# [KingSchlock](https://github.com/KingSchlock) / [**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)

**Private**

[**Code**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)[Issues](https://github.com/KingSchlock/UploadedCodeToPrintAgain/issues) [Pull requests](https://github.com/KingSchlock/UploadedCodeToPrintAgain/pulls) [Actions](https://github.com/KingSchlock/UploadedCodeToPrintAgain/actions) [Projects](https://github.com/KingSchlock/UploadedCodeToPrintAgain/projects) [Security](https://github.com/KingSchlock/UploadedCodeToPrintAgain/security) [Insights](https://github.com/KingSchlock/UploadedCodeToPrintAgain/network/dependencies) [Settings](https://github.com/KingSchlock/UploadedCodeToPrintAgain/settings)



**main**

[**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain) / **Shape.cs** / Jump to



**1** contributor

[**KingSchlock**](https://github.com/KingSchlock) [Add files via upload](https://github.com/KingSchlock/UploadedCodeToPrintAgain/commit/d2f8a51882f3a5733ba113f072ab1963db790164)

|  |  |
| --- | --- |
| 68 lines (55 sloc) 1.58 KB | |
| 1 | using SplashKitSDK; |
| 2 |  |
| 3 | namespace \_4.\_2P |
| 4 | { |
| 5 | public abstract class Shape |
| 6 | { |
| 7 | //! Field Declerations |
| 8 | private Color \_color; |
| 9 | private float \_x, \_y; |
| 10 | private bool \_selected; |
| 11 |  |
| 12 | //! Constructors |
| 13 | public Shape() |
| 14 | : this(Color.Red, false) |
| 15 | { |
| 16 |  |
| 17 | } |
| 18 |  |
| 19 | public Shape(Color color, bool selected) |
| 20 | { |
| 21 | this.\_color = color; |
| 22 | this.\_selected = selected; |
| 23 | } |
| 24 |  |
| 25 | public Shape(Color color, float x, float y, bool selected) |
| 26 | : this(color, selected) |
| 27 | { |
| 28 | this.\_x = x; |
| 29 | this.\_y = x; |
| 30 | } |
| 31 |  |
| 32 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 33 | |  |  | //! Properties | |
| 34 | |  |  | public Color Color | |
| 35 | |  |  | { | |
| 36 | |  |  | get { return this.\_color; } | |
| 37 | |  |  | set {this.\_color = value; } | |
| 38 | |  |  | } | |
| 39 | |  |  |  | |
| 40 | |  |  | public float X | |
| 41 | |  |  | { | |
| 42 | |  |  | get { return this.\_x; } | |
| 43 | |  |  | set { this.\_x = value; } | |
| 44 | |  |  | } | |
| 45 | |  |  |  | |
| 46 | |  |  | public float Y | |
| 47 | |  |  | { | |
| 48 | |  |  | get { return this.\_y; } | |
| 49 | |  |  | set { this.\_y = value; } | |
| 50 | |  |  | } | |
| 51 | |  |  |  | |
| 52 | |  |  | public bool Selected | |
| 53 | |  |  | { | |
| 54 | |  |  | get { return this.\_selected; } | |
| 55 | |  |  | set { this.\_selected = value; } | |
| 56 | |  |  | } | |
| 57 | |  |  |  | |
| 58 | |  |  | //! Methods | |
| 59 | |  |  | //? Takes a point and determines if said point lies within our shape | |
| 60 | |  |  | public abstract bool IsAt(Point2D mouseLocation); | |
| 61 | |  |  |  | |
| 62 | |  |  | //? Draws the outline of a rectangle | |
| 63 | |  |  | public abstract void DrawOutline(); | |
| 64 | |  |  |  | |
| 65 | |  |  | //? Draws a Rectangle based on parameters and outlines the rectangle if the shape is s | |
| 66 | |  |  | public abstract void Draw(); | |
| 67 | |  | } |  | |
| 68 | | } |  |  | |
|  |  | | | |  |

