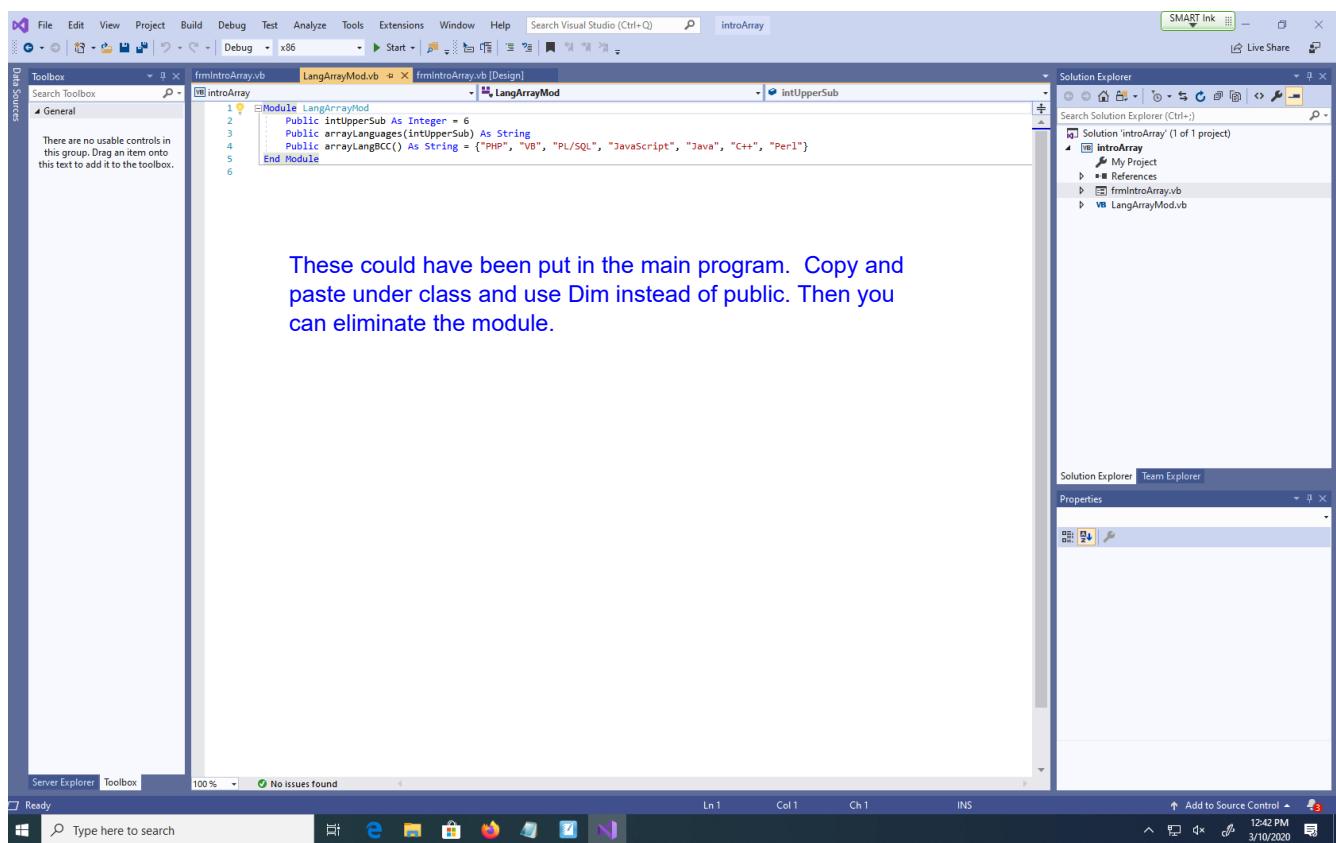


**Examples using VB 2010**

| Examples                                    | Click on links to retrieve   |
|---|--|
| Payroll Calculation (intro)                 | <a href="#">PayCalc.zip</a>  |
| Basic Project (chap #2)                     | <a href="#">cisBasicProj.zip</a>   |
| Variables and input (chap #3)               | <a href="#">infoCalcProj.zip</a><br><a href="#">varCalcProj.zip</a><br><a href="#">introExceptionProj.zip</a><br><a href="#">useGroupProg.zip</a><br><a href="#">projMath.zip</a>  |
| Decisions (chap #4)                         | <a href="#">projIntro1F.zip</a><br><a href="#">otherDecisions.zip</a><br><a href="#">projRadioCheck.zip</a><br><a href="#">inputVal.zip</a>  |
| Lists and Loops (chap #5)                   | <a href="#">useBoxes.zip</a><br><a href="#">basicLoops.zip</a><br><a href="#">diffBoxes.zip</a><br><a href="#">GuessGames2010.zip</a>  |
| Procedures and functions (chap #6)          | <a href="#">firstProcedures.zip</a><br><a href="#">firstFunctions.zip</a>  |
| VB functions                                | <a href="#">VBfunctions.zip</a>  |
| Multiple forms, modules and menus (chap #7) | <a href="#">basicMultiForms.zip</a><br><a href="#">menuGrades.zip</a>  |
| Arrays (chap #8)                            | <a href="#">introArray.zip</a><br><a href="#">calcArray.zip</a><br><a href="#">moreArrays.zip</a>  |
| Files (chap #9)                             | <a href="#">ReadVB2010.zip</a><br><a href="#">writeVB2010.zip</a>  |
| Database examples using SQLServer           | Basic<br>VBGrid<br>bindonclick<br>DonorFormSQL<br>DonorTwoTables<br>TwoGrids<br>combinewithSQL<br>ConvertedtoSQLServer contains all of the examples above  |
| ADO using Access 2010 - writing code        | DonorFormSQL12.zip OR DonorFormSQL12.zip<br>bookADO.zip OR bookADO.zip<br>ADOCurr12.zip OR ADOcurr12.zip<br>VBGrid12.zip OR VBGGrid12.zip<br>bindonclick12.zip<br>combinewithSQL.zip<br>SQLdiffcode12.zip<br>Donorthowtables12.zip |

