

projRadioCheck - Microsoft Visual Studio Express 2012 for Windows Desktop

```

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS TEST WINDOW HELP
Toolbox Search Toolbox General
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.
frmRadioCheck.vb [Design] frmRadioCheck.vb [Design]
Public Class frmRadioCheck
    Private Sub bntCalculate_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bntCalculate.Click
        Dim wkTotal As Double, msgAns As Integer
        If chkBlue.Checked = True Then
            wkTotal = wkTotal + 0.6 * CInt(txtQuan.Text)
        End If
        If chkPumpkin.Checked = True Then
            wkTotal = wkTotal + 0.85 * CInt(txtQuan.Text)
        End If
        If chkHoc.Checked = True Then
            wkTotal = wkTotal + 0.9 * CInt(txtQuan.Text)
        End If
        If chkBrain.Checked = True Then
            wkTotal = wkTotal + 0.75 * CInt(txtQuan.Text)
        End If
        If radMain.Checked = True Then
            wkTotal = wkTotal * 1.05
        Else
            If radElis.Checked = True Then
                wkTotal = wkTotal * 0.95
            End If
        End If
        txtTotal.Text = wkTotal.ToString("C")
        msgAns = MessageBox.Show("Is your order correct?", "Order to confirm", MessageBoxButtons.OKCancel, MessageBoxIcon.Question)
        If msgAns = Windows.Forms.DialogResult.Cancel Then
            txtTotal.Clear()
            MessageBox.Show("Your order for " & wkTotal.ToString("C") & " has been canceled")
        End If
    End Sub
    Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub
    Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
        radMain.Checked = False
    End Sub

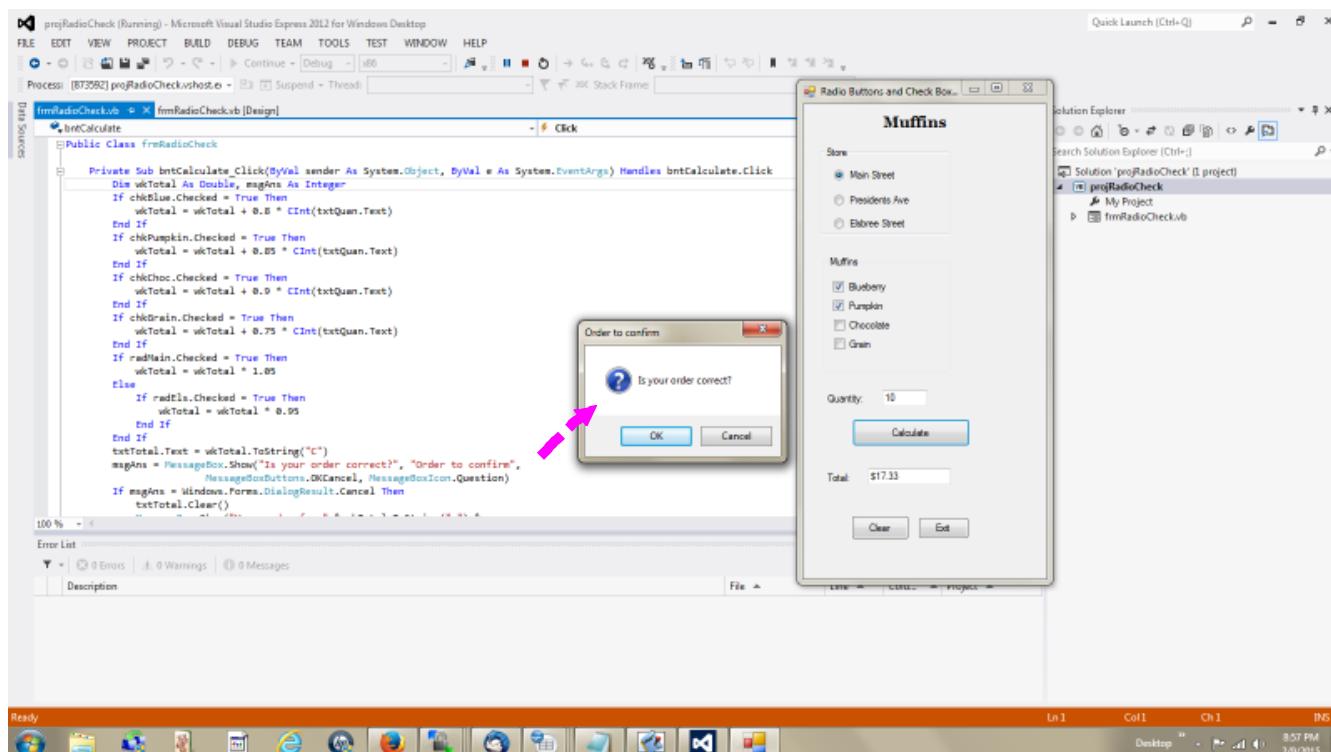
```

