

```
1 using SplashKitSDK;
2 using System.ComponentModel;
3 using System.Security.Cryptography.X509Certificates;
4
5 namespace ShapeDrawer
6 {
7     internal class Drawing
8     {
9         private readonly List<Shape> _shapes;
10        private Color _background;
11
12        public Color Background
13        {
14            get => _background;
15            set => _background = value;
16        }
17
18        public Drawing(Color background)
19        {
20            _shapes = new List<Shape>();
21            _background = background;
22        }
23
24        public Drawing() : this (Color.White)
25        {
26            _shapes = new List<Shape>();
27            _background = Color.White;
28        }
29
30        public int ShapeCount
31        {
32            get => _shapes.Count;
33        }
34
35        public void AddShape(Shape shape)
36        {
37            _shapes.Add(shape);
38        }
39
40        public void RemoveShape(Shape shape)
41        {
42            _ = _shapes?.Remove(shape);
43        }
44
45        public void Draw()
46        {
47            SplashKit.ClearScreen(_background);
48            for (int i = 0; i < _shapes.Count; i++)
49            {
```

```
50         if (_shapes[i].Selected)
51         {
52             _shapes[i].DrawOutline();
53         }
54         _shapes[i].Draw();
55     }
56 }
57
58 public void SelectShapesAt(Point2D point)
59 {
60     foreach (Shape s in _shapes)
61     {
62         s.Selected = s.IsAt(point);
63     }
64 }
65
66 public List<Shape> SelectedShapes
67 {
68     get
69     {
70         List<Shape> result = new List<Shape>();
71         foreach (Shape s in _shapes)
72         {
73             if (s.Selected)
74             {
75                 result.Add(s);
76             }
77         }
78         return result;
79     }
80 }
81 }
82 }
83
```