Type here to search 12:38 PM 3/10/2020

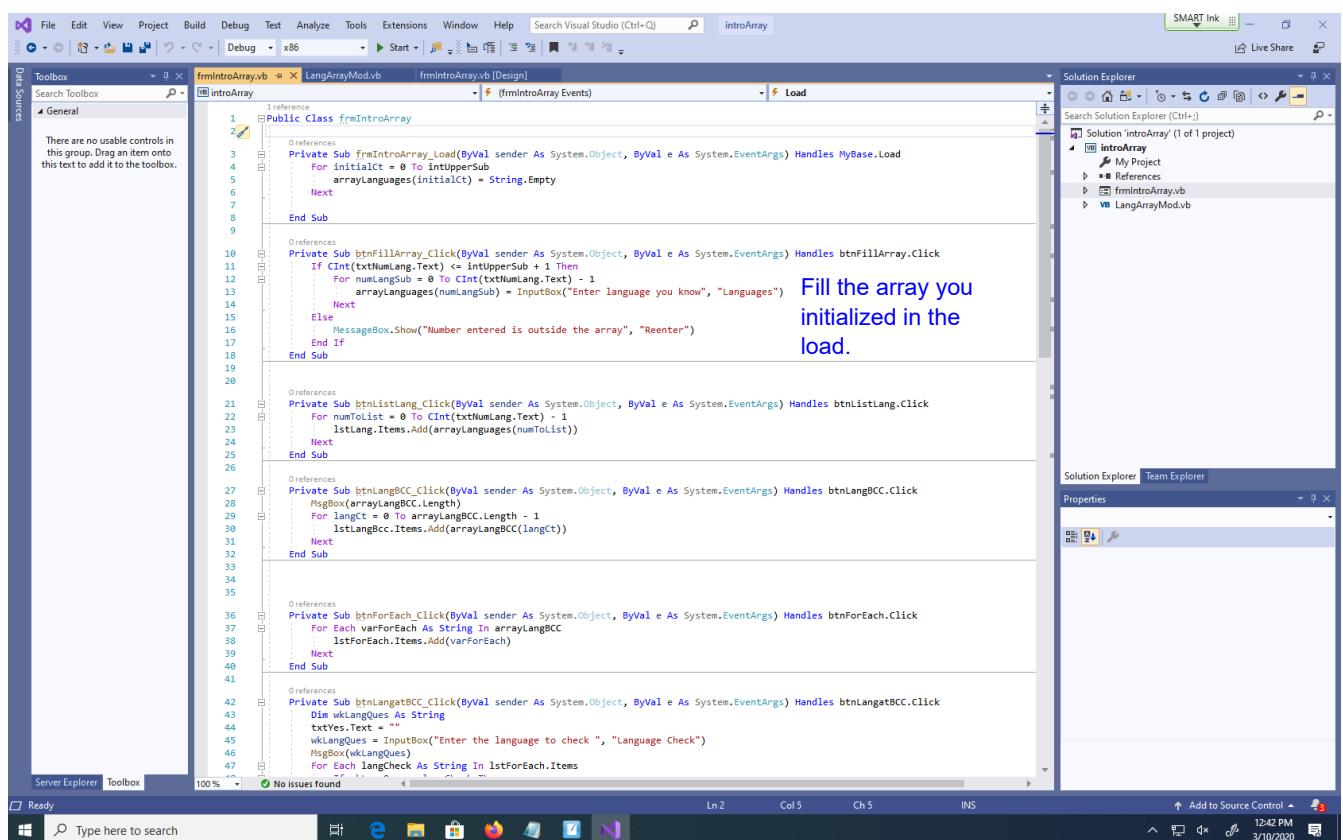


The screenshot shows the Microsoft Visual Studio IDE interface. The code editor displays the following VB.NET code:

```
Module LangArrayMod
    Public intUpperSub As Integer = 6
    Public arrayLanguage(intUpperSub) As String
    Public arrayLangBCC() As String = {"PHP", "VB", "PL/SQL", "JavaScript", "Java", "C++", "Perl"}
End Module
```

The Solution Explorer window on the right shows a single project named 'introArray' with files 'frmintroArray.vb', 'LangArrayMod.vb', and 'LangArrayMod.bak'. The Properties window is also visible at the bottom.

These could have been put in the main program. Copy and paste under class and use Dim instead of public. Then you can eliminate the module.

