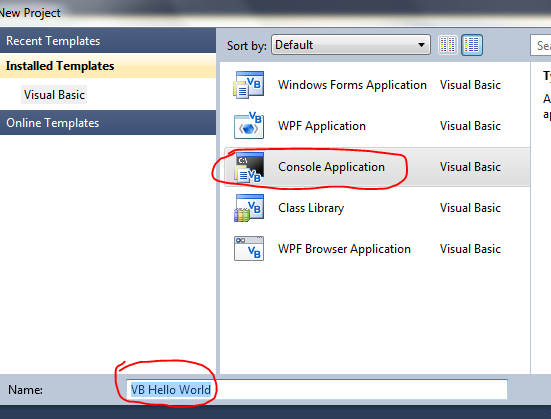
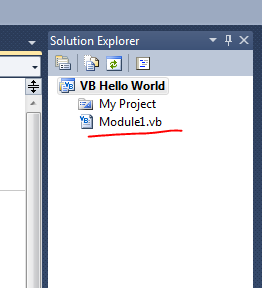
**Visual Basic -- Video Tutorials 200**

## 04 Visual Basic Tutorial - 4 - Hello World

New Project:



Tools 🡪 Settings 🡪 Expert Settings



Every program have to have a MAIN SUB.

If we do not print Console.ReadLine() then after we run the program the cmd window will disappear quickly.

Module Module1

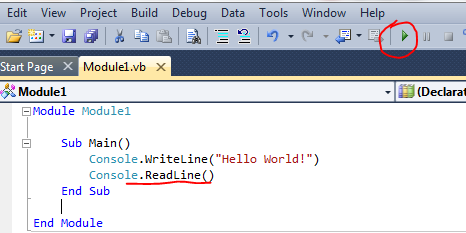
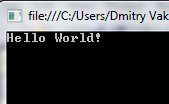
Sub Main()

Console.WriteLine("Hello World!")

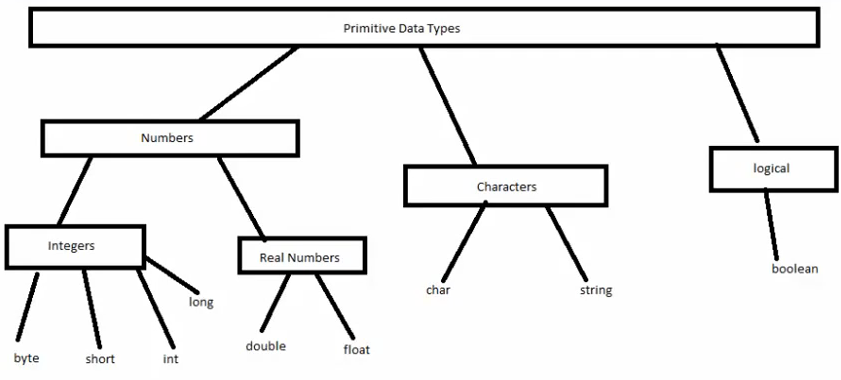
Console.ReadLine()

End Sub

End Module



## 5. Visual Basic Tutorial - 5 - Primitive Data Types



byte − 128 to 127

short −﻿ 32,768 to 32,767

int − 2,147,483,648 to 2,147,483,647

long − 9,223,372,036,854,775,808 to 9,223,372,036,854,775,807

## Visual Basic Tutorial - 6 - Comments And Whitespace

Module Module1

Sub Main()

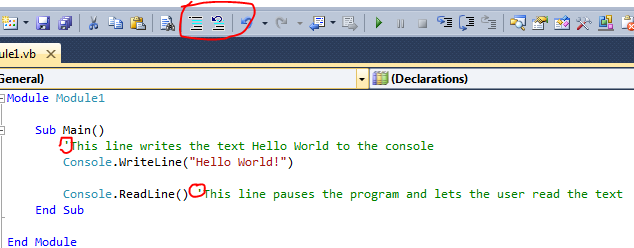
'This line writes the text Hello World to the console

Console.WriteLine("Hello World!")

Console.ReadLine() 'This line pauses the program and lets the user read the text

End Sub

End Module



## 07. Visual Basic Tutorial - 7 - Errors

**Three types of errors:**

1. Syntax errors (e.g. {} were omitted)
2. Runtime errors (for example when a program asks a user to print user name and password but they were wrong and he program gave us an error)
3. Logic errors (when a program was compiled. It runs fine but doesn’t provide the result we expected)

## 08. Visual Basic Tutorial - 8 - Variables

To declare a variable you need to start with a word “Dim”. You tell the compiler that you want to declare a variable.

Dim --- Declare and allocates storage space for one or more variables.

ToString() -- Converts numbers to numerical string representation.

It’s a good practice to convert numbers to strings before printing them out to the screen.

Sub Main()

Dim Var As Integer 'just to declare a variable

Dim myNum As Integer = 5 'defining a variable myNum

Dim myString As String = "Hello thenewboston fans!"

Dim myChar As Char = "A"

Dim myD As Double = 5.934

Dim myBoo As Boolean = True 'or False

Console.WriteLine(myNum)

Console.WriteLine(myString)

Console.WriteLine(myChar)

Console.WriteLine(myD)

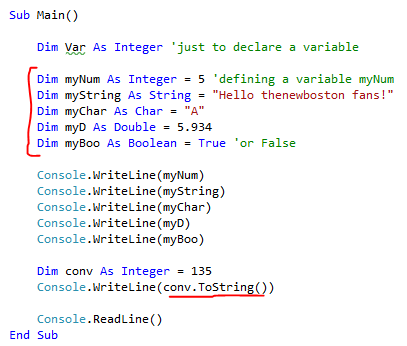
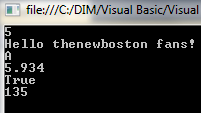
Console.WriteLine(myBoo)

Dim conv As Integer = 135

Console.WriteLine(conv.ToString())

Console.ReadLine()

End Sub



## 9. Visual Basic Tutorial - 9 - Math Operators

Sub Main()

Dim num1 As Integer = 5

Dim num2 As Integer = 3

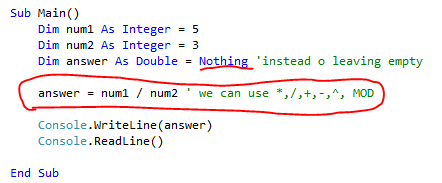
Dim answer As Double = Nothing 'instead of leaving empty

answer = num1 / num2 ' we can use \*,/,+,-,^, MOD

Console.WriteLine(answer)

Console.ReadLine()

