Aim
- To understand the purpose of variables and datatypes
- To understand and use symbolic constants and literals in programs
- To write VB.NET code that uses local and module level variables
- To understand the purpose of an InputBox and MessageBox.Show
- To be able to create your own InputBox and MessageBox

Exercise 1 (Should be completed before the class) (30mins for tutor assistance)

a) Explain what is meant by Option Explicit On and Option Strict On when writing VB.NET programs.

b) State the most suitable VB.NET datatype for each of the following:

i) daily rainfall

ii) a person’s name

iii) if you belong to a sport’s club

iv) a mobile number

c) A part-time waitress earns a salary based on the number of hours she works, and is paid at a fixed rate. Given the expression below that calculates the waitress’s net salary:

\[ Salary = 0.75 \times \text{fixed hourly rate} \times \text{hours worked} \]

Identify any literal constants, symbolic constants and variables.
Exercise 2 (30mins)

![User Interface to swap values](image)

- **a)** Draw a TOE chart for the above user interface.
- **b)** Design a similar user interface to that given in Figure 1, but which follows Window’s GUI standards.
- **c)** Write VB.NET code behind the Exit and Clear command buttons. Insert inline documentation.
- **d)** Write code behind the swap button that swaps integers. Remember to turn **Option Explicit On** and **Option Strict On**.
- **e)** Modify the code so that it can swap names.
- **f)** Modify the code so that it can swap single values.

**NOTE:** You will need to use the **Convert Class**.
Exercise 3 - Variables and Constants; local, module-level (30mins)

a) Design a similar user interface to that given in Figure 2, but which follows Window’s GUI standards.
b) Name the buttons, btnExit, btnClear, btnArea, btnPerim
c) Write the VB.NET code behind the Exit and Clear buttons.
d) Write the code behind the Area button to computes the area of a circle $A = \pi r^2$. Insert inline documentation and test it. [Remember to use Math.PI]
e) Identify any local variables, symbolic constants and literal constants that are used. List their names and purpose.
f) Write the code behind the Perimeter button. Insert inline documentation and test it.
Exercise 4: Consider the following program written in VB.NET and then answer the questions that follow on the next page. (30 minutes)

```vbnet
Public Class Form1
    Inherits System.Windows.Forms.Form

    Dim secret As String
    Dim counter As Integer = 0

    Private Sub Form1_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Load
        secret = "Apple"
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        secret = "Frog"
        counter = counter + 1
    End Sub

    Private Sub Blah(ByVal a As Integer)
        Dim secret As String
        secret = "House"
        a = a * 2
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        Dim magic_number As Integer = 9
        magic_number = magic_number + 1
        counter = counter + 1
        ListBox1.Items.Add(secret + ", " + magic_number.ToString())
        If counter >= 3 Then
            Blah(magic_number)
            counter = 0
        End If
    End Sub
End Class
```

Image showing the layout of controls for the program:
a) Write the total contents of the ListBox after the buttons have been clicked in the following order:
   Button2, Button2, Button1, Button2, Button1, Button2

b) What type of scope does the variable **counter** have?

c) What type of scope does the variable **magic_number** have?

d) Upon conclusion of the **Blah** sub procedure for the first time, what value is stored in **a**? How come this is the value stored in **a** at that point?

What is the value of **counter** after completing the above clicking sequence?