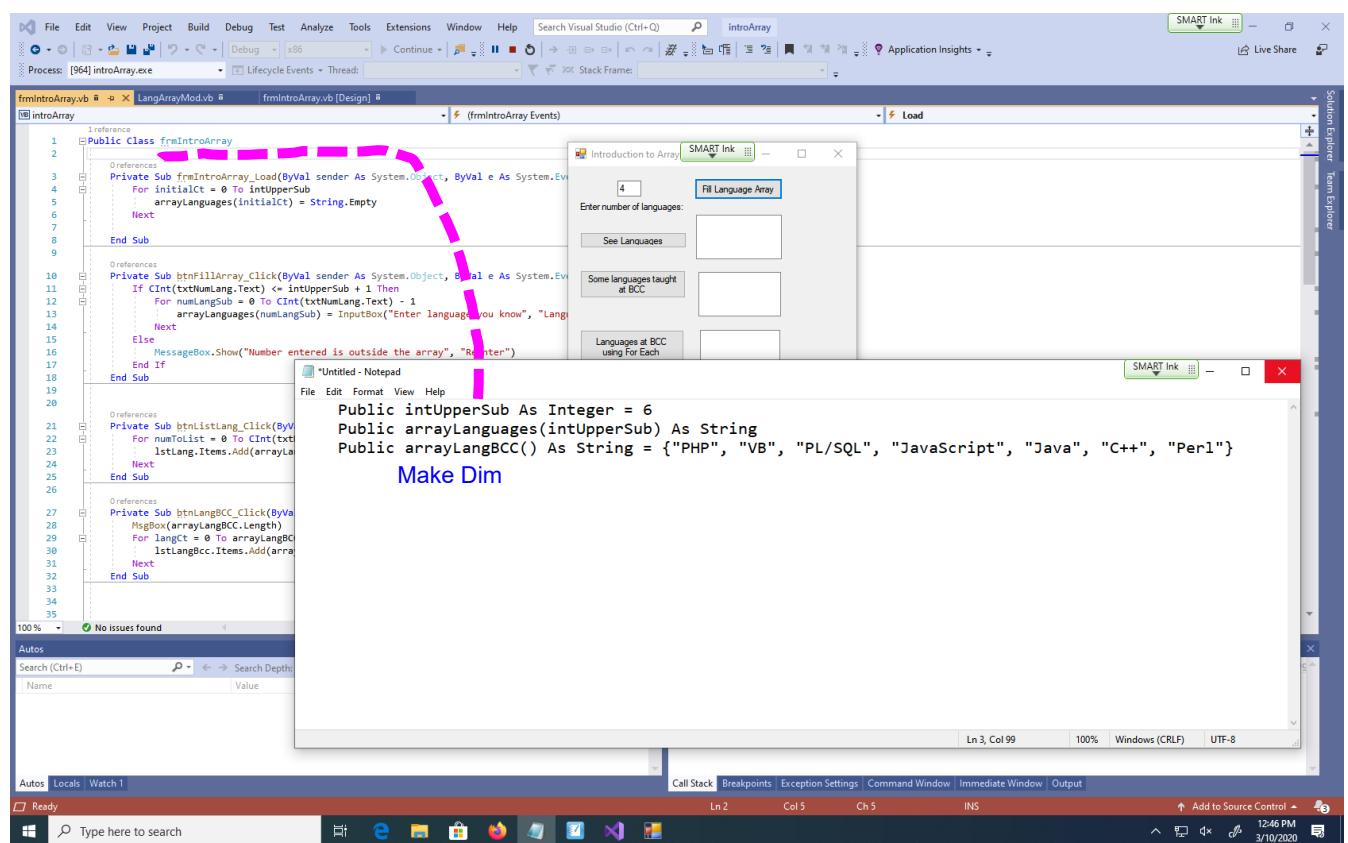


The screenshot shows the Microsoft Visual Studio IDE interface. The title bar reads "c156Mar1020.notebook" and "March 12, 2020". The main window displays the code for the "frmIntroArray" form. A blue annotation box highlights the following code in the "Load" event handler:

```
Private Sub frmIntroArray_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    For intUpperSub = 0 To intUpperSub
        arrayLanguages(intUpperSub) = String.Empty
    Next
End Sub
```

The code continues with other event handlers like "btnFillArray\_Click", "btnListLang\_Click", "btnLangBCC\_Click", "btnForEach\_Click", and "btnLangatBCC\_Click". The Solution Explorer on the right shows the project structure with files "introArray", "frmIntroArray.vb", and "LangArrayMod.vb". The status bar at the bottom shows "Ready", "Type here to search", and system information.

Fill the array you  
initialized in the  
load.



The screenshot shows the Microsoft Visual Studio IDE. The main window displays the code for `frmIntroArray.vb`. A modal dialog box titled "Language Check" is open, prompting the user to "Enter the language to check". In the background, a "SMART Ink" interface is visible, showing a list of languages: VB, PHP, Perl, PL/SQL, C++, VB, PL/SQL, JavaScript, and Java. The SMART Ink interface also includes sections for "Some languages taught at BCC", "Languages at BCC using For Each", and a search bar.

```

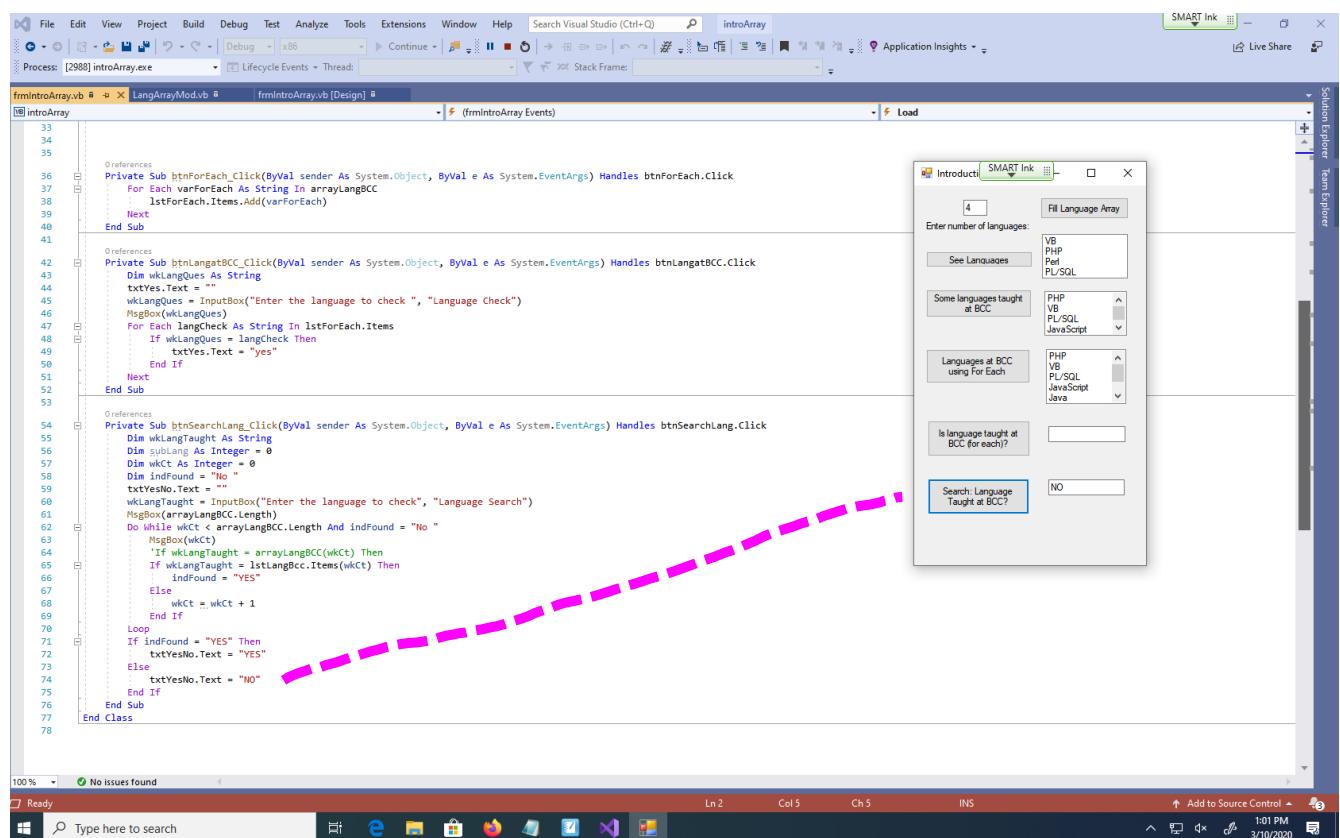
33
34
35
36  Private Sub btnForEach_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnForEach.Click
37      For Each varForEach As String In arrayLangBCC
38          lstForEach.Items.Add(varForEach)
39      Next
40  End Sub
41
42  Private Sub btnLangatBCC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLangatBCC.Click
43      Dim wkLangQues As String
44      wkLangQues = InputBox("Enter the language to check ", "Language Check")
45      MsgBox(wkLangQues)
46      For Each langCheck As String In lstForEach.Items
47          If wkLangQues = langCheck Then
48              txtYes.Text = "yes"
49          Else
50              txtYes.Text = "no"
51          End If
52      Next
53  End Sub
54
55  Private Sub btnSearchLang_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSearchLang.Click
56      Dim wkLangTaught As String
57      Dim subLang As Integer = 0
58      Dim wkCt As Integer = 0
59      Dim indFound = "No "
60      txtYesNo.Text = ""
61      wkLangTaught = InputBox("Enter the language to check", "Language Search")
62      Do While subLang < arrayLangBCC.Length And indFound = "No "
63          subLang = subLang + 1
64          If wkLangTaught = arrayLangBCC(wkCt).Length Then
65              If wkLangTaught = lstLangBcc.Items(wkCt) Then
66                  indFound = "YES"
67              Else
68                  wkCt = wkCt + 1
69              End If
70          Loop
71          If indFound = "YES" Then
72              txtYesNo.Text = "YES"
73          Else
74              txtYesNo.Text = "NO"
75          End If
76      End Sub
77  End Class
78

