

File Edit View History Bookmarks Tools Help

CIS156/56 Course Page

www.pgrocer.net/PFGCIS156.html

are available to you under Audio and Smartboard.

From week 4 loopscomboetc.zip

Use it.  
 Now where he is  
 to do for the Thursday class when we can review it. I decided since I have  
 n to arrays and then go back to procedures and functions.

under VB.NET in the break examples.  
 ors to report number of games user won, number of games computer won

ing all my classes on line this week.

have read those chapters or if you decided against the book that you have  
 and the examples for the topic under VB.NET as well.  
 Loops and I want to look at a series of programs involving Faculty and  
loopscomboetc.zip  
 which is in a loops folder which is in VBNETCIS56. We will also  
 ograms VB.NET and in the conditions group there are if examples with

uesday and Thursday.  
 above and look for him in K101 for the 11:00 class and the 12:30 class.  
 estions from what we went over last class, please ask! Please read the

chapter as well. If you do not have the book you should find information on line on IF statements. Remember, I have some older VB books that will definitely  
 help with logic if you want to see me about using one.  
**Some help with decision problems from chapter 4**  
 I strongly recommend that you reference the Smartboard and audio from last Spring and that you look at the examples including those under VB.NET  
**Assignments:**  
**In Class IF assignments** - everyone needs to do this and in fact all assignments posted under assignments  
**IF assignment**  
 As always, check back!

Either finish chapter #2 and then read chapter #3 or find another source to read about starting Visual Basic and using variables and input. The topics covered in  
 the examples are the things you should follow through and read about. We will slow down the chapter page now.

infoCalcProj.sln 1/28/2020 1:02 PM Continue without code →  
 E:\VBExamples\infoCalcProj\infoCalcProj

cisBasicProj.sln 1/28/2020 12:52 PM  
 E:\VBExamples\cisBasicProj\cisBasicProj

Week #3  
 Week of February 3rd

Type here to search

12:33 PM  
 2/18/2020

File Edit View History Bookmarks Tools Help

CIS156/56 Course Page

www.pgrocer.net/PFGCIS156.html

are available to you under Audio and Smartboard.

Open Project/Solution

VBExamples > loopscomboetc > VBNETCIS56 > loops > muffins

Name	Date modified	Type	Size
.vs	2/17/2020 3:51 PM	File folder	
Backup	2/17/2020 3:51 PM	File folder	
bin	2/13/2005 10:18 AM	File folder	
My Project	2/17/2020 3:51 PM	File folder	
obj	2/13/2005 10:18 AM	File folder	
muffins.sln	2/17/2020 3:51 PM	Visual Studio Solu...	2 KB
muffins	2/17/2020 3:51 PM	Visual Basic Projec...	5 KB

File name:  All Project Files (\*.sln;\*.dsw;\*.vc)

Do not load projects

Open Cancel

infoCalcProj.sln 1/28/2020 1:02 PM Continue without code →  
 E:\VBExamples\infoCalcProj\infoCalcProj

cisBasicProj.sln 1/28/2020 12:52 PM  
 E:\VBExamples\cisBasicProj\cisBasicProj

**About This Course**  
 Introduction to  
 Syllabus  
**Site Resources**  
 Notes, Handouts  
**VB 2019/2017**  
 Examples 2015  
 Smartboard 2015  
 Audio VB 2015  
**VB.NET (previous)**  
 Programs VB.NET  
 Presentations VB  
 Smartboard VB  
 Audio/lectures  
**VB.6 (previous)**  
 Programs written  
 Programs written  
 Presentations VB  
 Smartboard for  
 Video conferencing available  
**Weekly Schedule**  
 Weekly schedule  
**Links:**  
 Links & Tutorials

The information at this page is the property of PGrocer. The information is provided for your personal use only. No permission. ©Copyright

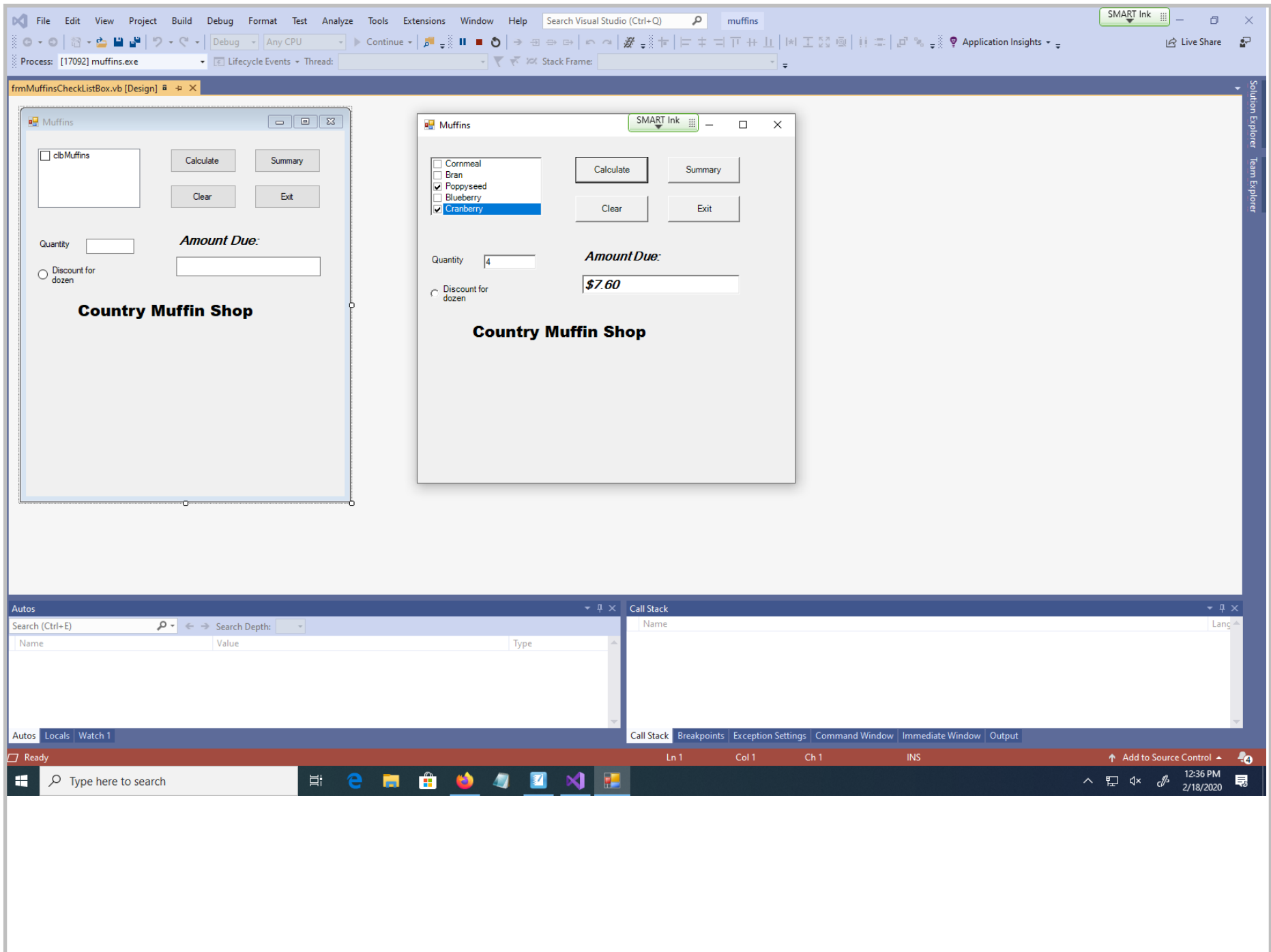
use it.  
 how where he is  
 to do for the Thursday class when we can review it. I decided since I have  
 n to arrays and then go back to procedures and functions.  
 under VB.NET in the break examples.  
 ors to report number of games user won, number of games computer won  
 ing all my classes on line this week.  
 have read those chapters or if you decided against the book that you have  
 nd the examples for the topic under VB.NET as well.  
 Loops and I want to look at a series of programs involving Faculty and  
**mboetc.zip**  
 muffins which is in a loops folder which is in VBNETCIS56. We will also  
 ograms VB.NET and in the conditions group there are if examples with  
 uesday and Thursday.  
 above and look for him in K101 for the 11:00 class and the 12:30 class.  
 estions from what we went over last class, please ask! Please read the

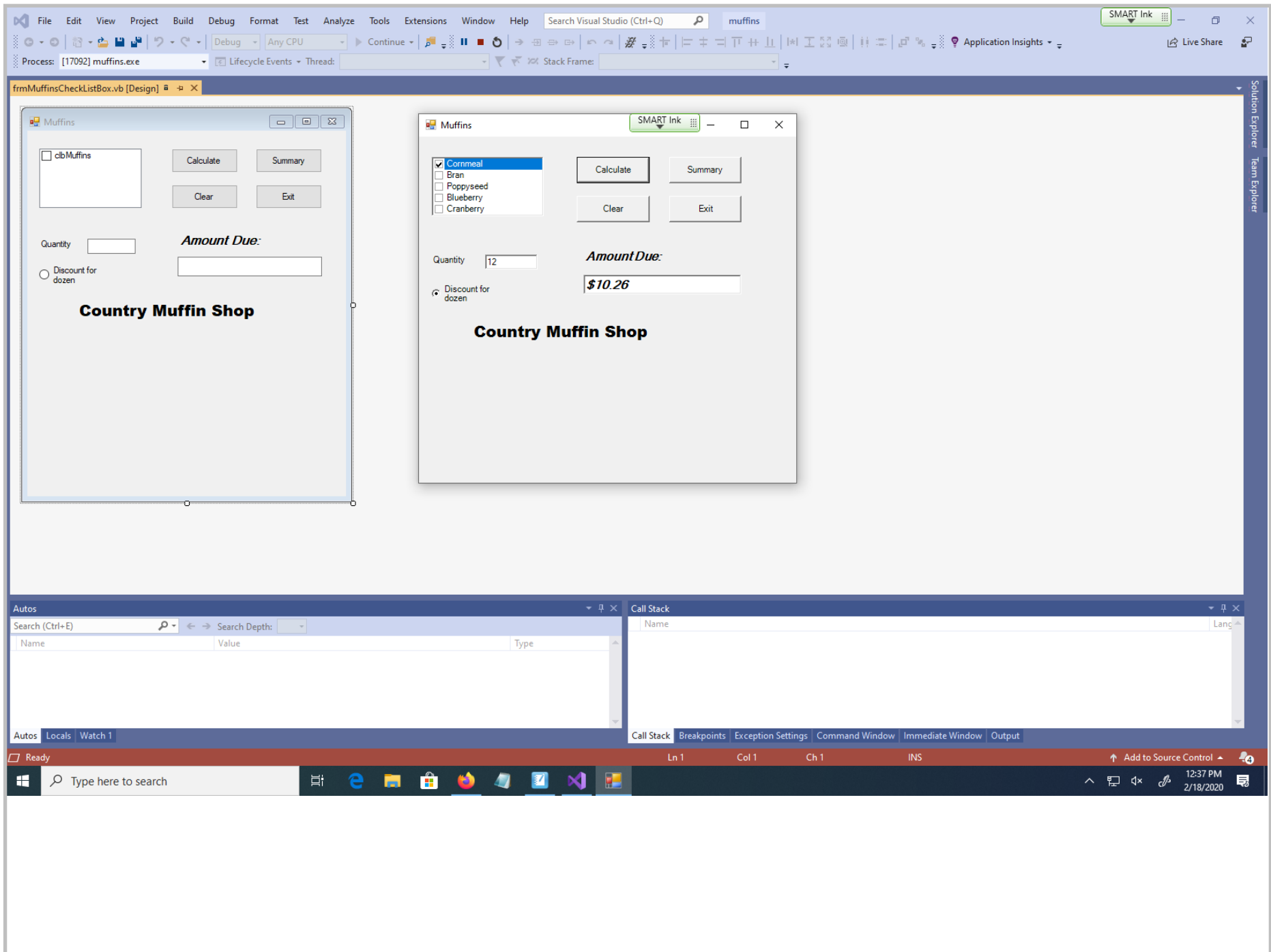
chapter as well. If you do not have the book you should find information on line on IF statements. Remember, I have some older VB books that will definitely help with logic if you want to see me about using one.  
**Some help with decision problems from chapter 4**  
 I strongly recommend that you reference the Smartboard and audio from last Spring and that you look at the examples including those under VB.NET  
**Assignments:**  
**In Class IF assignments** - everyone needs to do this and in fact all assignments posted under assignments  
**IF assignment**  
 As always, check back!