End Sub



## 10. Visual Basic Tutorial - 10 - More On Math Operators

Sub Main()

Dim myDouble As Double = 11

Console.WriteLine("Initial Value: 11 New Value: " & myDouble)

myDouble += 23 'the same as myDouble = myDouble + 23

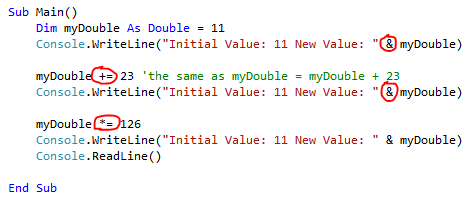
Console.WriteLine("Initial Value: 11 New Value: " & myDouble)

myDouble \*= 126

Console.WriteLine("Initial Value: 11 New Value: " & myDouble)

Console.ReadLine()

End Sub





## 11. Visual Basic Tutorial - 11 - Getting User Input

Sub Main()

Dim myName As String = Nothing 'in case a user will not input it

Dim myAge As Integer = Nothing

Dim mySalary As Double = Nothing

Console.WriteLine("What is your name?")

myName = Console.ReadLine()

Console.WriteLine("What is your age?")

myAge = Console.ReadLine()

Console.WriteLine("What is your salary?")

mySalary = Console.ReadLine()

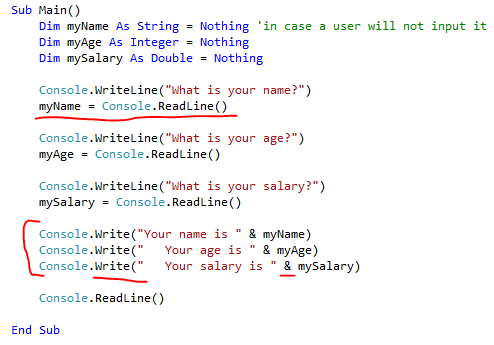
Console.Write("Your name is " & myName)

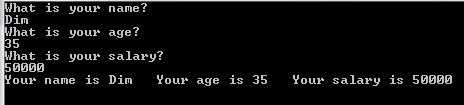
Console.Write(" Your age is " & myAge)

Console.Write(" Your salary is " & mySalary)

Console.ReadLine()

End Sub





## 12. Visual Basic Tutorial - 12 - Creating A Basic Calculator

Sub Main()

Console.Write("Enter your first number: ")

Dim num1 As Double = Console.ReadLine()

Console.WriteLine("First number: " & num1)

Console.Write("Enter your second number: ")

Dim num2 As Double = Console.ReadLine()

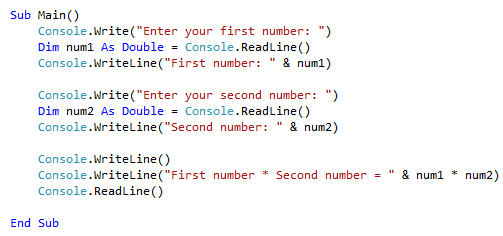
Console.WriteLine("Second number: " & num2)

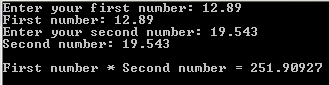
Console.WriteLine()

Console.WriteLine("First number \* Second number = " & num1 \* num2)

Console.ReadLine()

End Sub





## 13. Visual Basic Tutorial - 13 - If Statement

Sub Main()

Console.Write("Enter your username: ")

Dim userName As String = Console.ReadLine()

Console.Write("Enter your password: ")

Dim password As String = Console.ReadLine()

Console.WriteLine()

If userName = "Dmitry" Then

Console.WriteLine("Hello Dmitry!")

End If

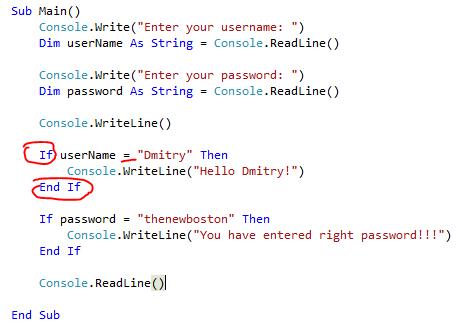
If password = "thenewboston" Then

Console.WriteLine("You have entered right password!!!")

End If

Console.ReadLine()

End Sub



## 14. Visual Basic Tutorial - 14 - Else If And Else

Sub Main()

Console.Write("Enter your name: ")

Dim userName As String = Console.ReadLine()

Console.Write("Enter your password: ")

Dim password As String = Console.ReadLine

If userName = "Dmitry" Then

Console.WriteLine("Welcome Dmitry!!!")

ElseIf userName = "Sam" Then

Console.WriteLine("Hello Sam!")

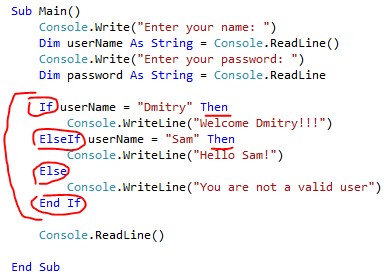
Else

Console.WriteLine("You are not a valid user")

End If

Console.ReadLine()

End Sub



## 15. Visual Basic Tutorial - 15 - Nested If Statements

Sub Main()

Dim Age As Integer = Nothing

Dim hasInsurance As Boolean = Nothing

Console.Write("Enter your age: ")

Age = Console.ReadLine()

Console.Write("Do you have insurance true/false: ")

hasInsurance = Console.ReadLine()

If Age >= 16 Then

If hasInsurance = True Then

Console.WriteLine("You can drive")

Else

Console.WriteLine("You'd not better stopped by cops")

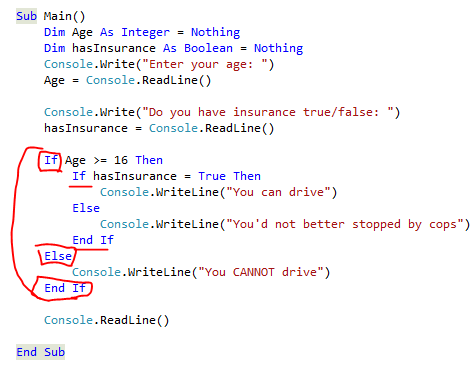
End If

Else

Console.WriteLine("You CANNOT drive")

End If

Console.ReadLine()



## 16. Visual Basic Tutorial - 16 - Conditional Operators

<> -- not equal

>=

<=

>

<

## 17. Visual Basic Tutorial - 17 - Logical Operators

Sub Main()

Dim userName As String = Nothing

Dim password As String = Nothing

Console.Write("Enter your name: ")

userName = Console.ReadLine()