```

The screenshot shows the Microsoft Visual Studio IDE with the project 'frmIntroArray' open. The code editor displays the file 'frmIntroArray.vb' with several subroutines:

```
33
34
35
36 Private Sub btnForEach_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnForEach.Click
37     For Each varForEach As String In arrayLangBCC
38         lstForEach.Items.Add(varForEach)
39     Next
40 End Sub
41
42 Preferences
43 Private Sub btnLangatBCC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLangatBCC.Click
44     Dim wkLangQues As String
45     txtLangQues.Text = "Enter the language to check"
46     wkLangQues = InputBox("Enter the language to check", "Language Check")
47     For Each langCheck As String In lstForEach.Items
48         If wkLangQues = langCheck Then
49             txtYes.Text = "yes"
50         End If
51     Next
52 End Sub
53
54 Preferences
55 Private Sub btnSearchLang_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSearchLang.Click
56     Dim wkLangTaught As String
57     Dim subLang As Integer = 0
58     Dim wkCt As Integer = 0
59     Dim indFound = "No"
60     txtYesNo.Text = ""
61     wkLangTaught = InputBox("Enter the language to check", "Language Search")
62     Do While wkCt < arrayLangBCC.Length And indFound = "No"
63         msgLangSub(subLang)
64         If wkLangTaught = arrayLangBCC(wkCt) Then
65             If wkLangTaught = lstLangBcc.Items(wkCt) Then
66                 indFound = "YES"
67             Else
68                 wkCt = wkCt + 1
69             End If
70         Loop
71         If indFound = "YES" Then
72             txtYesNo.Text = "YES"
73         Else
74             txtYesNo.Text = "NO"
75         End If
76     End Sub
77 End Class
78
```

A SMART Ink annotation is overlaid on the code editor. It includes a form with fields for entering the number of languages (4), selecting languages taught at BCC (PHP, VB, PL/SQL, JavaScript), and searching for languages taught at BCC (VB, PHP, PL/SQL, JavaScript). A question box asks if the language is taught at BCC for each, with a yes/no input field set to yes. A search button is also present.



The screenshot shows the Microsoft Visual Studio IDE with the following details:

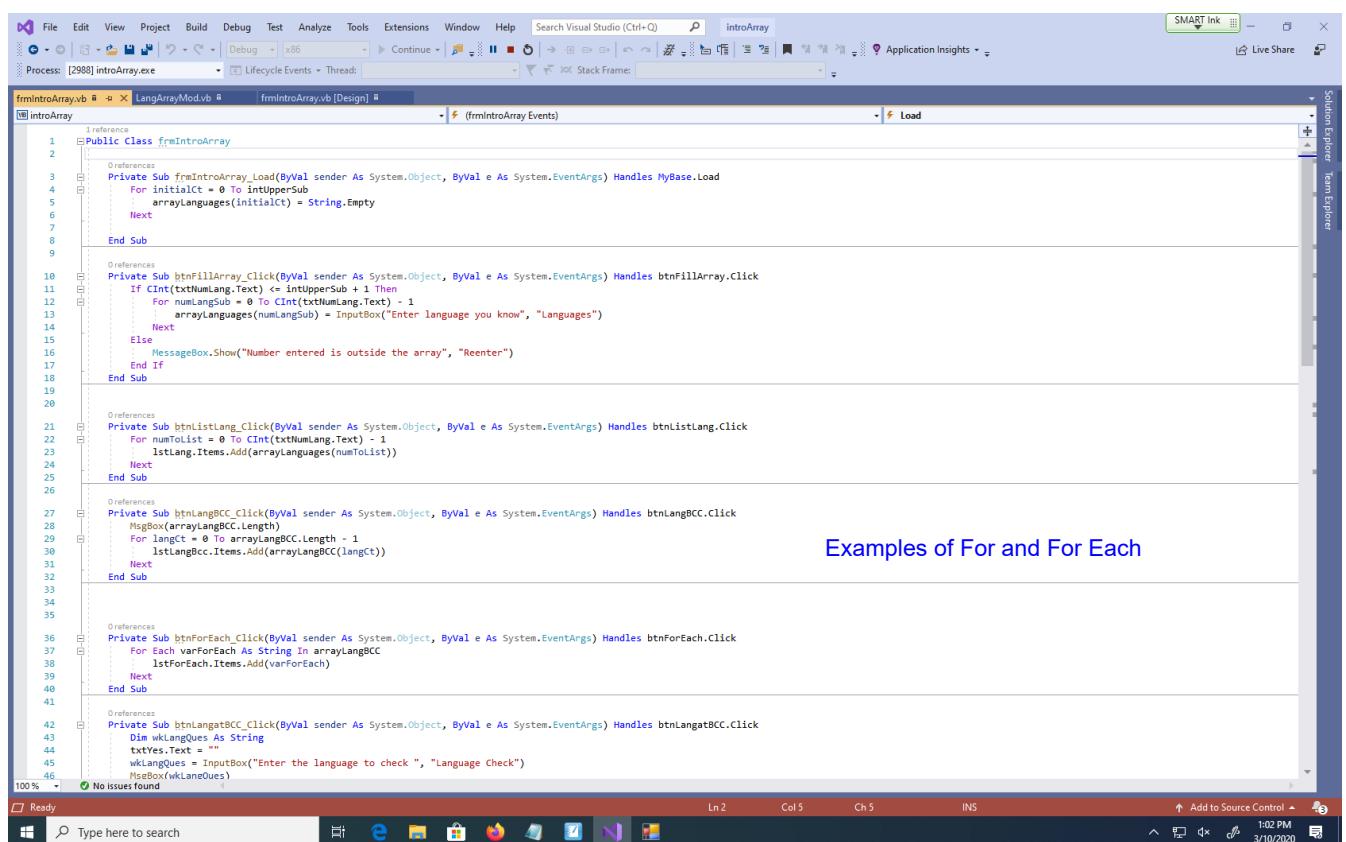
- File Menu:** File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help.
- Search Bar:** Search Visual Studio (Ctrl+Q).
- Toolbars:** Standard, Debug, Windows, Tools, Help.
- Project Explorer:** Processes [2980] introArray.exe, Lifecycle Events - Thread: Stack Frame:.
- Code Editor:** frmIntroArray.vb (Design) window. The code is as follows:

```

33
34
35
36    Private Sub btnForEach_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnForEach.Click
37        For Each varForEach As String In arrayLangBCC
38            lstForEach.Items.Add(varForEach)
39        Next
40    End Sub
41
42    Private Sub btnLangatBCC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLangatBCC.Click
43        Dim wkLangQues As String
44        txtYes.Text = ""
45        wkLangQues = InputBox("Enter the language to check", "Language Check")
46        For Each langCheck As String In lstForEach.Items
47            If wkLangQues = langCheck Then
48                txtYes.Text = "yes"
49            End If
50        Next
51    End Sub
52
53    Private Sub btnSearchLang_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSearchLang.Click
54        Dim wkLangTaught As String
55        Dim subLang As Integer = 0
56        Dim wkCt As Integer = 0
57        Dim indFound = "No"
58        txtYesNo.Text = ""
59        wkLangTaught = InputBox("Enter the language to check", "Language Search")
60        MsgBox(arrayLangBCC.Length)
61        Do While wkCt < arrayLangBCC.Length And indFound = "No"
62            If wkLangTaught = arrayLangBCC(wkCt) Then
63                If wkLangTaught = lstLangBCC.Items(wkCt) Then
64                    indFound = "YES"
65                Else
66                    wkCt = wkCt + 1
67                End If
68            Loop
69            If indFound = "YES" Then
70                txtYesNo.Text = "YES"
71            Else
72                txtYesNo.Text = "NO"
73            End If
74        End Sub
75    End Class
76
77

```

- Toolbox:** SMART Ink, Application Insights, Live Share.
- Annotations:** A pink dashed arrow points from the line 'If wkLangTaught = arrayLangBCC(wkCt) Then' in the code editor to the 'Some languages taught at BCC' dropdown in the SMART Ink overlay.
- SMART Ink Overlay:** Shows the following interface:
  - Enter number of languages:  Fill Language Array
  - See Languages: VB, PHP, Perl, PL/SQL
  - Some languages taught at BCC: PHP, VB, PL/SQL, JavaScript
  - Languages at BCC using For Each: PHP, VB, PL/SQL, JavaScript, Java
  - Is language taught at BCC for each?:
  - Search Language Taught at BCC?
  - Result: NO
- Status Bar:** 100%, No issues found, Ready, Type here to search, Ln 2 Col 5 Ch 5 INS, Add to Source Control, 1:01 PM, 3/10/2020.



```
1 reference
2
3  Public Class frmIntroArray
4
5    References
6      Private Sub frmIntroArray_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
7        For initialCt = 0 To intUpperSub
8          arrayLanguages(initialCt) = String.Empty
9        Next
10       End Sub
11
12      References
13      Private Sub btnFillArray_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnFillArray.Click
14        If CInt(txtNumLang.Text) <= intUpperSub + 1 Then
15          For numLangSub = 0 To CInt(txtNumLang.Text) - 1
16            arrayLanguages(numLangSub) = InputBox("Enter language you know", "Languages")
17          Next
18        Else
19          MessageBox.Show("Number entered is outside the array", "Reenter")
20        End If
21      End Sub
22
23      References
24      Private Sub btnListLang_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnListLang.Click
25        For numToList = 0 To CInt(txtNumLang.Text) - 1
26          lstLang.Items.Add(arrayLanguages(numToList))
27        Next
28      End Sub
29
30      References
31      Private Sub btnLangBCC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLangBCC.Click
32        MsgBox(arrayLangBCC.Length)
33        For langCt = 0 To arrayLangBCC.Length - 1
34          lstLangBCC.Items.Add(arrayLangBCC(langCt))
35        Next
36      End Sub
37
38      References
39      Private Sub btnForEach_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnForEach.Click
40        For Each varForEach As String In arrayLangBCC
41          lstForEach.Items.Add(varForEach)
42        Next
43      End Sub
44
45      References
46      Private Sub btnLangatBCC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLangatBCC.Click
47        Dim wkLangQues As String
48        txtYes.Text = ""
49        wkLangQues = InputBox("Enter the language to check ", "Language Check")
50        MsgBox(wkLangQues)
```

