Review for Test 3
CPT 186
Add to the End of a List Box or Combo Box

- Write the statement to add an item to a list box or combo box.
  - Format: `ObjectName.Items.Add("Item to be Added")`

- Example
  - `CoffeeComboBox.Items.Add(“Mountain Blend”)`
Add at a Specific Index

- Write the statement to add the value stored in a text box to a list box or combo box at a specific Index value.
- Format:
  
  `ObjectName.Items.Insert(IndexValue, _ TextBoxName.Text)`

- Example
  
  `SyrupListBox.Items.Insert(3,SyrupTextBox.Text)`
Add the Currently Typed Item

- Write the statement to add the item that the user has just typed into the text box portion of a combo box to the same combo box.

- Format:
  ObjectName.Items.Add(ObjectName.Text)

- Example:
  CoffeeComboBox.Items.Add(CoffeeComboBox.Text)
Count the Number of Items

- Write the statement to place the number of items contained in a list box or combo box into an integer variable.

- Format:
  CountInteger = ObjectName.Items.Count

- Example:
  CountInteger = CoffeeComboBox.Items.Count
Deselect Items in a List or Combo Box

- Write the statement to deselect all items for a list box or combo box so that none of the items in the list box or combo box are selected.

- Format:
  ObjectName.SelectedIndex = -1

- Example:
  CoffeeComboBox.SelectedIndex = -1
Write the statement to display a given message followed by the number of items in a list box or combo box in a message box.

Format:
MessageBox.Show("The number of items in the list is " & ObjectName.Items.Count)

Example:
MessageBox.Show("The number of items in the list is " & CoffeeComboBox.Items.Count)
What is the SelectedIndex property used for?
- to select an item in the list or to determine which item is selected.

Given the number of items in a list box or combo box, be able to determine the
- a) Items.Count property
- b) the highest SelectedIndex property
- c) the lowest SelectedIndex property.

Given that there are 20 items in the list
- a) Items.Count is 20
- b) Highest SelectedIndex property is 19
- c) Lowest SelectedIndex property is 0
Remove an Item from the List or Combo Box

- How do you remove an individual item from a list box or combo box?
  - Use the RemoveAt method and specify the index of the item
    `CoffeeComboBox.Items.RemoveAt(3)`
  - Use the Remove method and specify the text
    `CoffeeComboBox.Items.Remove(“Kona Blend”)`
Display the Selected Item

- Write the code to display the selected item from the list box or combo box in a label.
- Format:
  ```csharp
  LabelNameLabel.Text = ObjectName.Text or
  LabelNameLabel.Text =
  ObjectName.Items(ObjectName.SelectedIndex)
  ```
- Example:
  ```csharp
  CoffeeLabel.Text = CoffeeComboBox.Text or
  SyrupLabel.Text =
  SyrupListBox.Items(SyrupListBox.SelectedIndex)
  ```
Remove the Item at a Specific Index

- Write the statement to remove the item at a specified index from a list box or combo box.

- Format:
  ObjectName.Items.RemoveAt(IndexInteger)

- Example:
  CoffeeComboBox.Items.RemoveAt(0)
Remove the Selected Item

- Write the statement to remove the **currently selected** item from a list box or combo box.

- Format:
  `ObjectName.Items.RemoveAt(ObjectName.SelectedIndex) OR
  ObjectName.Items.Remove(ObjectName.Text)`

- Example:
  `CoffeeComboBox.Items.RemoveAt(CoffeeComboBox.SelectedIndex) OR
  SyrupListBox.Items.Remove(SyrupListBox.Text)`
Write the statement to clear all of the items from a list box or combo box.

Format:
ObjectName.Items.Clear()

Example:
CoffeeComboBox.Items.Clear()
SyrupListBox.Items.Clear()
Do Until Pretest Loop

- Write the first and last lines of a Do Until loop and the statement inside the loop to increment a given counter by a given amount. Use a pretest loop and execute until the value in counter is equal to a specific value.

