
アプリケーション開発

お絵かきソフト

【目次】

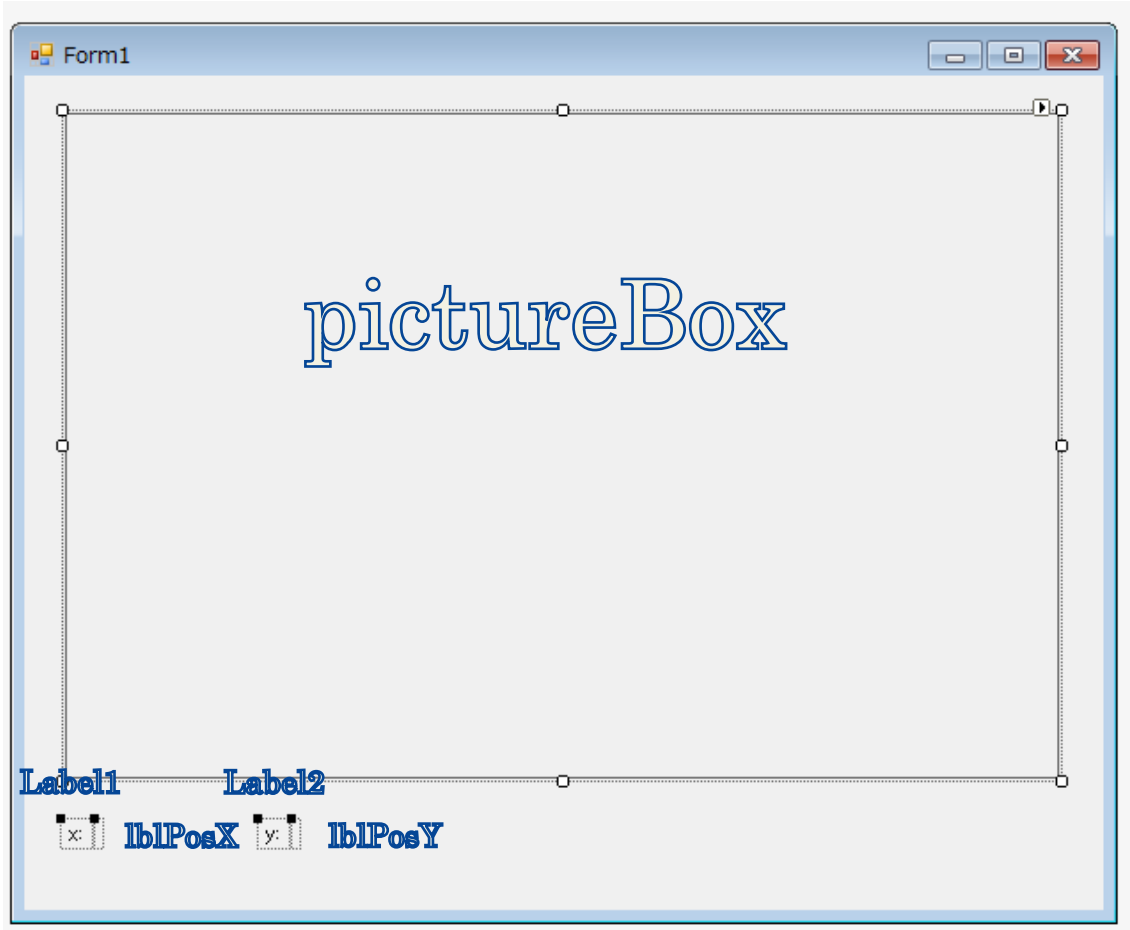
お絵かきソフトを作ってみよう.....	3
絵を書く枠と場所表示を作る.....	3
マウスの動きを見てもみよう.....	4
絵を書く準備をします.....	5
絵を書くとはどういうことか.....	5
では線画を描いてみよう.....	6
マウスをドラッグしたときだけ線を引くように改造する.....	8