### Examples of For and For Each

The screenshot shows the Microsoft Visual Studio IDE interface. The title bar reads "File Edit View Project Build Debug Test Analyze Tools Window Help Search Visual Studio (Ctrl+Q) introArray". The menu bar includes "Process: [2908] introArray.exe" and "Lifecycle Events Thread". The main code editor window displays VB.NET code for the "frmIntroArray" form. The code handles a button click event ("btnSearchLang\_Click") which searches for a language in a list and updates a text box ("txtYesNo"). The code uses loops and conditional statements to find the language and determine if it is present in the list.

```
47     For Each langCheck As String In lstLangForEach.Items
48         If wkLangQues = langCheck Then
49             txtYes.Text = "yes"
50         End If
51     Next
52 
```

```
53 End Sub
```

```
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 
```

```
frmintroArray.vb [Design]  LangArrayMod.vb  frmintroArray.vb [Design]  introArray
Process: [2988] introArray.exe  Lifecycle Events  Thread:  Stack Frame:  SMART Link  Live Share  Application Insights

frmintroArray.vb [Design]
introArray
47     For Each langCheck As String In lstForEach.Items
48         If wkLangQues = langCheck Then
49             txtYes.Text = "yes"
50         End If
51     Next
52 End Sub
53
54 References:
55 Private Sub btnSearchLang_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSearchLang.Click
56     Dim wkLangTaught As String
57     Dim subLang As Integer = 0
58     Dim wkCt As Integer = 0
59     Dim indFound = "No "
60
61     txtYesNo.Text =
62     wkLangTaught = InputBox("Enter the language to check", "Language Search")
63     MsgBox(arrayLangBCC.Length)
64     Do While wkCt < arrayLangBCC.Length And indFound = "No "
65         MsgBox(wkCt)
66         'If wkLangTaught = arrayLangBCC(wkCt) Then
67         If wkLangTaught = lstLangBCC.Items(wkCt) Then
68             indFound = "YES"
69         Else
70             wkCt = wkCt + 1
71         End If
72     Loop
73     If indFound = "YES" Then
74         txtYesNo.Text = "YES"
75     Else
76         txtYesNo.Text = "NO"
77     End If
78 End Sub
79 End Class
```

The screenshot shows the Microsoft Visual Studio IDE with the following details:

- Title Bar:** SMART Ink, File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help.
- Toolbar:** Standard icons for file operations like Open, Save, Print, etc.
- Status Bar:** Process: [1084] calcArray.exe, Lifecycle Events, Stack Frame.
- Code Editor:** The file is frmCalcArray.vb, and the code is as follows:

```

1  Public Class frmCalcArray
2      Dim numArray(5) As Single
3      Dim wkTotal As Double = 0
4
5      Private Sub frmCalcArray_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
6          numArray(0) = 12.34
7          numArray(1) = 23.45
8          numArray(2) = 34.56
9          numArray(3) = 45.67
10         numArray(4) = 56.78
11         numArray(5) = 67.89
12     End Sub
13
14     Private Sub btnShowArray_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnShowArray.Click
15         For Each showNum As Single In numArray
16             lstShowArray.Items.Add(showNum)
17         Next
18     End Sub
19
20     Private Sub btnTotal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnTotal.Click
21         For Each addnum As Single In numArray
22             wkTotal = wkTotal + addnum
23         Next
24         txtTotal.Text = Format(wkTotal, "n2")
25     End Sub
26
27     Private Sub btnAvg_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAvg.Click
28         For Each num As Single In numArray
29             wkTotal = wkTotal + num
30         Next
31         'MsgBox(numArray.Length)
32         txtAvg.Text = Format(wkTotal / numArray.Length, "n2")
33     End Sub
34
35     Private Sub btnHigh_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnHigh.Click
36         Dim wkHigh As Single = 0
37         For forct = 0 To numArray.Length - 1
38             If numArray(forct) > wkHigh Then
39                 wkHigh = numArray(forct)
40             End If
41         Next
42         txtHigh.Text = wkHigh
43     End Sub
44
45     Private Sub btnLow_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLow.Click
46         Dim wkLow As Single
47         wkLow = numArray(0)
    
```

- Calculate Window:** A modal window titled 'Calculate' containing the following data:

| Calculate    |   |
|--------------|---|
| Show Array   | 12.34<br>34.56<br>45.67<br>56.78<br>67.89 |
| Show Total   | 240.69                                    |
| Show Average | 48.14                                     |
| Highest      | 67.89                                     |
| Lowest       | 12.34                                     |
- Toolbars and Status Bar:** Standard Windows-style toolbars. Status bar shows Ln 21, Col 9, Ch 9, INS, and system information (1:12 PM, 3/10/2020).

The screenshot shows the Microsoft Visual Studio IDE interface. The main window displays the code for the frmMoreArrays.vb form. The code uses two arrays: numArray and wordArray, and a sorted array tosortArray. It handles the Load event of the form and the Click events of two buttons: btnShowArray and btnSearch.

```
3  Dim numArray(crsCt) As Integer
4  Dim wordArray(crsCt) As String
5  Dim toSortArray(7) As String
6
7  Private Sub frmMoreArrays_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
8      crsCt = 4
9      numArray(0) = 120
10     numArray(1) = 121
11     numArray(2) = 122
12     numArray(3) = 156
13     numArray(4) = 159
14     wordArray(0) = "Programming Logic"
15     wordArray(1) = "Operating Systems"
16     wordArray(2) = "Internet Developer"
17     wordArray(3) = "Visual Basic"
18     wordArray(4) = "PHP and MySQL"
19     toSortArray(0) = "First"
20     toSortArray(1) = "Second"
21     toSortArray(2) = "Third"
22     toSortArray(3) = "Fourth"
23     toSortArray(4) = "Fifth"
24     toSortArray(5) = "Sixth"
25     toSortArray(6) = "Seventh"
26     toSortArray(7) = "Eighth"
27
28  End Sub
29
30  Private Sub btnShowArray_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnShowArray.Click
31      Dim wkInfo As String
32      For ct = 0 To 4
33          wkInfo = numArray(ct) & " " & wordArray(ct)
34          lstShowArray.Items.Add(wkInfo)
35      Next
36  End Sub
37
38  Private Sub btnSearch_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSearch.Click
39      Dim wkHunIn As Integer
40      Dim wkSub As Integer = 0
41      Dim wkMatchInd As String = "NO"
42
```