- Given CountInteger = 0, increment by 1, repeat until CountInteger = 20

Do Until CountInteger = 20
    'statements in loop

    CountInteger += 1 'increment counter
Loop
**Do Until Posttest Loop**

- Write the first and last lines of a Do Until loop and the statement inside the loop to increment a given counter by a given amount. Use a posttest loop and execute until the value in counter is equal to a specific value.

- Given TotalInteger = 0, increment by 2, repeat until TotalInteger is greater than 10

    Do
        'statements in loop
        TotalInteger += 2  'increment the counter
    Loop Until TotalInteger > 10
For/Next that Counts Down

- Write the first and last lines of a For/Next that will start the value of the LoopIndex variable at a given value and will count down by a given value until the value in the LoopIndex reaches a given value.

  Given the LoopIndex is IndexInteger, start at 25 count down by 5’s until you reach 0

For IndexInteger = 25 to 0 Step -5
'statements in loop

Next IndexInteger
Write the first and last statements for a FOR NEXT loop that will execute as many times as there are items in a combo box or list box. You will be given the name of the list/combo box and the name of the loop counter variable.

For IndexInteger = 0 to CoffeeComboBox.Items.Count – 1

' Statements in loop

Next IndexInteger
Make Text in a Textbox Appear Selected

- Write the statements to make the entry in the text box appear selected when the text box's Enter event procedure occurs. The first and last lines of the procedure are given. You just need to write the single statement that will be placed inside the procedure to make the text appear selected.

```vbnet
Private Sub MessageTextBox_Enter(ByVal sender As Object, ByVal e As System.EventArgs) Handles MessageTextBox.Enter
    'Select any existing text
    MessageTextBox.SelectAll()
End Sub
```
Write the statement to use the DrawString method to send a given line to the graphics page. The font, the Brush, the X coordinate and the Y coordinate will be given.

- Given “This is a test” as the line, PrintFont as the font, Brushes.Black as the Brush, 300.0 as the X coordinate, and 300.0 as the Y coordinate. Assume that e is declared as System.Drawing.Printing.PrintPageEventArgs.

```csharp
e.Graphics.DrawString("This is a test", PrintFont, Brushes.Black, 300.0, 300.0)
```
Create a Static Variable

- Be able to write the statement to create a static variable given the name of the variable and the value to initialize it to.

```
Static LineCountInteger As Integer = 1
```
Declare an Array

- Write a statement to create a one-dimensional decimal array given the data type, array name and number of elements.

  Dim AmountDecimal(9) As Decimal

- Will create a decimal array called AmountDecimal with 10 elements
- The number in () is the highest subscript or index NOT the number of elements
For Each/Next Loop
Set a String Array Back to Null “”

- Write the For Each/Next loop to reinitialize a previously created string array to the null string ("""") or a previously created numeric array to 0.
  - Assume that the array has been loaded with values and your code is going to reinitialize it back to the null string or to 0. Include any Dim statements you may need with your loop.

Dim PartsString(10) As String (given)
Dim OneElementString As String

For Each OneElementString in PartsString
    OneElementString = ""
Next OneElementString
For Each/Next Loop
Set a Numeric Array Back to 0

Dim ValueInteger(10) As Integer (given)
Dim OneElementInteger As Integer

For Each OneElementInteger in ValueInteger
    OneElementInteger = 0
Next OneElementInteger
Process an Array with a For/Next Loop

- Given variable declarations, write a For/Next loop to add 3 to each element of the AmountInteger array. Assume that the array has already been loaded with integer values and your code will increment each array element by 3.