Console.Write("Enter your password: ")

password = Console.ReadLine()

If (userName = "Sam" Or userName = "Tim") And password = "thenewboston" Then

Console.WriteLine("Welcome " & userName)

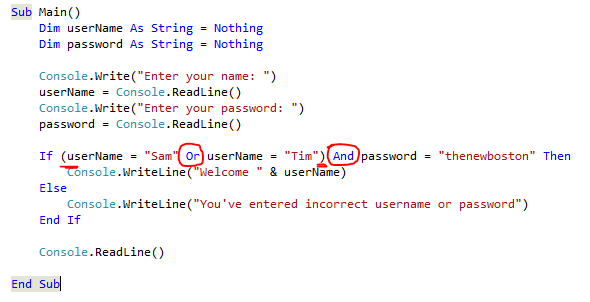
Else

Console.WriteLine("You've entered incorrect username or password")

End If

Console.ReadLine()

End Sub



## 18. Visual Basic Tutorial - 18 - Concatenate Strings

Use “&” in the console command

Use “+” in the variable command

Sub Main()

Dim userString As String = Nothing

Dim programString As String = " catfish"

Console.Write("Enter what you want: ")

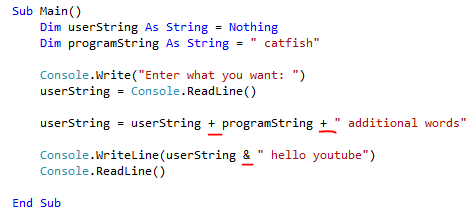
userString = Console.ReadLine()

userString = userString + programString + " additional words"

Console.WriteLine(userString & " hello youtube")

Console.ReadLine()

End Sub



## 19. Visual Basic Tutorial - 19 - Get Length Of Strings

Sub Main()

Dim userName As String = Nothing

Console.WriteLine("What is your user name?")

userName = Console.ReadLine()

If userName.Length <= 10 Then

Console.WriteLine("You have been granted access!")

Else

Console.WriteLine("You user name is not right length")

End If

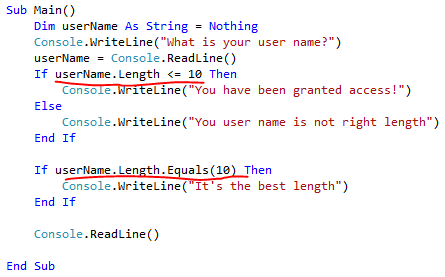
If userName.Length.Equals(10) Then

Console.WriteLine("It's the best length")

End If

Console.ReadLine()

End Sub



We can use both:

* userName.Length <= 10
* userName.Length.Equals(10)

## Visual Basic Tutorial - 20 - SubStrings

Sub Main()

Dim userString As String = Nothing

Console.WriteLine("Enter a string")

userString = Console.ReadLine()

Console.WriteLine()

Console.WriteLine(userString.Length.ToString())

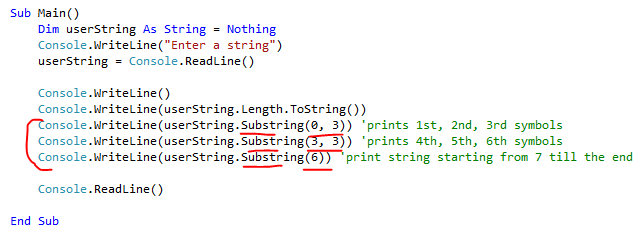
Console.WriteLine(userString.Substring(0, 3)) 'prints 1st, 2nd, 3rd symbols

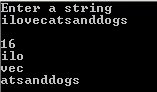
Console.WriteLine(userString.Substring(3, 3)) 'prints 4th, 5th, 6th symbols

Console.WriteLine(userString.Substring(6)) 'print string starting from 7 till the end

Console.ReadLine()

End Sub





## 21. Visual Basic Tutorial - 21 - Formatting Strings

Formatting Decimal and String output

Sub Main()

Console.WriteLine("Enter your string")

Dim myString As String = Console.ReadLine()

Console.WriteLine("Enter your decimal value")

Dim myDouble As Double = Console.ReadLine()

Console.WriteLine()

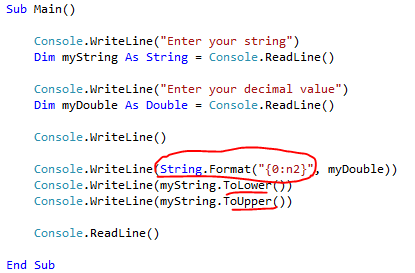
Console.WriteLine(String.Format("{0:n2}", myDouble))

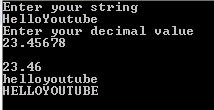
Console.WriteLine(myString.ToLower())

Console.WriteLine(myString.ToUpper())

Console.ReadLine()

End Sub





## 22. Visual Basic Tutorial - 22 - Replacing SubStrings

Sub Main()

Dim myString As String = Nothing

Dim finalString As String = Nothing

Console.WriteLine("Enter your string")

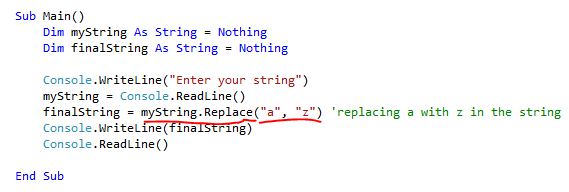
myString = Console.ReadLine()

finalString = myString.Replace("a", "z") 'replacing a with z in the string

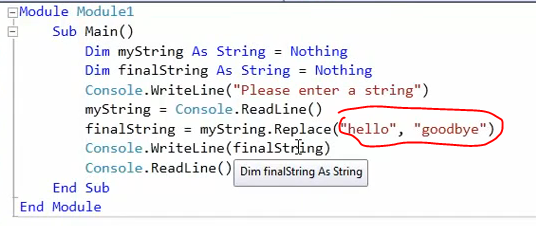
Console.WriteLine(finalString)

Console.ReadLine()

End Sub







## 23. Visual Basic Tutorial - 23 - Comparing Strings

String.Compare(str1, str2)

this﻿ compares string 1 and string 2 and tells you if string1 goes before or after string 2 when sorted in alphabetical order. Lowercase characters go before uppercase characters in the sort order, so passing true as a 3rd argument ignores casing

-1 = it should go before it

0 = they are the same string

1 = it should go after

so String.Compare("zzzz", "aaaa") will give you 1 because "zzzz" goes after "aaaa"

Sub Main()

