

# Visual Basic 程式設計

## 繪圖世界II

- ◆ 吸取顏色 - **Point**方法
- ◆ Shape控制項
- ◆ LoadPicture函數、SavePicture敘述  
與App.Path屬性

# 吸取顏色-Point方法

## ■ Point語法如下：

語法：[物件.] Point (x, y)

功能：取得表單或圖片方塊物件中某個指定點的顏色值。

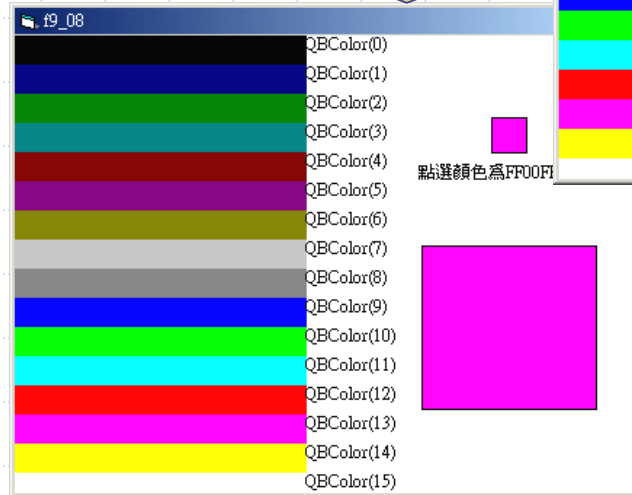
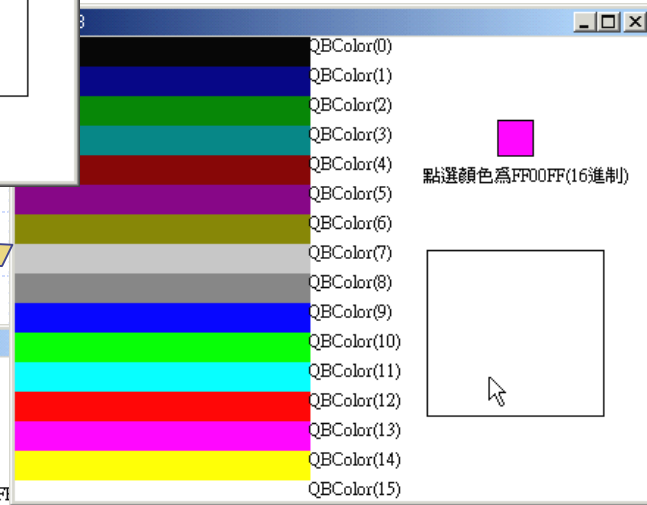
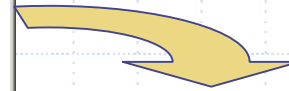
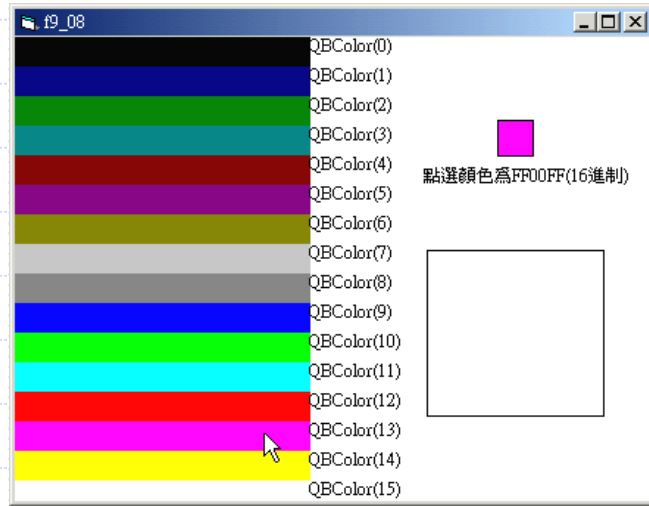
## ● 【語法說明】：