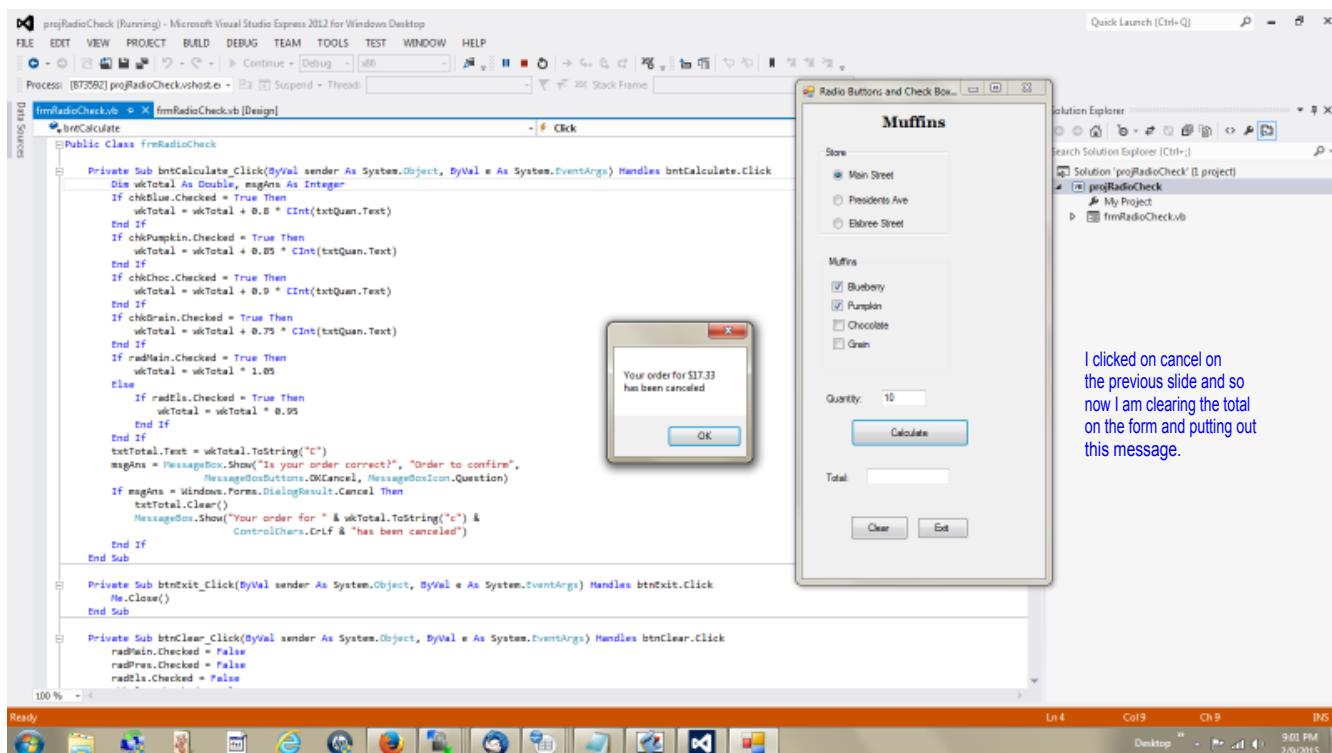
Notice that I checked each check box in a separate if statement because they could all be checked. If it is checked, I do the calcuation of price times quantity and added that to total.

Remember wkTotal = wkTotal + is the way that you take wkTotal, add to it and assign the solution to wkTotal.

Now I check the radio button named radElis. If it is checked then I am going to give a 5% discount.

Now I am assigning a messagebox to msgAns. I include the headers and make it the OKCancel button with a question mark icon. See next slide.





I clicked on cancel on the previous slide and so now I am clearing the total on the form and putting out this message.

When I clicked on OK I see the total.

The screenshot shows the Microsoft Visual Studio Express 2012 interface. On the left, the code editor displays the VB.NET code for the frmRadioCheck.vb file. The code handles button click events for calculate, exit, and clear, and performs calculations based on selected muffin types and quantities. On the right, a running application window titled "Muffins" is shown. This window has two sections: "Store" and "Muffins". In the "Store" section, there are three radio buttons: "Main Street" (selected), "Presidents Ave", and "Elbree Street". In the "Muffins" section, there is a list of muffin types with checkboxes: "Blueberry" (checked), "Pumpkin" (checked), "Chocolate" (unchecked), and "Grain" (unchecked). Below the muffin list is a "Quantity" input field set to 10, a "Calculate" button, and a "Total" output field showing \$17.33. At the bottom of the window are "Clear" and "Exit" buttons. The status bar at the bottom of the screen shows the date and time as 2/9/2015 9:02 PM.

