week11

April 25, 2018

1 Loops

- Another Conditional Structure
  - Has condition and code block
- Repeats the code block while the condition is true
  - How is this different from an if statement?
  - When does execution stop?
- Most useful with collection, to perform same action on collection items

2 Loops in VB

- While Loops: If like condition checked before the start of any iteration **(what is an iteration?)**
- Do While loops: If like condition checked at the end of every iteration
- For Loops: Used with collections, condition is implicit, works on all items in a collection until all are exhausted
  - We will mainly use for loops

3 For Loops

- Used to iterate over collection items

```
Dim nameList As New List(Of String) From {"mohammed", "ali", "sara"}

For Each x in nameList: ' Where is the condition?
    MsgBox(x)
Next 'end of the block
```

What is x? What is its type?

Important Note: Never change the list you are looping with for each by adding or dropping items.

4 Another Kind of For Loops

- Used to repeat an action a specific number of times

```
Dim nameList As New List(Of String) From {"mohammed", "ali", "sara"}

For x = 0 to 10: ' Where is the condition?
    MsgBox(x)
Next 'end of the block
```

What is the difference from the previous for loop? Can we use this type to iterate over nameList?

Remember: if you know the index, you can fetch an item or replace it directly from the list

5 Challenge

Create an application that: - Stores names of contacts - Allows user to capitalize all names - Allows user to remove names - Allows user to sort the names (use two different methods) - Allows the user to restore all the lost names and reset any changes done to the list

6 Useful Operations That USe Loops

- Counting
- Summing
- Find Min/Max
- Sorting
- Search/Filter

7 Counting

```
Dim myList As New List(Of Integer) From {5, 6, 72, 3}
Dim intCount As Integer = 0 'must have a counter initialized to zero

For Each x In myList
       intCount = intCount + 1
Next

MsgBox("Total count: " & intCount)
```

Can you modify to count only odd numbers?

8 Summing

```
Dim myList As New List(Of Integer) From {5, 6, 72, 3}
Dim intSum As Integer = 0 'must have a summation variable initialized to zero
```

```
For Each x In myList
       intSum = intSum + x
Next

MsgBox("Total count: " & intSum)
```

Can you modify to sum only even numbers? or numbers less than 10?

9 Maximum

```
Dim myList As New List(Of Integer) From {5, 6, 72, 3}
Dim maximum As Integer =  myList(0) 'Assume first value is minimum, look for bigge

For Each x In myList
    ' look for a value greater than maximum
    ' it should be the new maximum
    If x > maximum Then
        maximum = x
    End If
Next

MsgBox("Total count: " & maximum)
```

10 Minimum

```
Dim myList As New List(Of Integer) From {5, 6, 72, 3}
Dim minimum As Integer =  myList(0) 'Assume first value is minimum, look for smalle

For Each x In myList
    ' look for a value less than minimum
    ' it should be the new minimum
    If x < minimum Then
        minimum = x
    End If
Next

MsgBox("Total count: " & minimum)
```

11 Challenge

Create an application with the following features: - Allows instructor to enter student grades and display them as a list - Find the minimum, maximum, and average grade in class

12 Shopping List Exercise

- Create a shopping list that you add items, their quantity, and unit price to the list

- User should be able to remove items from the list
- Show the total price for the shopping list, total number of items, and average item price
- When selecting an item, the corrsponding quantity and unit price must be selected as well.
- When an item is selected, the show the total for that item.
- Allow user to update the values for selected entries.

Hint: The items that match in each list must remain in the same index across lists.
Potential problem: what if you sort the list?

```
In [ ]:
```