Dim userString As String = Nothing

Dim compString As String = "OnliveGamer"

Console.WriteLine("Enter your string")

userString = Console.ReadLine()

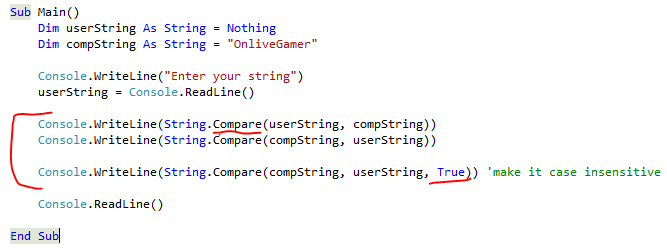
Console.WriteLine(String.Compare(userString, compString))

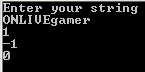
Console.WriteLine(String.Compare(compString, userString))

Console.WriteLine(String.Compare(compString, userString, True)) 'make it case insensitive

Console.ReadLine()

End Sub





## 24. Visual Basic Tutorial - 24 - Select Case

As it executes one particular CASE it doesn’t go to check other cases.

While in IF statement it will check all others.

Sub Main()

Dim myInt As Integer = Nothing

Console.WriteLine("Enter an Integer")

myInt = Console.ReadLine()

Select Case myInt

Case 0

Console.WriteLine("Hello")

Case 1

Console.WriteLine("Bye")

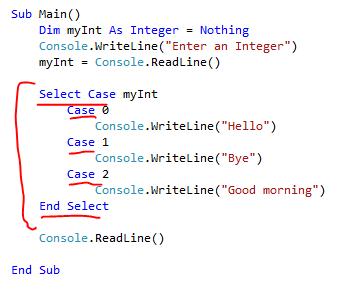
Case 2

Console.WriteLine("Good morning")

End Select

Console.ReadLine()

End Sub



## 25. Visual Basic Tutorial - 25 - Select Case Else

Sub Main()

Dim myString As String = Nothing

Console.WriteLine("Enter you string")

myString = Console.ReadLine()

Select Case myString.ToLower

Case "hello"

Console.WriteLine("Goodbye")

Case "fishing"

Console.WriteLine("Boat")

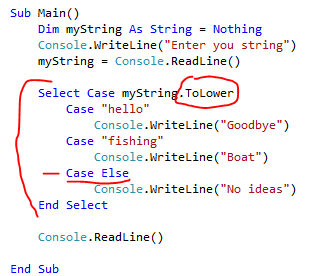
Case Else

Console.WriteLine("No ideas")

End Select

Console.ReadLine()

End Sub



## 26. Visual Basic Tutorial - 26 - For Next Loop

Sub Main()

Dim int1 As Integer

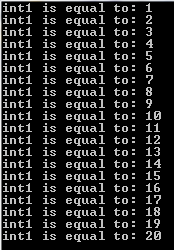
For int1 = 1 To 20

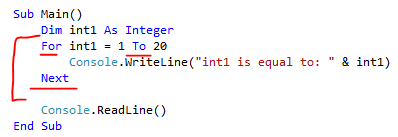
Console.WriteLine("int1 is equal to: " & int1)

Next

Console.ReadLine()

End Sub





## 27. Visual Basic Tutorial - 27 - Step Operator

Sub Main()

Console.WriteLine("Normal For Loop")

For num1 = 1 To 20

Console.WriteLine(num1)

Next

Console.WriteLine()

Console.WriteLine("Step 5 Loop")

For num2 = 1 To 20 Step 5

Console.WriteLine(num2)

Next

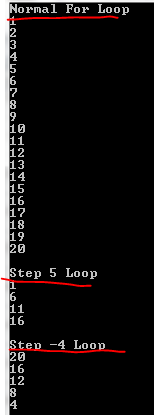
Console.WriteLine()

Console.WriteLine("Step -4 Loop")

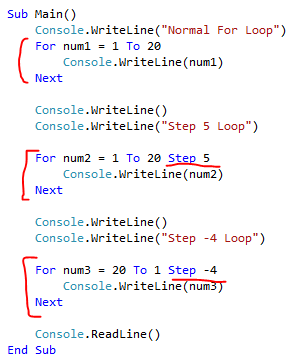
For num3 = 20 To 1 Step -4

Console.WriteLine(num3)

Next

 Console.ReadLine()

End Sub



## 28. Visual Basic Tutorial - 28 - Exiting For Loop

Sub Main()

Console.WriteLine("Exiting For Loop")

For num1 = 1 To 30 Step 4

If num1 > 23 Then

Exit For

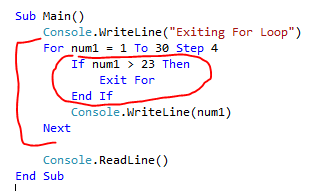
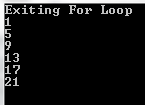
End If

Console.WriteLine(num1)

Next

Console.ReadLine()

End Sub



## 29. Visual Basic Tutorial - 29 - Continue For

It’s a way to skip some values from execution in the For Loop

It skips 5, 8, and 12 from printing.

Sub Main()

For num1 = 1 To 15

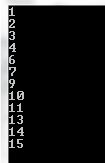
If num1 = 5 Or num1 = 8 Or num1 = 12 Then Continue For

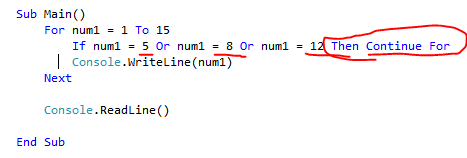
Console.WriteLine(num1)

Next

Console.ReadLine()

End Sub





## 30. Visual Basic Tutorial - 30 - Do Until Loop

Use with “=” sign.

Sub Main()

Dim num1 As Integer = 0

Do Until num1 = 10

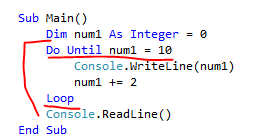
Console.WriteLine(num1)

num1 += 2

Loop

Console.ReadLine()

End Sub



## 31. Visual Basic Tutorial - 31 - Do While Loop

Use with “<, >, <=, >=” signs.