The screenshot shows the Microsoft Visual Studio IDE with the project 'moreArrays' open. The code editor displays the file 'frmMoreArrays.vb' in Design mode. A context menu is open over the code, specifically over the 'Sort' button in the 'Sort' dialog box. The menu has the following options:

- Show Array
- Search
- Visual Basic
- Add Course
- Sort

The 'Sort' option is highlighted with a pink arrow. The 'Sort' dialog box is visible, showing the following list of items:

- Eighth
- Fifth
- First
- Fourth
- Second
- Seventh
- Sixth
- Third

The code in 'frmMoreArrays.vb' is as follows:

```
43     If wkNumIn = numArray(wkSub) Then
44         : wkMatchInd = "YES"
45     Else
46         wkSub = wkSub + 1
47     End If
48 Loop
49 If wkMatchInd = "YES" Then
50     txtWord.Text = wordArray(wkSub)
51 Else
52     txtWord.Text = "Did not find match"
53 End If
54 End Sub

55
56 Private Sub btnSort_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSort.Click
57     'Sort(toSortArray)
58     For ct = 0 To 7
59         lstSortResults.Items.Add(toSortArray(ct))
60     Next
61 End Sub

62
63 Private Sub btnAddCourse_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddCourse.Click
64     Dim inputNum As Integer
65     Dim inputWord As String
66     Dim wkInfo As String
67     Dim crsCt As Integer
68     crsCt = crsCt + 1
69     ReDim Preserve numArray(crsCt)
70     ReDim Preserve wordArray(crsCt)
71     inputNum = InputBox("Enter the course number", "Course Number")
72     inputWord = InputBox("Enter the course name", "Course Name")
73     numArray(crsCt) = inputNum
74     wordArray(crsCt) = inputWord
75     lstWithAdd.Items.Clear()
76     For ct = 0 To crsCt
77         wkInfo = numArray(ct) & " " & wordArray(ct)
78         lstWithAdd.Items.Add(wkInfo)
79     Next
80 End Sub
81 End Class
```

```

File Edit View Project Build Debug Test Analyze Tools Window Help Search Visual Studio (Ctrl+Q) firstProcedures
Process: [12856] firstProcedures.exe Lifecycle Events Threads Stack Frame Application Insights
firstProc.vb [Design] firstProc
26
    References
27     Private Sub btnPassByRef_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPassByRef.Click
28         Dim wkAmt As Single
29         Dim wkPer As Single
30         Dim wkMsg As String
31         wkAmt = InputBox("Enter the amount of the transaction", "Transaction Amount")
32         wkPer = InputBox("Enter the percent you are using", "Percent")
33         CalcByRef(wkAmt, wkPer)
34         wkMsg = "Back in btnPassByRef from CalcByRef " & wkAmt
35         lstTrans.Items.Add(wkMsg)
36         OtherCalcByRef(wkAmt, wkPer)
37         wkMsg = "Back in btnPassByRef from OtherCalcByRef " & wkAmt
38         lstTrans.Items.Add(wkMsg)
39     End Sub
40
41     References
42     Sub CalcByRef(ByRef sentAmt As Single, ByRef sentPer As Single)
43         sentAmt = sentAmt * sentPer
44         lstTrans.Items.Add("The result from CalcByRef is " & sentAmt)
45     End Sub
46
47     References
48     Sub OtherCalcByRef(ByRef wkAmt As Single, ByVal wkPer As Single)
49         wkAmt = wkAmt * wkPer
50         lstTrans.Items.Add("The result from CalcByRef is " & wkAmt)
51     End Sub
52
53     References
54     Private Sub btnPassByVal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPassByVal.Click
55         Dim wkAmt As Single
56         Dim wkPer As Single
57         Dim wkMsg As String
58         wkAmt = InputBox("Enter the amount of the transaction", "Transaction Amount")
59         wkPer = InputBox("Enter the percent you are using", "Percent")
60         CalcByVal(wkAmt, wkPer)
61         wkMsg = "Back in btnPassByVal from CalcByVal " & wkAmt
62         lstTransVal.Items.Add(wkMsg)
63         lstTransVal.Items.Add(OtherCalcByVal(wkAmt, wkPer))
64         wkMsg = "Back in btnPassByVal from OtherCalcByVal " & wkAmt
65         lstTransVal.Items.Add(wkMsg)
66     End Sub
67
68     References
69     Sub CalcByVal(ByVal sentAmt As Single, ByVal sentPer As Single)
70         sentAmt = sentAmt * sentPer
71         lstTransVal.Items.Add("The result from CalcByVal is " & sentAmt)
72     End Sub
73     Sub OtherCalcByVal(ByVal wkAmt As Single, ByVal wkPer As Single)
74         wkAmt = wkAmt * wkPer
75         lstTransVal.Items.Add("The result from CalcByVal is " & wkAmt)
76     End Sub
77 End Class

```

No issues found.

Ready Type here to search

Ln 14 Col 50 Ch 50 INS

1:27 PM 3/10/2020

```

100% No issues found
    Ready
    Type here to search
    File Edit View Project Build Debug Test Analyze Tools Window Help
    Process: [12856] firstProcedures.exe
    SMART Link
    Live Share
    Application Insights
    firstProc.vb [Design]
    firstProcedures
    firstProc
    References
    Private Sub btnPassByRef_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPassByRef.Click
        Dim wkAmt As Single
        Dim wkPer As Single
        Dim wkMsg As String
        wkAmt = InputBox("Enter the amount of the transaction", "Transaction Amount")
        wkPer = InputBox("Enter the percent you are using", "Percent")
        CalcByRef(wkAmt, wkPer)
        wkMsg = "Back in btnPassByRef from CalcByRef " & wkAmt
        lstTrans.Items.Add(wkMsg)
        OtherCalcByRef(wkAmt, wkPer)
        wkMsg = "Back in btnPassByRef from OtherCalcByRef " & wkAmt
        lstTrans.Items.Add(wkMsg)
    End Sub
    References
    Sub OtherCalcByRef(ByVal sentAmt As Single, ByRef sentPer As Single)
        sentAmt = sentAmt * sentPer
        lstTrans.Items.Add("The result from CalcByRef is " & sentAmt)
    End Sub
    References
    Sub CalcByRef(ByRef wkAmt As Single, ByVal wkPer As Single)
        Dim wkamt As Single
        Dim wkPer As Single
        Dim wkMsg As String
        wkamt = InputBox("Enter the amount of the transaction", "Transaction Amount")
        wkPer = InputBox("Enter the percent you are using", "Percent")
        CalcByVal(wkamt, wkPer)
        wkMsg = "Back in btnPassByRef from CalcByVal " & wkAmt
        lstTrans.Items.Add(wkMsg)
        OtherCalcByVal(wkamt, wkPer)
        wkMsg = "Back in btnPassByRef from OtherCalcByVal " & wkAmt
        lstTrans.Items.Add(wkMsg)
    End Sub
    References
    Sub CalcByVal(ByVal sentAmt As Single, ByVal sentPer As Single)
        sentAmt = sentAmt * sentPer
        lstTransVal.Items.Add("The result from CalcByVal is " & sentAmt)
    End Sub
    References
    Sub OtherCalcByVal(ByVal wkAmt As Single, ByVal wkPer As Single)
        sentAmt = wkAmt * wkPer
        lstTransVal.Items.Add("The result from CalcByVal is " & wkAmt)
    End Sub
    End Class