# [KingSchlock](https://github.com/KingSchlock) / [**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)

**Private**

[**Code**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)[Issues](https://github.com/KingSchlock/UploadedCodeToPrintAgain/issues) [Pull requests](https://github.com/KingSchlock/UploadedCodeToPrintAgain/pulls) [Actions](https://github.com/KingSchlock/UploadedCodeToPrintAgain/actions) [Projects](https://github.com/KingSchlock/UploadedCodeToPrintAgain/projects) [Security](https://github.com/KingSchlock/UploadedCodeToPrintAgain/security) [Insights](https://github.com/KingSchlock/UploadedCodeToPrintAgain/network/dependencies) [Settings](https://github.com/KingSchlock/UploadedCodeToPrintAgain/settings)



**main**

[**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain) / **MyRectangle.cs** / Jump to



**1** contributor

[**KingSchlock**](https://github.com/KingSchlock) [Add files via upload](https://github.com/KingSchlock/UploadedCodeToPrintAgain/commit/d2f8a51882f3a5733ba113f072ab1963db790164)

|  |  |
| --- | --- |
| 63 lines (53 sloc) 1.46 KB | |
| 1 | using SplashKitSDK; |
| 2 |  |
| 3 | namespace \_4.\_2P |
| 4 | { |
| 5 | public class MyRectangle : Shape |
| 6 | { |
| 7 | //! Fields |
| 8 | private int \_width, \_height; |
| 9 |  |
| 10 | //! Constructor(s) |
| 11 | public MyRectangle() |
| 12 | { |
| 13 |  |
| 14 | } |
| 15 |  |
| 16 | public MyRectangle(Color color, float x, float y, bool selected, int width, int height |
| 17 | { |
| 18 | \_width = width; |
| 19 | \_height = height; |
| 20 | } |
| 21 |  |
| 22 | //! Properties |
| 23 | public int Width |
| 24 | { |
| 25 | get { return \_width; } |
| 26 | set { \_width = value; } |
| 27 | } |
| 28 |  |
| 29 |  |
| 30 | public int Height |
| 31 | { |
| 32 | get { return \_height; } |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 33 | |  |  | set { \_height = value; } | |
| 34 | |  |  | } | |
| 35 | |  |  |  | |
| 36 | |  |  | //! Method(s) | |
| 37 | |  |  | public override bool IsAt(Point2D mouseLocation) | |
| 38 | |  |  | { | |
| 39 | |  |  | if (X < mouseLocation.X && mouseLocation.X < (X + Width) && Y < mouseLocation.Y && | |
| 40 | |  |  | { | |
| 41 | |  |  | return true; | |
| 42 | |  |  | } | |
| 43 | |  |  | else | |
| 44 | |  |  | { | |
| 45 | |  |  | return false; | |
| 46 | |  |  | } | |
| 47 | |  |  | } | |
| 48 | |  |  | public override void DrawOutline() | |
| 49 | |  |  | { | |
| 50 | |  |  | SplashKit.DrawRectangle(Color.Black, (X - 2), (Y - 2), (Width + 4), (Height + 4)); | |
| 51 | |  |  | } | |
| 52 | |  |  |  | |
| 53 | |  |  | public override void Draw() | |
| 54 | |  |  | { | |
| 55 | |  |  | SplashKit.FillRectangle(Color, X, Y, Width, Height); | |
| 56 | |  |  |  | |
| 57 | |  |  | if (Selected) | |
| 58 | |  |  | { | |
| 59 | |  |  | DrawOutline(); | |
| 60 | |  |  | } | |
| 61 | |  |  | } | |
| 62 | |  | } |  | |
| 63 | | } |  |  | |
|  |  | | | |  |