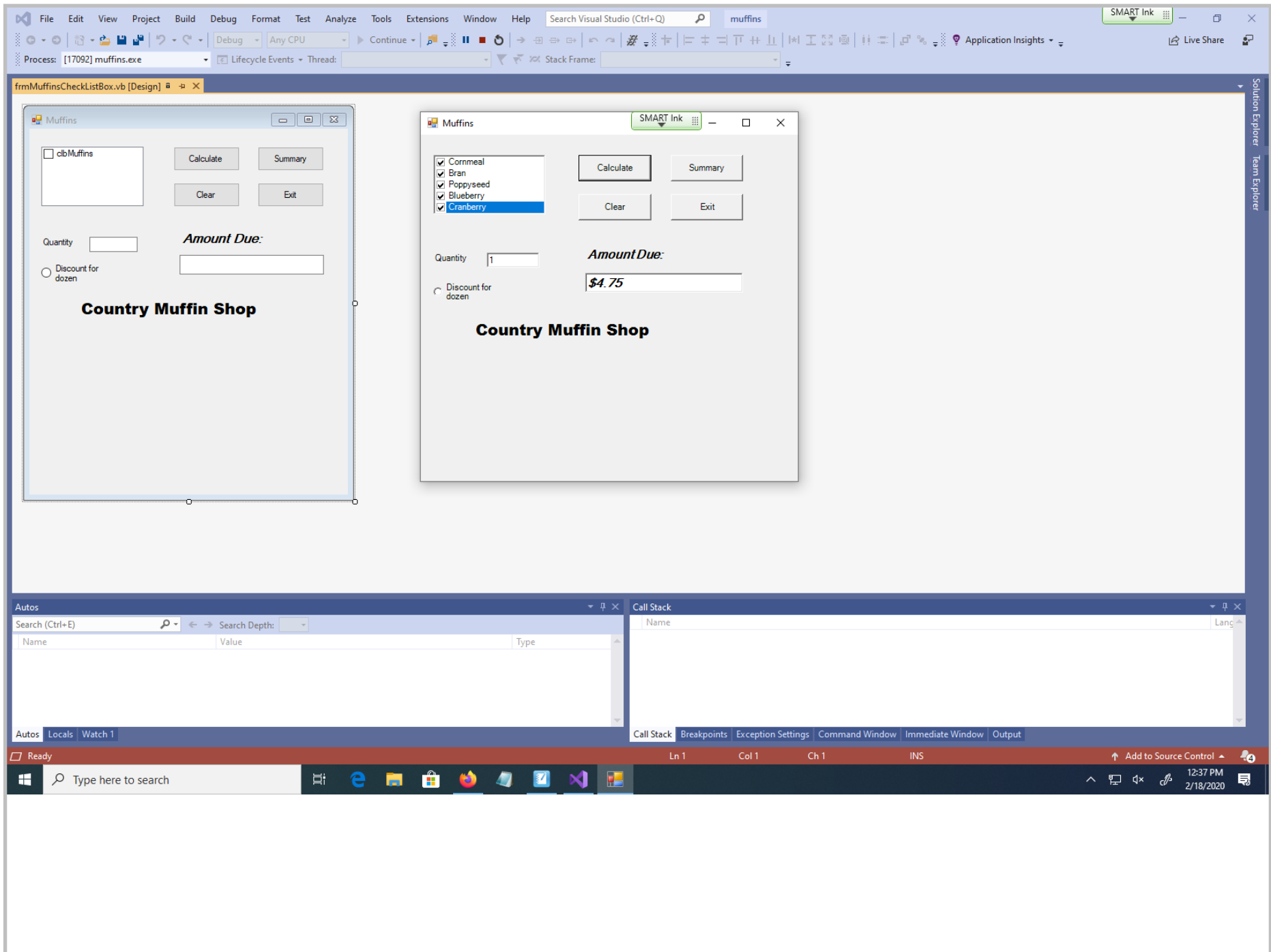
Week #3  
 Week of February 3rd

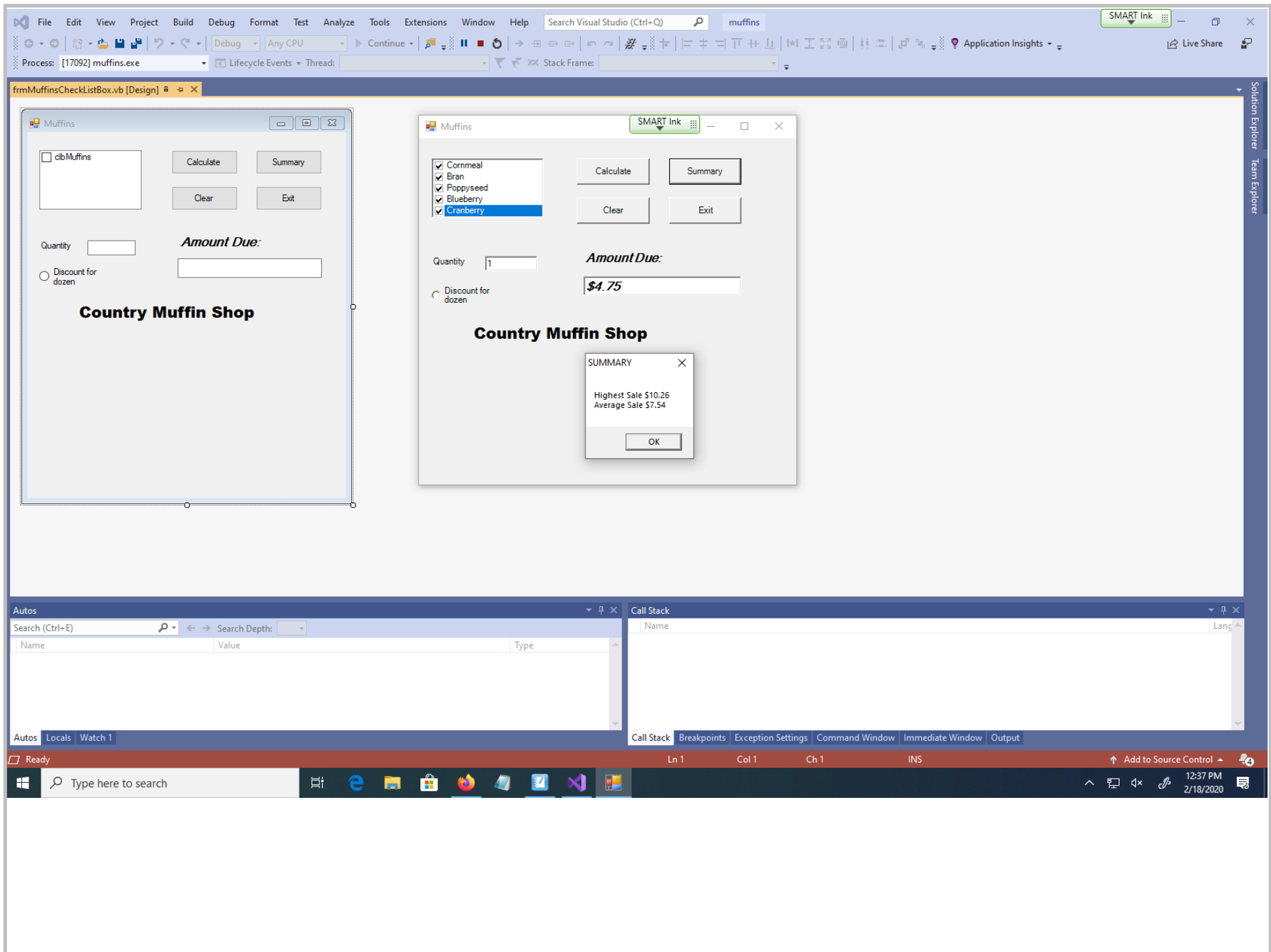
Either finish chapter #2 and then read chapter #3 or find another source to read about starting Visual Basic and using variables and input. The topics covered in the examples are the things you should follow through and read about. We will slow down the chapter pace now.

