

SWINBURNE UNIVERSITY OF TECHNOLOGY

COS20007 OBJECT ORIENTED PROGRAMMING

Drawing Program - A Basic Shape

PDF generated at 19:38 on Monday 14th August, 2023

```
1  using System;
2  using SplashKitSDK;
3
4  namespace ShapeDrawer
5  {
6      public class Program
7      {
8          public static void Main()
9          {
10             // Create a window for the shape drawer
11             Window window = new Window("Shape Drawer", 800, 600);
12             Shape myShape = new Shape(Color.Green, 0, 0, 100, 100);
13             do
14             {
15                 // Processes events from user input
16                 SplashKit.ProcessEvents();
17                 SplashKit.ClearScreen();
18
19                 Point2D pt = SplashKit.mousePosition();
20
21                 // Set x to be mouseX and y to be MouseY if mouse clicked is true
22                 if (SplashKit.MouseClicked(MouseButton.LeftButton))
23                 {
24                     myShape.X = SplashKit.MouseX();
25                     myShape.Y = SplashKit.MouseY();
26                 }
27
28                 if (myShape.IsAt(pt) && SplashKit.KeyTyped(KeyCode.SpaceKey))
29                 {
30                     myShape.color = Color.RandomRGB(255);
31                 }
32
33                 myShape.Draw();
34
35                 SplashKit.RefreshScreen();
36             } while (!window.CloseRequested);
37         }
38     }
39 }
```

```
1  using SplashKitSDK;
2
3  namespace ShapeDrawer
4  {
5      public class Shape
6      {
7          private Color _color;
8          private float _x;
9          private float _y;
10         private int _width;
11         private int _height;
12
13         public Shape(Color color, float x, float y, int width, int height)
14         {
15             _color = color;
16             _x = x;
17             _y = y;
18             _width = width;
19             _height = height;
20         }
21
22         public Color color
23         {
24             get { return _color; }
25             set { _color = value; }
26         }
27
28         public float X
29         {
30             get { return _x; }
31             set { _x = value; }
32         }
33
34         public float Y
35         {
36             get { return _y; }
37             set { _y = value; }
38         }
39
40         public int Width
41         {
42             get { return _width; }
43             set { _width = value; }
44         }
45
46         public int Height
47         {
48             get { return _height; }
49             set { _height = value; }
50         }
51
52         public void Draw()
53         {
```

```
54         SplashKit.FillRectangle(_color, _x, _y, _width, _height);
55     }
56     // Check if a point is within the boundaries of the shape
57     public bool IsAt(Point2D pt)
58     {
59
60         if (_x < pt.X && pt.X < (_x + _width) && _y < pt.Y && pt.Y < (_y +
61             _height))
62         {
63             return true;
64         }
65         else
66         {
67             return false;
68         }
69     }
70 }
```

The screenshot shows a software development environment with a code editor and a terminal window.

Solution Explorer: Shows a project named "ShapeDrawer" with files: Program.cs, Shape.cs, and myeasylog.log.

Code Editor: Displays the content of Program.cs. The code is as follows:

```
1  using System;
2  using SplashKitSDK;
3
4  namespace ShapeDrawer
5  {
6      public class Program
7      {
8          public static void Main()
9          {
10             // Create a window for the shape drawer
11             Window w = new Window("Shape Drawer");
12             Shape myShape = new Shape();
13             do
14             {
15                 // Paint the splash screen
16                 SplashKitSDK.Splash();
17                 Point p = myShape.GetPosition();
18                 // Set the position of the shape
19                 if (p.X < 0)
20                 {
21                     p.X = 0;
22                 }
23                 if (p.Y < 0)
24                 {
25                     p.Y = 0;
26                 }
27                 if (p.X > 800)
28                 {
29                     p.X = 800;
30                 }
31                 if (p.Y > 600)
32                 {
33                     p.Y = 600;
34                 }
35                 myShape.SetPosition(p);
36             } while (true);
37         }
38     }
39 }
40
```

Terminal: Shows the command "git status" output:

```
master master no changes Build successful.
```