Sub Main()

Dim num1 As Integer = 0

Do While num1 <= 15

Console.WriteLine(num1)

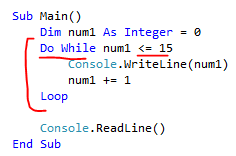
num1 += 1

Loop



Console.ReadLine()

End Sub



## 32. Visual Basic Tutorial - 32 - Exit Do Loops

Sub Main()

Dim num1 As Integer = 0

Do Until num1 = 15

If num1 = 5 Then Exit Do

Console.WriteLine(num1)

num1 += 1

Loop

Console.WriteLine()

num1 = 0

Do While num1 <= 15

If num1 = 7 Then Exit Do

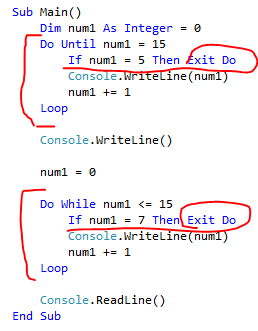
Console.WriteLine(num1)

num1 += 1

Loop

Console.ReadLine()

End Sub



## 33. Visual Basic Tutorial - 33 - More On Do Loops

We can make Do While or Do Until Loops run at least once.

Sub Main()

Dim num1 As Integer = 0

Console.WriteLine("Normal Do Loop")

Do Until num1 = 5

Console.WriteLine(num1)

num1 += 1

Loop

num1 = 10

Console.WriteLine()

Console.WriteLine("Run at least once")

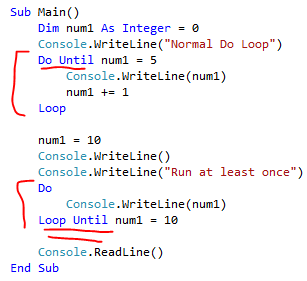
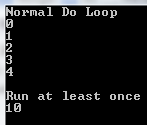
Do

Console.WriteLine(num1)

Loop Until num1 = 10

Console.ReadLine()

End Sub



## 34. Visual Basic Tutorial - 34 - Nested Loops

Sub Main()

For num1 = 1 To 10

For num2 = 1 To 10

Console.Write(num2)

Console.Write(" ")

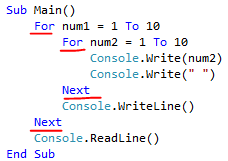
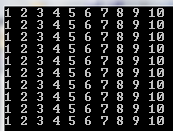
Next

Console.WriteLine()

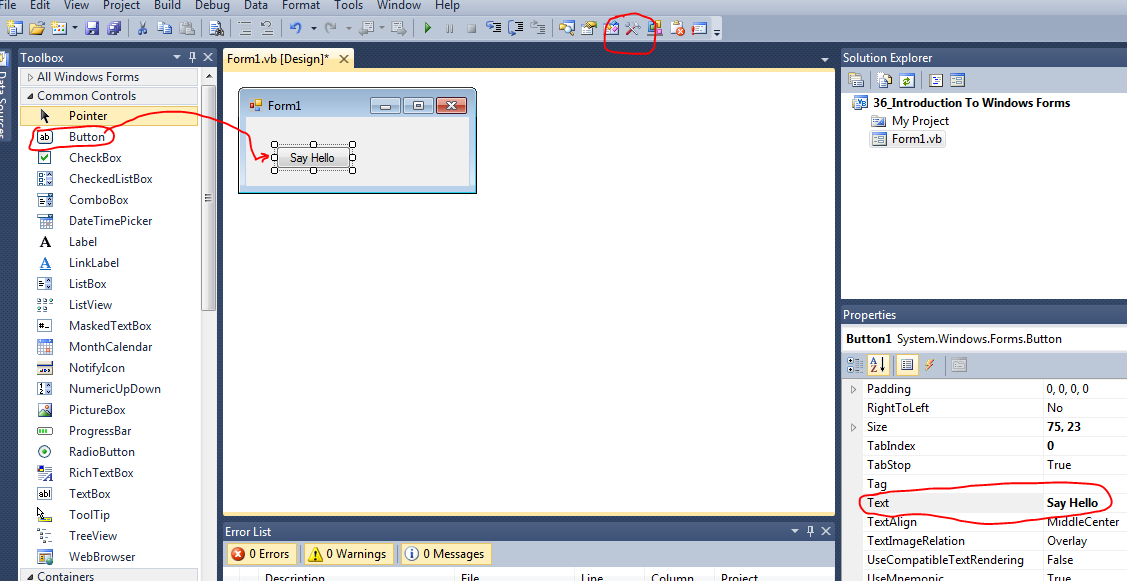
Next

Console.ReadLine()

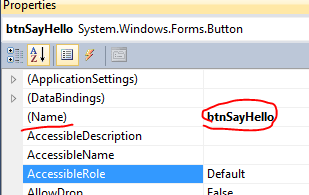
End Sub

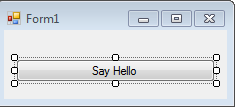


## 36. Visual Basic Tutorial - 36 - Introduction To Windows Forms

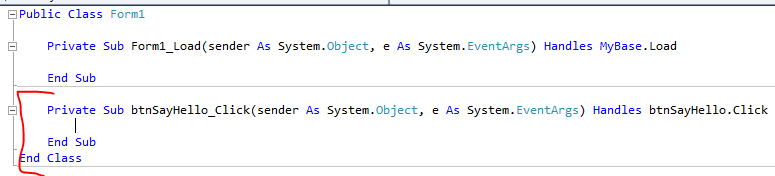


**Change the button name in order not to be confused in the future.**

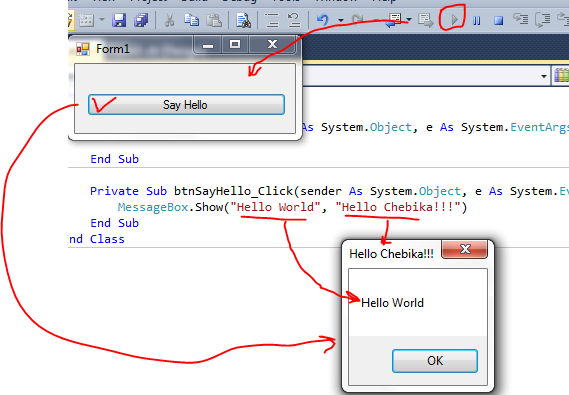




Double click on Say Hello button.

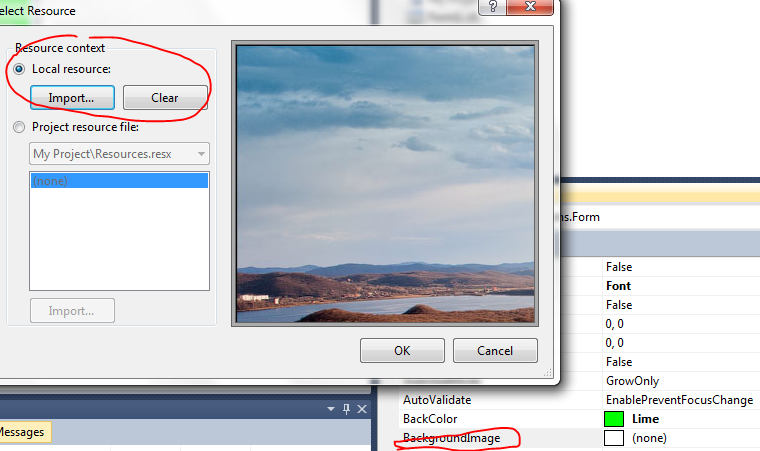


Run the program. The Form1 will appear. Press the “Say Hello” button.