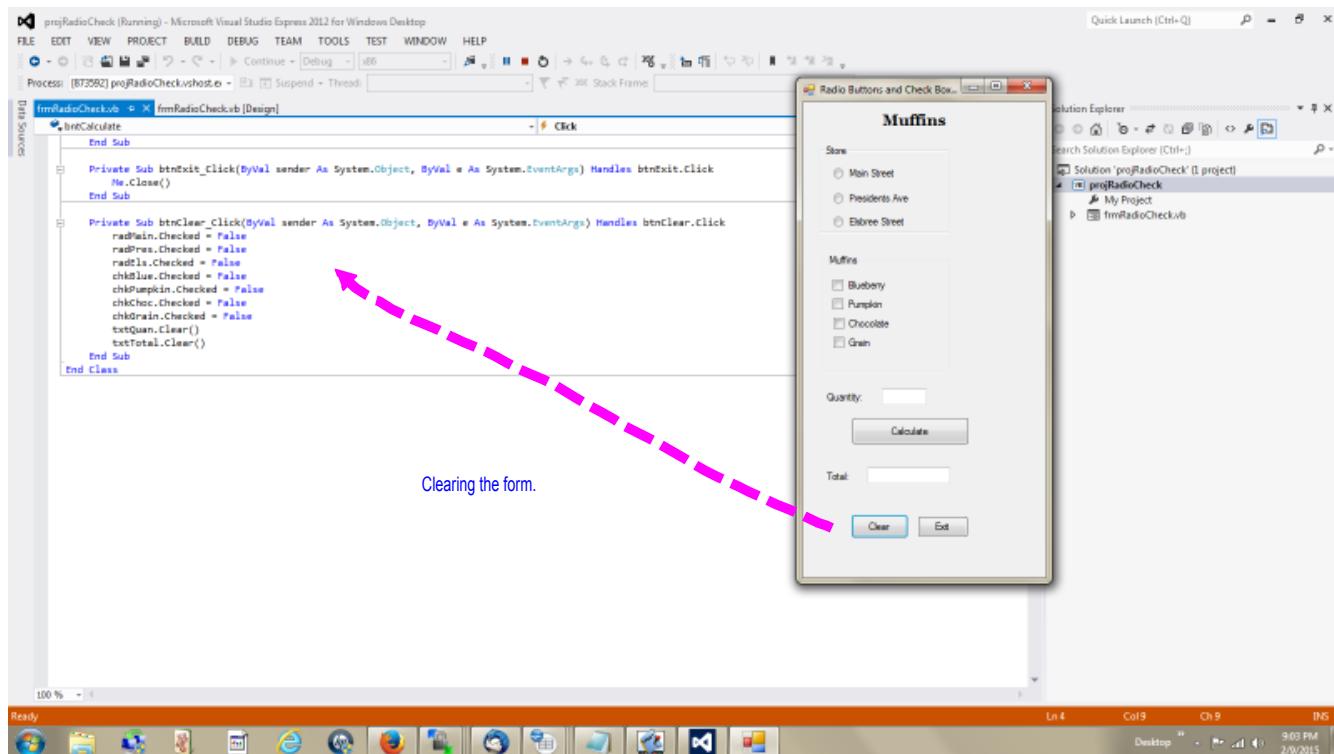
```

Private Sub bntCalculate_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bntCalculate.Click
    Dim wkTotal As Double, msgAns As Integer
    If chkBlue.Checked = True Then
        wkTotal = wkTotal + 0.8 * CInt(txtQuan.Text)
    End If
    If chkPumpkin.Checked = True Then
        wkTotal = wkTotal + 0.05 * CInt(txtQuan.Text)
    End If
    If chkChocolate.Checked = True Then
        wkTotal = wkTotal + 0.9 * CInt(txtQuan.Text)
    End If
    If chkGrain.Checked = True Then
        wkTotal = wkTotal + 0.75 * CInt(txtQuan.Text)
    End If
    If radMain.Checked = True Then
        wkTotal = wkTotal * 1.05
    Else
        If radRes.Checked = True Then
            wkTotal = wkTotal * 0.95
        End If