- 使用Point方法可吸取特定點的顏色。
- (2) Point方法所吸取的特定點顏色值為長整數。

## ● 【實作Try it!!】：設計一個繪圖軟體提供的吸取顏色功能。

- (1) 到指定點吸取顏色。
- (2) 將吸取的顏色指定到其他的區域。

# Try it



# Visual Basic 程式設計

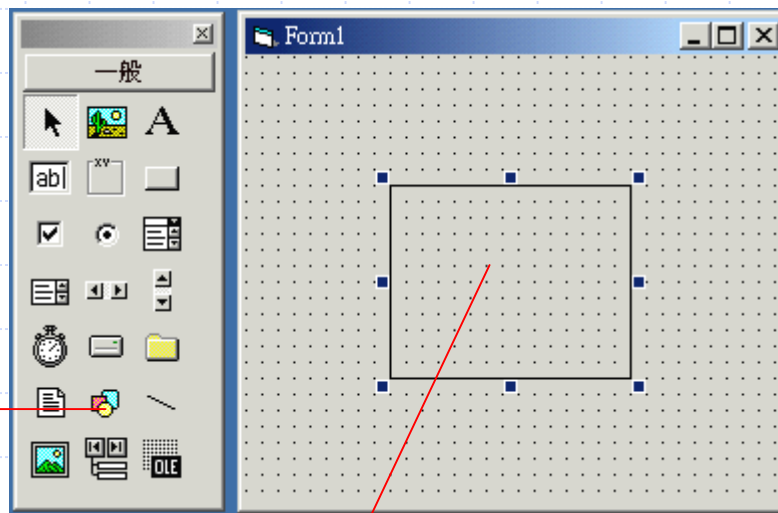
## 繪圖世界II

- ◆ 吸取顏色 - Point方法
- ◆ **Shape**控制項
- ◆ LoadPicture函數、SavePicture敘述  
與App.Path屬性

# 形狀(shape)控制項

- 『形狀』控制項一共可以製作6種不同形狀的圖形。並且和直線控制項一樣，形狀控制項沒有任何的事件，僅僅能夠做為繪圖之用，同樣也會隨母件移動。
- 建立形狀控制項

1: 選取  
形狀工具



2: 在表單上拉出形狀控制項大小

# 形狀(shape)控制項 (cont.)

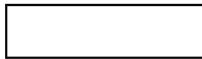


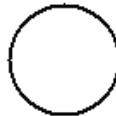
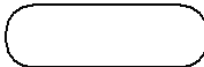

## ■ 形狀的屬性

屬性	說明
<b>Shape</b>	設定形狀控制項的形狀。
<b>DrawMode</b>	設定形狀控制項的顯示效果。
<b>BackColor</b>	設定形狀控制項的背景顏色
<b>BackStyle</b>	設定形狀控制項的背景形式： 0-透明(預設值) 1-不透明
<b>BorderColor</b>	設定形狀控制項的邊框顏色。
<b>BorderStyle</b>	設定形狀控制項的邊框樣式，屬性值為0~6(預設值為0)
<b>BorderWidth</b>	設定形狀控制項的邊框寬度。
<b>FillColor</b>	設定形狀控制項內的填滿顏色。
<b>FillStyle</b>	設定形狀控制項內的填充條紋樣式。

- 形狀控制項可設定的屬性比起直線控制項還多了許多，因此可以產生更多變化。

# 形狀(shape)控制項 (cont.)

- Shape屬性是用來設定形狀控制項的形狀，屬性值為0~5，各代表不同形狀：

Shape屬性值	說明	
0	長方形(預設值)	
1	正方形	
2	橢圓形	
3	圓形	
4	圓角矩形 (類似流程圖之開始結束符號)	
5	圓角正方形	

# 形狀(shape)控制項 (cont.)

## ◆ DrawMode屬性

DrawMode屬性是用來決定畫筆與螢幕顏色的組合關係。由於表單、圖片方塊等物件都具有DrawMode屬性，因此當執行Pset、Line及Circle方法時，不同的DrawMode屬性會產生不同組合效果。

而直線控制項和形狀控制項也有DrawMode屬性，下表是常用的DrawMode屬性值及組合效果。

屬性值	說明	屬性值	說明
1	黑色	9	背景顏色 And 前景顏色
2	Not(背景顏色Or前景顏色)	10	Not (背景顏色 Xor 前景顏色)
3	背景顏色And (Not前景顏色)	11	沒有動作
4	Not (前景顏色)	12	背景顏色 Or (Not前景顏色)
5	(Not前景顏色)And前景顏色	13	前景顏色(預設值)
6	Not(背景顏色)	14	(Not 背景顏色) Or 前景顏色
7	前景顏色 Xor 背景顏色	15	背景顏色 Or 前景顏色
8	Not (背景顏色 And 前景顏色)	16	白色



