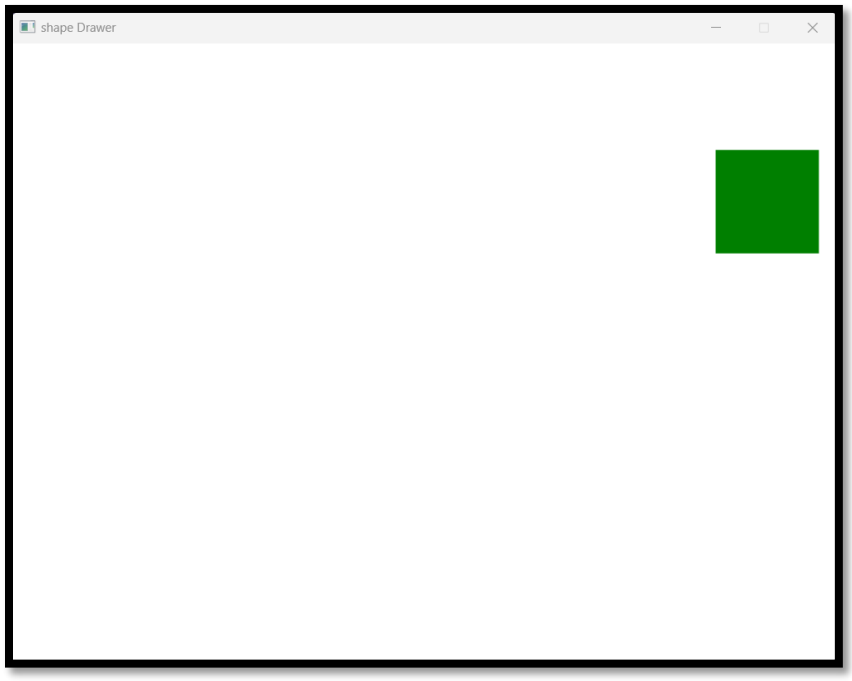
```
1  using System;
2  using SplashKitSDK;
3
4  namespace Drawing
5  {
6      public class Program
7      {
8          public static void Main()
9          {
10             Window window = new Window("shape Drawer", 800, 600);
11             Shape myShape = new Shape();
12             do
13             {
14                 SplashKit.ProcessEvents();
15                 SplashKit.ClearScreen();
16                 if (SplashKit.MouseClicked(MouseButton.LeftButton))
17                 {
18                     myShape.X = (float)SplashKit.MouseX();
19                     myShape.Y = (float)SplashKit.MouseY();
20                 }
21                 if (myShape.IsAt(SplashKit.MousePosition()))
22                 {
23                     if (SplashKit.KeyDown(KeyCode.SpaceKey))
24                     {
25                         myShape.Color = Color.Red;
26                     }
27                     myShape.Draw();
28                 }
29                 SplashKit.RefreshScreen();
30
31             } while (!SplashKit.WindowCloseRequested(window));
32         }
33     }
34 }
35
```

```
 1  using SplashKitSDK;
 2  using System;
 3  using System.Collections.Generic;
 4  using System.Drawing;
 5  using System.Linq;
 6  using System.Text;
 7  using System.Threading.Tasks;
 8
 9  namespace Drawing
10  {
11      public class Shape
12      {
13          private SplashKitSDK.Color _color;
14          private float _x;
15          private float _y;
16          private int _width;
17          private int _height;
18          public Shape()
19          {
20              _color = SplashKitSDK.Color.Green;
21              _x = 0;
22              _y = 0;
23              _width = 100;
24              _height = 100;
25          }
26          public SplashKitSDK.Color Color
27          {
28              get { return _color; }
29              set { _color = value; }
30          }
31
32          public float X
33          {
34              get { return _x; }
35              set { _x = value; }
36          }
37          public float Y
38          {
39              get { return _y; }
40              set { _y = value; }
41          }
42          public void Draw()
43          {
44              SplashKit.FillRectangle(_color, _x, _y, _width, _height);
45          }
46          public bool IsAt(Point2D p)
47          {
48              return SplashKit.PointInRectangle(p, SplashKit.RectangleFrom(X, ↵
                    Y, _width, _height));
```

```
49          }
50
51      }
52  }
53
```

1st Output



2nd Output