        txtTotal.Text = wkTotal.ToString("C")
        msgAns = MessageBox.Show("Is your order correct?", "Order to confirm",
                               MessageBoxButtons.OKCancel, MessageBoxIcon.Question)
        If msgAns = Windows.Forms.DialogResult.Cancel Then
            txtTotal.Clear()
            MsgBox.Show("Your order for " & wkTotal.ToString("c") &
                       ControlChars.CrLf & "has been canceled")
        End If
    End If
End Sub

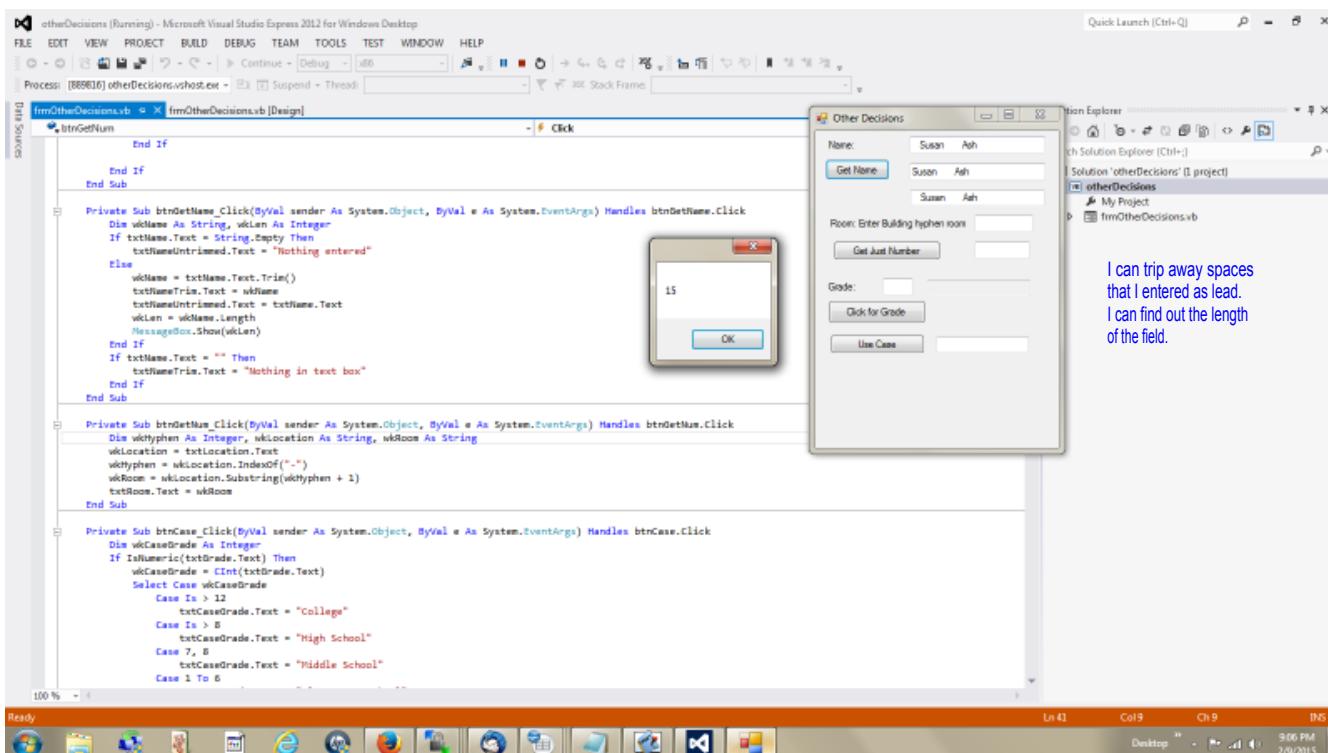
Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
    Me.Close()
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
    radMain.Checked = False
    radRes.Checked = False
    radFls.Checked = False

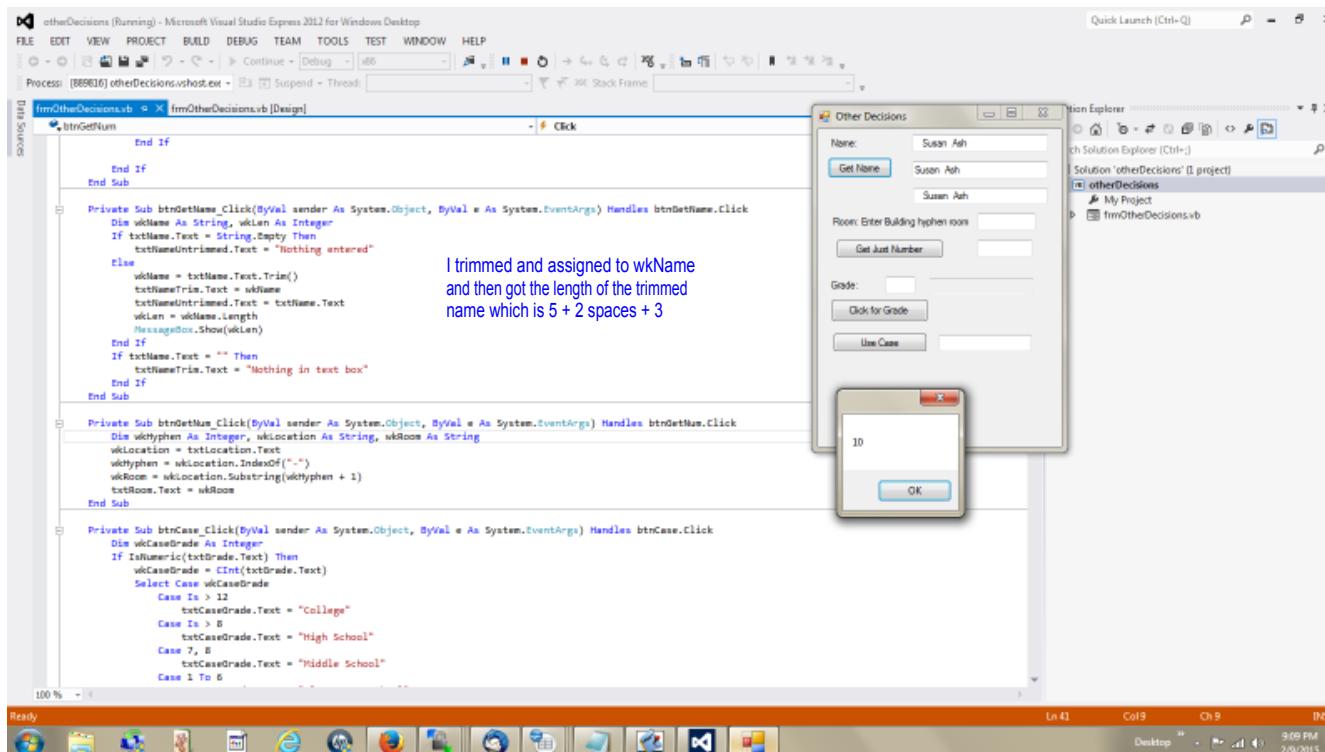
```

