

```
1 using System;
2 using SplashKitSDK;
3
4 namespace ShapeDrawer
5 {
6     public class Program
7     {
8         public static void Main()
9         {
10             Window window = new Window("Shape Drawer", 800, 600);
11             Shape myShape = new Shape();
12             do
13             {
14                 SplashKit.ProcessEvents();
15                 SplashKit.ClearScreen();
16                 if(SplashKit.MouseClicked(MouseButton.LeftButton))
17                 {
18                     myShape.X = SplashKit.MouseX();
19                     myShape.Y = SplashKit.MouseY();
20                 }
21                 Point2D pt = SplashKit.MousePosition();
22                 if (myShape.IsAt(pt) && SplashKit.KeyTyped
23                     (KeyCode.SpaceKey))
24                 {
25                     myShape.Color = SplashKit.RandomRGBColor(255);
26                 }
27                 myShape.Draw();
28                 SplashKit.RefreshScreen();
29             } while (!window.CloseRequested);
30         }
31     }
32 }
```

```
1 using System;
2 using SplashKitSDK;
3
4 namespace ShapeDrawer
5 {
6     public class Shape
7     {
8         private Color _color;
9         private float _x, _y;
10        private int _width, _height;
11
12        public Shape()
13        {
14            _color = Color.Green;
15            _x = 0;
16            _y = 0;
17            _width = 100;
18            _height = 100;
19        }
20        public Color Color
21        {
22            get { return _color; }
23            set { _color = value; }
24        }
25        public float X
26        {
27            get { return _x; }
28            set { _x = value; }
29        }
30        public float Y
31        {
32            get { return _y; }
33            set { _y = value; }
34        }
35        public void Draw()
36        {
37            SplashKit.FillRectangle(_color, _x, _y, _width, _height);
38        }
39        public bool IsAt(Point2D pt)
40        {
41            return SplashKit.PointInRectangle(pt, SplashKit.RectangleFrom
42                (X, Y, _width, _height));
43        }
44    }
45 }
```



