Практична 3.

Завдання 1.

```
using System;

class Title
{
    public string Text { get; }

    public Title(string text)
    {
        Text = text;
    }

    public void Show()
    {
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("Title: " + Text);
        Console.ResetColor();
    }
}
```

```
class Author
{
    public string Name { get; set; }

    public Author(string name)
    {
        Name = name;
    }

    public void Show()
    {
        Console.ForegroundColor = ConsoleColor.Blue;
        Console.WriteLine("Author: " + Name);
        Console.ResetColor();
    }
}

class Content
{
    public string Text { get; set; }

    public Content(string text)
```

```
    {
        Text = text;
    }

    public void Show()
    {
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.WriteLine("Content: " + Text);
        Console.ResetColor();
    }
}

class Book
{
    public Title BookTitle { get; }
    public Author BookAuthor { get; set; }
    public Content BookContent { get; set; }

    public Book(string title, string author, string content)
    {
        BookTitle = new Title(title);
        BookAuthor = new Author(author);
```

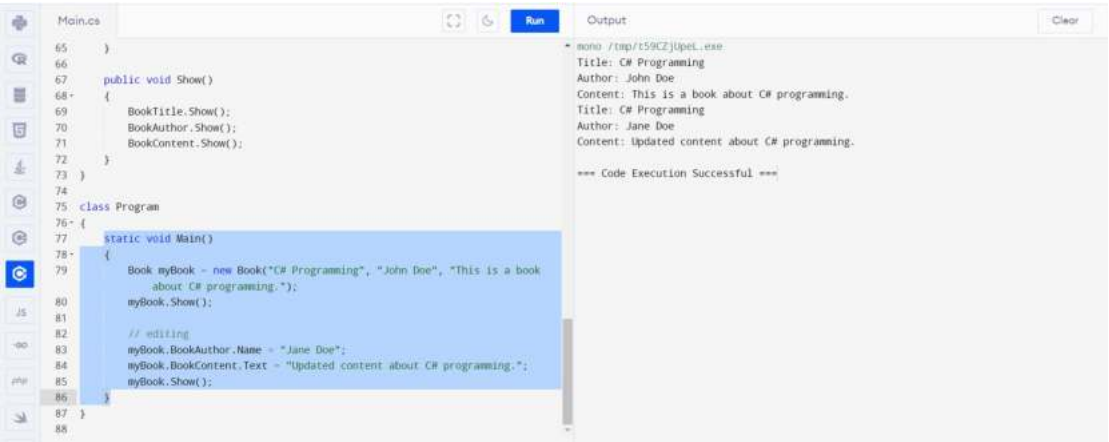
```
        BookContent = new Content(content);
    }

    public void Show()
    {
        BookTitle.Show();
        BookAuthor.Show();
        BookContent.Show();
    }
}

class Program
{
    static void Main()
    {
        Book myBook = new Book("C# Programming", "John Doe", "This is a book about C# programming.");
        myBook.Show();

        // editing
        myBook.BookAuthor.Name = "Jane Doe";
        myBook.BookContent.Text = "Updated content about C# programming.";
```

```
        myBook.Show();
    }
}
```



Завдання 2.

```
using System;

class Point
{
    public int X { get; }
    public int Y { get; }
    public string Name { get; }

    public Point(int x, int y, string name)
    {
```

```
        X = x;
        Y = y;
        Name = name;
    }
}

class Figure
{
    private Point[] points;

    public Figure(params Point[] points)
    {
        if (points.Length < 3 || points.Length > 5)
        {
            throw new ArgumentException("A figure must have between 3 and 5
points.");
        }

        this.points = points;
    }

    public double LengthSide(Point A, Point B)
    {
        return Math.Sqrt(Math.Pow(B.X - A.X, 2) + Math.Pow(B.Y - A.Y, 2));
```

```
    }

    public void PerimeterCalculator()
    {
        double perimeter = 0;

        for (int i = 0; i < points.Length; i++)
        {
            perimeter += LengthSide(points[i], points[(i + 1) % points.Length]);
        }

        Console.WriteLine($"Perimeter of the polygon: {perimeter}");
    }
}

class Program
{
    static void Main()
    {
        Point A = new Point(0, 0, "A");
        Point B = new Point(0, 4, "B");
        Point C = new Point(3, 0, "C");
        Point D = new Point(3, 4, "D");
```

```
        Figure figure = new Figure(A, B, C, D);
        figure.PerimeterCalculator();
    }
}
```