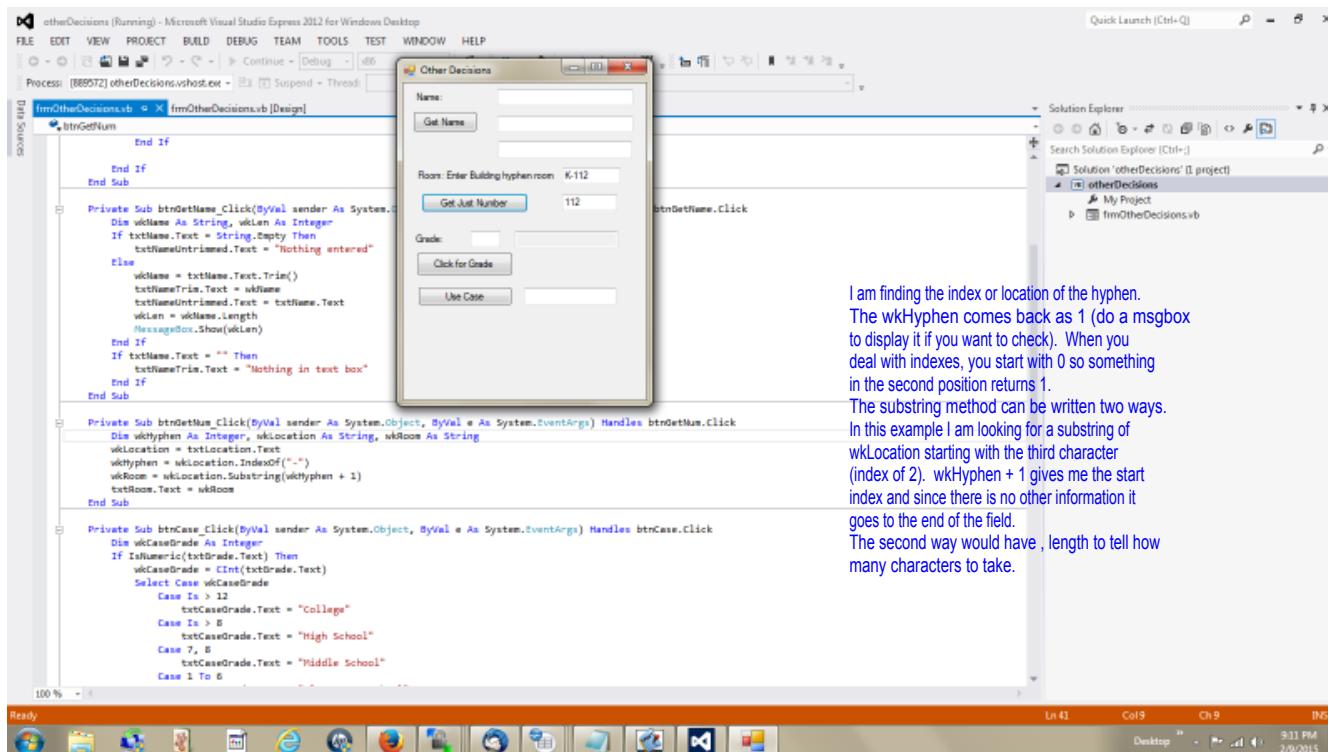


Clearing the form.



I can trip away spaces
that I entered as lead.
I can find out the length
of the field.





I am finding the index or location of the hyphen.
 The wkHyphen comes back as 1 (do a msgbox to display if you want to check). When you deal with indexes, you start with 0 so something in the second position returns 1.
 The substring method can be written two ways.
 In this example I am looking for a substring of wkLocation starting with the third character (index of 2). wkHyphen + 1 gives me the start index and since there is no other information it goes to the end of the field.
 The second way would have .length to tell how many characters to take.

Screenshot of Microsoft Visual Studio Express 2012 for Windows Desktop showing code and a running application.

The code in the editor is for a Windows application named "otherDecisions". It includes three event handlers:

- btnGetName_Click:** Handles the click event of the "Get Name" button. It checks if the text box is empty. If not, it trims the text, removes the first three characters, and shows a message box with the result.
- btnGetNum_Click:** Handles the click event of the "Get Just Number" button. It extracts the 4th character from the input string and displays it in a message box.
- btnCase_Click:** Handles the click event of the "Click for Grade" button. It checks if the input is numeric. If so, it converts it to an integer and uses a select case statement to determine the grade based on the value.

A note in the code states: "I put in code to show substring. Notice that it takes the 4th character until the end."

The application window titled "Other Decisions" is shown running in the foreground. It has three buttons: "Get Name", "Get Just Number", and "Click for Grade". A message box is displayed with the text "DEFG".

The Solution Explorer shows the project structure with a single file "frmOtherDecisions.vb".