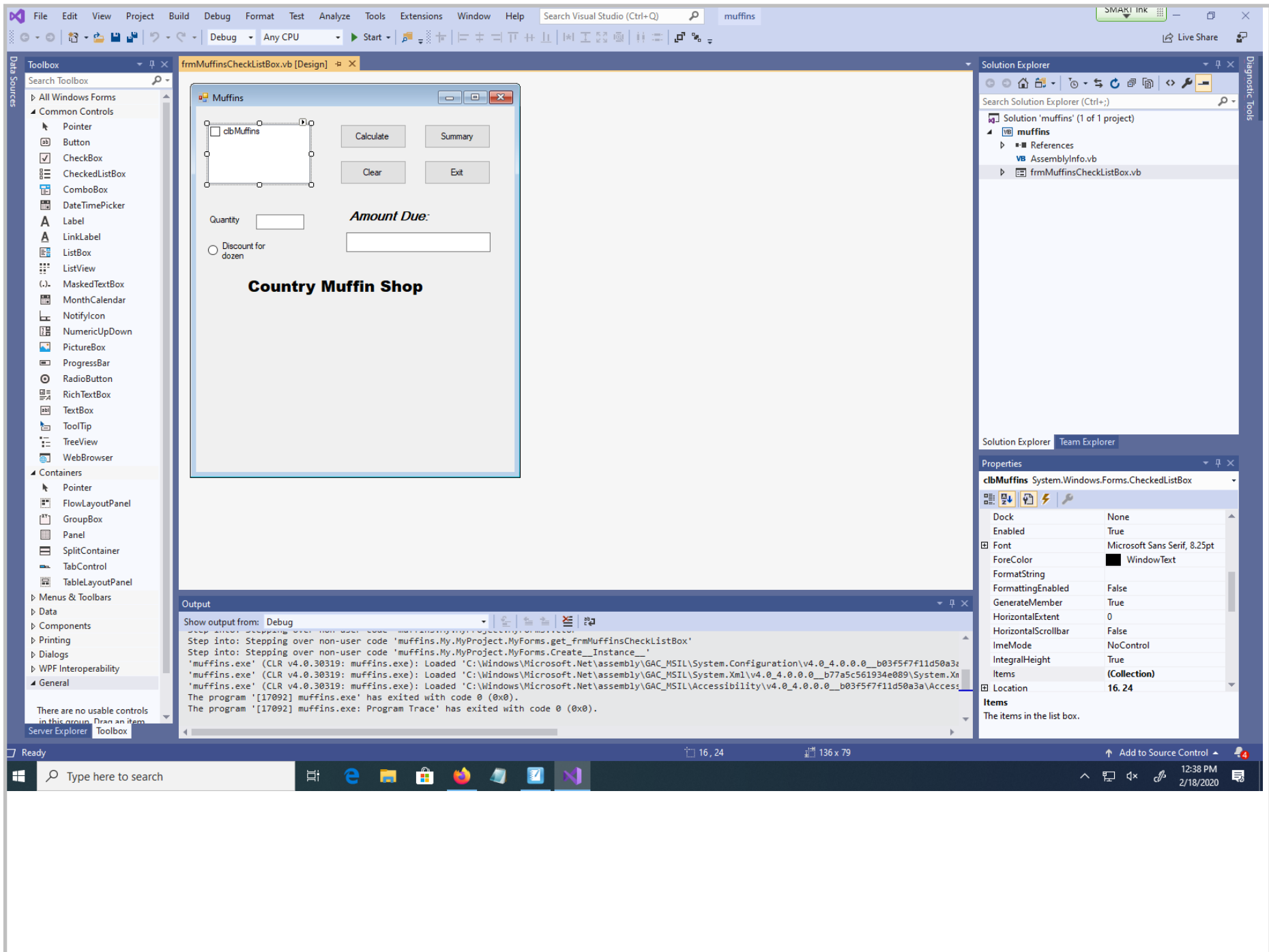
12:33 PM  
 2/18/2020

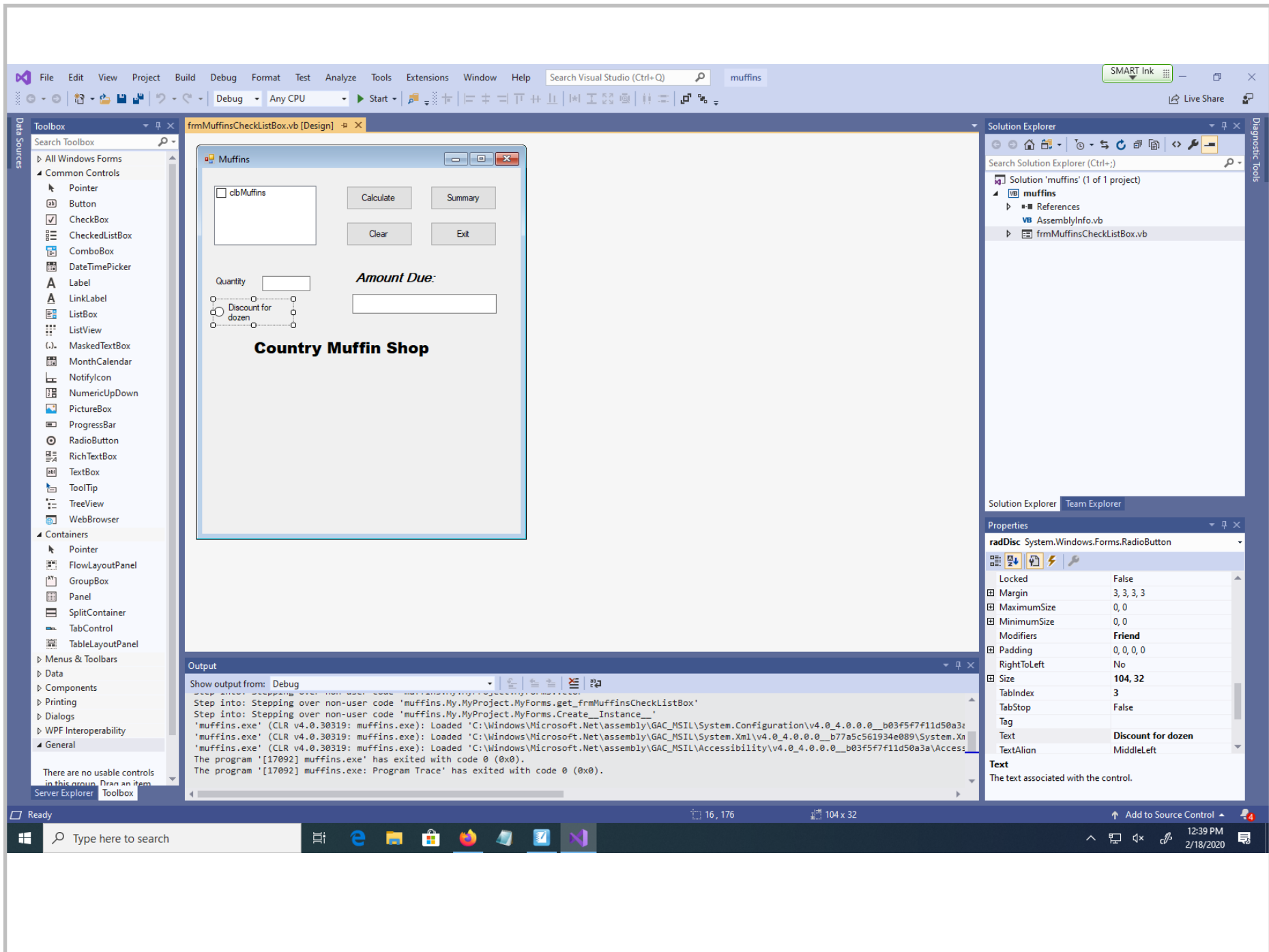




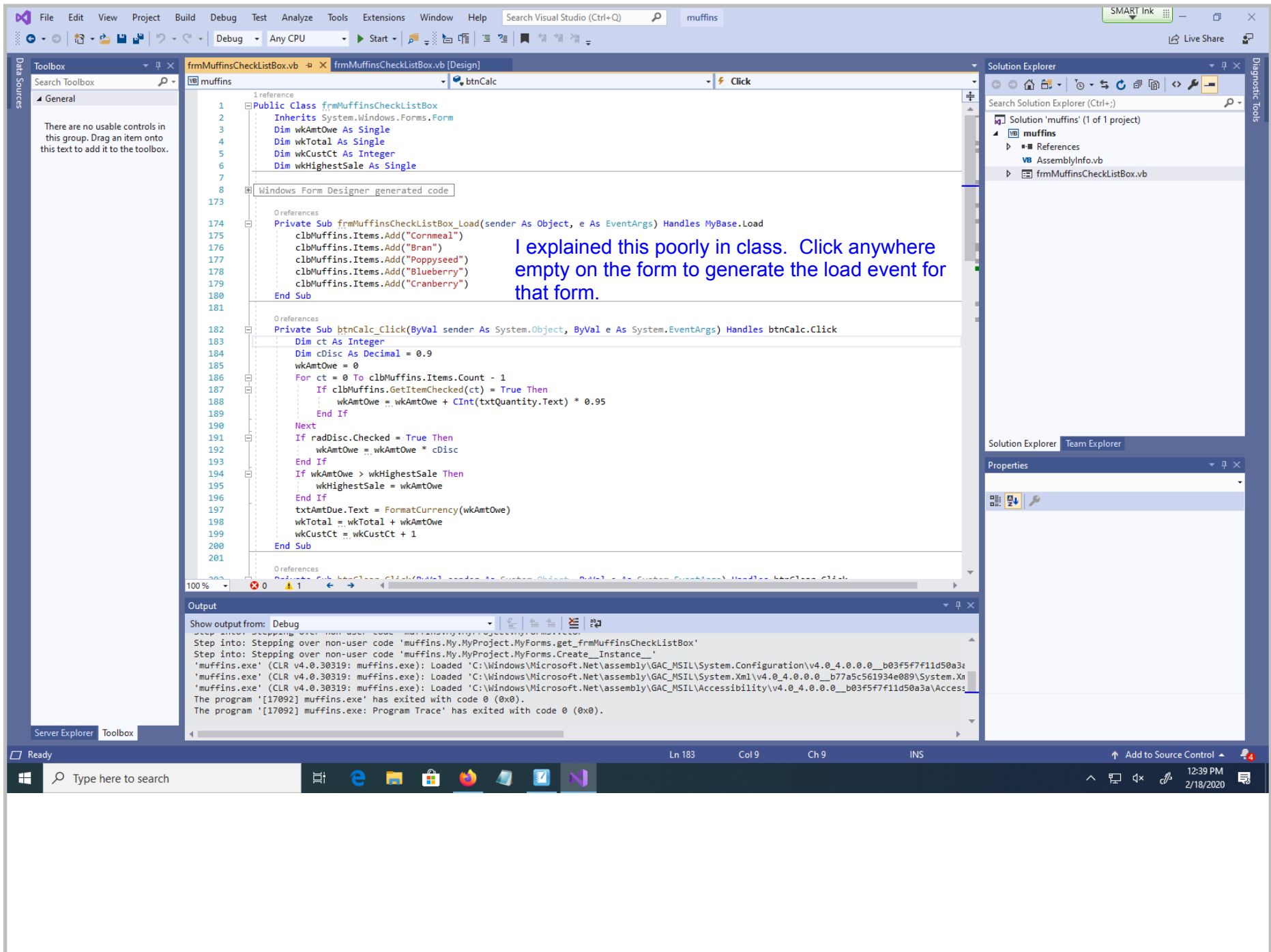












File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search Visual Studio (Ctrl+Q) muffins SMART Ink

Process: [25028] muffins.exe Lifecycle Events Thread: Stack Frame: Application Insights Live Share

frmMuffinsCheckListBox.vb [Design] (frmMuffinsCheckListBox Events) Load

```

1 reference
2 Public Class frmMuffinsCheckListBox
3     Inherits System.Windows.Forms.Form
4     Dim wkAmtOwe As Single
5     Dim wkTotal As Single
6     Dim wkCustCt As Integer
7     Dim wkHighestSale As Single
8
9     Windows Form Designer generated code
10
113
12
13 References
14 Private Sub frmMuffinsCheckListBox_Load(sender As Object, e As EventArgs) Handles MyBase.Load
15     clbMuffins.Items.Add("Cornmeal")
16     clbMuffins.Items.Add("Bran")
17     clbMuffins.Items.Add("Poppyseed")
18     clbMuffins.Items.Add("Blueberry")
19     clbMuffins.Items.Add("Cranberry")
20 End Sub
21
22 References
23 Private Sub btnCalc_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCalc.Click
24     Dim ct As Integer
25     Dim cDisc As Decimal = 0.9
26     wkAmtOwe = 0
27     For ct = 0 To clbMuffins.Items.Count - 1
28         If clbMuffins.GetItemChecked(ct) = True Then
29             wkAmtOwe = wkAmtOwe + CInt(txtQuantity.Text) * 0.95
30         End If
31     Next
32     If radDisc.Checked = True Then
33         wkAmtOwe = wkAmtOwe * cDisc
34     End If
35     If wkAmtOwe > wkHighestSale Then
36         wkHighestSale = wkAmtOwe
37     End If
38     txtAmtDue.Text = FormatCurrency(wkAmtOwe)
39     wkTotal = wkTotal + wkAmtOwe
40     wkCustCt = wkCustCt + 1
41 End Sub
42
43

```

Handwritten note:  $+1.90$   
 $7.90$   
 $+1.90$   
 $\hline$   
 $3.80$   
*wh Amt Owe*

Muffins SMART Ink

Cornmeal  
 Bran  
 Poppyseed  
 Blueberry  
 Cranberry

Quantity:

Discount for dozen

Calculate Summary  
Clear Exit

**Amount Due:**  
\$3.80

**Country Muffin Shop**

Autos Search (Ctrl+E) Search Depth: Autos Locals Watch 1

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output

Ln 175 Col 41 Ch 41 INS Add to Source Control

Type here to search 12:52 PM 2/18/2020

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search Visual Studio (Ctrl+Q) muffins SMART Ink Live Share

Process: [25028] muffins.exe Lifecycle Events Thread: Stack Frame: Load

frmMuffinsCheckListBox.vb [Design] (frmMuffinsCheckListBox Events)

```

1 reference
2 Public Class frmMuffinsCheckListBox
3     Inherits System.Windows.Forms.Form
4     Dim wkAmtOwe As Single
5     Dim wkTotal As Single
6     Dim wkCustCt As Integer
7     Dim wkHighestSale As Single
8
9     Windows Form Designer generated code
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201

```

```

174 Private Sub frmMuffinsCheckListBox_Load(sender As Object, e As EventArgs) Handles MyBase.Load
175     clbMuffins.Items.Add("Cornmeal")
176     clbMuffins.Items.Add("Bran")
177     clbMuffins.Items.Add("Poppyseed")
178     clbMuffins.Items.Add("Blueberry")
179     clbMuffins.Items.Add("Cranberry")
180 End Sub

```

```

182 Private Sub btnCalc_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCalc.Click
183     Dim ct As Integer
184     Dim cDisc As Decimal = 0.9
185     wkAmtOwe = 0
186     For ct = 0 To clbMuffins.Items.Count - 1
187         If clbMuffins.GetItemChecked(ct) = True Then
188             wkAmtOwe = wkAmtOwe + CInt(txtQuantity.Text) * 0.95
189         End If
190     Next
191     If radDisc.Checked = True Then
192         wkAmtOwe = wkAmtOwe * cDisc
193     End If
194     If wkAmtOwe > wkHighestSale Then
195         wkHighestSale = wkAmtOwe
196     End If
197     txtAmtDue.Text = FormatCurrency(wkAmtOwe)
198     wkTotal = wkTotal + wkAmtOwe
199     wkCustCt = wkCustCt + 1
200 End Sub

```

Muffins

SMART Ink

Cornmeal  
 Bran  
 Poppyseed  
 Blueberry  
 Cranberry

Quantity: 2

Amount Due: \$3.80

Discount for dozen:

Calculate Summary

Clear Exit

Country Muffin Shop

Summary

Highest Sale \$3.80  
Average Sale \$3.80

OK

Handwritten notes:

+1.90 = wk Amt Owe

7.90 = wk Total

1.90 = wk CustCt

3.80

Autos

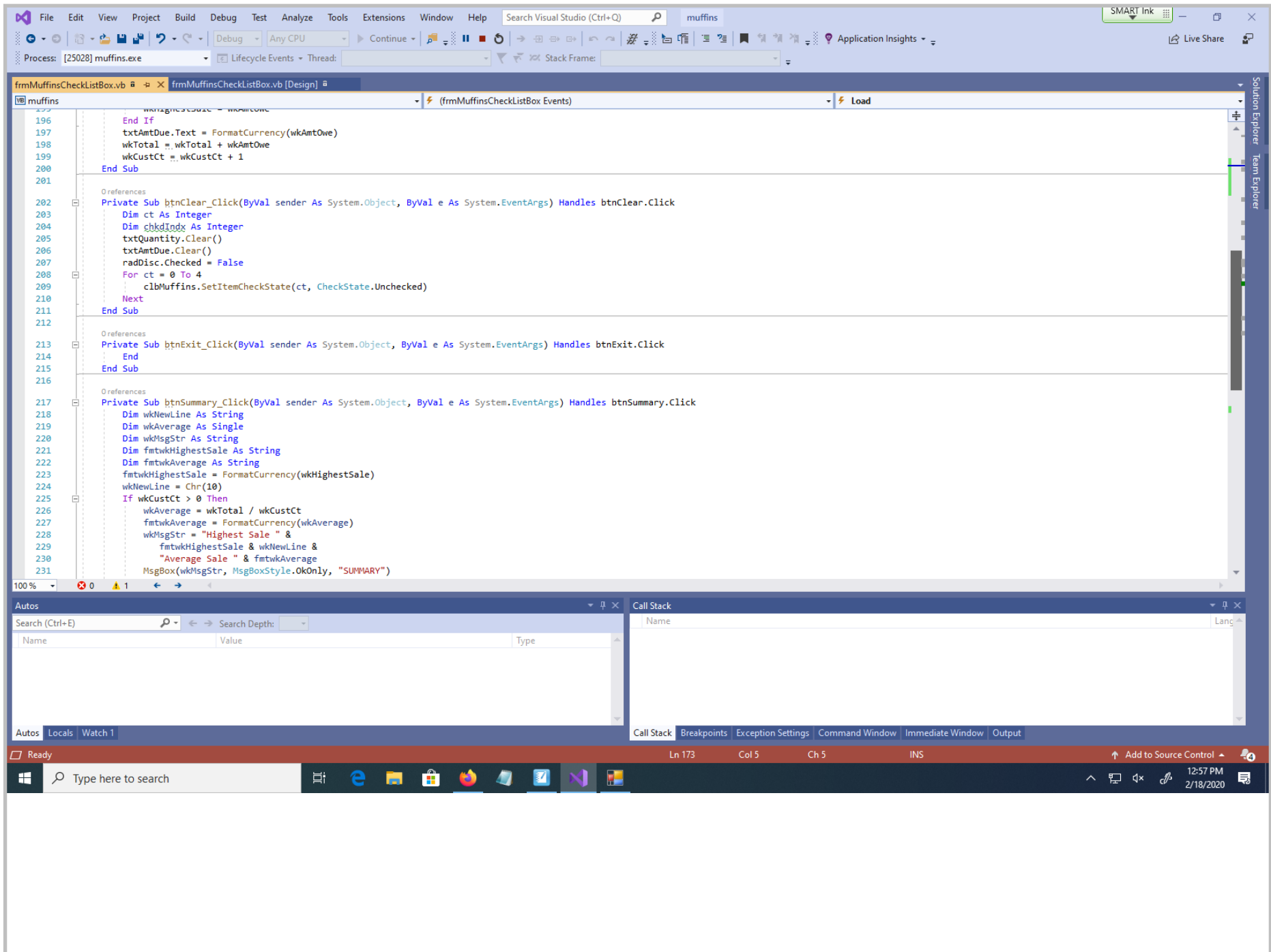
Search (Ctrl+E) Search Depth: Name Value Type

