Programming Mouse Events

The mouse is central to most windows-based applications. VB provides programmers complete control of mouse related events. Programmers can create powerful and very useful applications by mastering the mouse events. Some important events associated with a mouse are: Click, Double Click, MouseUp, MouseDown and MouseMove.

Create a new application. Open the mousemove event of the form. The argument e provides the exact location of the mouse on the form. Add the following code to evaluate this:

```vbnet
Me.Text = e.X & ", " & e.Y
```

We can also easily evaluate which mouse key is pressed. Try adding the following code:

```vbnet
If e.Button = MouseButtons.Left Then
    Me.Text = Me.Text & " and you have pressed the left mouse key"
End If
```

We will now use some of these concepts to create a simple application to allow a user to free-draw on a form. To do this we need to use the graphics capabilities in Visual Basic. We want the user to be able to draw on the form by moving the mouse. Whenever the mouse is moved we want to draw a line connecting two points.

Define the following form-level variables:

```vbnet
Dim xpos, ypos As Single
Dim g As System.Drawing.Graphics
Dim newpen As Pen
```

In the MouseDown event of the form, set xpos and ypos variables to e.X and e.Y, respectively. Try typing the following code inside the MouseMove event of the form:

```vbnet
If e.Button = MouseButtons.Left Then
    g = Me.CreateGraphics
    newpen = New Pen(Color.Red, 2)
    xpos = e.X
    ypos = e.Y
End If
```

You can also change the shape of the cursor through the cursor property. For example,

```vbnet
```

Save the application.

Now, add drawing color and drawing width options.