

The screenshot shows a Firefox browser window with the address bar displaying 'www.pgrocer.net/Cis17/cis17programs.html'. The page title is 'Programs for CIS17 - Programming: Logic, Design and Implementation'. The content is organized into two main sections:

- Visual Basic (2010)**:
  - firstSp12.zip - introduction to using VB2010
  - FirstMath.zip - calculations
  - basicMath.zip - calculations
  - loopswhile120.zip - introduction to loops
  - workareas.zip - introduction to variables
  - userInput.zip - introduction to user input
  - deptArray.zip - introduction to arrays
  - NextVB.zip - two programs in this zip
  - VBF11CIS120.zip - multiple programs
  - projWriteText.zip - creating a file
  - readProj.zip - reading a file
- JavaScript**:
  - hello.html
  - multiply.html
  - multiplyans.html
  - addnum.html
  - addnumcont.html
  - mathans.html
  - ifwith.html
  - MAorRI.html
  - prob2.html
  - whileloop.html
  - doloop.html
  - ifwithloop.html
  - finaltotals.html
  - Loop comparison:
  - Math facts while loop
  - Math facts nested while loop
  - Math facts do...while loop
  - Math facts nested do...while loop
  - Math facts for loop
  - Math facts nested for loop
  - Arrays:
  - thedata.html
  - thedata12.html
  - thedata12another.html
  - thedata11.html
  - soupparrays1.html Search using JavaScript

A pink arrow points from the text 'These are the examples we will look at today.' to the file 'loopswhile120.zip' in the Visual Basic section.

The screenshot shows the Visual Studio IDE with the code for `frmDeptArray.vb` in Design view. The code is as follows:

```
Public Class frmDeptArray
    Dim DeptName(5) As String
    Private Sub frmDeptArray_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        DeptName(1) = "Girls"
        DeptName(2) = "Boys"
        DeptName(3) = "Infants"
        DeptName(4) = "Toys"
        DeptName(5) = "Books"
    End Sub

    Private Sub btnGetName_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGetName.Click
        If Val(txtDeptNo.Text) > 0 And Val(txtDeptNo.Text) < 6 Then
            txtDeptName.Text = DeptName(txtDeptNo.Text)
        Else
            txtDeptName.Text = ("Invalid Department Number")
        End If
    End Sub

    Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
        txtDeptNo.Clear()
        txtDeptName.Clear()
    End Sub

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

Handwritten annotations in blue ink include:

- "House Keeping" with an arrow pointing to the `frmDeptArray_Load` event handler.
- "Wrapup" with an arrow pointing to the `End Class` statement.
- "Processing" with a bracket pointing to the `btnGetName_Click`, `btnClear_Click`, and `btnExit_Click` event handlers.

Other annotations include:

- "Define the array name and the top element of 5." pointing to `Dim DeptName(5) As String`.
- "Click anywhere on the form and you create the form\_load - the load event which happens when the page is loaded." pointing to the `Handles MyBase.Load` clause.
- "The wrapup - End or Me.Close()" pointing to the `End` statement in `btnExit_Click`.

deptArray - Microsoft Visual Studio (Administrator)

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM SQL TOOLS TEST ARCHITECTURE ANALYZE WINDOW HELP

Toolbox Search Toolbox

frmDeptArray.vb frmDeptArray.vb [Design] (frmDeptArray Events) Load

```
Public Class frmDeptArray
    Dim DeptName(5) As String
    Private Sub frmDeptArray_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
        DeptName(1) = "Girls"
        DeptName(2) = "Boys"
        DeptName(3) = "Infants"
        DeptName(4) = "Toys"
        DeptName(5) = "Books"
    End Sub

    Private Sub btnGetName_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        If Val(txtDeptNo.Text) > 0 And Val(txtDeptNo.Text) < 6 Then
            txtDeptName.Text = DeptName(txtDeptNo.Text)
        Else
            txtDeptName.Text = ("Invalid Department Number")
        End If
    End Sub

    Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        txtDeptNo.Clear()
        txtDeptName.Clear()
    End Sub

    Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        End
    End Sub
End Class
```

DeptName is repeat so you have to use the index in parenthesis to specify which.

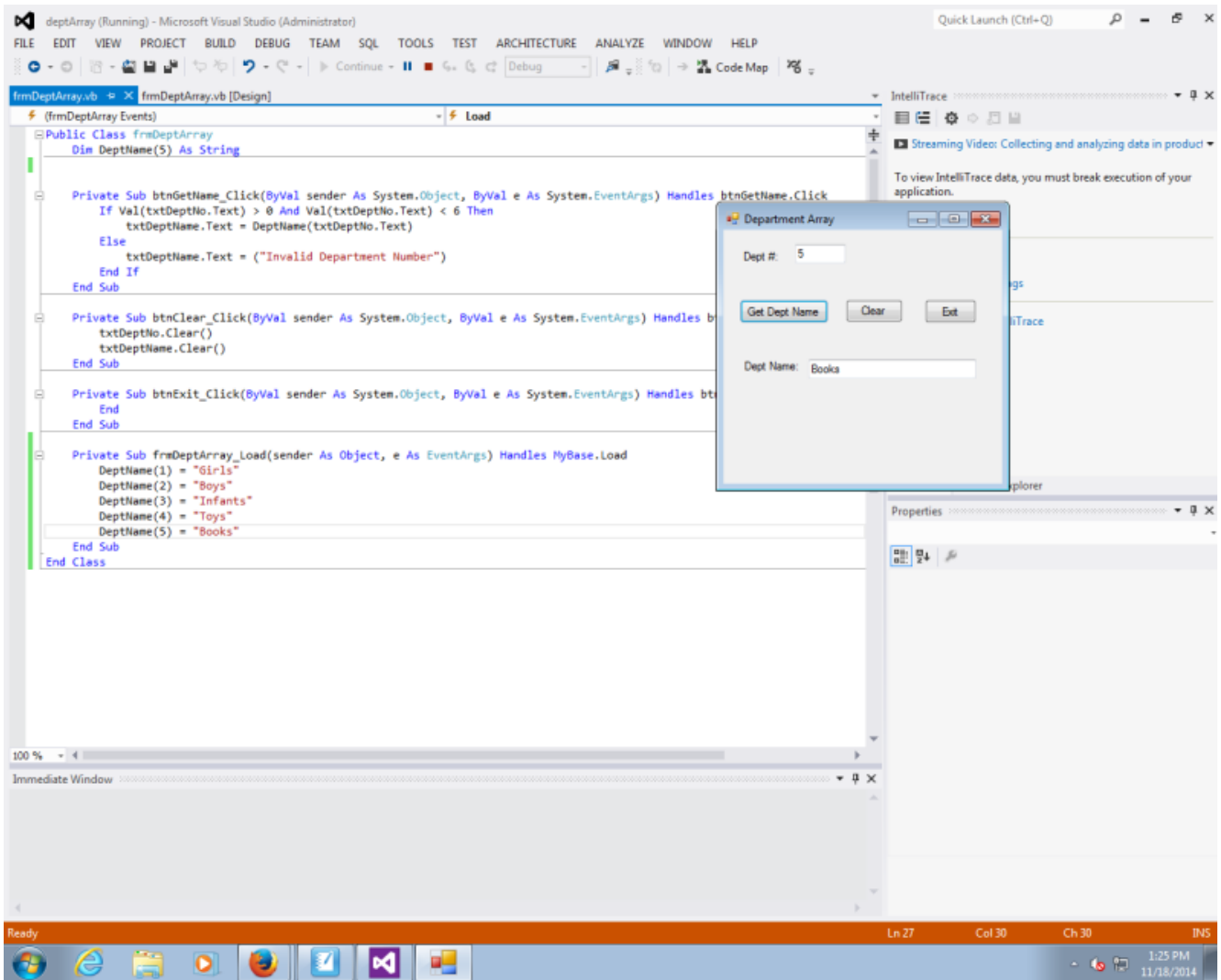
I only want a txtDeptNo.text that is greater than 0 and less than 6 since this fits with the array I created.

Code to clear a field.

Index  
Pointer  
Subscript

Properties

Ready Ln 3 Col 30 Ch 30 INS



The screenshot shows the Visual Studio IDE with the following code in `frmWhile.vb`:

```
Public Class frmWhile
    Dim wkCt As Integer = 1
    Dim wkTotal As Double = 0

    Private Sub btnProcess_Click(sender As System.Object, e As System.EventArgs) Handles btnProcess.Click
        Do While wkCt <= CInt(txtHowMany.Text)
            wkTotal = wkTotal + txtAmt.Text
            wkCt = wkCt + 1
        Loop
        txtResult.Text = wkTotal
        'Actually there is a while that can test before and after
    End Sub

    Private Sub btnUntil_Click(sender As System.Object, e As System.EventArgs) Handles btnUntil.Click
        Do
            wkTotal = wkTotal + txtAmt.Text
            wkCt = wkCt + 1
        Loop Until wkCt > CInt(txtHowMany.Text)
        txtResult.Text = wkTotal
        'Actually there is an until that can test before and after
    End Sub

