery items were loaded into the list.

Note that each item in listbox should always correspond to each item in the List of grocery items except for the extra header line at the top in the listbox but not in the List of Grocery Items. Every ith item in the listbox will thus correspond to (i-1)the item in the list of Grocery objects.

The screenshot shows a Windows application window titled "Form1" with the title "Local Grocery Application". The window contains a table of grocery items, three control panels, and a status bar.

ItemName	ItemId	UnitPrice	StartQty	QtyMinForRestock	QtySold	QtyRestocked	QtyHand	Sales
Shampoo	11-559-3026	\$8.83	31	13	0	0	31	\$0.00
Conditioner	36-389-7424	\$8.72	60	10	0	0	60	\$0.00
Dandruff Shampoo	64-090-4668	\$8.09	66	13	0	0	66	\$0.00
Body Wash	29-054-1549	\$9.01	34	6	0	0	34	\$0.00
Bar Soap	76-866-0130	\$8.76	78	14	0	0	78	\$0.00
Hand Soap	12-575-1269	\$6.43	61	12	0	0	61	\$0.00
Razor Blades	51-854-4250	\$3.96	97	14	0	0	97	\$0.00
Shaving Cream	38-416-2990	\$4.04	41	10	0	0	41	\$0.00
Feminine Products	00-499-9472	\$3.64	95	5	0	0	95	\$0.00
Women's Deodorant	90-222-4482	\$3.56	19	20	0	0	19	\$0.00
Men's Deodorant	71-280-8443	\$6.32	12	14	0	0	12	\$0.00
Toothpaste	36-559-4086	\$3.04	52	7	0	0	52	\$0.00
Toothbrushes	23-363-4831	\$2.37	85	6	0	0	85	\$0.00
Floss	14-459-7070	\$3.67	50	11	0	0	50	\$0.00
Lotion	88-225-9449	\$2.64	85	19	0	0	85	\$0.00

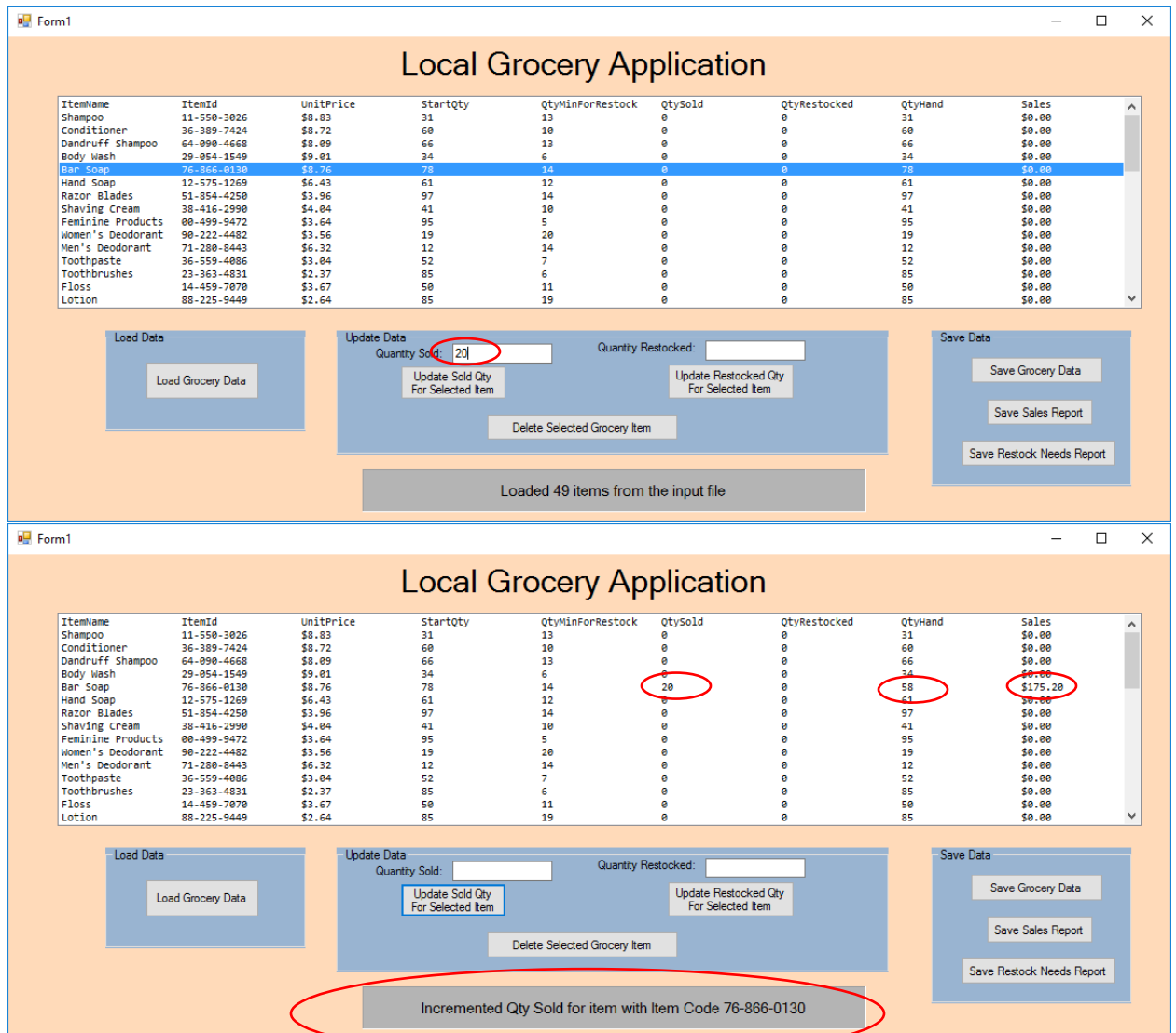
The application interface includes three main control panels:

- Load Data:** A button labeled "Load Grocery Data".
- Update Data:** Contains two text boxes for "Quantity Sold:" and "Quantity Restocked:". Below them are buttons for "Update Sold Qty For Selected Item", "Update Restocked Qty For Selected Item", and "Delete Selected Grocery Item".
- Save Data:** Contains buttons for "Save Grocery Data", "Save Sales Report", and "Save Restock Needs Report".

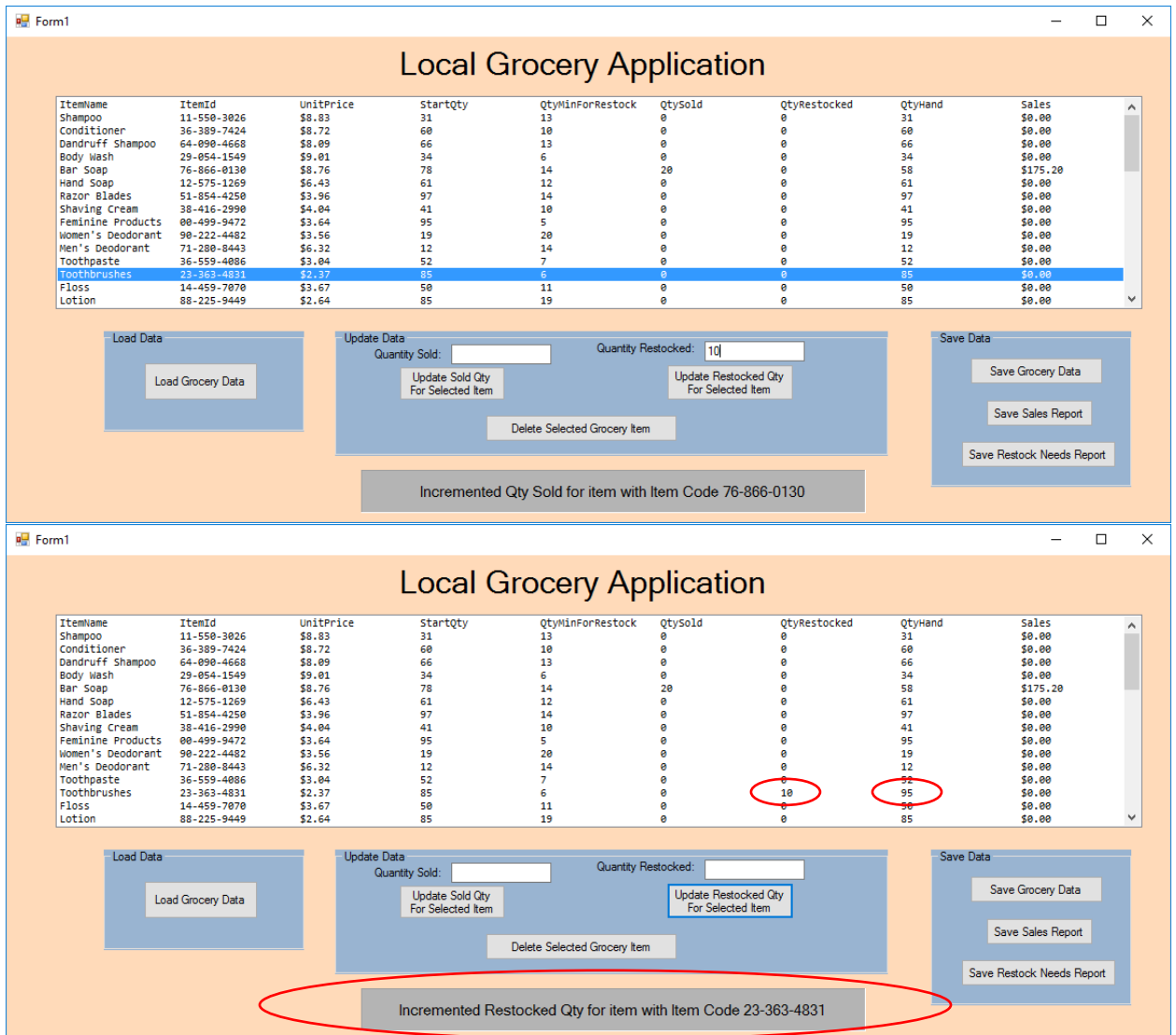
A status bar at the bottom of the window displays the message: "Loaded 49 items from the input file".

Feel free to use methods as you see fit.

- Update Sold Qty For Selected Item:** When this button is clicked, and if no item or if the headerline is selected from the listbox, you should display an error message in the status label indicating that *"Please selected a grocery item to increment sold qty"*. If a grocery item is selected from the listbox, then TryParse the sold Qty from the QtySold TextBox. If the TryParse fails or if the Qty Sold from TextBox input is less than or equal to 0 or if QtySold from TextBox is greater than Qty on Hand, then output relevant error messages in the status label. If the parse succeeds, and if the Qty Sold from the TextBox is greater than 0 and if Qty Sold from TextBox is lesser than Qty on Hand for that item, then increment the QtySold field by that amount given for the corresponding grocery item from the list. Qty on Hand should also reflect this change. Display a message in the status label saying *"Sold Qty has been increased for the item with item code ABC"*. See sample below.



- Update Restocked Qty For Selected Item Button:** When this button is clicked, and if no item or if the headerline is selected from the listbox, you should display an error message in the status label indicating that *“Please selected a grocery item to increment restocked qty”*. If a grocery item is selected from the listbox, then TryParse the Qty Restocked from the TextBox. If the TryParse fails or if the Qty Restocked from TextBox input is less than or equal to 0, then output relevant error messages in the status label. If the parse succeeds, and if the Qty Restocked from the TextBox is greater than 0, then increment the QtyRestocked field by that amount given for the corresponding grocery item from the list. Qty on Hand needs to reflect this change. Display a message in the status label saying *“Incremented Restocked Qty for item with item code ABC”*. See sample below.



