I101/B100
Problem Solving with Computers

By:

Dr. Hossein Hakimzadeh
Computer Science and Informatics
IU South Bend

(c) Copyright 2022, H. Hakimzadeh
What do we need to learn in order to write computer programs?

- Fundamental programming constructs:
  - Variables,
  - Arithmetic operators,
  - Input and output
  - Conditionals,
  - Loops,
  - Procedures and functions,
  - Arrays,
  - Structures, classes and objects,
  - Files
Procedures

Syntax:

Private Sub procedureName([OptionalParameterList])

[Statements]

End Sub
Procedure calls:

- No parameter needed
- Specification of work
- Architectural drawing
- Room number
- House Location
Procedures **without** Parameters:

- **Example:**

  ```csharp
  Private Sub DisplayMessage1()
    Console.WriteLine("Hello World")
  End Sub
  ```

  Note: This procedure has **no parameters**.
Passing Parameters to a Procedure

- Parameters can be sent to a procedure either ‘**By Value**’ or ‘**By Reference**’.

- **Pass By Value** is used if the sub-program has no reason to modify the content of the parameter.

- **Pass By Reference** is used if the sub-program is expected to modify the content of the parameter.
Passing parameters **By Value:**

- **Example:**

```csharp
Private Sub DisplayMessage2(ByVal Message As String)
    Console.WriteLine(Message)
End Sub
```

```csharp
Sub Main()
    DisplayMessage2("hello") 'Pass by value
End Sub
```
Passing Parameters **By Reference**:

- This sub procedure will read the “**User’s Name**”, and pass it back to the calling module.

```vbnet
Private Sub GetUserName(ByRef strName As String)
    Console.Write("Enter your name: ")
    strName = Console.ReadLine()
End Sub

Sub Main()
    Dim Name As String
    GetUserName(Name) 'Pass by Reference
    Console.WriteLine("Name = {0}", Name)
End Sub
```
Passing Parameters **By Reference**:

- This sub procedure will read the “**User’s Name**”, and pass it back to the calling module.

```vbnet
Private Sub GetUserName(ByVal strName As String)
    Console.Write("Enter your name: ")
    strName = Console.ReadLine()
End Sub

Sub Main()
    Dim Name As String
    GetUserName(Name)        'Pass by Reference
    Console.WriteLine("Name = {0}", Name)
End Sub
```
Passing Parameters **By Reference**:

- This sub procedure will read the “**User’s Name**”, and pass it back to the calling module.

```vbnet
Private Sub GetUserName(ByRef strName As String)
    Console.Write("Enter your name: ")
    strName = Console.ReadLine()
End Sub

Sub Main()
    Dim Name As String
    GetUserName(Name) 'Pass by Reference
    Console.WriteLine("Name = {0}", Name)
End Sub
```
Option Explicit On
Option Strict On

Module Module1

    Sub Main()
        Dim Name, Age As String
        Name = ""
        Age = ""
        GetUserInfo(Name, Age)    'Pass by Reference
        Console.WriteLine("Name = {0}, Age = {1}", Name, Age)
    End Sub

    Private Sub GetUserInfo(ByRef strName As String, ByRef strAge As String)
        Console.Write("Enter your name: ")
        strName = Console.ReadLine()
        Console.Write("Enter your age: ")
        strAge = Console.ReadLine()
    End Sub

End Module
Combining (Pass ByVal and Pass ByRef)

Option Explicit On
Option Strict On

Module Module1

Sub Main()

Dim Name, Age As String
Name = "...."
Age = "...."

GetUserInfo(Name, Age)  ' What will be the result?!!!

Console.WriteLine("Name = {0}, Age = {1}", Name, Age)
End Sub

Private Sub GetUserInfo(ByRef strName As String, ByVal strAge As String)
    Console.Write("Enter your name: ")
    strName = Console.ReadLine()
    Console.Write("Enter your age: ")
    strAge = Console.ReadLine()
End Sub

End Module
Combining Pass By Value and Reference

Option Explicit On
Option Strict On

Module Module1

Sub Main()
    Dim Name, Age As String
    Name = "...."
    Age = "...."

    GetUserInfo(Name, Age)

    Console.WriteLine("Name = {0}, Age = {1}", Name, Age)
End Sub

Private Sub GetUserInfo(ByRef strName As String, ByVal strAge As String)
    Console.Write("Enter your name: ")
    strName = Console.ReadLine()
    Console.Write("Enter your age: ")
    strAge = Console.ReadLine()
End Sub

End Module
Event Procedures

- Also known as **Event Handlers**

```vbnet
Private Sub BtnQuit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BtnQuit.Click
    Application.Exit()
End Sub
```