

National Taipei University of Technology

Windows Programming (Fall 2019)

Homework #7

Deadline: 12/20 (Fri), before 24:00 PM

壹、注意

請遵守以下規則，否則單次成績以 0 分計算

1. 準時繳交作業，逾時不候
2. 不准抄襲他人作業，請自己完成

貳、主題

根據上一次的作業內容，本次新增一個新的 Shape 為六邊形以及 Command，Redo 和 Undo 等功能，如圖 1 所示，Windows Store App 比照辦理。

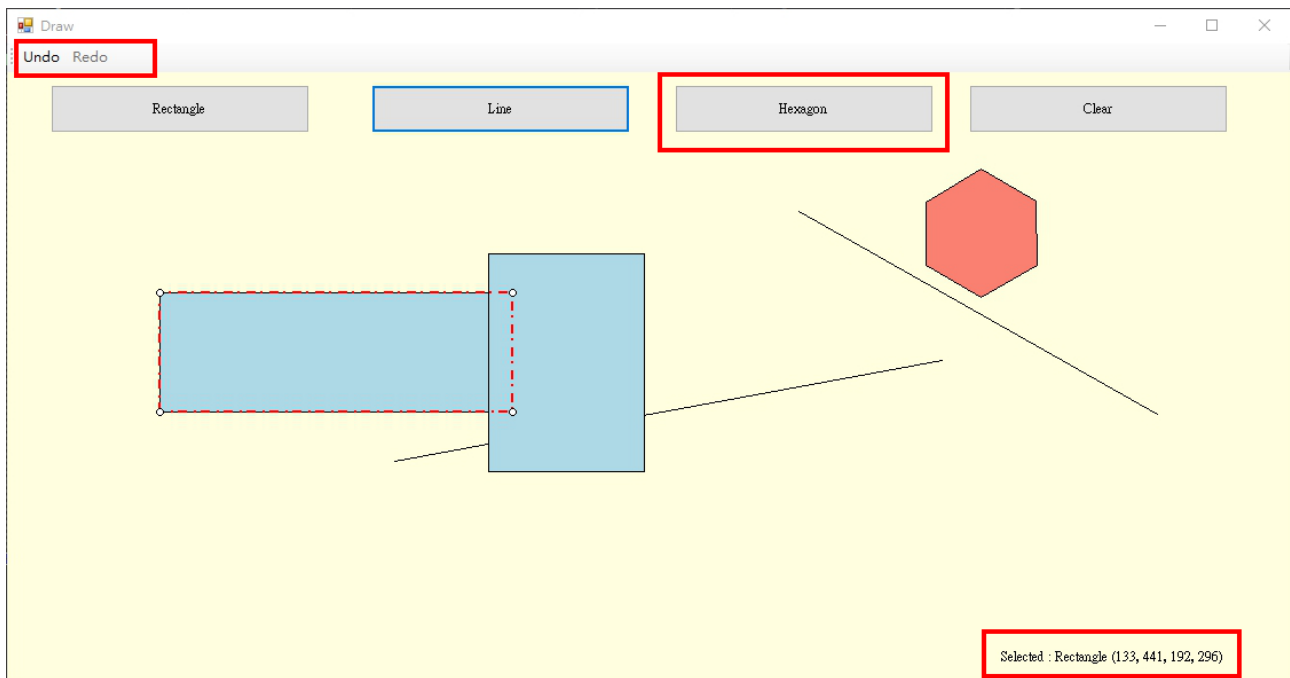


圖 1 Windows Form GUI

參、題目介紹

一、[2 pts] Drawing Hexagon

請在兩個 View 分別新增 Hexagon Button，點擊 Hexagon Button 時，你可以畫出六邊形，並且將該按鈕設定為 disable。注意：點擊任何畫圖按鈕時，須將其他畫圖按鈕設定為 enable，且將本身按鈕設定為 disable。

二、[1 pts] Factory Pattern

在此作業中，可以在 Canvas 上繪製三種形狀（線、矩形和六邊形）。當繪製形狀時，請使用簡單工廠(simple factory)模式來創建不同形狀的 instance。

三、[3 pts] 點選 Shape

使用者可以在畫面上點選 Shape。當使用者點擊某個形狀時，將選擇該形狀，被選擇到的形狀必須有虛線外框以及四個點。（未來被選擇到的形狀可能會新增不同行為，例如：放大、移動、刪除等等）。

