- **Attendance/download Day02 from D2L**

- **PowerPoint with Illustrations:**
  - **New objects**
    - **Horizontal & Vertical TrackBars**
    - **NumericUpDown**
    - **CheckBox**
  - **New procedure and function**
    - **Random number generation**
      - **Randomize()**
      - **Rnd()**

- **Demo Problem: Bouncing Ball**

- **Practice Problem: Moving Letter**

- **REMINDER: FOR EACH NEW PROJECT:**
  - **Launch Visual Basic Express 2010**
  - **Create New Project**
  - **Use Toolbox to create objects in form/window class**
  - **Use Properties window to set object properties, such as text, color, size, and location**
  - **Save in folder for day (Day02, Day03, etc.)**
  - **Use Code window to attach methods to objects**
  - **Run/debug project/program**

- Launch Visual Basic Express 2010
- Create New Project
- Use Toolbox to create objects in form/window class

- **Form1 (the window) is an object automatically created**
- **Use Toolbox to create objects in form/window class**
    - **Label (for Ball)**
    - **Two Buttons (for Go & Exit)**
    - **Horizontal & Vertical TrackBars**
    - **Two NumericUpDowns (for Speed and Ball character)**
    - **CheckBox (for random motion)**
    - **Two more labels (for NumericUpDowns**
    - **Two Timers (for Clock and flicker effect)**

- **Use Properties window to set object properties, such as name, text, color, size, and location**

If the Properties window is not showing, use View Menu – Properties or right click on the form or one of its objects and select Properties

- **For the form itself, let the name remain as form1.**
- **For the Text property, enter**
  - **Bouncing Ball**
  - **This becomes the caption in the title bar**
- **Change the Size property to 800,600**
- **For the Color property, select BackColor, Web colors tab, then Yellow**

- Save in new folder called Demos
- EXTREMELY IMPORTANT:
  - You have ONLY ONE chance to name it!
  - Name it BouncingBall

- **Go to the Toolbox to get objects into your window**
  - **Drag a Label from the Toolbox to your window**
  - **Change properties**
    - **Name = Ball**
    - **Color & Text do not matter – they will be set by code (default will be black but flicker will alternate between Red and Blue)**
    - **Font = Bold 72**
    - **Change Autosize to False**
      - **This normally is used to fit the label's text**
    - **Size =100,100 (a ball, not an ellipse)**

- **More objects from the Toolbox**
    - **Drag two Button from the Toolbox to your window**
    - **Change properties**
        - **Names = btnExit and btnGoPause**
        - **Texts = EXIT and Go**
        - **btnGoPause is so named because its Text will toggle between Go & PAUSE**
        - **Font = Bold 14**
        - **Size: drag the handles to resize so the text fits!**
        - **Location: drag the buttons**

- **Two TrackBars**
    - **Drag two Trackbars from the Toolbox**
    - **Change properties**
        - **Name = htbEastWest & vtbNorthSouth**
        - **Orientation = horizontal & vertical**
        - **Maximum = 10 & minimum = -10**
        - **Location: drag the Trackbars as needed**

- **Two NumericUpDowns**
  - **Drag two NumericUpDowns from the Toolbox**
  - **Change properties**
    - **Names = nudInterval & nudBall**
    - **Maximum = 255 & minimum = 1**
    - **Value = 100 and 7**
    - **Location: drag the NumericUpDowns as needed**

- **One CheckBox**
  - **Drag one CheckBox from the Toolbox**
  - **Change properties**
    - **Name = chkRandom**
    - **Checked: default is False (unchecked)**
    - **Font = Bold 16**
    - **Text = Random**
    - **Location: drag the Checkbox as needed**

- Two Labels
  - Drag two Labels from the Toolbox
  - Change properties
    - Name: no change required – generic names Label1 & Label 2 are OK – not used in code
    - Font = *Bold Italic 12*
    - Text = *SPEED* and *"BALL"* : use quotes to emphasize that what we call the Ball can actually be any symbol given by its character code number
    - Location: drag the Labels as needed