```

The screenshot shows the Visual Studio IDE with the project 'firstProcedures' open. The code editor displays 'firstProc.vb [Design]'. The code implements two buttons: 'btnPassByRef\_Click' and 'btnPassByVal\_Click'. Both buttons prompt for transaction amount and percentage, calculate the total, and add the result to a list. The 'btnPassByRef\_Click' button uses 'CalcByRef' and 'OtherCalcByRef' subroutines, while the 'btnPassByVal\_Click' button uses 'CalcByVal' and 'OtherCalcByVal' subroutines. Handwritten annotations in blue highlight variable names like 'wkAmt', 'wkPer', and 'wkMsg' throughout the code.

```

26
27  References
28  Private Sub btnPassByRef_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPassByRef.Click
29    Dim wkAmt As Single
30    Dim wkPer As Single
31    Dim wkMsg As String
32    wkAmt = InputBox("Enter the amount of the transaction", "Transaction Amount")
33    wkPer = InputBox("Enter the percent you are using", "Percent")
34    CalcByRef(wkAmt, wkPer)
35    wkMsg = "Back in btnPassByRef from CalcByRef " & wkAmt
36    lstTrans.Items.Add(wkMsg)
37    OtherCalcByRef(wkAmt, wkPer)
38    wkMsg = "Back in btnPassByRef from OtherCalcByRef " & wkAmt
39    lstTrans.Items.Add(wkMsg)
End Sub

40  Sub CalcByRef(ByRef sentAmt As Single, ByRef sentPer As Single)
41    sentAmt = sentAmt * sentPer
42    lstTrans.Items.Add("The result from CalcByRef is " & sentAmt)
End Sub

43  Sub CalcByVal(ByRef wkAmt As Single, ByVal wkPer As Single)
44    wkAmt = wkAmt * wkPer
45    lstTrans.Items.Add("The result from CalcByVal is " & wkAmt)
46    lstTrans.Items.Add("The result from CalcByRef is " & wkAmt)
End Sub

47  References
48  Private Sub btnPassByVal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPassByVal.Click
49    Dim wkAmt As Single
50    Dim wkPer As Single
51    Dim wkMsg As String
52    wkAmt = InputBox("Enter the amount of the transaction", "Transaction Amount")
53    wkPer = InputBox("Enter the percent you are using", "Percent")
54    CalcByVal(wkAmt, wkPer)
55    wkMsg = "Back in btnPassByVal from CalcByVal " & wkAmt
56    lstTransVal.Items.Add(wkMsg)
57    OtherCalcByVal(wkAmt, wkPer)
58    wkMsg = "Back in btnPassByVal from OtherCalcByVal " & wkAmt
59    lstTransVal.Items.Add(wkMsg)
60
61 End Sub

62  Sub CalcByVal(ByRef sentAmt As Single, ByVal sentPer As Single)
63    sentAmt = sentAmt * sentPer
64    lstTransVal.Items.Add("The result from CalcByVal is " & sentAmt)
End Sub

65  References
66  Sub OtherCalcByVal(ByVal wkAmt As Single, ByVal wkPer As Single)
67    wkAmt = wkAmt * wkPer
68    lstTransVal.Items.Add("The result from CalcByVal is " & wkAmt)
69  End Sub
End Class

```

The running application window titled 'First Procedures' shows the results of the calculations. It includes a 'Greet' message, a 'Total' section with three items, and a 'Pass By Ref' and 'Pass By Val' section each containing four items. The results show the original inputs and the calculated totals (12, 45, 2, 200, 40).

The screenshot shows the Microsoft Visual Studio IDE with the following details:

- Project:\*\*** firstProcedures.exe
- File:\*\*** FirstProcedures.vb
- Design View:\*\*** The code editor displays the VB.NET code for the `firstProc` class. The code includes two event handlers: `btnPassByRef_Click` and `btnPassByVal_Click`, both of which call the `CalcByRef` and `OtherCalcByRef` subroutines. The `CalcByVal` subroutine is also defined. Handwritten annotations in red and green are present in the code, such as "1000", "200", "40", and "1000".
- Output Window:\*\*** Shows the output of the application. The application window title is "First Procedures". Inside, a message box says "Hello, it is 3/10/2020 1:23:51 PM". Below it, a "Total" button is shown, followed by three text boxes:
  - "Pass By Ref": "The result from CalcByRef is 200  
Back in btnPassByRef from CalcByRef 200  
The result from CalcByRef is 40  
Back in btnPassByRef from OtherCalcByRef 40"
  - "Pass By Val": "The result from CalcByVal is 200  
Back in btnPassByVal from CalcByVal 200  
The result from CalcByVal is 200  
Back in btnPassByVal from OtherCalcByVal 200"
- Solution Explorer:\*\*** Shows the project structure.
- Status Bar:\*\*** Shows "100% No issues found".
- Bottom Bar:\*\*** Includes a search bar, pinned icons for File, Home, Recent, and Help, and status indicators for Line (Ln 14), Column (Col 50), Character (Ch 50), and Insert (INS).