

```
1 using SplashKitSDK;
2
3 namespace DrawingShape
4 {
5     internal class MyRectangle : Shape
6     {
7         private int _width, _height;
8
9         public MyRectangle() : base(color: Color.Green)
10        {
11            _width = 100;
12            _height = 100;
13        }
14
15        public MyRectangle(Color color, int x, int y, int width, int height) : base(color)
16        {
17            Color = color;
18            X = x;
19            Y = y;
20            Width = width;
21            Height = height;
22        }
23
24        public int Width
25        {
26            get { return _width; }
27            set { _width = value; }
28        }
29
30        public int Height
31        {
32            get { return _height; }
33            set { _height = value; }
34        }
35
36        public override void SaveTo(StreamWriter writer)
37        {
38            writer.WriteLine("Rectangle");
39            base.SaveTo(writer);
40            writer.WriteLine(_width);
41            writer.WriteLine(_height);
42        }
43
44        public override void LoadFrom(StreamReader reader)
45        {
46            base.LoadFrom(reader);
47            _width = reader.ReadInteger();
48            _height = reader.ReadInteger();
```

```
49     }
50
51     public override void Draw()
52     {
53         if (Selected)
54         {
55             DrawOutline();
56         }
57         SplashKit.FillRectangle(Color, X, Y, _width, _height);
58     }
59
60     public override void DrawOutline()
61     {
62         SplashKit.FillRectangle(Color.Black, X - 2, Y - 2, _width + 4, ↗
            _height + 4);
63     }
64
65     public override bool IsAt(Point2D pt)
66     {
67         return SplashKit.PointInRectangle(pt, SplashKit.RectangleFrom ↗
            (X, Y, _width, _height));
68     }
69 }
70 }
71
```