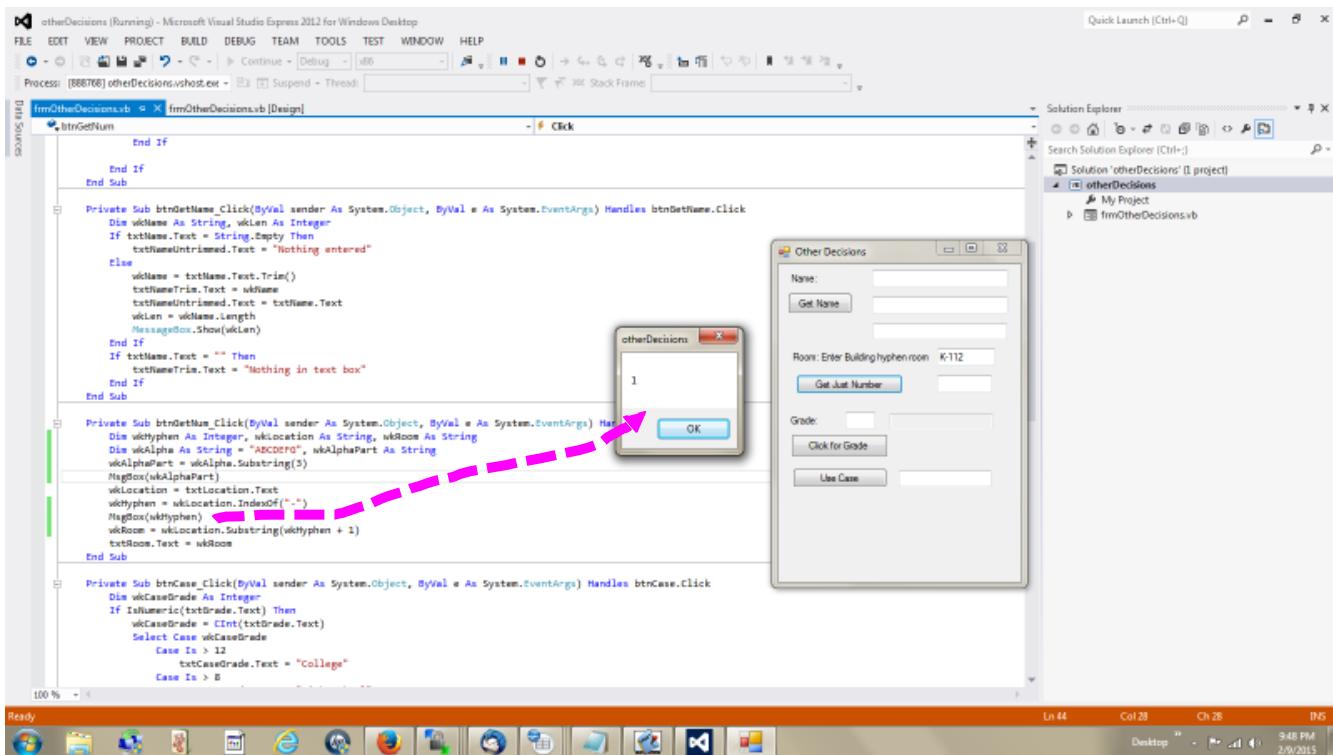
```

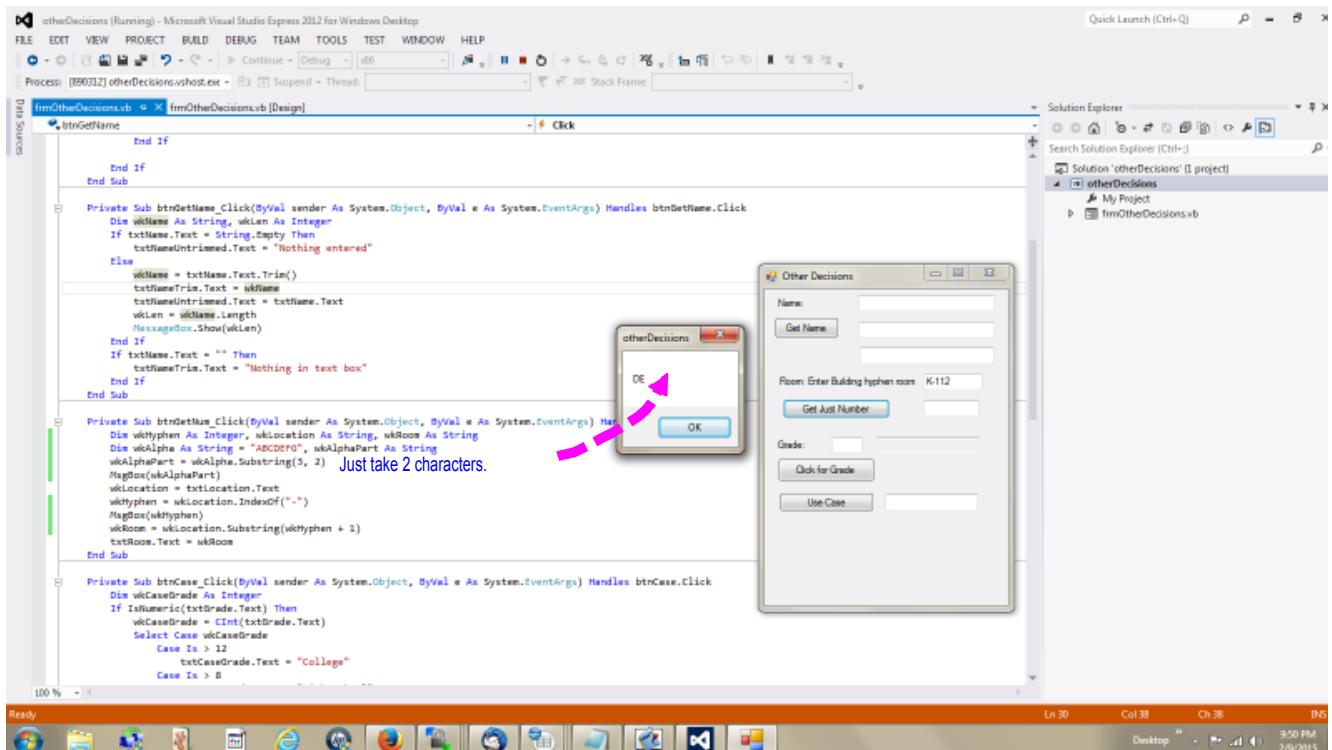
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS TEST WINDOW HELP
Process [888708] otherDecisions.vshost.exe - Continue | Debug | 100 | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Stack Frame
Quick Launch (Ctrl+Q) ×
Solution Explorer
Search Solution Explorer (Ctrl+F)
Solution 'otherDecisions' [1 project]
  otherDecisions
    My Project
      frmOtherDecisions.vb
  Other Decisions
    frmOtherDecisions.vb [Design]
      btnGetName_Click()
      btnGetNum_Click()
      btnCase_Click()
    frmOtherDecisions.vb [Code]
      Private Sub btnGetName_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGetName.Click
        Dim wkName As String, wkLen As Integer
        If txtName.Text = "" Then
          txtNameTrimmed.Text = "Nothing entered"
        Else
          wkName = txtName.Text.Trim()
          txtNameTrimmed.Text = wkName
          wkLen = wkName.Length
          MessageBox.Show(wkLen)
        End If
        If txtName.Text = "" Then
          txtNameTrimmed.Text = "Nothing in text box"
        End If
      End Sub

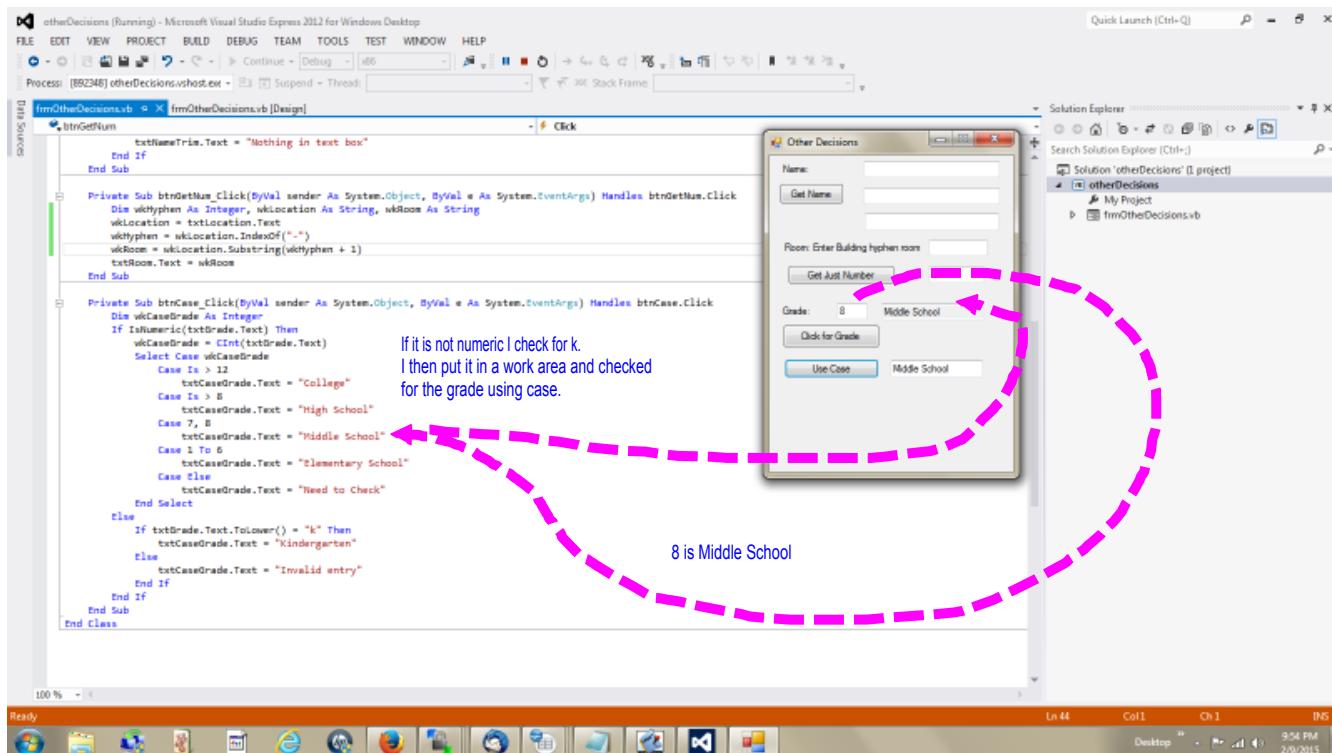
      Private Sub btnGetNum_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGetNum.Click