お絵かきソフトを作ってみよう

今回は、お絵かきソフトを作ってみましょう。マウスを動かして線画を書いてみましょう。

絵を書く枠と場所表示を作る

PictureBox と Label を4つフォームに配置します。



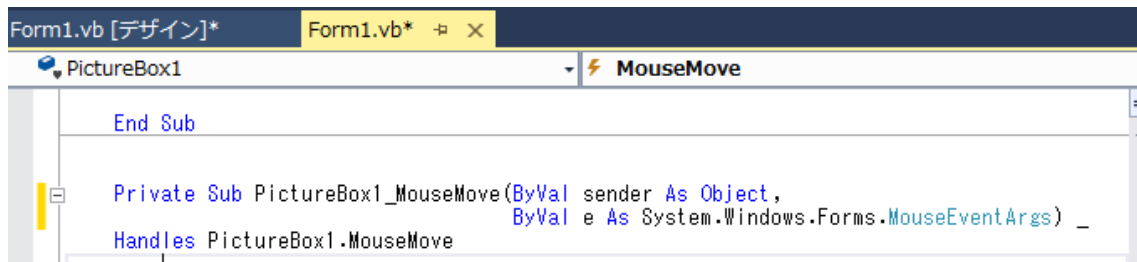
Label は左下に4つ並べて作成。

名前	Text	プロパティ
Label1	x:	
lblPosX	なし	
Label2	y:	
lblPosY	なし	
PictureBox	—	BordarStyle: FixedSingle

マウスの動きを見てみよう

PictureBox の上をマウスを動かした時の X,Y の位置を表示するようにしてみましょう。コードを表示して、PictureBox1_MouseMove のイベントを作ります。

コード画面の上の左側のコンボボックスを PictureBox1 に、右側のコンボボックスを MouseMove に切り替えるとソースコードが自動生成されます。



```
Form1.vb [デザイン]* Form1.vb*
PictureBox1 MouseMove
End Sub
Private Sub PictureBox1_MouseMove(ByVal sender As Object,
                                   ByVal e As System.Windows.Forms.MouseEventArgs) _
    Handles PictureBox1.MouseMove
```

```
Private Sub PictureBox1_MouseMove(ByVal sender As Object,
                                   ByVal e As System.Windows.Forms.MouseEventArgs) _
    Handles PictureBox1.MouseMove
    lblPosX.Text = e.X
    lblPosY.Text = e.Y
End Sub
```

赤字の部分だけ入力します。

これで実行してみましょう。マウスを PictureBox の上で動かした時だけ x,y の値が表示されます。左上が 0,0,右下が大きな値になります。



絵を書く準備をします

いままでも、何回かグラフィックを書いています。同様にまず絵を書く準備をします。

Public Class Form1 の直下にモジュール変数を作ります。

g as Graphics の部分です。

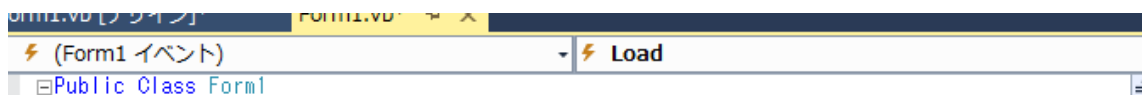
```
Public Class Form1
    Dim g As Graphics 'グラフィックオブジェクト

    Private Sub Form1_Load(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles MyBase.Load
        PictureBox1.Image = New Bitmap(PictureBox1.Width, _
            PictureBox1.Height)

        'グラフィックオブジェクトを作成します。
        g = Graphics.FromImage(PictureBox1.Image)

    End Sub
```

続いて、Form_Load のイベントを作ります。



そして、赤字の部分を入力します。

これで g というグラフィック用のオブジェクトができますので、これを使って描画していきます。

絵を書くとはどういうことか

```
g.DrawLine(Pens.Black, intX, intY, e.X, e.Y)
```

上の命令が線画を書くための命令です。

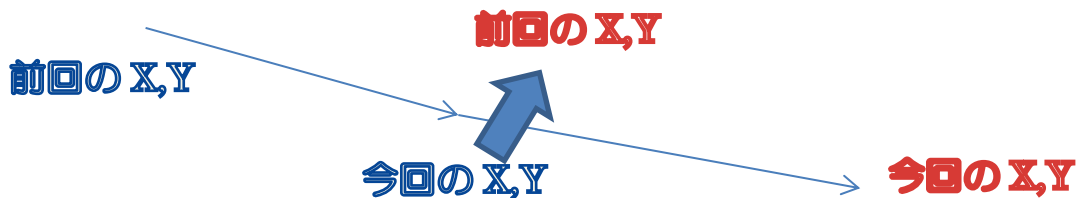
g というオブジェクトの DrawLine というメソッドを使って線を書きます。

5つのパラメータがあります。

項目番号	仮引数	意味
1つ目	Pens.Black	ペンの色、これは黒
2つ目	intX	前回の位置 X
3つ目	intY	前回の位置 Y
4つ目	e.X	現在の位置 X
5つ目	e.Y	現在の位置 Y