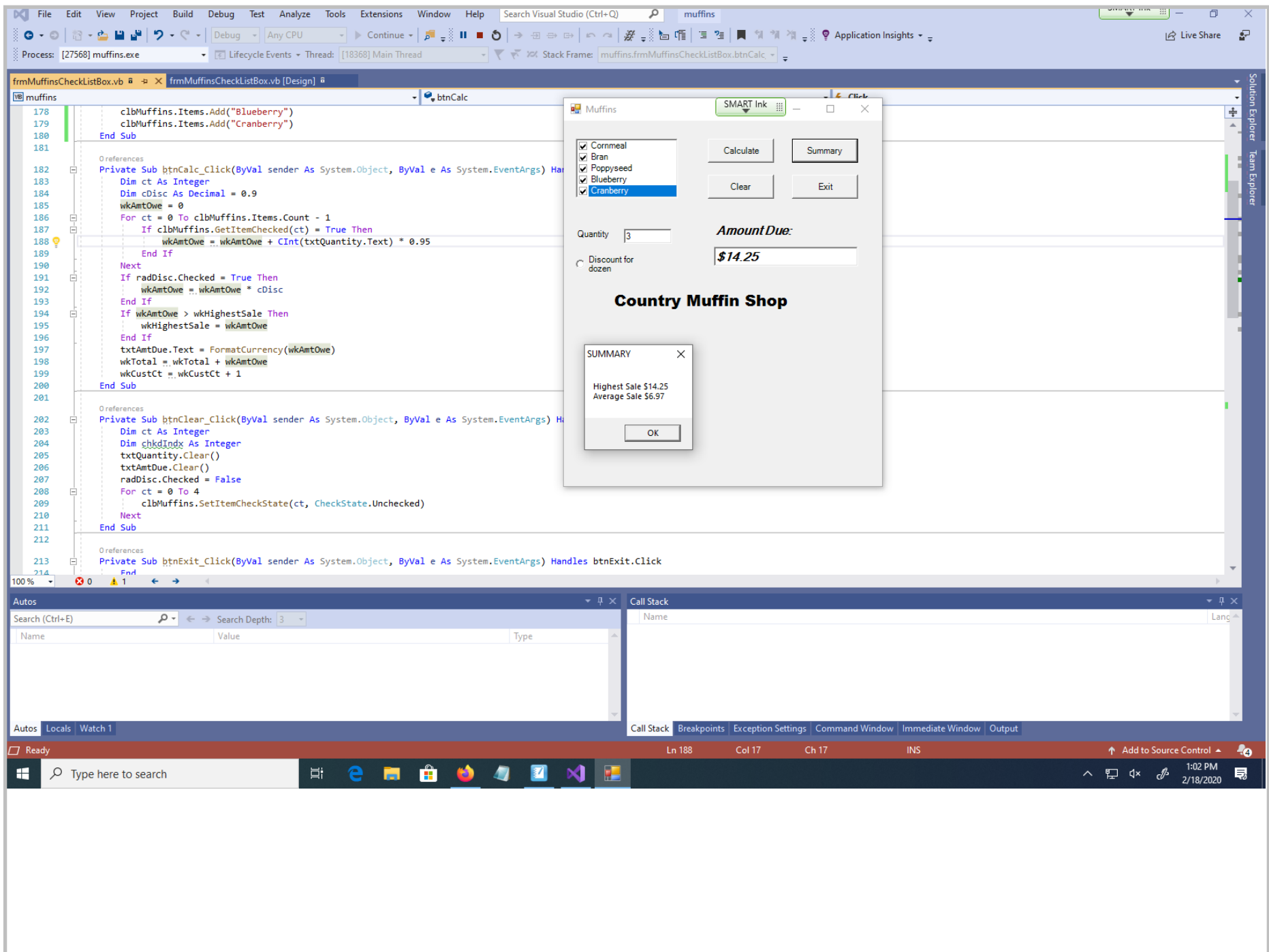
Call Stack

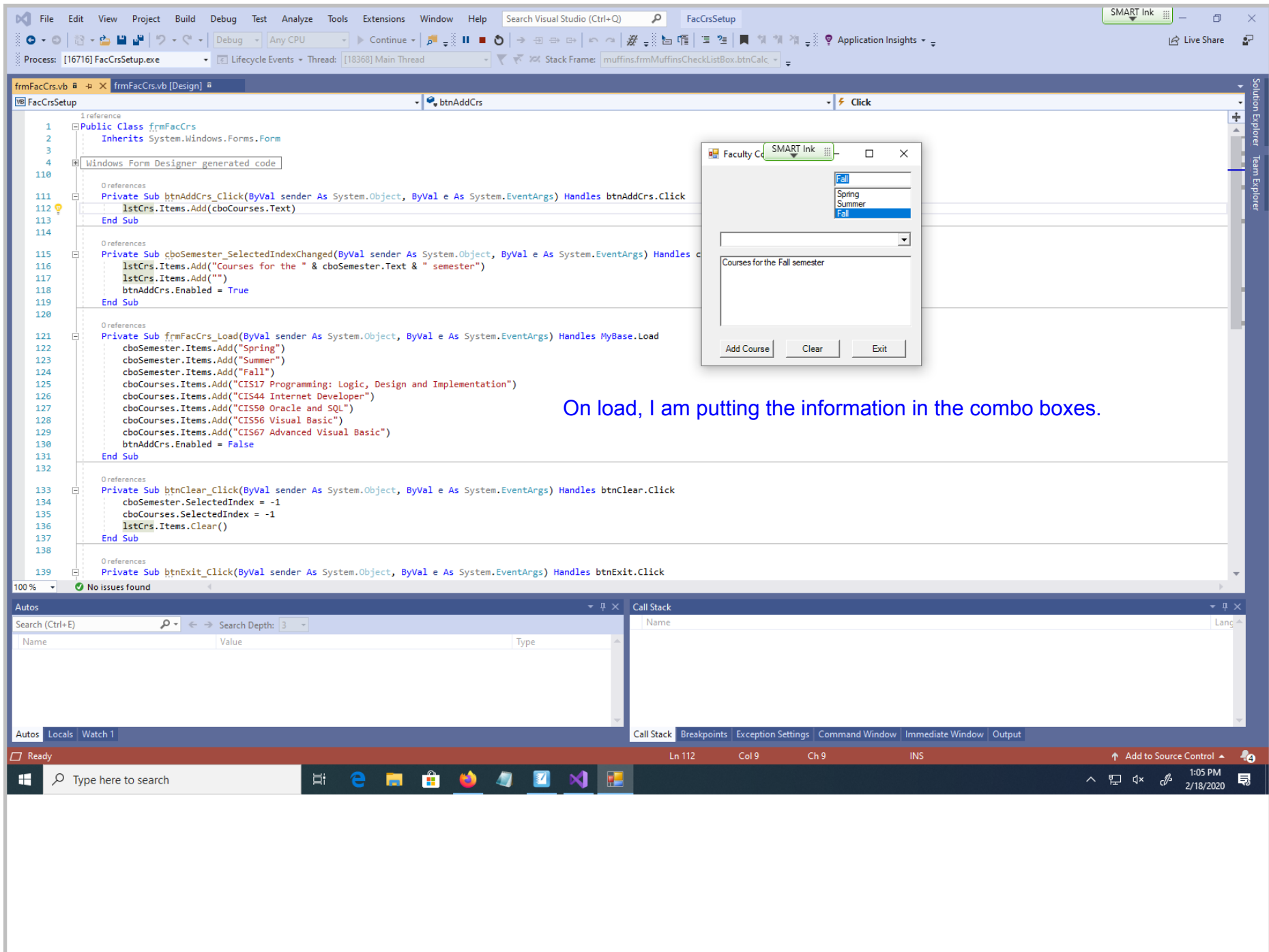
Call Stack Breakpoints Exception Settings Command Window Immediate Window Output

Ready Ln 175 Col 41 Ch 41 INS Add to Source Control

Type here to search 12:54 PM 2/18/2020







The screenshot displays the Visual Studio IDE with the following components:

- Code Editor:** Shows the source code for `frmFacCrsvb`. The code includes a class declaration, a `Load` event handler that populates `cboSemester` and `cboCourses`, and several click event handlers for `btnAddCrsvb`, `btnClear`, and `btnExit`.
- Application Window:** A window titled "Faculty C..." is open, showing a dropdown menu for semester selection (Fall, Spring, Summer) with "Fall" selected. Below it is a list box containing "Courses for the Fall semester" with items: "CIS50 Oracle and SQL" and "CIS56 Visual Basic". Buttons for "Add Course", "Clear", and "Exit" are at the bottom.
- Autos Window:** Empty, showing a search depth of 3.
- Call Stack Window:** Empty.
- Taskbar:** Shows the Windows taskbar with the time 1:06 PM on 2/18/2020.

The screenshot displays the Visual Studio IDE with a Windows Forms application named 'Faculty Courses' in design mode. The form contains a text box, a list box labeled 'lstCrs', and three buttons: 'Add Course', 'Clear', and 'Exit'. A pink dashed arrow points from the 'lstCrs' list box to the 'DropDownStyle' property in the Properties window. The Properties window shows 'DropDownStyle' set to 'DropDown'. The Output window shows debug output for the application.