- **Two Timers**
  - **Drag two Timers from the Toolbox to your window**
  - **They automatically positions BELOW your window, out of sight, in the Component Tray**
  - **Change properties**
    - **Names = tmrClock & tmrFlicker**
    - **Intervals = 100 (milliseconds) & 200**
      - **Reminder: this is how often the timers "tick" when their Enabled properties are set to be True**

Now use View Menu – Code in order to write the code for the project
•**Reminder: you will attach methods to each of the objects in one of two ways:**
  •**In the design window, double-click the object**
  •**In the code window, pull down the object name on the left and the method name on the right**

```vb
Public Class Form1

    Private Sub btnExit_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub
End Class
```

Above the btnExit code but below the declaration of form1 as a class, you need to declare two four global variables:

```
Public Class Form1
    Dim EastWest, NorthSouth As Integer
    Dim ewChange, nsChange As Integer

    Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
```

"Dim" reserves space, or dimension, in this case for integers, which can be referred to by name throughout the form1 class. "As" is also a key word. Let Intellisense help you with this!

```vb
 Private Sub Form1_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
        Ball.Text = Chr(7)
        Randomize() 'random seed for random number generator
Rnd()

 End Sub
```

You get to this section of code by either double-clicking the form or pulling down Form1 Events & Load. This code executes BEFORE anything else happens! On many computers, the 7 is the numerical code for a ball. Randomize() makes sure that no two program executions are identical.

```
Private Sub htbEastWest_Scroll(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
htbEastWest.Scroll
        EastWest = htbEastWest.Value
End Sub

Private Sub vtbNorthSouth_Scroll(ByVal sender As Object,
ByVal e As System.EventArgs) Handles vtbNorthSouth.Scroll
        NorthSouth = vtbNorthSouth.Value
End Sub
```

```
 Private Sub nudInterval_ValueChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
nudInterval.ValueChanged
        tmrClock.Interval = nudInterval.Value

 End Sub


 Private Sub nudBall_ValueChanged(ByVal sender As Object,
ByVal e As System.EventArgs) Handles nudBall.ValueChanged
        Ball.Text = Chr(nudBall.Value)
 End Sub
```

```vbnet
 Private Sub btnGoPause_Click(ByVal sender As Object, ByVal
e As System.EventArgs) Handles btnGoPause.Click
        If tmrClock.Enabled = False Then
            tmrClock.Enabled = True 'toggle
            btnGoPause.Text = "PAUSE"
            tmrFlicker.Enabled = True 'toggle
        Else
            tmrClock.Enabled = False
            btnGoPause.Text = "Go"
            tmrFlicker.Enabled = False
        End If
  End Sub
```

```vbnet
Private Sub tmrFlicker_Tick(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles tmrFlicker.Tick
        If Ball.ForeColor = Color.Blue Then
            Ball.ForeColor = Color.Red 'toggle
        Else
            Ball.ForeColor = Color.Blue
        End If
End Sub
```

```
 Private Sub tmrClock_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles tmrClock.Tick
        Dim currentx, currenty, movedx, movedy As Integer
        If chkRandom.Checked Then
            ewChange = -20 + Rnd() * 41
            nsChange = -20 + Rnd() * 41
        Else
            ewChange = nsChange = 0
        End If
        currentx = Ball.Location.X
        currenty = Ball.Location.Y
       movedx = currentx + EastWest + ewChange
       movedy = currenty - NorthSouth + nsChange
```

Rnd() is a function returning a random decimal value between 0 and 1, so ewChange and nsChange are randomly selected between -20 and plus 20. Ball.Location.x and Ball.Location.y gives the current location, which is then adjusted by the normal movement plus the random movement (if it is turned on).

```vb
    'Check for east or west border
        If (movedx > 750) Or (movedx < -50) Then
            EastWest = -EastWest
        ElseIf (movedy > 500) Or (movedy < -50) Then 'Check
for north or south border
            NorthSouth = -NorthSouth
        Else
            Ball.Location = New System.Drawing.Point(movedx,
movedy)
        End If

    End Sub
```

```
Public Class Form1

    Private Sub btnExit_Click(ByVal sender As Object, ByVal
e As System.EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub
End Class
```

• You will name this project **MovingLetter** and save it in your **Day02** folder.

• It should run like the demo project **MovingLetter.exe** already in your Day02 folder.