        Dim wkHyphen As Integer, wkLocation As String, wkRoom As String
        Dim wkAlpha As String = "ABCDE0", wkAlphaPart As String
        wkAlphaPart = wkAlpha.Substring(5)
        MessageBox(wkAlphaPart)

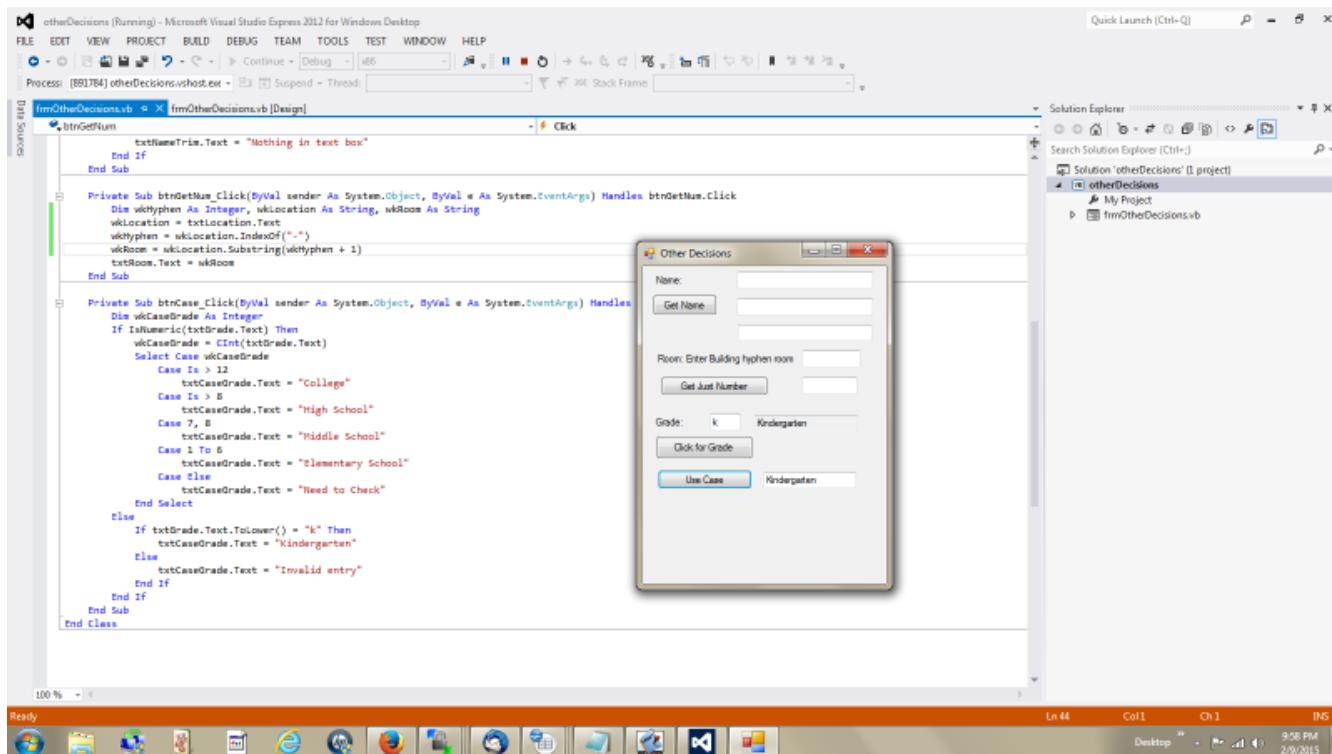
        wkLocation = txtLocation.Text
        wkHyphen = wkLocation.IndexOf("-")
        MessageBox(wkHyphen)
        wkRoom = wkLocation.Substring(wkHyphen + 1)
        txtRoom.Text = wkRoom
      End Sub

      Private Sub btnCase_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCase.Click
        Dim wkCaseGrade As Integer
        If IsNumeric(txtGrade.Text) Then
          wkCaseGrade = Int(txtGrade.Text)
          Select Case wkCaseGrade
            Case Is > 10
              txtCaseGrade.Text = "College"
            Case Is > 8
              txtCaseGrade.Text = "High School"
            Case Is > 6
              txtCaseGrade.Text = "Middle School"
            Case Is > 4
              txtCaseGrade.Text = "Elementary School"
            Case Is > 2
              txtCaseGrade.Text = "Preschool"
            Case Is <= 2
              txtCaseGrade.Text = "Not in School"
          End Select
        End If
      End Sub
    
```