## 37. Visual Basic Tutorial - 37 - Form Properties







## 38. Visual Basic Tutorial - 38 - ToolBox

## 39. Visual Basic Tutorial - 39 - MessageBoxes

To fill the full form with a button





Public Class btnShowMessage

Private Sub Button1\_Click(sender As System.Object, e As System.EventArgs) Handles Button1.Click

MessageBox.Show("This is a bare message")

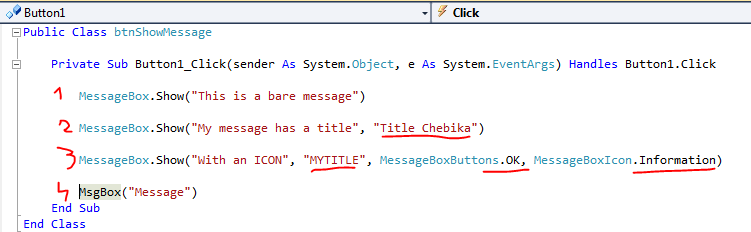
MessageBox.Show("My message has a title", "Title Chebika")

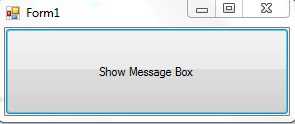
MessageBox.Show("With an ICON", "MYTITLE", MessageBoxButtons.OK, MessageBoxIcon.Information)

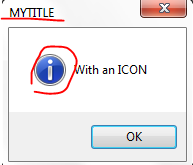
MsgBox("Message")

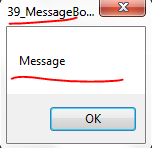
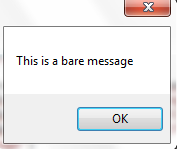
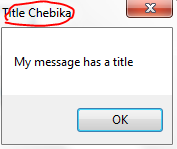
End Sub

End Class









## 40. Visual Basic Tutorial - 40 - MessageBox Input

Run a form.

If a user selects the”Abort” option then the “Aborted” message will show up.

It will show nothing for other options.

1 -- Message in the dialog box

2 -- Title of the dialog box

3 -- buttons in the dialog box

4 -- icon in the dialog box

Public Class Form1

Private Sub Button1\_Click(sender As System.Object, e As System.EventArgs) Handles Button1.Click

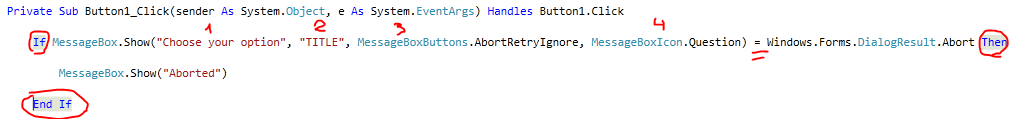
If MessageBox.Show("Choose your option", "TITLE", MessageBoxButtons.AbortRetryIgnore, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Abort Then

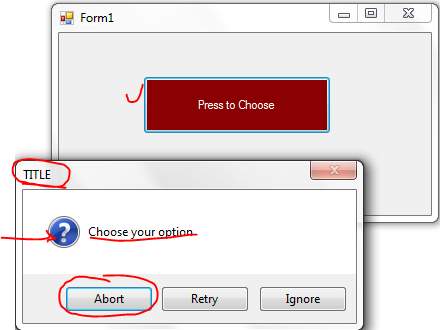
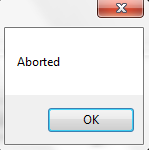
MessageBox.Show("Aborted")

End If

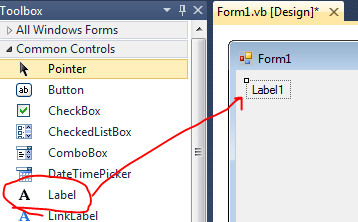
End Sub

End Class

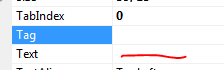




## 41. Visual Basic Tutorial - 41 - Input Box







Public Class Form1

Private Sub Form1\_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load

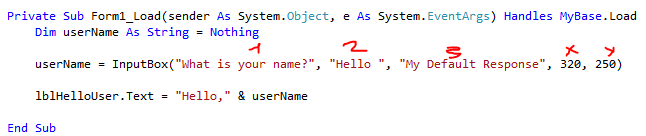
Dim userName As String = Nothing

userName = InputBox("What is your name?", "Hello ", "My Default Response", 320, 250)

lblHelloUser.Text = "Hello," & userName

End Sub

End Class

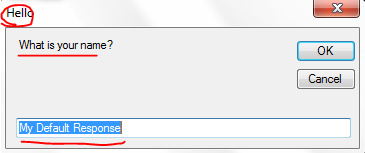


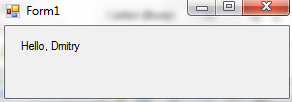
1 -- Message in the inbox

2 -- Title

3 -- Default message

4 (X,Y) -- coordinates on the screen





We use it as we created a Lable in the very beginning and named it **lblHelloUser**



## 42. Visual Basic Tutorial - 42 - User Defined Subs

Public Class Form1

Private Sub btnRunSub\_Click(sender As System.Object, e As System.EventArgs) Handles btnRunSub.Click

addNumbers()

End Sub

'Private means that this sub cannot be accessed by any other class

'outside of the form Form1.vb

Private Sub addNumbers()

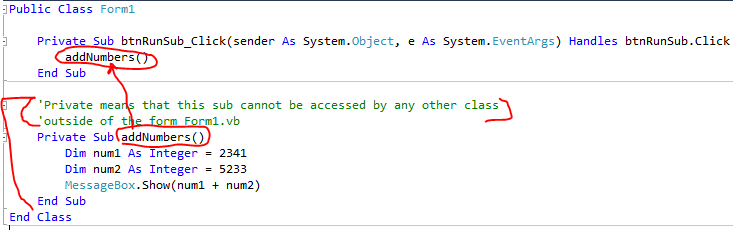
Dim num1 As Integer = 2341

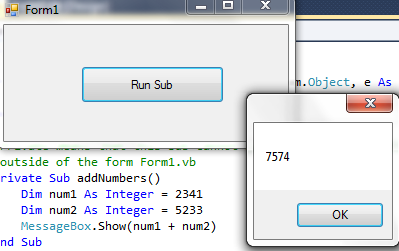
Dim num2 As Integer = 5233

MessageBox.Show(num1 + num2)