# 形狀(shape)控制項 (cont.)

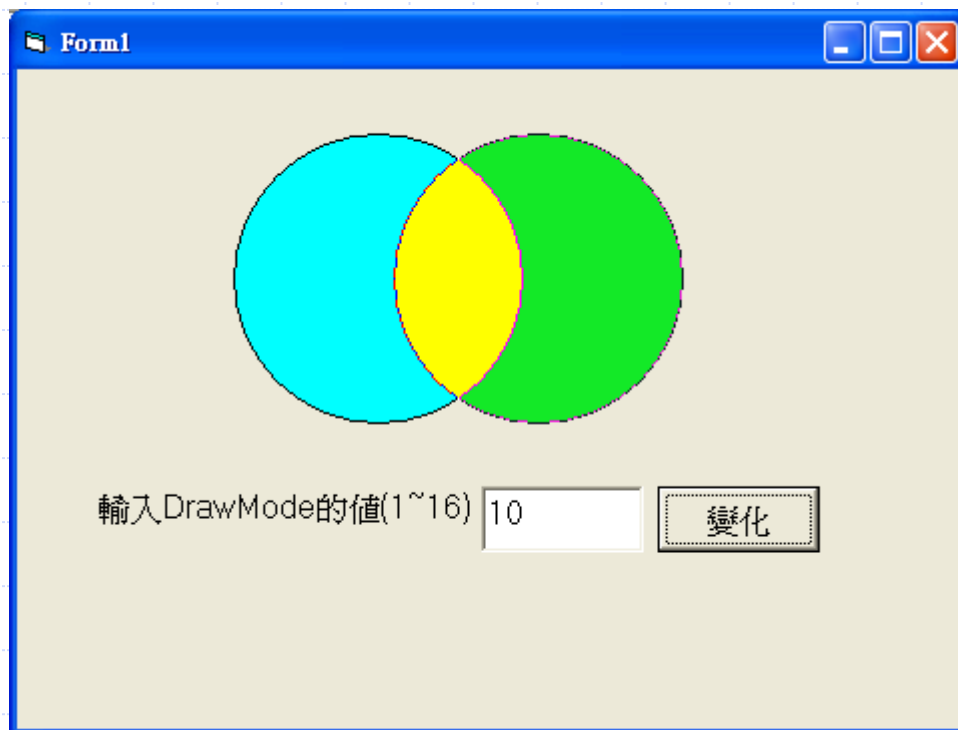
- ◆ FillStyle屬性可以設定圖形內部的填充樣式，有多種花紋可供選擇如下表：

FillStyle屬性值	說明
0	實心 
1	透明(預設值) 
2	水平線 
3	垂直線 
4	左上到右下的斜線 
5	左下到右上的斜線 
6	垂直交叉線 
7	對角交叉線 

# 形狀(shape)控制項 (cont.)

- 形狀控制項的程式設計

- 由於DrawMode的屬性值效果很難了解，因此我們直接用範例來看看不同的DrawMode屬性值會造成什麼樣的效果。



# Visual Basic 程式設計

## 繪圖世界II

- ◆ 吸取顏色 - Point方法
- ◆ Shape控制項
- ◆ **LoadPicture**函數、**SavePicture**敘述與**App.Path**屬性

# LoadPicture 函數、SavePicture 敘述與 App.Path 屬性

- 如果不是要開發繪圖軟體，則可以使用別人開發的繪圖軟體製作所需要的精美圖片，再將該圖片載入到我們所需要的程式中即可。載入圖片使用的是 LoadPicture 函數，而儲存圖片使用的則是 SavePicture 敘述。當遇到與實體檔案（例如圖檔）有關的敘述時，常常會希望在程式執行時動態得到該檔案的路徑，此時則可以利用 App.Path 屬性取得檔案路徑。
- **LoadPicture 函數**

語法：[物件.] Picture = LoadPicture ([filename])

功能：將圖形檔載入表單、影像或圖片方塊物件中。

- **【語法說明】：**
  - filename：要載入圖形的檔名，如果該圖檔位置與應用程式位處於不同資料夾，則需要加上路徑名稱，例如：`Form1.Picture = LoadPicture ("C:\VB6\PIC\TEST.bmp")`
  - LoadPicture 函數可以載入 .bmp、.wmf、.ico 等格式的圖檔。
  - 若省略 filename，則會清除畫面。  
例如：`Image1.Picture = LoadPicture ()` 就會將 Image1 的圖形清除掉，