4. **Delete Selected Item Button:** When this button is clicked, and if no item or if the headerline is selected from the listbox, you should display an error message in the status label indicating that *“Please selected a grocery item to delete”*. If a grocery item is selected from the listbox, then delete this grocery item from the listbox (and from the list of grocery item objects). Also, clear the output label if necessary. Display a message in the status label indicating *“Deleted Item with item code ABC”*. See sample with Deleted item Shampoo (1st item in the original list).

Form1

Local Grocery Application

ItemName	ItemId	UnitPrice	StartQty	QtyMinForRestock	QtySold	QtyRestocked	QtyHand	Sales
Shampoo	11-550-3026	\$8.83	31	13	0	0	31	\$0.00
Conditioner	36-389-7424	\$8.72	60	10	0	0	60	\$0.00
Dandruff Shampoo	64-090-4668	\$8.09	66	13	0	0	66	\$0.00
Body Wash	29-054-1549	\$9.01	34	6	0	0	34	\$0.00
Bar Soap	76-866-0130	\$8.76	78	14	20	0	58	\$175.20
Hand Soap	12-575-1269	\$6.43	61	12	0	0	61	\$0.00
Razor Blades	51-854-4250	\$3.96	97	14	0	0	97	\$0.00
Shaving Cream	38-416-2990	\$4.04	41	10	0	0	41	\$0.00
Feminine Products	00-499-9472	\$3.64	95	5	0	0	95	\$0.00
Women's Deodorant	90-222-4482	\$3.56	19	20	0	0	19	\$0.00
Men's Deodorant	71-280-8443	\$6.32	12	14	0	0	12	\$0.00
Toothpaste	36-559-4086	\$3.04	52	7	0	0	52	\$0.00
Toothbrushes	23-363-4831	\$2.37	85	6	0	10	95	\$0.00
Floss	14-459-7070	\$3.67	50	11	0	0	50	\$0.00
Lotion	88-225-9449	\$2.64	85	19	0	0	85	\$0.00

Load Data

Load Grocery Data

Update Data

Quantity Sold: Quantity Restocked:

Update Sold Qty For Selected Item Update Restocked Qty For Selected Item

Delete Selected Grocery Item

Save Data

Save Grocery Data

Save Sales Report

Save Restock Needs Report

Incremented Restocked Qty for item with Item Code 23-363-4831

Form1

Local Grocery Application

ItemName	ItemId	UnitPrice	StartQty	QtyMinForRestock	QtySold	QtyRestocked	QtyHand	Sales
Conditioner	36-389-7424	\$8.72	60	10	0	0	60	\$0.00
Dandruff Shampoo	64-090-4668	\$8.09	66	13	0	0	66	\$0.00
Body Wash	29-054-1549	\$9.01	34	6	0	0	34	\$0.00
Bar Soap	76-866-0130	\$8.76	78	14	20	0	58	\$175.20
Hand Soap	12-575-1269	\$6.43	61	12	0	0	61	\$0.00
Razor Blades	51-854-4250	\$3.96	97	14	0	0	97	\$0.00
Shaving Cream	38-416-2990	\$4.04	41	10	0	0	41	\$0.00
Feminine Products	00-499-9472	\$3.64	95	5	0	0	95	\$0.00
Women's Deodorant	90-222-4482	\$3.56	19	20	0	0	19	\$0.00
Men's Deodorant	71-280-8443	\$6.32	12	14	0	0	12	\$0.00
Toothpaste	36-559-4086	\$3.04	52	7	0	0	52	\$0.00
Toothbrushes	23-363-4831	\$2.37	85	6	0	10	95	\$0.00
Floss	14-459-7070	\$3.67	50	11	0	0	50	\$0.00
Lotion	88-225-9449	\$2.64	85	19	0	0	85	\$0.00
Excema Cream	24-443-3243	\$3.14	104	6	0	0	104	\$0.00

Load Data

Load Grocery Data

Update Data

Quantity Sold: Quantity Restocked:

Update Sold Qty For Selected Item Update Restocked Qty For Selected Item

Delete Selected Grocery Item

Save Data

Save Grocery Data

Save Sales Report

Save Restock Needs Report

Deleted record for item with item code 11-550-3026

5. **Save Grocery Data (updatedgrocery.csv):** When the user clicks on this button, **for those items that are still in the list box (and the list of grocery objects)**, write the ItemCode, ItemName, StartingQty, QtyMinForRestock, QtySold, QtyRestocked, UnitPrice onto a comma-separated text file in the default folder location using file name updatedgrocery.csv. The comma separated text file will have each column separated by a comma. The file should have a header row at the top that gives the name of each field. Finally, the status label should display log messages indicating how many grocery items were written to the file (see sample).