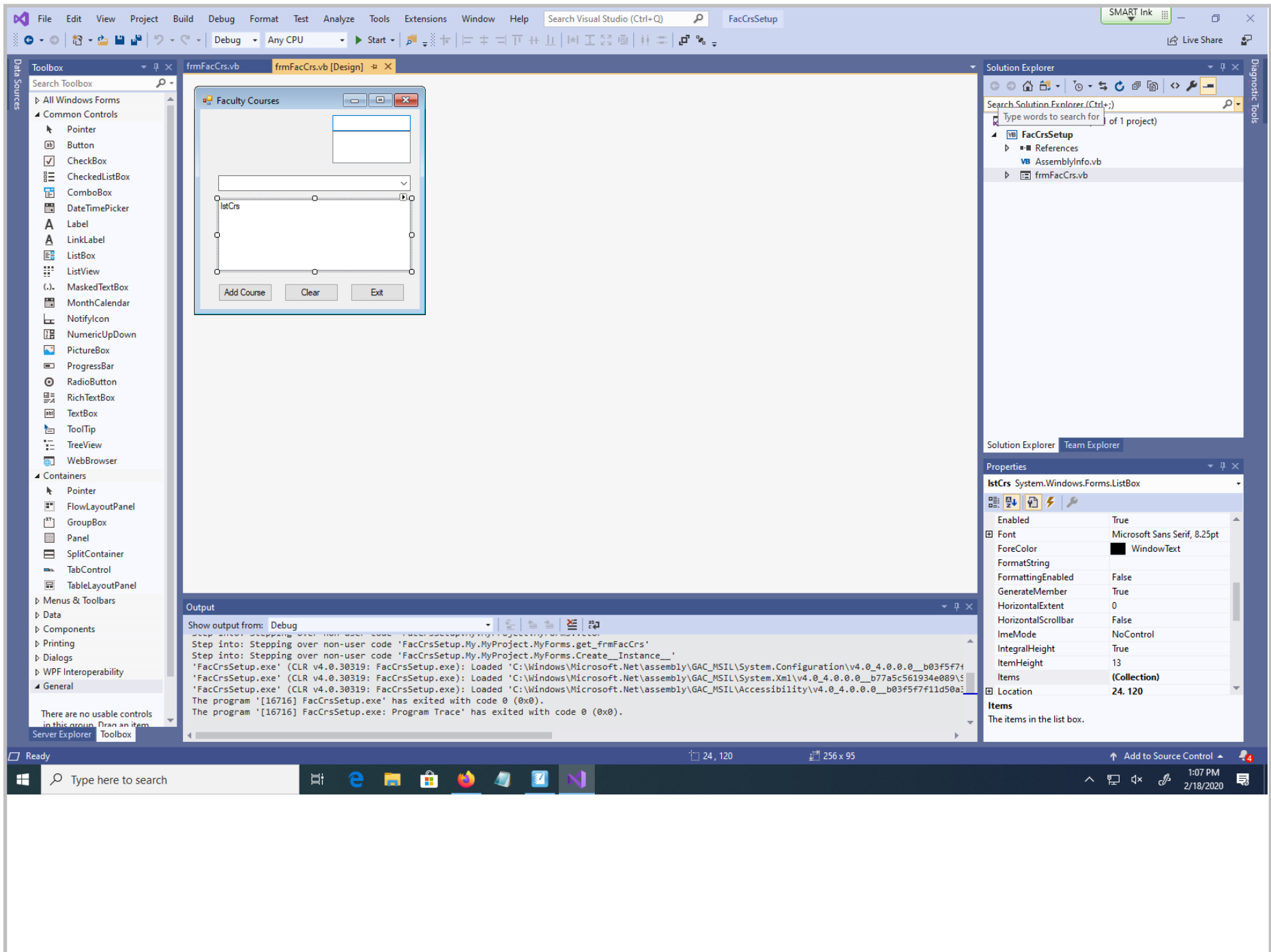
**Properties Window:**

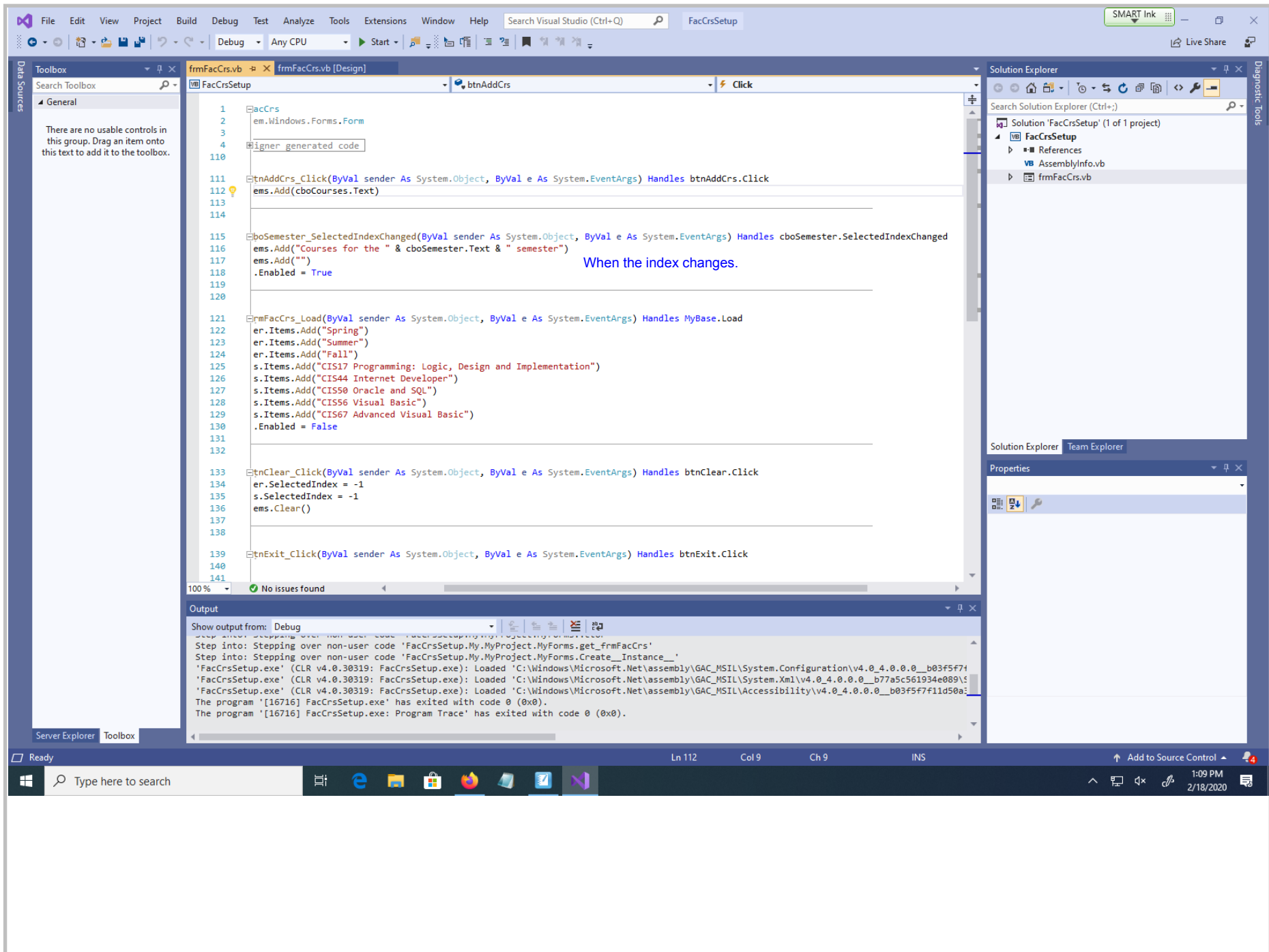
Property	Value
ContextMenuStrip	(none)
Cursor	Default
DataSource	(none)
DisplayMember	(none)
Dock	None
DrawMode	Normal
DropDownHeight	106
<b>DropDownStyle</b>	<b>DropDown</b>
DropDownWidth	256
Enabled	True
FlatStyle	Standard
Font	Microsoft Sans Serif, 8.25pt
ForeColor	WindowText

**Output Window:**

```
Show output from: Debug
Step into: Stepping over non-user code 'FacCrSetup.My.MyProject.MyForms.get_frmFacCr...'
Step into: Stepping over non-user code 'FacCrSetup.My.MyProject.MyForms.get_frmFacCr...'
Step into: Stepping over non-user code 'FacCrSetup.My.MyProject.MyForms.Create_Instance...'
'FacCrSetup.exe' (CLR v4.0.30319: FacCrSetup.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Configuration\v4.0.4.0.0__b03f5f71
'FacCrSetup.exe' (CLR v4.0.30319: FacCrSetup.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Xml\v4.0.4.0.0__b77a5c561934e089\
'FacCrSetup.exe' (CLR v4.0.30319: FacCrSetup.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Xml\v4.0.4.0.0__b77a5c561934e089\
'FacCrSetup.exe' (CLR v4.0.30319: FacCrSetup.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\Accessibility\v4.0.4.0.0__b03f5f711d50a
The program '[16716] FacCrSetup.exe' has exited with code 0 (0x0).
The program '[16716] FacCrSetup.exe: Program Trace' has exited with code 0 (0x0).
```

