

實做題

1. 參考 10-3-1 的類別圖寫出 c#程式的 IPrice 介面和 car 類別以便顯示車輛的價格

```
using System;

namespace VehicleApp
{
    public interface IPrice
    {
        double GetPrice();
    }

    public class Car : IPrice
    {
        public string Name { get; set; }
        public double Price { get; set; }

        public Car(string name, double price)
        {
            Name = name;
            Price = price;
        }

        public double GetPrice()
        {
            return Price;
        }

        public string GetName()
        {
            return Name;
        }
    }

    class Program
    {
        static void Main(string[] args)
        {
            Car car1 = new Car("Toyota Corolla", 750000);
            Car car2 = new Car("Tesla Model 3", 1600000);
            Console.WriteLine($"車名: {car1.GetName()}, 價格: {car1.GetPrice():C}");
            Console.WriteLine($"車名: {car2.GetName()}, 價格: {car2.GetPrice():C}");
        }
    }
}
```

Microsoft Visual Studio 偵錯主控台

車名: Toyota Corolla, 價格: NT\$750,000.00
車名: Tesla Model 3, 價格: NT\$1,600,000.00

C:\Users\427\source\repos\ConsoleApp1\ConsoleApp1\Program.cs
若要在偵錯停止時自動關閉主控台，請啟用「工作時關閉主控台」。
按任意鍵關閉此視窗...

2. 請建立 2 個過載的類別方法 `Cube()`, 可以分別計算 `int` 和 `double` 參數的平方, 然後建立 2 個過載 `MinElement()` 類別方法, 傳入 3 個或 4 個 `int` 參數, 其傳回值是參數中的最小值。


```
using System;
namespace OverloadExample
{
    public class Calculator
    {
        public int Cube(int value)
        {
            return value * value * value;
        }

        public double Cube(double value)
        {
            return value * value * value;
        }

        public int MinElement(int a, int b, int c)
        {
            return Math.Min(a, Math.Min(b, c));
        }

        public int MinElement(int a, int b, int c, int d)
        {
            return Math.Min(Math.Min(a, b), Math.Min(c, d));
        }
    }

    class Program
    {
        static void Main(string[] args)
        {
            Calculator calc = new Calculator();
            Console.WriteLine("Cube(3) = " + calc.Cube(3));
            Console.WriteLine("Cube(2.5) = " + calc.Cube(2.5));
            Console.WriteLine("MinElement(4, 7, 2) = " + calc.MinElement(4, 7, 2));
            Console.WriteLine("MinElement(9, 3, 6, 1) = " + calc.MinElement(9, 3, 6, 1));
        }
    }
}
```

 選取 Microsoft Visual Studio 偵錯器
Cube(2.5) = 15.625
MinElement(4, 7, 2) = 2
MinElement(9, 3, 6, 1) = 1
C:\Users\427\source\repos\Cd
若要在偵錯停止時自動關閉主控
按任意鍵關閉此視窗...

3. 請建立單位轉換的 C# 程式, 一英尺有 12 英吋: 一英碼等於 3 英尺, `FeetToInches()` 方法可以將英尺轉換成英吋; `YardsToInches()` 方法將英碼轉換成英吋, 程式在輸入英吋後, 使用委派來動態執行轉換方法。

```
using System;
namespace UnitConverter
{
    public class Converter
    {
        public double FeetToInches(double feet)
        {
            return feet * 12;
        }

        public double YardsToInches(double yards)
        {
            return yards * 3 * 12;
        }
    }

    class Program
    {
        delegate double ConversionDelegate(double value);
        static void Main(string[] args)
        {
            Converter converter = new Converter();
            ConversionDelegate convert;
            Console.WriteLine("請選擇轉換方式:");
            Console.WriteLine("1. 英尺轉英吋");
            Console.WriteLine("2. 英碼轉英吋");
            Console.Write("請輸入選項 (1 或 2): ");
            string option = Console.ReadLine();
            if (option == "1")
            {
                convert = converter.FeetToInches;
                Console.Write("請輸入英尺數: ");
            }
            else if (option == "2")
            {
                convert = converter.YardsToInches;
                Console.Write("請輸入英碼數: ");
            }
            else
            {
                Console.WriteLine("無效的選項。");
                return;
            }
            if (double.TryParse(Console.ReadLine(), out double input))
            {
                double inches = convert(input);
                Console.WriteLine($"轉換結果: {inches} 英吋");
            }
            else
            {
                Console.WriteLine("輸入格式錯誤, 請輸入數值。");
            }
        }
    }
}
```

Microsoft Visual Studio 偵錯

請選擇轉換方式:
1. 英尺轉英吋
2. 英碼轉英吋
請輸入選項 (1 或 2): 1
請輸入英尺數: 100
轉換結果: 1200 英吋


C:\Users\427\source\re
若要在偵錯停止時自動關
按任意鍵關閉此視窗...

4. 請建立 C# 應用程式,在表單使用標籤控制項模擬紅綠燈,預設背景是紅色 代表紅燈,按滑鼠左鍵改為黃色的黃燈;右鍵改為綠色的綠燈。

```
using System;
using System.Drawing;
using System.Reflection.Emit;
using System.Windows.Forms;
namespace WinFormsApp3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            InitializeTrafficLight();
        }

        private void InitializeTrafficLight()
        {
            label1.Text = "紅燈";
            label1.BackColor = Color.Red;
            label1.ForeColor = Color.White;
            label1.TextAlign = ContentAlignment.MiddleCenter;
            label1.Font = new Font("Arial", 16, FontStyle.Bold);
            label1.Dock = DockStyle.Fill;
            label1.MouseClick += Label1_MouseClick;
        }

        private void Label1_MouseClick(object sender, MouseEventArgs e)
        {
            if (e.Button == MouseButtons.Left)
            {
                label1.BackColor = Color.Yellow;
                label1.ForeColor = Color.Black;
                label1.Text = "黃燈";
            }
            else if (e.Button == MouseButtons.Right)
            {
                label1.BackColor = Color.Green;
                label1.ForeColor = Color.White;
                label1.Text = "綠燈";
            }
        }
    }
}
```



The screenshot shows a Windows Form titled "Form" with a yellow background and the text "黃燈" (Yellow Light) in black, bold font, centered. This represents the state of the traffic light after a left mouse click.

簡答題

第四題。

在 `FileInfo` 物件建立新文字檔是使用 `CreateText()` 方法:新增文字內容至檔尾 是呼叫 `AppendText()` 方法來開啟檔案。。

第五題

檔案對話方塊依用途分為 2 種控制項:`OpenFileDialog` 和 `SaveFileDialog`