- 虛線外框必須在最上層，不管點選的圖片是否為最上層，如圖 1。
- 畫面右下角新增一個 Label，點選到的圖形請在該 Label 上顯示被點選圖形之形狀及座標(Windows Form 與 Store 兩個視窗都必須顯示)，例如顯示「Rectangle (1, 1, 2, 2)」，表示該形狀為矩形且其左上角座標為(1, 1)、右下角座標為(2, 2)，請參考圖 1。

四、[8 pts] Undo and Redo

請使用 Command Pattern 來實作 redo 和 undo 功能，請參考圖 2。

- 如果是第一步，undo button 為 disable。
- 如果是最後一步，redo button 為 disable。
- 畫圖需要 redo、undo 功能，點選則不需 redo、undo。

五、[6 pts] Unit Testing

請為每個 Model class 的每個 method 撰寫 test case，且必須全部通過測試，Unit Test 是根據 test-coverage 去評分。

六、[3 pts] GUI Testing

請使用自動化 UI 測試為你的 GUI 編寫測試，測試案例(Test case)的規定如下：

- 只有 Windows Form 需要做 GUI Testing。
- 請將畫面上所有按鈕分別測試一次(畫出正方形、畫出線、畫出六邊形、清空畫布)，畫出一個形狀後，請點選這個形狀，並 Assert 畫面上的 Label 所顯示的值與所繪的圖形吻合(assert 圖 1 的 Label 為正確的)。
- 請整合所有按鈕畫出一個具體的圖形，並執行適當的 Assert 後清除並關閉。

七、[3 pts] MVC Pattern

你的設計將根據 MVC 架構評分，讓 UI 盡可能薄，必須將 UI 以及 Model 切開，強制實行單向依賴，建議遵循圖 2 Class Diagram 設計。

八、[2 pts] Adaptor Pattern

在本作業中，您需要使用同一 Model 實現兩個不同的 view。這兩個 view 具有不兼容的圖形界面。請使用 Adaptor Pattern 來解決該問題。

九、[2 pts] Observer pattern and event-delegation model

您需要使用帶有 C# event-delegation model 的 observer pattern 來實現 MVC 架構。

十、[9 pts] Code Quality

Code Quality 將取決於你的程式碼是否有 bad smell，你的分數將取決於你的 code smell 密度。當你使用 Windows Form Designer 產生程式碼時，其預設的變數命名、副程式命名等可能違反 coding standard，請修改以符合本課程之規定。