この命令で、前回の位置から今回の位置まで線が引かれます。

前回の線とはどこでしょう。それは MouseMove のイベントが前回発生したときの位置で、現在のイベントで線を引いた後、その値がそのまま前回の位置になります。



線を引く前に数値だけ表示した時のように、1ポイントでも割り込みは発生しますので、線はほとんど点になります。

では線画を描いてみよう

前回位置の変数を追加します。

```
Public Class Form1
    Dim g As Graphics 'グラフィックオブジェクト
    Dim intX As Integer 'マウスポインタX
    Dim intY As Integer 'マウスポインタY
```

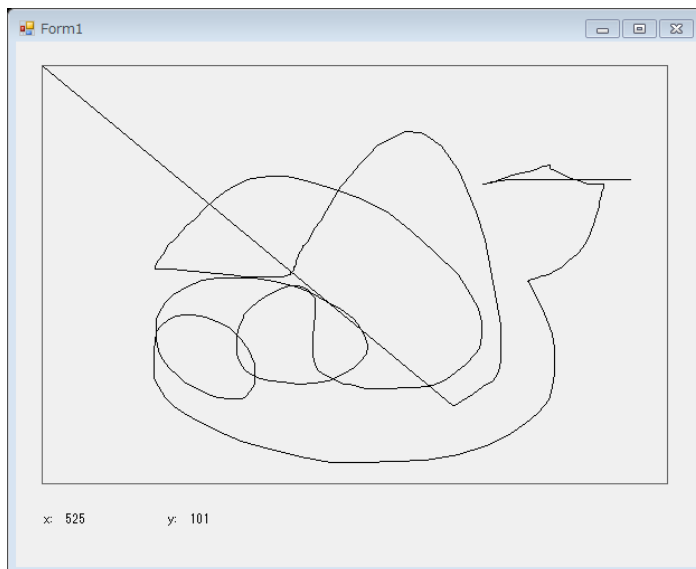
mousemove イベントの中に赤字の部分を追加しましょう。

```
IblPosX.Text = e.X  
IblPosY.Text = e.Y  
  
' ドローモード  
g.DrawLine(Pens.Black, intX, intY, e.X, e.Y)  
' 描画  
PictureBox1.Refresh()  
' ポインタをレジスト  
intX = e.X  
intY = e.Y
```

また、MouseDown で、今の位置を前回の位置として記憶します。

```
Private Sub PictureBox1_MouseDown(ByVal sender As Object, _  
ByVal e As System.Windows.Forms.MouseEventArgs) _  
Handles PictureBox1.MouseDown  
    intX = e.X  
    intY = e.Y  
End Sub
```

実行すると



これで線画が書けるようになります。

マウスをドラッグしたときだけ線を引くように改造する

今のままでは、ずっと線が書かれてしまいますので、マウスを押してドラッグしたときだけ線が書けるように改造しましょう。

MouseDown で書き始めて、MouseUp で終わるようにすればいいですね。

そこで MouseDown で描画の旗を立てて、MouseUp で旗を降ろすようにしてみましょう。

まず旗の変数をモジュール変数として宣言します

```
Public Class Form1
    Dim DrawFlg As Integer 'ドローモード
```

```
    Private Sub PictureBox1_MouseDown(ByVal sender As Object, _
        ByVal e As System.Windows.Forms.MouseEventArgs) _
        Handles PictureBox1.MouseDown
        DrawFlg = True
        intX = e.X
        intY = e.Y
    End Sub
```

また、MouseUp のイベントを追加して旗を降ろすコードを書きます。

```
    Private Sub PictureBox1_MouseUp(ByVal sender As Object, _
        ByVal e As System.Windows.Forms.MouseEventArgs) _
        Handles PictureBox1.MouseUp
        'ドローモード初期化
        DrawFlg = False
    End Sub
```

そして、DrawFlg が True の時だけ線を描きます。MouseMove を赤字の部分を追加します。


```
lblPosX.Text = e.X  
lblPosY.Text = e.Y  
  
If DrawFlg = False Then  
    Exit Sub  
End If  
  
'ドローモード  
g.DrawLine(Pens.Black, intX, intY, e.X, e.Y)
```

課題：ここまでできたら、色を変える、線の太さを変える、ペンの形を変えるなどの改造を試みよう。

課題：出来上がった絵を保存できるかチャレンジしてみよう。