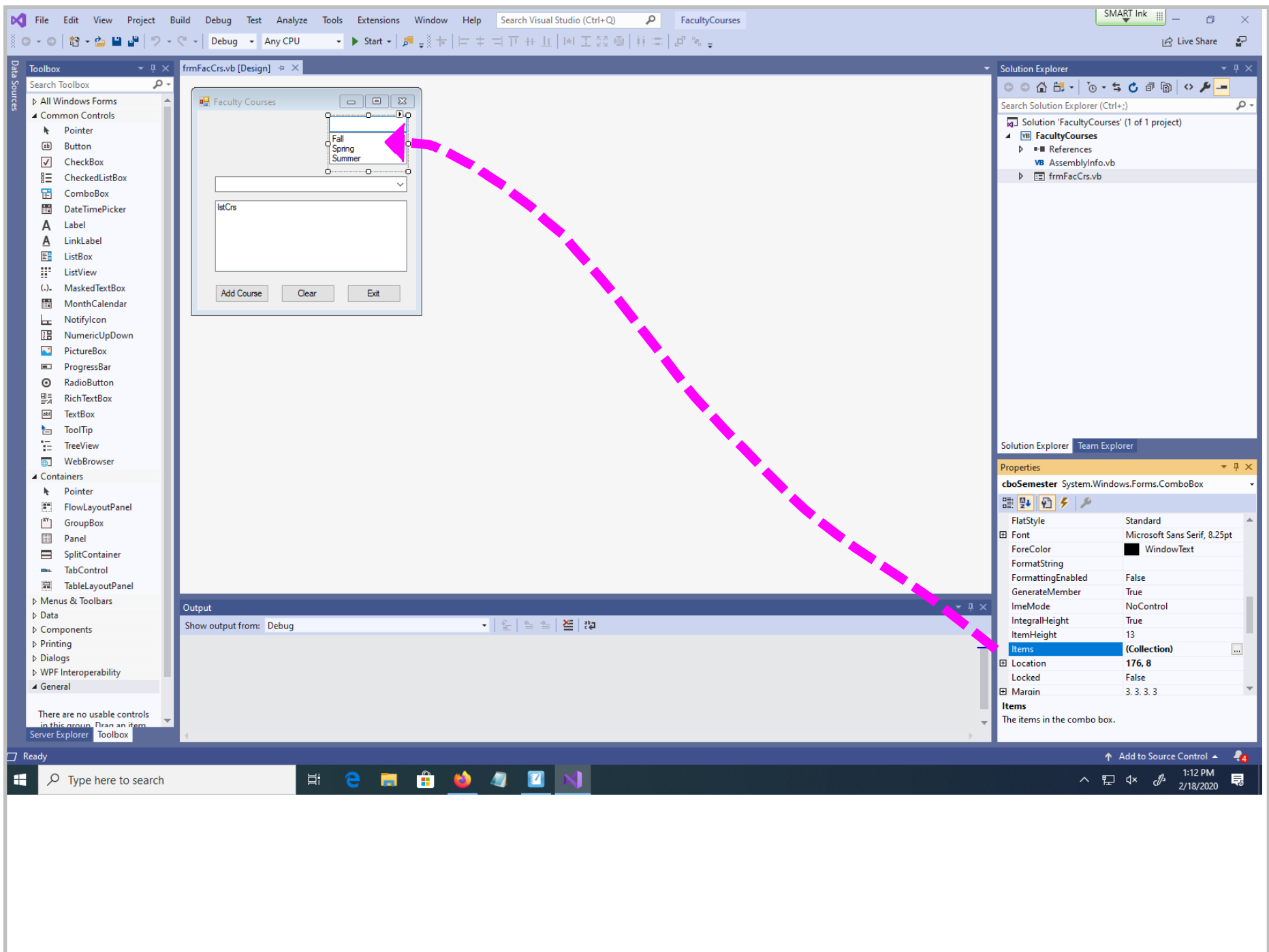


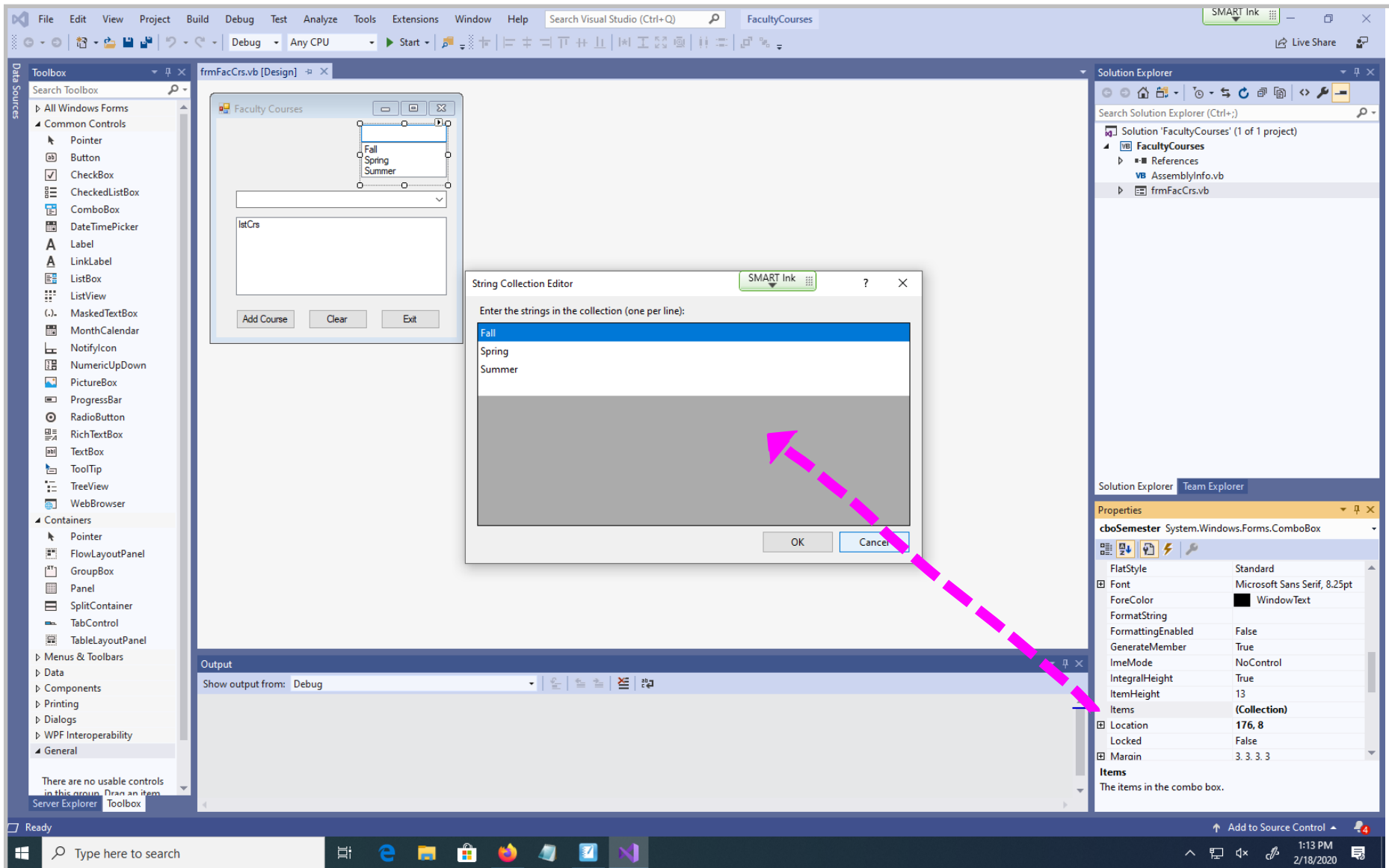




The image shows a screenshot of the Visual Studio IDE. The main window displays the code for `frmFacCrS.vb` in Design mode, with the `btnAddCrS` control selected. The code includes several event handlers for `btnAddCrS`, `cboSemester`, and `btnClear`. A small application window titled "Faculty C..." is overlaid on the code, showing a list of semesters (Fall, Spring, Summer, Fall) and a list of courses for the selected semester (Fall). The courses listed are "CIS17 Programming: Logic, Design and Implementati..." and "CIS56 Visual Basic". The application window has "Add Course", "Clear", and "Exit" buttons.

```
3
4
110
111 Private Sub btnAddCrS_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddCrS.Click
112     . . . lstCrS.Items.Add(cboCourses.Text)
113 End Sub
114
115 Private Sub cboSemester_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cboSemester.SelectedIndexCh
116     . . . lstCrS.Items.Add("Courses for the " & cboSemester.Text & " semester")
117     . . . lstCrS.Items.Add("")
118     . . . btnAddCrS.Enabled = True
119 End Sub
120
121 Private Sub frmFacCrS_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
122     . . . cboSemester.Items.Add("Spring")
123     . . . cboSemester.Items.Add("Summer")
124     . . . cboSemester.Items.Add("Fall")
125     . . . cboCourses.Items.Add("CIS17 Programming: Logic, Design and Implementation")
126     . . . cboCourses.Items.Add("CIS44 Internet Developer")
127     . . . cboCourses.Items.Add("CIS50 Oracle and SQL")
128     . . . cboCourses.Items.Add("CIS56 Visual Basic")
129     . . . cboCourses.Items.Add("CIS67 Advanced Visual Basic")
130     . . . btnAddCrS.Enabled = False
131 End Sub
132
133 Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
134     . . . cboSemester.SelectedIndex = -1
135     . . . cboCourses.SelectedIndex = -1
136     . . . lstCrS.Items.Clear()
137 End Sub
138
139 Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
140     . . . End
141 End Sub
142 End Class
```





The season and courses are set up through the properties collection for use. In the previous example, they were loaded at run time. In the next example, they will be entered using user input.

String Collection Editor

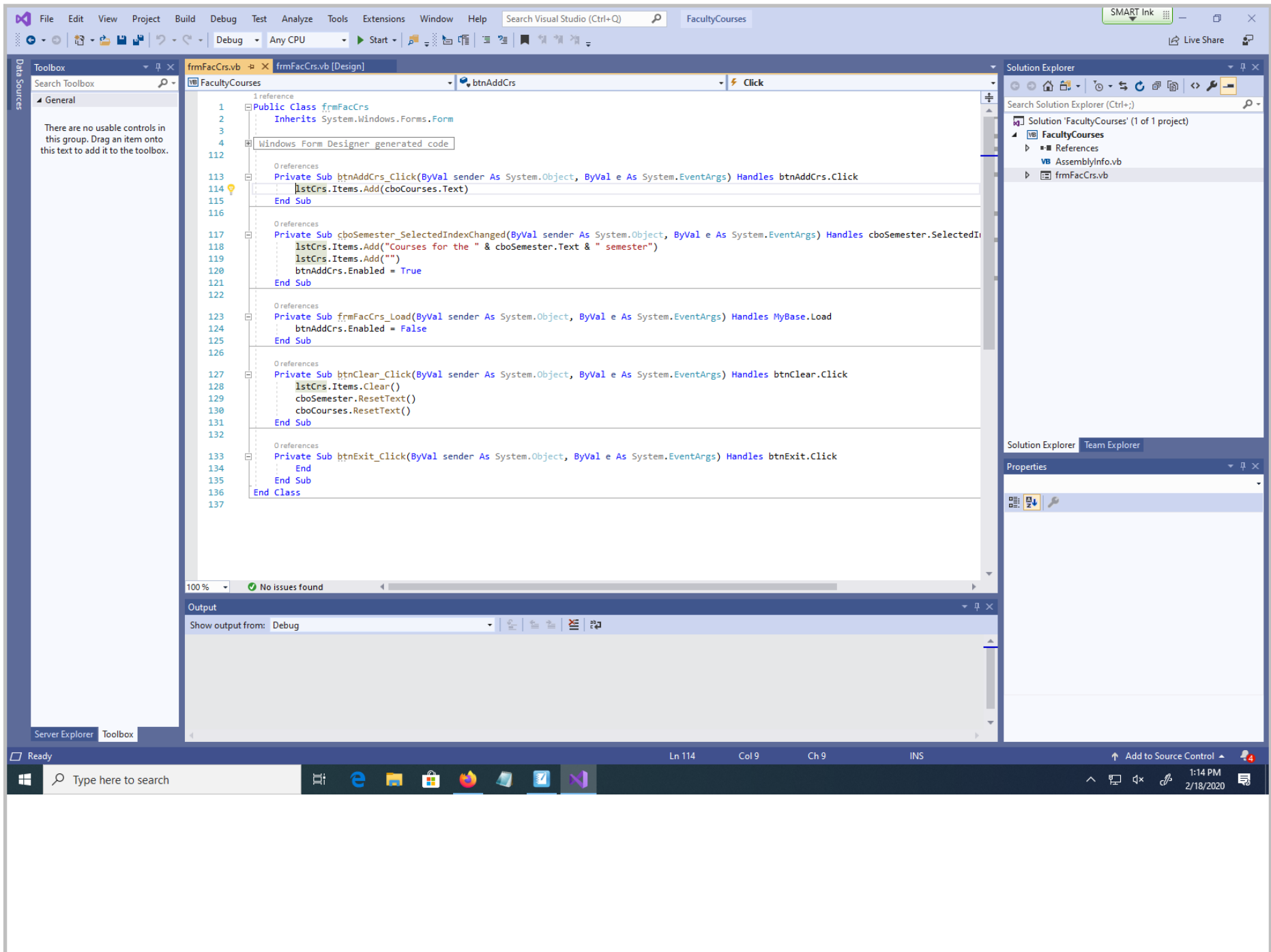
Enter the strings in the collection (one per line):

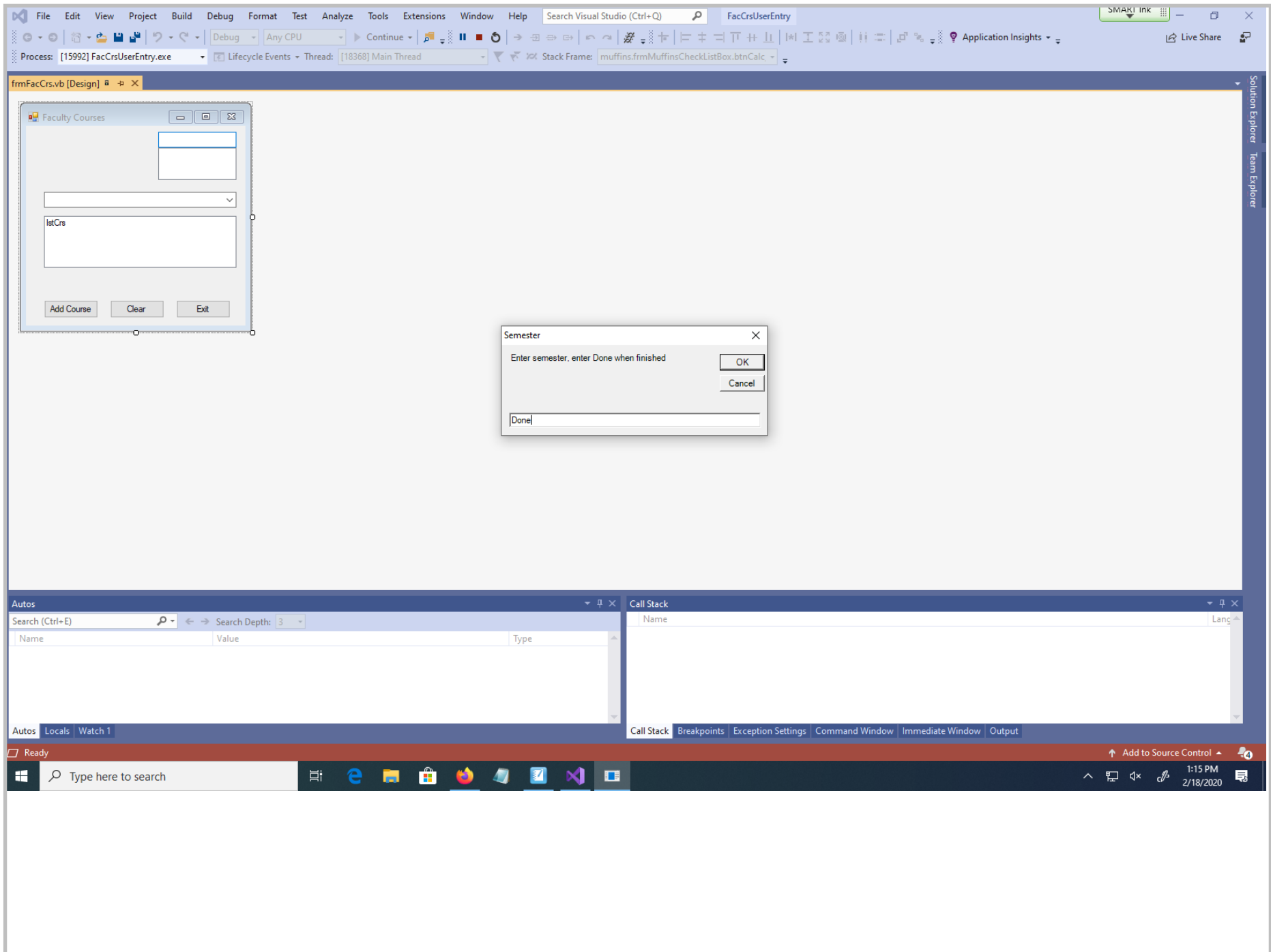
- CIS17 Programming: Logic Design and Implementation
- CIS44 Internet Developer
- CIS50 Oracle and SQL
- CIS56 Visual Basic
- CIS67 Advanced Visual Basic

Properties

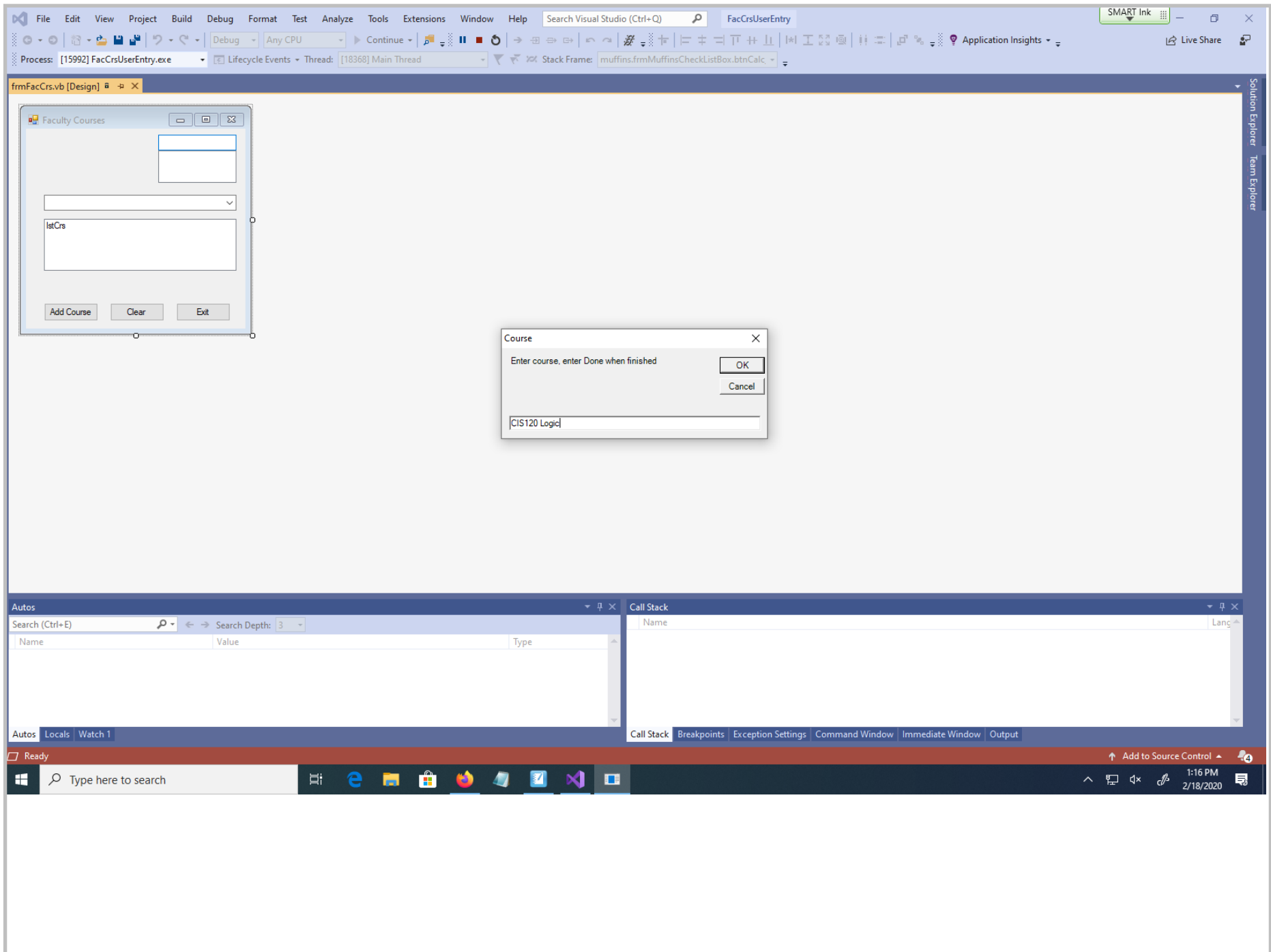
FlatStyle	Standard
Font	Microsoft Sans Serif, 8.25pt
ForeColor	Window Text
FormatString	
FormattingEnabled	False
GenerateMember	True
ImeMode	NoControl
IntegralHeight	True
ItemHeight	13
Items	(Collection)
Location	24, 88
Locked	False
Margin	3, 3, 3, 3

