Graphics Code

class Point

```
public class Point
{

    private int x;
    private int y;
    public Point(int x, int y)
    {
        this.x = x;
        this.y = y;

    }
    public int getX()
    {
        return x;
    }
    public int getY()
    {
        return y;
    }
    public void setx(int x)
    {
        this.x = x;

    }
    public void sety(int y)
    {
        this.y = y;

    }
}
```

class Line

```
using Raylib_cs;

internal class Line:Shape
{
```

```
    Point Start;
    Point End;
    public Line()
    {
        Start = new Point(0, 0);
        End = new Point(0, 0);
        Co = Raylib_cs.Color.Black;
    }
    public Line(int x1,int y1,int x2,int y2,Raylib_cs.Color c)
    {
        Start=new Point(x1,y1);
        End=new Point(x2,y2);
        Co = c;
    }
    public Line(Point p1,Point p2,Raylib_cs.Color c)
    {
        Start = p1;
        End = p2;
        Co = c;
    }
    public override void Draw()
    {
        Raylib.DrawLine(Start.getX(),Start.getY(),End.getX(),End.getY(
    }
}
```

class Circle

```
using Raylib_cs;

class Circle : Shape
{
    Point Center;
    int radius;

    public Circle()
    {
        Center = new Point(0, 0);
        radius = 10;
        Co = Color.Black;

    }
    public Circle(int x, int y, int r, Color c)
    {
```

```
            Center = new Point(x, y);
            radius = r;
            Co = c;

        }
        public Circle(Point p, int r)
        {
            Center = p;
            radius = r;

        }
        public override void Draw()
        {
           Raylib.DrawCircle(Center.getX(),Center.getY(), radius, Co);
        }


    }
```

class Rect

```
    internal class Rect:Shape
    {
        Point UL;
        Point LR;


        public Rect()
        {
            UL = new Point(0,0);
            LR = new Point(0,0);
            Co = Color.Black;
        }
        public Rect( int x1, int y1, int x2, int y2, Color c)
        {
            UL = new Point(x1, y1);
            LR = new Point(x2, y2);
            Co = c;

        }

        public override void Draw()
        {
            int width=LR.getX()-UL.getX();
```

```
        int height=LR.getY()-UL.getY();
        Raylib.DrawRectangle(UL.getX(), UL.getY(), width,height, Co);
    }
}
```

Program.cs

```
using Raylib_cs;

namespace ConsoleAppGraphics
{
    internal class Program
    {
        static void Main(string[] args)
        {
            // Console.WriteLine("Hello, World!");
            Raylib.InitWindow(800, 600, "Hello World");

            while (!Raylib.WindowShouldClose())
            {
                Raylib.BeginDrawing();
                Raylib.ClearBackground(Color.White);
                #region test Drawing
                //Raylib.ClearBackground(Color.White);

                //Raylib.DrawText("Hello, world!", 12, 12, 20, Color.B
                //draw a rectangle
                // Raylib.DrawRectangle(100, 100, 200, 150, Color.Red)

                //draw a circle
                //   Raylib.DrawCircle(400, 300, 50, Color.Blue);

                //draw a line
                // Raylib.DrawLine(100, 100, 300, 300, Color.Green);
                //draw a triangle
                //     Raylib.DrawTriangle(new Vector2(400, 100), new

                // Draw a triangle
                // Vector2 p1 = new Vector2(400, 100);
                // Vector2 p2 = new Vector2(350, 200);
                // Vector2 p3 = new Vector2(450, 200);
                // Raylib.DrawTriangle(p1, p2, p3, Color.Green);
                #endregion
```

```
                Circle c = new Circle(70, 70, 100, Color.Red);
                Circle c2 = new Circle(200, 200, 100, Color.Blue);
                Line l = new Line(500, 500, 240, 240,Color.Magenta);
                Rect r = new Rect(10, 10, 150, 100, Color.Yellow);
                Shape[] shapes = new Shape[4];

                shapes[0] = c;
                shapes[1] = l;
                shapes[2] = r;
                shapes[3] = c2;

                DrawShapes(shapes);
                Raylib.EndDrawing();
            }

            Raylib.CloseWindow();
        }
        public static void DrawShapes(Shape[] shapes)
        {
            foreach (Shape s in shapes)
            {
                s.Draw();
            }
        }
    }
}
```