End Sub

End Class





## 43. Visual Basic Tutorial - 43 - Functions

## Functions return the value.

Public Class Form1

Private Sub btnRunFunction\_Click(sender As System.Object, e As System.EventArgs) Handles btnRunFunction.Click

Dim answer As Double = solveMath() 'assigning the returning value to the variable

MessageBox.Show(answer)

End Sub

Private Function solveMath() As Double

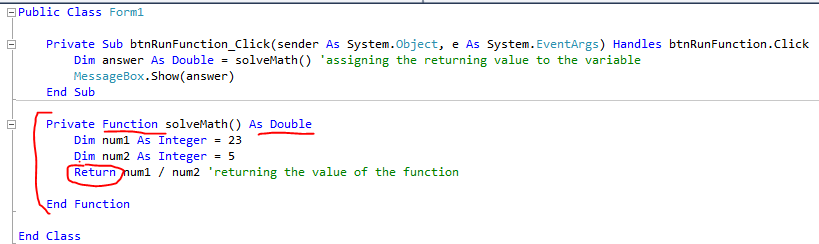
Dim num1 As Integer = 23

Dim num2 As Integer = 5

Return num1 / num2 'returning the value of the function

End Function

End Class



## 44. Visual Basic Tutorial - 44 - ByVal

Giving your function or sub a variable to use from the another part of the program. But ths value will not be changed outside of the function

Public Class Form1

Private Sub Button1\_Click(sender As System.Object, e As System.EventArgs) Handles btnSubtract.Click

Dim answer As Double = subtractNumbers(TextBox1.Text, TextBox2.Text)

MessageBox.Show(answer)

End Sub

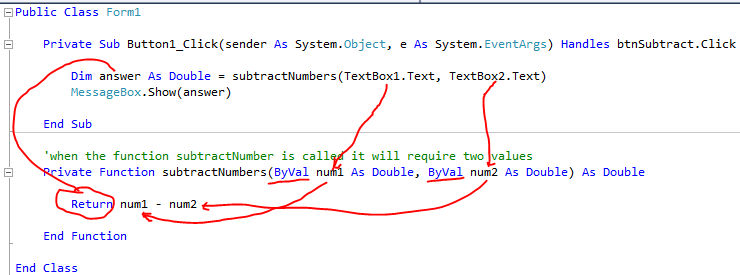
'when the function subtractNumber is called it will require two values

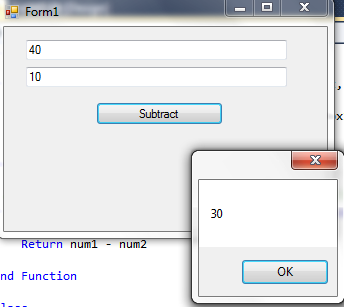
Private Function subtractNumbers(ByVal num1 As Double, ByVal num2 As Double) As Double

Return num1 - num2

End Function

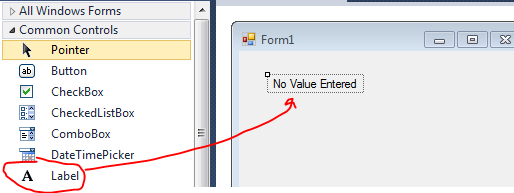
End Class





## 45. Visual Basic Tutorial - 45 - ByRef

You can reference a variable that is outside of your sub or function and change it.





Public Class Form1

Private Sub Form1\_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load

Dim num1 As Integer = 0

MessageBox.Show(num1.ToString())

lblVariable.Text = num1.ToString()

incrementVariable(num1)

MessageBox.Show(num1.ToString())

lblVariable.Text = num1.ToString

End Sub

Private Sub incrementVariable(ByRef x As Integer)

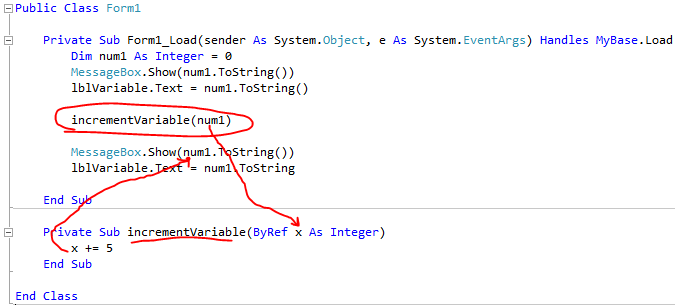
x += 5

End Sub

End Class

We call the Sub incrementVariable using the num1 variable.

This function will take the argument and add 5 to it.



## 46. Visual Basic Tutorial - 46 - Optional ByVal

Public Class Form1

Private Sub btnShowMessage\_Click(sender As System.Object, e As System.EventArgs) Handles btnShowMessage.Click

Dim mesg As String = txtMessage.Text 'it will take the input from the second field

Dim title As String = Nothing

'we have two forms: txtTitle and txtMessage

If txtTitle.TextLength > 0 Then

title = txtTitle.Text

showMessage(mesg, title) 'running a sub with message and title

Else

showMessage(mesg) 'running a sub with message and DEFAULT Title

End If

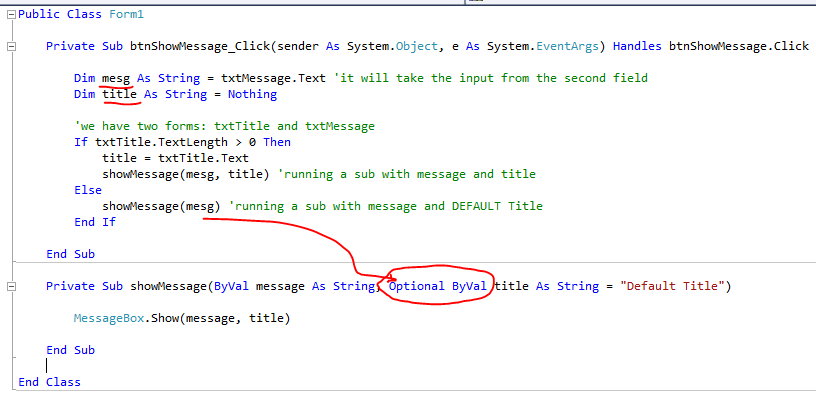
End Sub

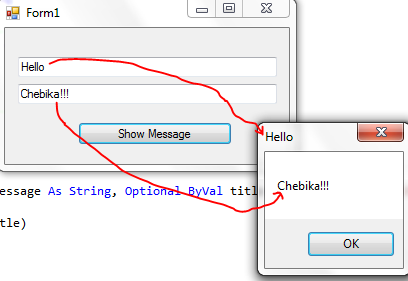
Private Sub showMessage(ByVal message As String, Optional ByVal title As String = "Default Title")