# LoadPicture函數、SavePicture敘述與App.Path屬性 (cont.)

## ■ SavePicture敘述

語法：App.Path

功能：取得目前執行中的程式的所在路徑。

### ● 【語法說明】：

- 若未指定物件，則SavePicture會將目前表單內的圖形存檔。  
例如：SavePicture Image, "C:\PIC\TEST2.bmp"。
- 儲存圖形的控制項時，AutoRedraw屬性為True。
- 必須選擇Image或Picture，若指定為Image，則儲存時會以Bitmap(.bmp)格式存檔；若指定為Picture，則會以載入該物件時的格式當作存檔格式。  
例如：picTEST1圖片方塊的Picture屬性載入了一個JPEG(\*.jpg)檔，若執行SavePicture picTEST1.Picture, "TEST2"，則TEST1會是一個JPEG圖檔；若執行SavePicture picA.Image, "TEST3"，則TEST3會是一個點陣圖檔。

# LoadPicture函數、SavePicture敘述與App.Path屬性 (cont.)

## ■ App.Path屬性

語法：App.Path

功能：取得目前執行中的程式的所在路徑。

### ● 【語法說明】：

- 假設目前執行的程式位於『C:\VB6\aa\』目錄，則App.Path屬性值就是"C:\VB6\aa"字串，所以若要使用LoadPicture來載入與執行程式同一目錄的TEST.bmp圖檔，則可使用下列敘述加以完成：

```
picTEST = LoadPicture(App.Path+"\TEST.bmp")
```

若TEST.bmp圖檔位於執行程式的父目錄，則可使用"/../"取得上一層的路徑，敘述如下：

```
picTEST = LoadPicture(App.Path+"\..\TEST.bmp")
```

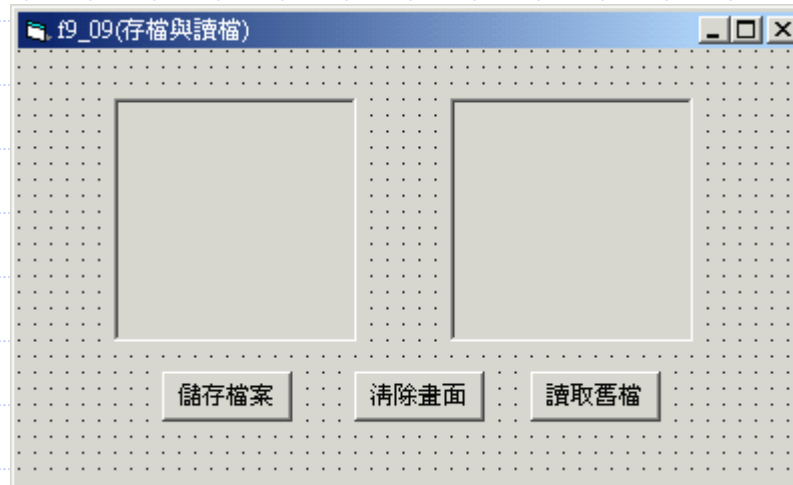
若TEST.bmp圖檔位於執行程式的兄弟目錄，即『C:\VB6\bb\』目錄，則可使用下列敘述完成：

```
picTEST = LoadPicture(App.Path+"\..\bb\TEST.bmp")
```

# LoadPicture 函數、SavePicture 敘述與 App.Path 屬性 (cont.)

## ◆ Try it!

- ◆ 在左邊圖片方塊中，提供繪圖功能，按下【儲存檔案】鈕，會將圖片存檔，按下【開啟舊檔】鈕，則會將剛才存檔的圖片讀入右邊的圖片方塊中。



# LoadPicture函數、SavePicture敘述與App.Path屬性 (cont.)

◆ 用滑鼠繪圖簡易版程式碼：

```
Private Sub picSource_MouseDown(Button As Integer, Shift As Integer,  
                                X As Single, Y As Single)  
    If Button = 1 Then  
        picSource.PSet (X, Y)  
    End If  
End Sub  
  
Private Sub picSource_MouseMove(Button As Integer, Shift As Integer,  
                                X As Single, Y As Single)  
    If Button = 1 Then  
        picSource.Line -(X, Y)  
    End If  
End Sub
```



# LoadPicture 函數、SavePicture 敘述與 App.Path 屬性 (cont.)

