week7

February 21, 2018

1 Functions, Subroutines, and Methods

- Way to group statements together and run them by calling the name of the function or subroutine
- In VB, functions return values, subroutines do not
  - Functions can be used in mathematical expressions as they are replaced by the value they return
- Method is a name for subroutines and functions in OOP

2 Creating a Subroutine

```
Sub SubName()
    MSGBox("You called SubName")
End Sub
```

3 Creating a Function

```
Sub FuncName() As Integer
    MSGBox("You called FuncName")
    Return 5  'Can any value, variable, or expresstion that
              'is an integer
End Sub
```

4 Using Functions/Subroutines

- Just call the name of the Func/Sub with () to run it

```
SubName() 'will execute SubName
MSGBox("The result from FuncName is "& FuncName()) 'doesnt work
                                                   'with sub
```

5 Parameters/Arguments

- Way to pass values/variables to the function/sub
  - You include them inside the parantheses ()
- They are just variables
- Allows us to perform the actions in the Func/Sub on different values, variables, controls ..etc
- Way that different parts of the program talk to one another and pass data
  - e.g: windows sending information to your program

6 Parameter Example

```
Function Add(x As Integer, y As Integer) As Integer
    return x+y
End If

' to use it
MsgBox("result is: " & Add(5,6))
```

7 Example: Validation of TextBoxes

```
Sub ValidateTextbox1()
    if TextBox1.text = "" Then
        MsgBox("Textbox is Empty")
    End If
End Sub
```

8 Problem with ValidateTextbox1

```
ValidateTextbox1()
' The next statement should run only if input is valid
MsgBox("The Valid Input is:" & TextBox1.Text)
' Will it run?
```

9 Problem with ValidateTextbox1

- No way to prevent display of MsgBox if input is invalid
- Works only with Textbox1

10 Use Function To Fix Validation

```
Function ValidateTextbox2() As Boolean
    if TextBox1.text = "" Then
        MsgBox("Textbox is Empty")
```

```
        Return False ' If there is a problem return False
    End If
    Return True ' Return True if everything is OK!
End Sub
```

11 Use Function To Fix Validation

```
If ValidateTextbox2() Then
    ' Now it will only run if ValidateTextbox2() return True
    MsgBox("The Valid Input is:" & TextBox1.Text)
End If
```

12 Problem with ValidateTextbox2

- Fixed one problem
- Still, works only with TextBox1

13 Fixing ValidateTextbox2 With Parameters

```
Function ValidateTextbox3(ByRef txtBox As TextBox) As Boolean
    if txtBox.text = "" Then
        MsgBox("Textbox is Empty")
        Return False ' If there is a problem return False
    End If
    Return True ' Return True if everything is OK!
End Sub
```

14 Using Function with Different TextBoxes

```
If ValidateTextbox3(TextBox1) Then
    MsgBox("The Valid Input is:" & TextBox1.Text)
End If

If ValidateTextbox3(TextBox2) Then
    MsgBox("The Valid Input is:" & TextBox2.Text)
End If
```

15 Some Notes

- Use **ByRef** when you want to pass controls as parameters
  - No need to use it with other datatypes

16 Variable scope

- The parts of a program in which a variable you defined can be used
- Can be local or global

17 Local Scope

- Variable exists and can be used in specific Func/Sub
- Variables defined inside a Func/Sub have local scope
- Parameters have local scope
  - What does this mean?
- What if we need a value inside a Func/Sub to reach another Func/Sub?

18 Global Scope

- Variables exists and can be used in all Func/Sub in the file
- Variable defined inside Class/Module have global scope
  - The variable is defined outside any func/sub
- Controls you create with the designer have global scope, why?

19 Challenge

- Create an application that implements the following requirements:
  - User can calculate areas of: Square, Rectangle, Circle
  - Application will validate input of user and check it for:
    - missing values, numeric values, and correct values
  - Validation must be performed using a single function
  - Area calculations must be done using functions

20 Challenge

- Create currency conversion calculator
- Gets amount from user
- Converts amount to USD, KWD, SAR
- Use functions to validate input, and calculate converted amount