

[KingSchlock](#) / [UploadedCodeToPrintAgain](#) Private[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)[main](#) ▾

...

[UploadedCodeToPrintAgain](#) / [Drawing.cs](#) / <> Jump to ▾

KingSchlock Add files via upload



1 contributor

94 lines (79 sloc) | 2.25 KB

...

```
1  using System;
2  using System.Collections.Generic;
3  using SplashKitSDK;
4
5  namespace _4._2P
6  {
7      class Drawing
8      {
9          ///! Fields
10         private readonly List<Shape> _shapes;
11         private Color _background;
12
13
14         ///! Constructors
15         ///? Default constructor, should draw a white background when initialised.
16         public Drawing(Color background)
17         {
18             _background = background;
19             _shapes = new();
20         }
21
22         public Drawing()
23             : this(Color.White)
24         {
25
26         }
27
28         ///! Properties
29         public Color Background
30         {
31             get { return _background; }
32         }
```

```
33         set { _background = value; }
34     }
35
36     ///? Readonly
37     public int ShapeCount
38     {
39         get { return _shapes.Count; }
40     }
41
42     ///? Readonly, adds a selected shape to the selectedShapes array
43     public List<Shape> SelectedShapes
44     {
45         get
46         {
47             List<Shape> selectedShapes = new();
48
49             foreach(Shape genericShape in _shapes)
50             {
51                 if (genericShape.Selected)
52                 {
53                     selectedShapes.Add(genericShape);
54                 }
55             }
56             return selectedShapes;
57         }
58     }
59
60     ///! Methods and Fields
61     public void AddShape(Shape genericShape)
62     {
63         _shapes.Add(genericShape);
64     }
65
66     public void RemoveShape(Shape genericShape)
67     {
68         _shapes.Remove(genericShape);
69     }
70
71     ///? Turns selected to true if shape is at mouselocation
72     public void SelectShapesAt(Point2D mouseLocation)
73     {
74         foreach(Shape genericShape in _shapes)
75         {
76             if (!genericShape.Selected)
77             {
78                 genericShape.Selected = genericShape.IsAt(mouseLocation);
79             }
80         }
81     }
82
83     ///? Draw da shapes
84     public void Draw()
```

```
85     {
86         SplashKit.ClearScreen(Background);
87
88         foreach (Shape genericShape in _shapes)
89         {
90             genericShape.Draw();
91         }
92     }
93 }
94 }
```