Local Grocery Application

ItemName	ItemId	UnitPrice	StartQty	QtyMinForRestock	QtySold	QtyRestocked	QtyHand	Sales
Conditioner	36-389-7424	\$8.72	60	10	0	0	60	\$0.00
Dandruff Shampoo	64-090-4668	\$8.09	66	13	0	0	66	\$0.00
Body Wash	29-054-1549	\$9.01	34	6	0	0	34	\$0.00
Bar Soap	76-866-0130	\$8.76	78	14	20	0	58	\$175.20
Hand Soap	12-575-1269	\$6.43	61	12	0	0	61	\$0.00
Razor Blades	51-854-4250	\$3.96	97	14	0	0	97	\$0.00
Shaving Cream	38-416-2990	\$4.04	41	10	0	0	41	\$0.00
Feminine Products	00-499-9472	\$3.64	95	5	0	0	95	\$0.00
Women's Deodorant	90-222-4482	\$3.56	19	20	0	0	19	\$0.00
Men's Deodorant	71-280-8443	\$6.32	12	14	0	0	12	\$0.00
Toothpaste	36-559-4086	\$3.04	52	7	0	0	52	\$0.00
Toothbrushes	23-363-4831	\$2.37	85	6	0	10	95	\$0.00
Floss	14-459-7070	\$3.67	50	11	0	0	50	\$0.00
Lotion	88-225-9449	\$2.64	85	19	0	0	85	\$0.00
Excema Cream	24-443-3243	\$3.14	104	6	0	0	104	\$0.00

Load Data

Load Grocery Data

Update Data

Quantity Sold: Quantity Restocked:

Save Data

Saved 48 records into the output inventory file

updatedgrocery - Notepad

File Edit Format View Help

```

ItemName,ItemCode,UnitPrice,StartingQty,QtyMinRestck,QtySold,QtyRStcked
Conditioner,36-389-7424,8.72,60,10,0,0
Dandruff Shampoo,64-090-4668,8.09,66,13,0,0
Body Wash,29-054-1549,9.01,34,6,0,0
Bar Soap,76-866-0130,8.76,78,14,20,0
Hand Soap,12-575-1269,6.43,61,12,0,0
Razor Blades,51-854-4250,3.96,97,14,0,0
Shaving Cream,38-416-2990,4.04,41,10,0,0
Feminine Products,00-499-9472,3.64,95,5,0,0
Women's Deodorant,90-222-4482,3.56,19,20,0,0
Men's Deodorant,71-280-8443,6.32,12,14,0,0
Toothpaste,36-559-4086,3.04,52,7,0,0
Toothbrushes,23-363-4831,2.37,85,6,0,10
Floss,14-459-7070,3.67,50,11,0,0
Lotion,88-225-9449,2.64,85,19,0,0
Excema Cream,24-443-3243,3.14,104,6,0,0
Hydrocortizone Cream,57-097-8098,7.04,54,9,0,0
Diapers,16-801-1792,4.10,112,9,0,0
Wipes,38-094-2745,5.72,17,5,0,0
Baby Wash,80-606-2379,2.11,31,6,0,0
Lightbulbs,79-593-0652,6.16,104,8,0,0
Dish Soap,10-553-0064,9.98,105,5,0,0
Laundry Detergent,57-181-7386,5.71,45,17,0,0
Dishwasher Detergent,20-698-7128,3.19,20,10,0,0
                    
```

6. **Save Sales Report (groceriesales.csv):** When the user clicks on this button, **for those items that are still in the list box (and the list of grocery objects) and have QtySold greater than 0**, write the ItemCode, ItemName QtySold, UnitPrice and Sales onto a comma-separated text file in the default folder location using the name groceriesales.csv. The comma separated text file will have each column separated by a comma. The file should have a header row at the top that gives the name of each field. Finally, the status label should display log messages indicating how many sale items were written to the sales report file (see sample).