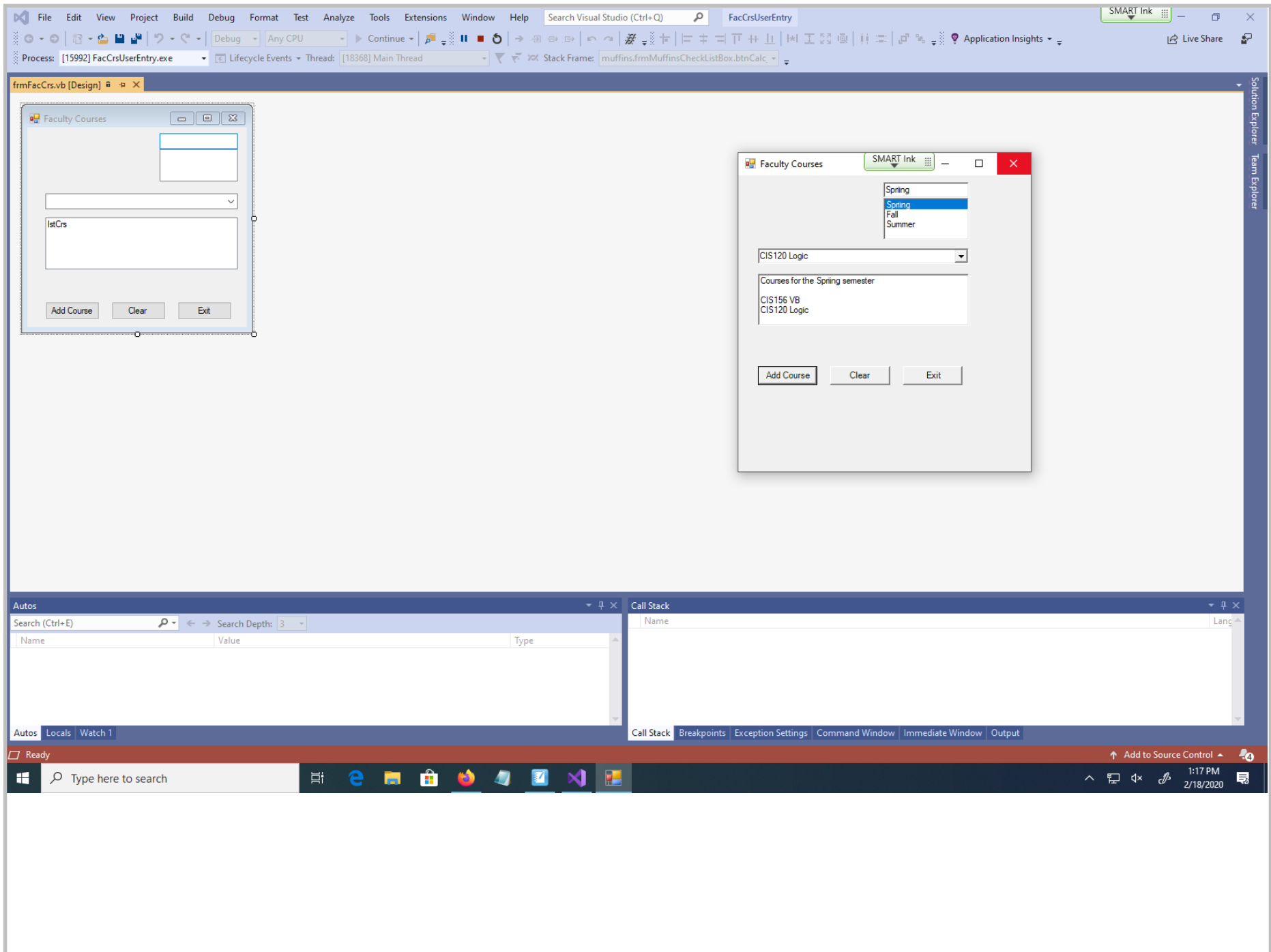
Items  
The items in the combo box.

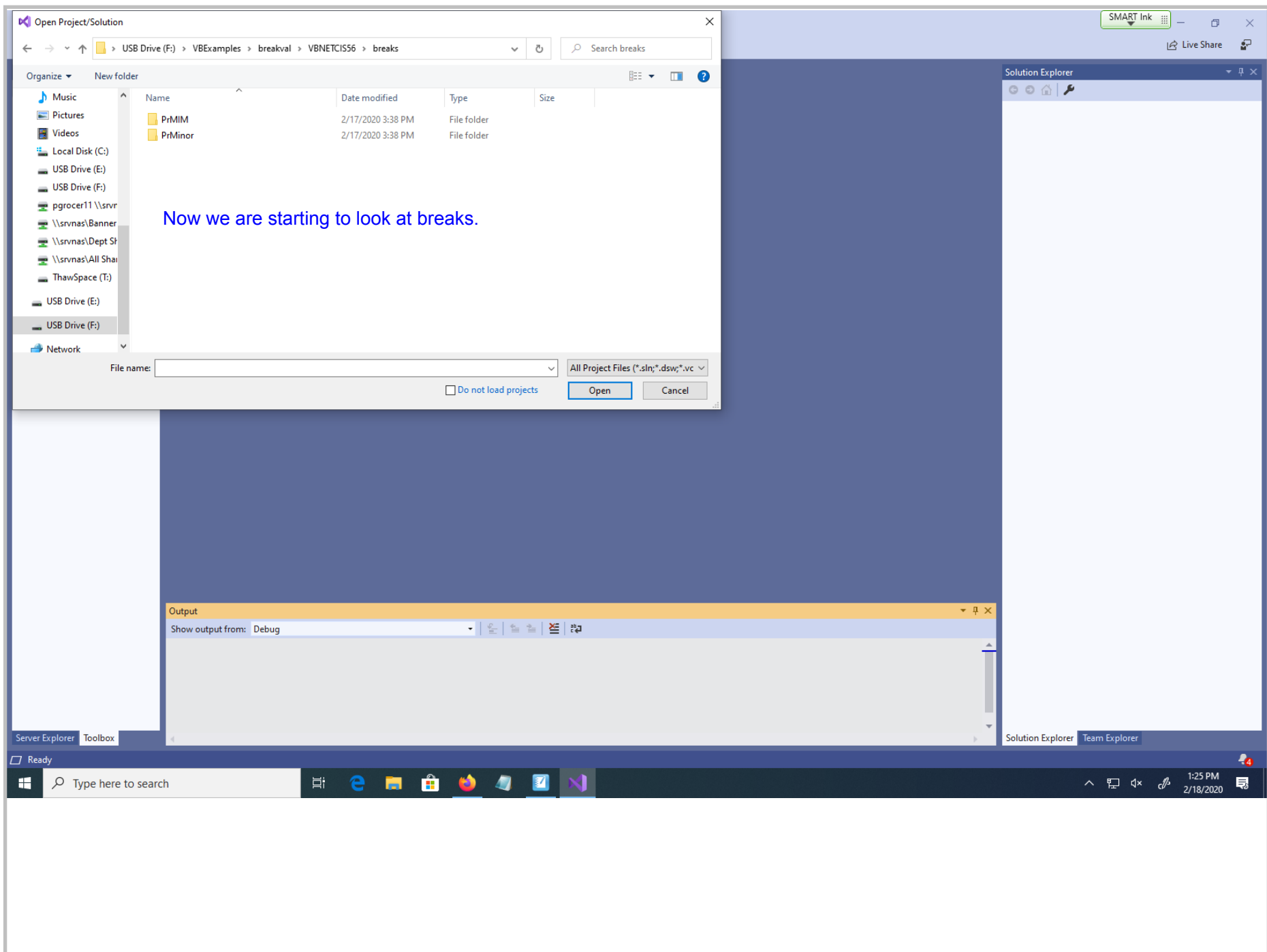


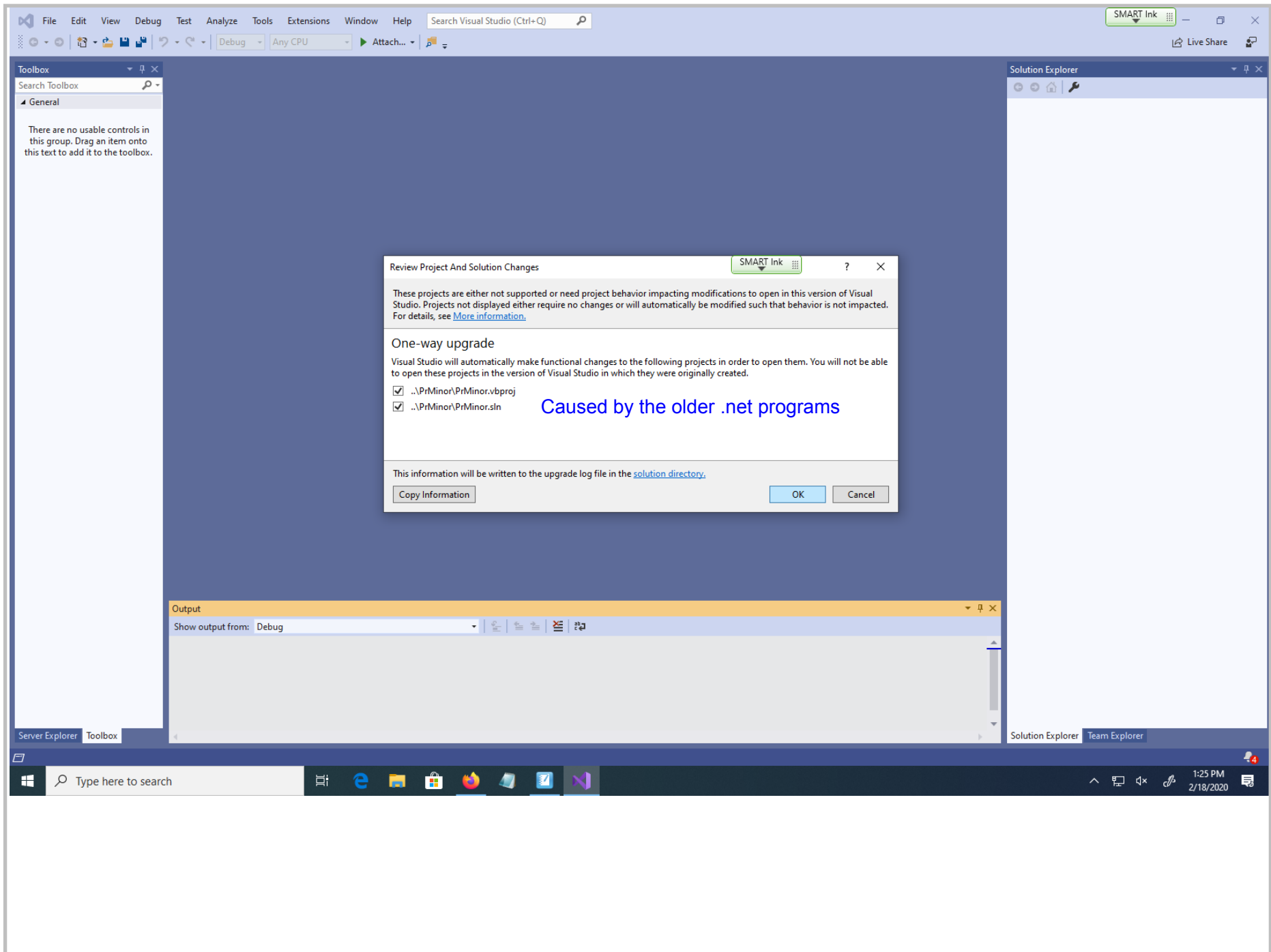












The screenshot shows a Visual Studio IDE with the following components:

- Code Editor:** Displays the code for `frmMinor`. The `btnProcess_Click` event handler reads a file named `Minor.txt` and processes its contents. The code uses `Input` to read department numbers and amounts, and `lstMinor.Items.Add` to display the results. The code also includes logic to update `wkMinTot` and `wkFinTot` based on the input.
- Minor Totals Window:** A small application window titled "Minor Totals" with a list of items and two buttons: "Process" and "Exit". The list contains the following data:
 

12	\$500.00
12	\$100.00
12	\$50.00
Dept Total: \$650.00	
15	\$200.00
15	\$100.00
15	\$300.00
15	\$60.00
Dept Total: \$660.00	
17	\$20.00
17	\$240.00
Dept Total: \$260.00	
Final Total: \$1,570.00	
- Minor - Notepad:** A Notepad window showing the contents of `Minor.txt`, which are pairs of department numbers and amounts, ordered by department number:
 

```
"12", "500"
"12", "100"
"12", "050"
"15", "200"
"15", "100"
"15", "300"
"15", "060"
"17", "020"
"17", "240"
```
- Autos Window:** Located at the bottom, it shows the current state of variables in the application.