    Private Sub btnFor_Click(sender As System.Object, e As System.EventArgs) Handles btnFor.Click
        For wkCt = 1 To txtHowMany.Text Step 1
            wkTotal = wkTotal + txtAmt.Text
        Next
        txtResult.Text = wkTotal
    End Sub
End Class
```

Hand-drawn green circles highlight the loop conditions: `wkCt <= CInt(txtHowMany.Text)` in the `Do While` loop, `wkCt > CInt(txtHowMany.Text)` in the `Loop Until` loop, and `1 To txtHowMany.Text` in the `For` loop.

Looking at three kinds of loops.  
The Do While which in this example has the condition at the top of the loop.  
The Do Until which has the condition at the bottom of the loop.  
The FOR loop.

There is a problem with wkCt and wkTotal that we are going to discuss in this class.

loopswhile120 (Running) - Microsoft Visual Studio (Administrator)

Public Class frmWhile  
Dim wkCt As Integer = 1  
Dim wkTotal As Double = 0  
  
Private Sub btnProcess\_Click(sender As System.Object, e As System.EventArgs) Handles btnProcess.Click  
wkCt = 1  
Do While wkCt <= CInt(txtHowMany.Text)  
wkTotal = wkTotal + txtAmt.Text  
wkCt = wkCt + 1  
Loop  
txtResult.Text = wkTotal  
'Actually there is a while that can test before and after  
End Sub  
  
Private Sub btnUntil\_Click(sender As System.Object, e As System.EventArgs) Handles btnUntil.Click  
wkCt = 1  
Do  
wkTotal = wkTotal + txtAmt.Text  
wkCt = wkCt + 1  
Loop Until wkCt > CInt(txtHowMany.Text)  
txtResult.Text = wkTotal  
'Actually there is an until that can test before and after  
End Sub  
  
Private Sub btnFor\_Click(sender As System.Object, e As System.EventArgs) Handles btnFor.Click  
For wkCt = 1 To CInt(txtHowMany.Text) Step 1  
wkTotal = wkTotal + txtAmt.Text  
Next  
txtResult.Text = wkTotal  
End Sub  
End Class

While Loop  
How many times: 5  
Amount to Add: 12  
Result: 120  
Process  
Other Loops: Until For

I initialized wkCt in each Sub so that it would start over at 1 each time the button was pressed. But... I did not move 0 to wkTotal so work total continues to grow. I could initialize wkTotal = 0 at the top of the Sub right before the loop or at the end of the Sub right before End Sub.

IntelliTrace  
Streaming Video: Collecting and analyzing data in product  
To view IntelliTrace data, you must break execution of your application.  
Break all  
More options:  
Open IntelliTrace settings  
Learn more about IntelliTrace  
IntelliTrace Solution Explorer  
Properties

100%  
Immediate Window

Ready Ln 16 Col 17 Ch 17 INS 12:03 PM 11/18/2014

deptArray - Microsoft Visual Studio (Administrator)

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM SQL TOOLS TEST ARCHITECTURE ANALYZE WINDOW HELP

Toolbox Search Toolbox

frmDeptArray.vb frmDeptArray.vb [Design]\*

(frmDeptArray Events)

Public Class frmDeptArray *I changed the name and size of the array.*  
Dim crsName(7) As String

Private Sub btnGetName\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
If Val(txtDeptNo.Text) > 0 And Val(txtDeptNo.Text) < 6 Then  
txtDeptName.Text = crsName(txtDeptNo.Text)  
Else  
txtDeptName.Text = ("Invalid Department Number")  
End If  
End Sub

Private Sub btnClear\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
txtDeptNo.Clear()  
txtDeptName.Clear()  
End Sub

Private Sub btnExit\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
End  
End Sub

Private Sub frmDeptArray\_Load(sender As Object, e As EventArgs) Handles MyBase.Load  
*Me.lose()*  
crsName(1) = "CIS110"  
crsName(2) = "CIS111"  
crsName(3) = "CIS112"  
crsName(4) = "CIS113"  
crsName(5) = "CIS150"  
crsName(6) = "CIS156"  
crsName(7) = "CIS159"  
End Sub  
End Class

*I decided to do a different array using crsName - I deleted the first one, clicked in a blank area on the form to create the Load event and put the array back in - order does not matter.*

Solution Explorer

Search Solution Explorer (Ctrl+Q)

Solution 'deptArray' (1 project)

deptArray

My Project

frmDeptArray.vb

Output

Show output from: Debug

Managed (v4.0.30319): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC\_...  
'deptArray.vshost.exe' (Managed (v4.0.30319)): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC\_...  
'deptArray.vshost.exe' (Managed (v4.0.30319)): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC\_...  
The thread '<No Name>' (0x21f4) has exited with code 0 (0x0).  
The thread 'vshost.RunParkingWindow' (0x27dc) has exited with code 0 (0x0).  
The thread '<No Name>' (0x1250) has exited with code 0 (0x0).  
The program '[8604] deptArray.vshost.exe: Program Trace' has exited with code 0 (0x0).  
The program '[8604] deptArray.vshost.exe: Managed (v4.0.30319)' has exited with code 0 (0x0).

Ready Ln 29 Col 30 Ch 30 INS