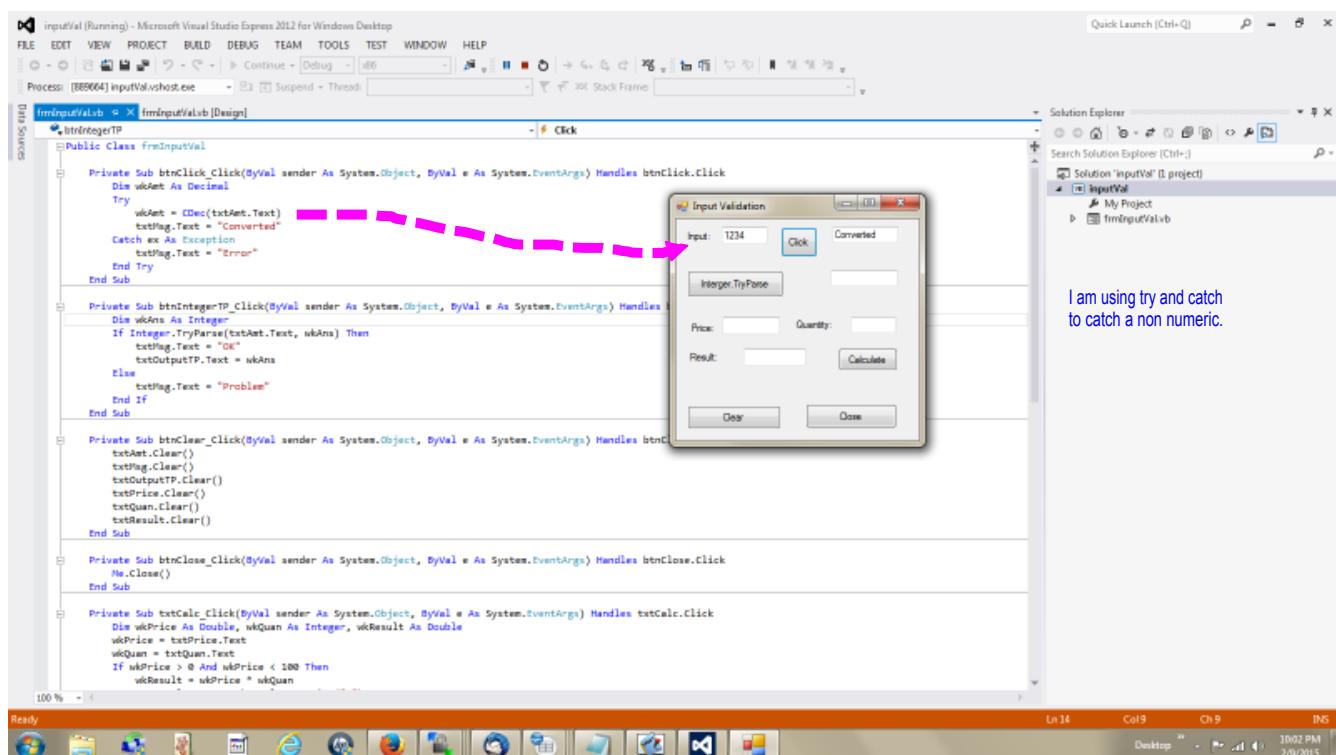


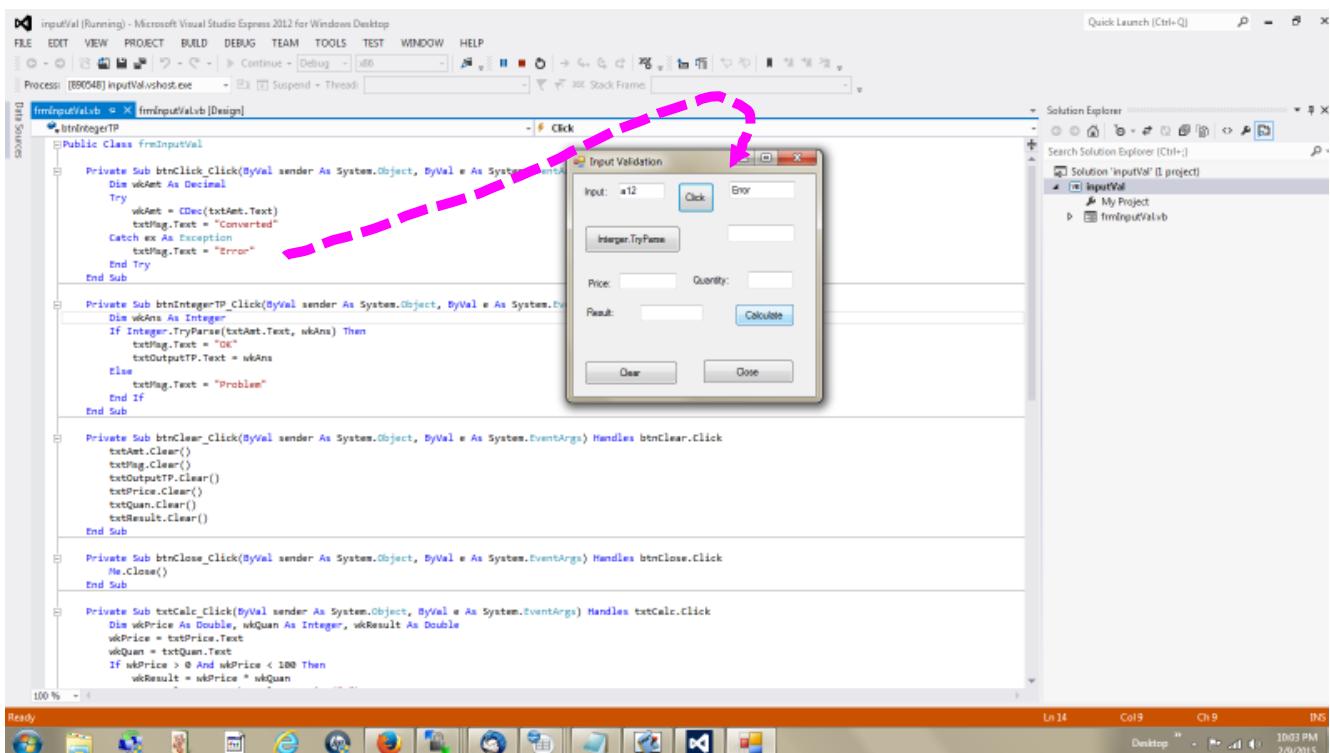


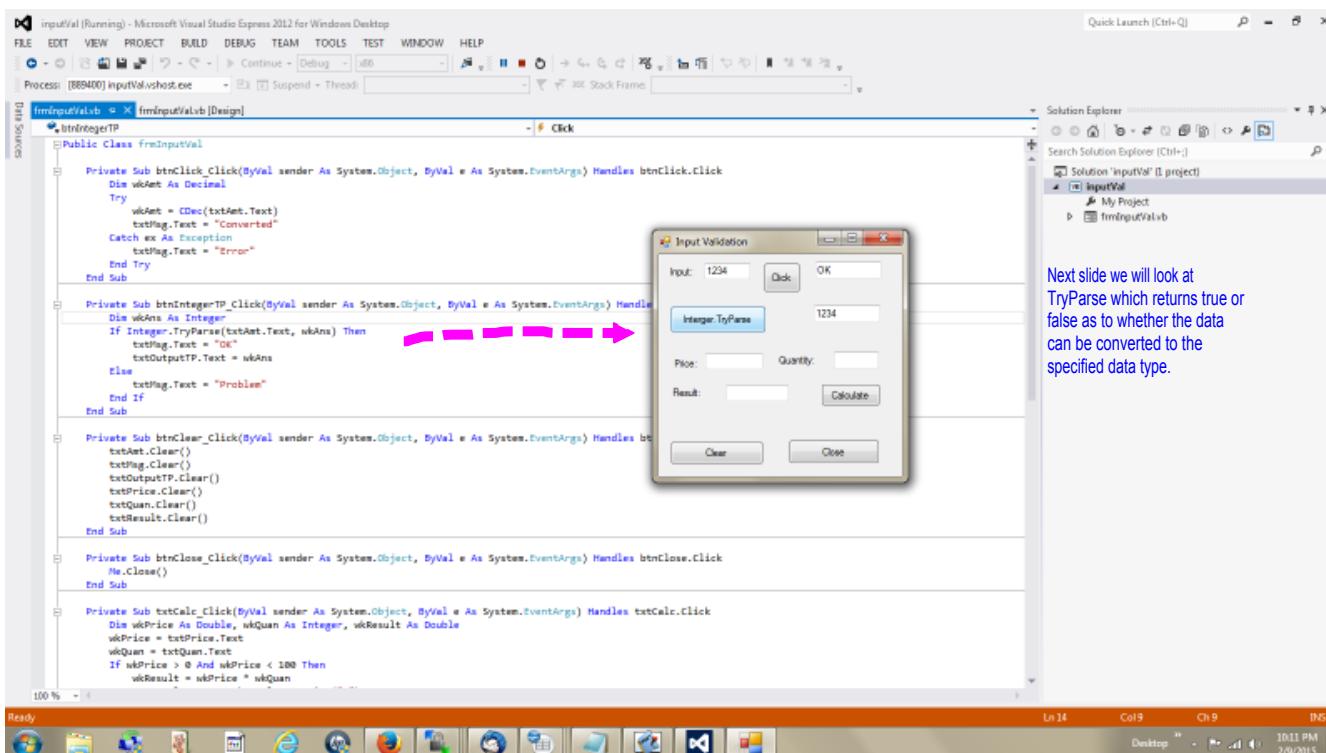
```
Private Sub btnGrade_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGrade.Click
    If IsNumeric(txtGrade.Text) Then
        If txtGrade.Text > 12 Then
            txtGradeWord.Text = "College"
        ElseIf txtGrade.Text > 8 Then
            txtGradeWord.Text = "High School"
        ElseIf txtGrade.Text > 6 Then
            txtGradeWord.Text = "Middle School"
        Else
            txtGradeWord.Text = "Elementary School"
        End If
    Else
        If txtGrade.Text.ToLower() = "k" Then
            txtGradeWord.Text = "Kindergarten"
        Else
            txtGradeWord.Text = "Invalid entry"
        End If
    End If
End Sub

Private Sub btnGetName_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGetName.Click
    Dim vName As String, vLen As Integer
    If txtName.Text = String.Empty Then
        txtNameUntrimmed.Text = "Nothing entered"
    Else
        vName = txtName.Text.Trim()
        txtNameTrim.Text = vName
        txtNameUntrimmed.Text = txtName.Text
        vLen = vName.Length
        MessageBox.Show(vLen)
    End If
    If txtName.Text = "" Then
        txtNameTrim.Text = "Nothing in text box"
    End If
End Sub
```

You see two answers because I used the case and I also did it using ElseIf.







Next slide we will look at TryParse which returns true or false as to whether the data can be converted to the specified data type.