  - Given:
    Dim AmountInteger(5) As Integer
    Dim IndexInteger As Integer
  - Write:

    For IndexInteger = 0 to 5
      AmountInteger(IndexInteger) += 3
    Next IndexInteger
Create a Structure

- Be able to write a structure given a description of the structure.
- For Example: Write a structure called Employee that contains the following items:
  Last Name (string), First Name (string), Employee ID (string), Department Name (string), Pay Rate. (decimal), Hours Worked (decimal) and Total YTD Gross (decimal)

Structure Employee

Dim LastNameString As String
Dim FirstNameString As String
Dim EmployeeIDString As String
Dim DeptNameString As String
Dim PayRateDecimal As Decimal
Dim HoursDecimal As Decimal
Dim TotalYTDGrossDecimal As Decimal
End Structure
Create a Structure Variable

- Given a structure and a structure variable created from the structure, be able to assign a value to one of the elements of the structure.
  - For Example:
    Structure Inventory
      Dim DescString As String
      Dim ProdIDString As String
      Dim QuantityInteger As Integer
      Dim PriceDecimal As Decimal
    End Structure

    Dim PliersInventory As Inventory
    PliersInventory.PriceDecimal = 24.99D
Create an Array of Structures

- Given a structure declaration, write a statement to create an array of a given number of Item variables.
  - Create an array of 8 Inventory items given the following:
    Structure Inventory
      Dim DescString As String
      Dim ProdIDString As String
      Dim QuantityInteger As Integer
      Dim PriceDecimal As Decimal
    End Structure

    Dim InvenInventory(7) As Inventory

    - Remember to subtract 1 because your subscripts start at 0
Given the declaration for a 2 dimensional array, write the statements to initialize the elements of the array to 0.0. Assume that the array has already been loaded with decimal values and your code will be used to reinitialize the array back to 0. Include the Dim statements for any additional variables that you might need.

- Given Dim SalesDecimal(4, 4) As Decimal
  Dim RowInteger As Integer
  Dim CollInteger As Integer

  For RowInteger = 0 To 4
    For CollInteger = 0 To 4
      SalesDecimal(RowInteger, CollInteger) = 0.0D
    Next CollInteger
  Next RowInteger
With a For Each/Next Loop

Dim OneElementDecimal as Decimal

For Each OneElementDecimal in SalesDecimal
    OneElementDecimal = 0.0D
Next OneElementDecimal
Accumulate Row Totals

- Given the following declarations, write the code to add each row of a 2 dimensional array `AmountDecimal` and store the row totals in the one dimensional array `TotalDecimal`. Assume that the values have already been loaded into the `AmountDecimal` array. Include any Dim statements for any additional variables that you might need to create.

  - Given the following:
    Dim `AmountDecimal(3, 5)` As Decimal
    Dim `TotalDecimal(3)` As Decimal
    Dim `RowInteger` As Integer
    Dim `ColInteger` As Integer
  - Write:
    For `RowInteger = 0` to `3`
      For `ColInteger = 0` to `5`
        `TotalDecimal(RowInteger)` +=
        `AmountDecimal(RowInteger, ColInteger)`
      Next `ColInteger`
    Next `RowInteger`
Test 3 Coding

- The 20 point coding part of the test will consist of a program using the direct reference method of table lookup for a 2 dimensional table. It will be similar to the program code on p. 346 for the example except that you are to use 2 separate nested If statements so that each combo box has its own error message. Following is the code on p. 346 for the LookupButton_Click event rewritten in the way I want your test code to be written.
Dim WeightIndexInteger, ZoneIndexInteger as Integer
WeightIndexInteger = WeightListBox.SelectedIndex
ZoneIndexInteger = ZoneListBox.SelectedIndex
If WeightIndexInteger <> -1 Then
    If ZoneIndexInteger <> -1 Then
        ChargesTextBox.Text = RateDecimal(WeightIndexInteger, ZoneIndexInteger).ToString("N")
    Else
        MessageBox.Show("Select the zone.", "Information Missing", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
    End If
Else
    MessageBox.Show("Select the weight.", "Information Missing", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
End If