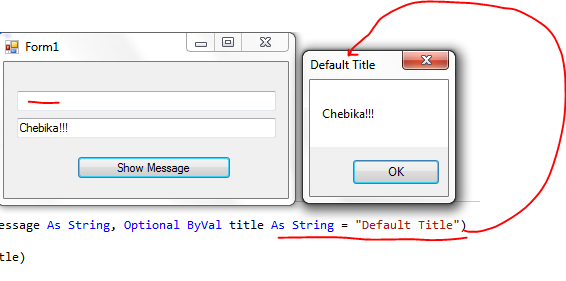
MessageBox.Show(message, title)

End Sub

End Class







## 47. Visual Basic Tutorial - 47 - Coercion

Visual Basic will automatically select which sub will work best for you.

We have two subs (**showDataType**) with the same name but the first is using an Integer as an argument while the second uses a Double.

When we assign a value to the myNum variable (let's say as Double = 23.56) and then call the subshoDataType the VB will automatically call the sub that works with double values.

Public Class Form1

Private Sub btnCoercion\_Click(sender As System.Object, e As System.EventArgs) Handles btnCoercion.Click

'Dim myNum As Integer = 23

Dim myNum As Double = 23.56

showDataType(myNum)

End Sub

Private Sub **showDataType**(ByVal num As Integer)

MessageBox.Show("This is an integer")

End Sub

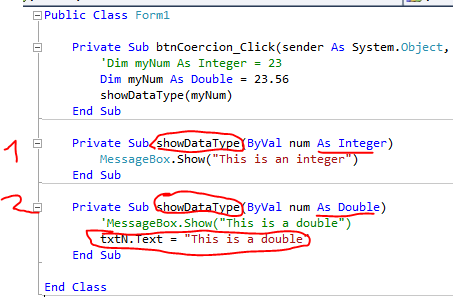
Private Sub **showDataType**(ByVal num As Double)

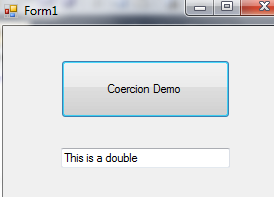
'MessageBox.Show("This is a double")

txtN.Text = "This is a double"

End Sub

End Class





## 48. Visual Basic Tutorial - 48 - Exit Subs

Public Class Form1

Private Sub btnExitSub\_Click(sender As System.Object, e As System.EventArgs) Handles btnExitSub.Click

Dim number As Integer = 10

count(number)

End Sub

Private Sub count(ByVal num As Integer)

While True

listNumbers.Items.Add(num)

num += 1

If num > 54 Then

Exit Sub

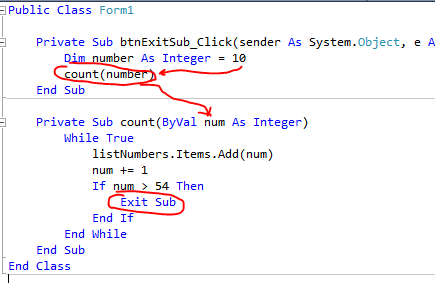
End If

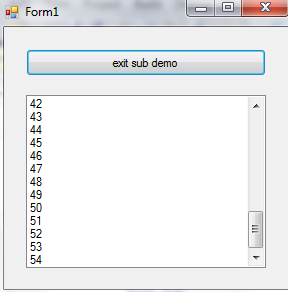
End While

End Sub

End Class

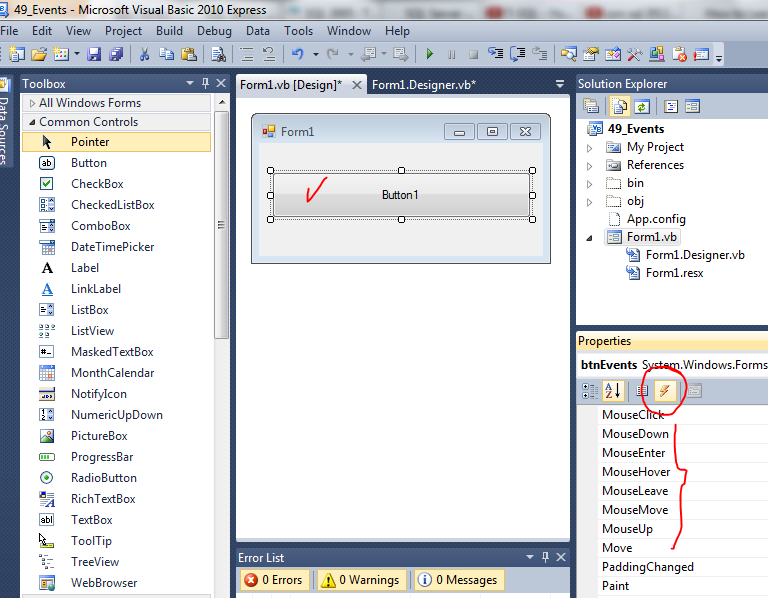
Call the "count" sub with the starting value of 10. Exit the sub when reaches the value of 54.

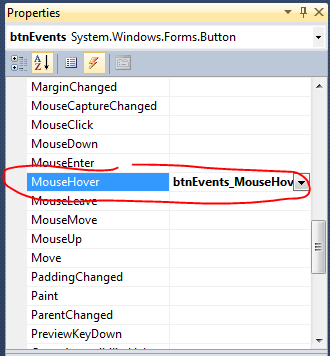




## 49. Visual Basic Tutorial - 49 - Events

Events that could be assigned to the form





Public Class Form1

Private Sub btnEvents\_Click(sender As System.Object, e As System.EventArgs) Handles btnEvents.Click

MessageBox.Show(sender.ToString()) 'checking what "sender" is in the argument

MessageBox.Show(e.ToString()) ' checking what "e" is in the argument

End Sub

Private Sub btnEvents\_MouseHover(sender As System.Object, e As System.EventArgs) Handles btnEvents.MouseHover

btnEvents.Text = "Text changed"

'MessageBox.Show(e.ToString())

End Sub

End Class