Reading a file that has been ordered by dept number.

Visual Studio interface showing the code for `frmMinor` and a running application window titled "Minor Totals".

```

1 reference
2 Public Class frmMinor
3     Inherits System.Windows.Forms.Form
4     Dim wkMinTot As Integer
5     Dim wkFinTot As Integer
6     Dim wkHoldDept As String
7
81 Windows Form Designer generated code
82
83 Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
84     FileOpen(1, "Minor.txt", OpenMode.Input)
85     wkHoldDept = ""
86 End Sub
87
88 Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
89     Dim wkDeptNo As String, wkAmt As String
90     If Not EOF(1) Then
91         Input(1, wkDeptNo)
92         Input(1, wkAmt)
93         If wkHoldDept <> wkDeptNo Then
94             If wkHoldDept <> "" Then
95                 lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
96                 wkHoldDept = wkDeptNo
97                 wkMinTot = 0
98             Else
99                 wkHoldDept = wkDeptNo
100            End If
101            lstMinor.Items.Add(wkDeptNo & " " & FormatCurrency(wkAmt))
102            wkMinTot = wkMinTot + wkAmt
103            wkFinTot = wkFinTot + wkAmt
104        Else
105            MsgBox("End of File", vbOKOnly, "EOF")
106            lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
107            lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
108            FileClose(1)
109        End If
110    End Sub
111
112 Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
113     End
114 End Sub
115 End Class
116

```

The "Minor Totals" application window displays the following data:

12	\$500.00
12	\$100.00
12	\$50.00
Dept Total:	\$650.00
15	\$200.00
15	\$100.00
15	\$300.00
15	\$90.00
Dept Total:	\$660.00
17	\$20.00
17	\$240.00
Dept Total:	\$260.00
Final Total:	\$1,570.00

A Notepad window titled "Minor - Notepad" contains the following text:

```

"12", "500"
"12", "100"
"12", "050"
"15", "200"
"15", "100"
"15", "300"
"15", "060"
"17", "020"
"17", "240"

```

Handwritten annotations in blue ink on the Notepad window:

- `wk Hold Dept` and `wk Dept No` labels with arrows pointing to the first two columns of data.
- An arrow pointing from the value `12` in the second column to the value `12` in the first column.
- An arrow pointing from the value `15` in the second column to the value `15` in the first column.
- The text `Write total` written vertically between the two arrows.

Visual Studio screenshot showing the code for `frmMinor` and `btnProcess`. The code includes logic for handling file operations and calculating departmental totals. Handwritten annotations in blue ink include "Write total" with an arrow pointing to the `btnProcess` sub, "broke" near the `wkHoldDept` logic, and "move to hold reset tot to 0 minor" near the `Input` statements. A `Minor - Notepad` window displays a list of departmental entries, and a `Minor Totals` dialog box shows a summary of these totals with `Process` and `Exit` buttons.

```

1 Public Class frmMinor
2     Inherits System.Windows.Forms.Form
3     Dim wkMinTot As Integer
4     Dim wkFinTot As Integer
5     Dim wkHoldDept As String
6
7     'Windows Form Designer generated code
81
82     Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
83         FileOpen(1, "Minor.txt", OpenMode.Input)
84         wkHoldDept = ""
85     End Sub
86
87     Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
88         Dim wkDeptNo As String, wkAmt As String
89         If Not EOF(1) Then
90             Input(1, wkDeptNo)
91             Input(1, wkAmt)
92             If wkHoldDept <> "" Then
93                 If wkHoldDept <> "" Then
94                     lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
95                     wkHoldDept = wkDeptNo
96                     wkMinTot = 0
97                 Else
98                     wkHoldDept = wkDeptNo
99                 End If
100             End If
101             lstMinor.Items.Add(wkDeptNo & " " & FormatCurrency(wkAmt))
102             wkMinTot = wkMinTot + wkAmt
103             wkFinTot = wkFinTot + wkAmt
104         Else
105             MsgBox("End of File", vbOKOnly, "EOF")
106             lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
107             lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
108             FileClose(1)
109         End If
110     End Sub
111
112     Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
113     End
114     End Sub
115 End Class
116

```

**Minor - Notepad**

```

"12", "500"
"12", "100"
"12", "050"
"15", "200"
"15", "100"
"15", "300"
"15", "060"
"17", "020"
"17", "240"

```

**Minor Totals**

```

12 $500.00
12 $100.00
12 $50.00
Dept Total: $650.00
15 $200.00
15 $100.00
15 $300.00
15 $60.00
Dept Total: $660.00
17 $20.00
17 $240.00
Dept Total: $260.00
Final Total: $1,570.00

```

**Handwritten Annotations:**

- `Write total` (with arrow pointing to `btnProcess`)
- `broke` (near `wkHoldDept` logic)
- `move to hold reset tot to 0 minor` (near `Input` statements)

**Diagram:**

Diagram illustrating the relationship between `Wk Hold Dept` and `Wk Dept No` values:

```

Wk Hold Dept: 12 ← Wk Dept No: 12
                15 ← Wk Dept No: 15

```

Additional notes: `Write total` is written near the `15` values.

**Code Snippets:**

```

1 reference
2 Public Class frmMinor
3     Inherits System.Windows.Forms.Form
4     Dim wkMinTot As Integer
5     Dim wkFinTot As Integer
6     Dim wkHoldDept As String
7
81 Windows Form Designer generated code
82
83 Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
84     FileOpen(1, "Minor.txt", OpenMode.Input)
85     wkHoldDept = ""
86 End Sub
87
88 Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
89     Dim wkDeptNo As String, wkAmt As String
90     If Not EOF(1) Then
91         Input(1, wkDeptNo)
92         Input(1, wkAmt)
93         If wkHoldDept <> wkDeptNo Then
94             If wkHoldDept <> "" Then
95                 lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
96                 wkHoldDept = wkDeptNo
97                 wkMinTot = 0
98             Else
99                 wkHoldDept = wkDeptNo
100            End If
101            lstMinor.Items.Add(wkDeptNo & " " & FormatCurrency(wkAmt))
102            wkMinTot = wkMinTot + wkAmt
103            wkFinTot = wkFinTot + wkAmt
104        Else
105            MsgBox("End of File", vbOKOnly, "EOF")
106            lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
107            lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
108            FileClose(1)
109        End If
110    End Sub
111
112 Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
113     End
114 End Sub
115 End Class
116

```

**Minor Totals Window:**

```

12 $500.00
12 $100.00
12 $50.00
Dept Total: $650.00
15 $200.00
15 $100.00
15 $300.00
15 $60.00
Dept Total: $660.00
17 $20.00
17 $240.00
Dept Total: $260.00
Final Total: $1,570.00

```

**Minor - Notepad:**

```

File Edit Format View Help
"12", "500"
"12", "100"
"12", "050"
"15", "200"
"15", "100"
"15", "300"
"15", "060"
"17", "020"
"17", "240"

```

**Handwritten Notes:**

- broke* (blue)
- move to hold* (blue)
- reset tot to 0* (blue)
- process* (red)
- Wk Hold Dept* (blue)
- Write total* (blue)
- the record that caused break or next record* (red)



The image shows a Visual Studio IDE with a VB.NET project named 'PrMinor'. The code editor displays the 'frmMinor' class with handwritten annotations in green and red ink. The annotations include:

- Wk MinTot**: 500, ~~600~~, ~~650~~, 200
- Wk FinTot**: 500, ~~600~~, ~~650~~, 850
- Wk Hold Dept**: 12, 15

The code in the editor includes:

```

1 Public Class frmMinor
2     Inherits System.Windows.Forms.Form
3     Dim wkMinTot As Integer
4     Dim wkFinTot As Integer
5     Dim wkHoldDept As String
6
7     'Windows Form Designer generated code
81
82     Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
83         FileOpen(1, "Minor.txt", OpenMode.Input)
84         wkHoldDept = ""
85     End Sub
86
87     Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
88         Dim wkDeptNo As String, wkAmt As String
89         If Not EOF(1) Then
90             Input(1, wkDeptNo)
91             Input(1, wkAmt)
92             If wkHoldDept <> wkDeptNo Then
93                 If wkHoldDept <> "" Then
94                     lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
95                     wkHoldDept = wkDeptNo
96                     wkMinTot = 0
97                 Else
98                     wkHoldDept = wkDeptNo
99                 End If
100             End If
101             lstMinor.Items.Add(wkDeptNo & " " & FormatCurrency(wkAmt))
102             wkMinTot = wkMinTot + wkAmt
103             wkFinTot = wkFinTot + wkAmt
104         Else
105             MsgBox("End of File", vbOKOnly, "EOF")
106             lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
107             lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
108             FileClose(1)
109         End If
110     End Sub
111
112     Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
113     End
114 End Sub
115 End Class
116

```

Overlaid on the code editor is a 'Minor Totals' dialog box with the following content:

```

12 $500.00
12 $100.00
12 $50.00
Dept Total: $650.00
15 $200.00

```

Buttons: Process, Exit

Below the dialog box is a 'Minor - Notepad' window containing the following text:

```

"12", "500"
"12", "100"
"12", "050"
"15", "200"
"15", "100"
"15", "300"
"15", "060"
"17", "020"
"17", "240"

```

The Windows taskbar at the bottom shows the system tray with the date and time: 1:47 PM, 2/18/2020.

Visual Studio screenshot showing the development of a VB.NET application named PrMinor. The code in frmMinor.vb [Design] includes:

```

1 reference
2 Public Class frmMinor
3     Inherits System.Windows.Forms.Form
4     Dim wkMinTot As Integer
5     Dim wkFinTot As Integer
6     Dim wkHoldDept As String
7
81 Windows Form Designer generated code
82
83 Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
84     FileOpen(1, "Minor.txt", OpenMode.Input)
85     wkHoldDept = ""
86 End Sub
87
88 Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
89     Dim wkDeptNo As String, wkAmt As String
90     If Not EOF(1) Then
91         Input(1, wkDeptNo)
92         Input(1, wkAmt)
93         If wkHoldDept <> wkDeptNo Then
94             If wkHoldDept <> "" Then
95                 lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
96                 wkHoldDept = wkDeptNo
97                 wkMinTot = 0
98             Else
99                 wkHoldDept = wkDeptNo
100             End If
101             End If
102             lstMinor.Items.Add(wkDeptNo & " " & FormatCurrency(wkAmt))
103             wkMinTot = wkMinTot + wkAmt
104             wkFinTot = wkFinTot + wkAmt
105         Else
106             MsgBox("End of File", vbOKOnly, "EOF")
107             lstMinor.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
108             lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
109             FileClose(1)
110         End If
111     End Sub
112
113 Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
114     End
115 End Sub
116 End Class

```

Handwritten annotations in green ink on the code include:

- $wkMinTot$  and  $wkFinTot$  with values 500, 600, 650, and 850.
- $wkHoldDept$  with values 200 and 250.

A small dialog box titled "EOF" with the message "End of File" and an "OK" button is visible over the code.

The "Minor Totals" window displays a list of entries and their totals:

12	\$500.00
12	\$100.00
12	\$50.00
Dept Total: \$650.00	
15	\$200.00
15	\$100.00
15	\$300.00
15	\$60.00
Dept Total: \$660.00	
17	\$20.00
17	\$240.00

The "Minor - Notepad" window shows the raw data entries:

```

"12", "500"
"12", "100"
"12", "050"
"15", "200"
"15", "100"
"15", "300"
"15", "060"
"17", "020"
"17", "240"

```

The Windows taskbar at the bottom shows the system tray with the date 2/18/2020 and time 1:47 PM.