十一、[2 pts] Summary

每一次寫完作業時都必須完成 homework summary，必須填寫本次作業花費時間，請在網站下載範本檔。

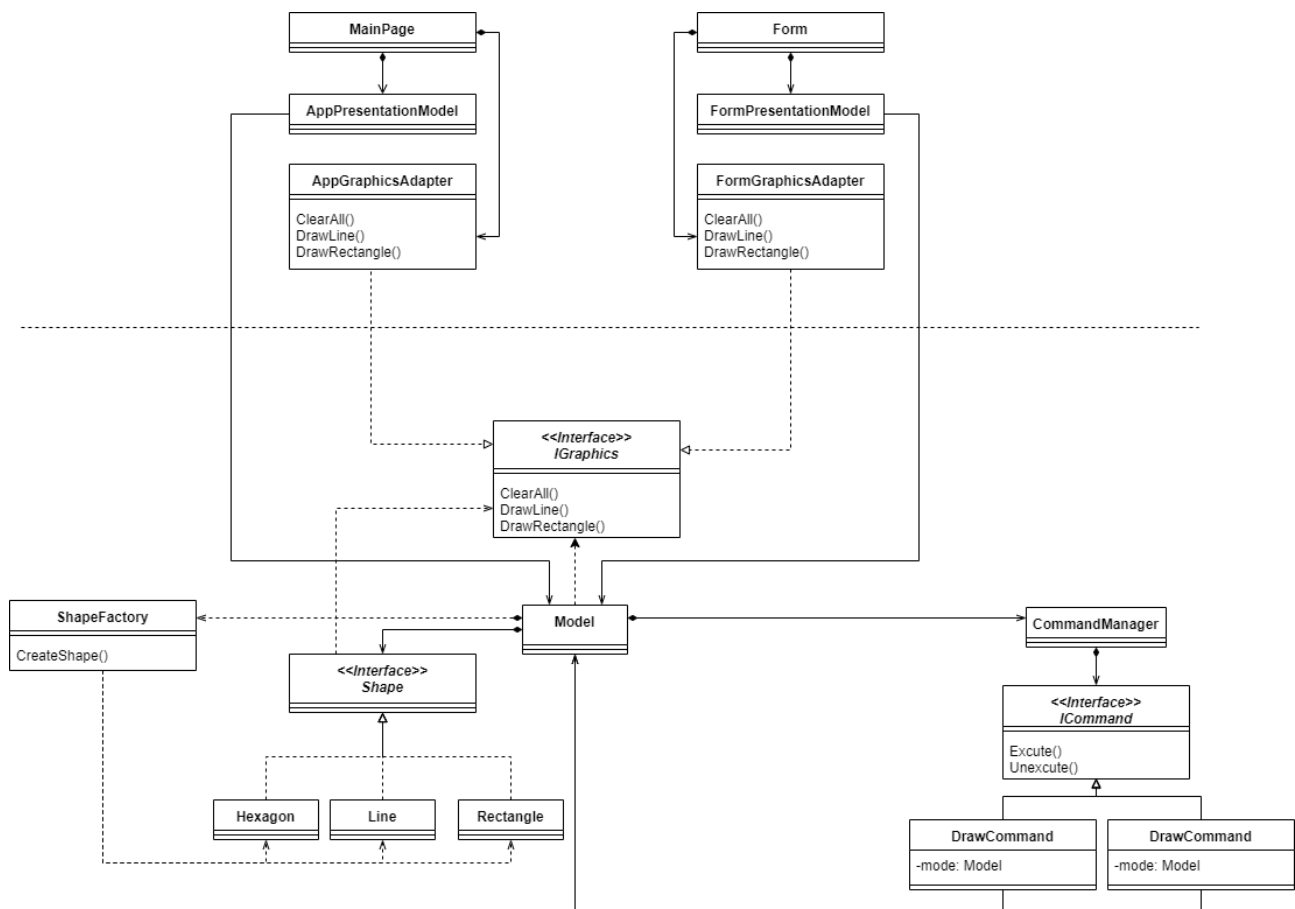


圖 2 Class Diagram

Additional Information

The following information may be useful to your homework. If you need more details, please visit MSDN website.

1. Draw Rectangle in Windows Form - You can use the following codes to draw diamond in Windows Store Form.

```
myRect = new System.Windows.Shapes.Rectangle();  
myRect.Stroke = System.Windows.Media.Brushes.Black;  
myRect.Fill = System.Windows.Media.Brushes.SkyBlue;  
myRect.HorizontalAlignment = HorizontalAlignment.Left;  
myRect.VerticalAlignment = VerticalAlignment.Center;  
myRect.Height = 50;  
myRect.Width = 50;  
myGrid.Children.Add(myRect);
```

For more information, please visit MSDN :

<https://docs.microsoft.com/en-us/dotnet/api/system.drawing.graphics.fillpolygon>

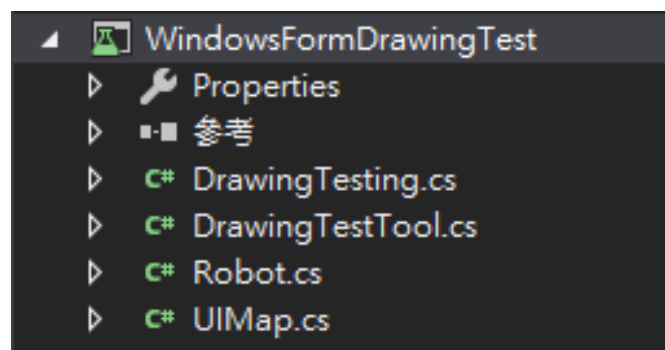
2. Draw Rectangle in Windows Store App - You can use the following codes to draw diamond in Windows Store App.

```
rectangle = new Rectangle();  
InitializeShape(rectangle, 0, 250, 150, 150, _blueColor);  
_canvas.Children.Add(rectangle);
```

For more information, please visit MSDN :

<https://docs.microsoft.com/en-us/uwp/api/windows.ui.xaml.shapes.polygon>

3. Add Two Class Robot and DrawingTestTool - You can use the following class to Do UI testing.



4. Drawing UI testing - You can use the following codes to Do UI testing.

```
[TestMethod]
public void DrawingRectangle()
{
    // PANEL_ACCESSIBLENAME為Panel的AccessibleName
    DrawingTestTool.DrawRectangle(PANEL_ACCESSIBLENAME, 50, 150, 50, 150);
    Assert.IsTrue(true);
}
```