Form1

Local Grocery Application

ItemName	ItemId	UnitPrice	StartQty	QtyMinForRestock	QtySold	QtyRestocked	QtyHand	Sales
Conditioner	36-389-7424	\$8.72	60	10	0	0	60	\$0.00
Dandruff Shampoo	64-090-4668	\$8.09	66	13	0	0	66	\$0.00
Body wash	29-054-1549	\$9.01	34	6	0	0	34	\$0.00
Bar Soap	76-866-0130	\$8.76	76	14	20	0	58	\$175.20
Hand Soap	12-575-1269	\$6.43	61	12	0	0	61	\$0.00
Razor Blades	51-854-4250	\$3.96	97	14	0	0	97	\$0.00
Shaving Cream	38-416-2990	\$4.04	41	10	0	0	41	\$0.00
Feminine Products	00-499-9472	\$3.64	95	5	0	0	95	\$0.00
Women's Deodorant	90-222-4482	\$3.56	19	20	0	0	19	\$0.00
Men's Deodorant	71-280-8443	\$6.32	12	14	0	0	12	\$0.00
Toothpaste	36-559-4086	\$3.04	52	7	0	0	52	\$0.00
Toothbrushes	23-363-4031	\$2.37	95	6	0	10	95	\$0.00
Floss	14-459-7070	\$3.67	50	11	0	0	50	\$0.00
Lotion	88-225-9449	\$2.64	85	19	0	0	85	\$0.00
Excema Cream	24-443-3243	\$3.14	104	6	0	0	104	\$0.00

Load Data

Load Grocery Data

Update Data

Quantity Sold: Quantity Restocked:

Save Data

Saved 1 records into the output sales file

grocerysales - Notepad

File Edit Format View Help

```

ItemName,ItemCode,UnitPrice,QtySold,Sales
Bar Soap,76-866-0130,$8.76,20,$175.20
  
```

7. **Save Restock Needs Report (groceryrestocks.csv):** When the user clicks on this button, for those items that are still in the list box (and the list of grocery objects) and have QtyHand lesser than QtyMinForRestock, write the ItemCode, ItemName, QtyMinForRestock, QtyHand onto a comma-separated text file in the default folder location using the file name groceryrestocks.csv. The comma separated text file will have each column separated by a comma. The file should have a header row at the top that gives the name of each field. Finally, the status label should display log messages

indicating how many restock needed items were written to the restock needs report file (see sample).

The screenshot shows two windows. The top window is titled 'Local Grocery Application' and contains a table of grocery items with columns for ItemName, ItemId, UnitPrice, StartQty, QtyMinForRestock, QtySold, QtyRestocked, QtyHand, and Sales. Below the table are three main control panels: 'Load Data' with a 'Load Grocery Data' button; 'Update Data' with input fields for 'Quantity Sold' and 'Quantity Restocked', and buttons for 'Update Sold Qty For Selected Item', 'Update Restocked Qty For Selected Item', and 'Delete Selected Grocery Item'; and 'Save Data' with buttons for 'Save Grocery Data', 'Save Sales Report', and 'Save Restock Needs Report'. A status bar at the bottom of the application window reads 'Saved 4 records into the restocks needed output file'.

The bottom window is titled 'groceryrestocks - Notepad' and displays the following text:

```

ItemName,ItemCode,QtyHand,QtyMinRestck
Women's Deodorant,90-222-4482,19,20
Men's Deodorant,71-280-8443,12,14
Chapsticks,32-998-9786,3,19
Radish,60-641-7970,11,16
    
```

Note: Use additional methods as you see fit to complete the task.