# [KingSchlock](https://github.com/KingSchlock) / [**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)

**Private**

[**Code**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)[Issues](https://github.com/KingSchlock/UploadedCodeToPrintAgain/issues) [Pull requests](https://github.com/KingSchlock/UploadedCodeToPrintAgain/pulls) [Actions](https://github.com/KingSchlock/UploadedCodeToPrintAgain/actions) [Projects](https://github.com/KingSchlock/UploadedCodeToPrintAgain/projects) [Security](https://github.com/KingSchlock/UploadedCodeToPrintAgain/security) [Insights](https://github.com/KingSchlock/UploadedCodeToPrintAgain/network/dependencies) [Settings](https://github.com/KingSchlock/UploadedCodeToPrintAgain/settings)



**main**

[**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain) / **MyCircle.cs** / Jump to



**1** contributor

[**KingSchlock**](https://github.com/KingSchlock) [Add files via upload](https://github.com/KingSchlock/UploadedCodeToPrintAgain/commit/d2f8a51882f3a5733ba113f072ab1963db790164)

|  |  |
| --- | --- |
| 55 lines (47 sloc) 1.33 KB | |
| 1 | using SplashKitSDK; |
| 2 |  |
| 3 | namespace \_4.\_2P |
| 4 | { |
| 5 | public class MyCircle : Shape |
| 6 | { |
| 7 | //! Fields |
| 8 | private int \_radius; |
| 9 |  |
| 10 | //! Constructor(s) |
| 11 | public MyCircle() |
| 12 | { |
| 13 |  |
| 14 | } |
| 15 |  |
| 16 | public MyCircle(Color color, float x, float y, bool selected, int radius) : base(color |
| 17 | { |
| 18 | this.\_radius = radius; |
| 19 | } |
| 20 |  |
| 21 | //! Properties |
| 22 | public int Radius |
| 23 | { |
| 24 | get { return \_radius; } |
| 25 | set { \_radius = value; } |
| 26 | } |
| 27 |  |
| 28 | //! Method(s) |
| 29 | public override bool IsAt(Point2D mouseLocation) |
| 30 | { |
| 31 | Point2D origin = new() |
| 32 | { |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 33 | |  |  | //Had to set the points or they kept changing with the mouse location, real pr | |
| 34 | |  |  | X = X, | |
| 35 | |  |  | Y = Y | |
| 36 | |  |  | }; | |
| 37 | |  |  |  | |
| 38 | |  |  | Circle circle = SplashKit.CircleAt(origin, \_radius); | |
| 39 | |  |  | return SplashKit.PointInCircle(mouseLocation, circle); | |
| 40 | |  |  | } | |
| 41 | |  |  |  | |
| 42 | |  |  | public override void DrawOutline() | |
| 43 | |  |  | { | |
| 44 | |  |  | SplashKit.DrawCircle(Color.Black, X, Y, (Radius + 2)); | |
| 45 | |  |  | } | |
| 46 | |  |  | public override void Draw() | |
| 47 | |  |  | { | |
| 48 | |  |  | SplashKit.FillCircle(Color, X, Y, Radius); | |
| 49 | |  |  | if (Selected) | |
| 50 | |  |  | { | |
| 51 | |  |  | DrawOutline(); | |
| 52 | |  |  | } | |
| 53 | |  |  | } | |
| 54 | |  | } |  | |
| 55 | | } |  |  | |
|  |  | | | |  |

# [KingSchlock](https://github.com/KingSchlock) / [**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)

**Private**

[**Code**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)[Issues](https://github.com/KingSchlock/UploadedCodeToPrintAgain/issues) [Pull requests](https://github.com/KingSchlock/UploadedCodeToPrintAgain/pulls) [Actions](https://github.com/KingSchlock/UploadedCodeToPrintAgain/actions) [Projects](https://github.com/KingSchlock/UploadedCodeToPrintAgain/projects) [Security](https://github.com/KingSchlock/UploadedCodeToPrintAgain/security) [Insights](https://github.com/KingSchlock/UploadedCodeToPrintAgain/network/dependencies) [Settings](https://github.com/KingSchlock/UploadedCodeToPrintAgain/settings)



**main**

[**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain) / **MyLine.cs** / Jump to



**1** contributor

[**KingSchlock**](https://github.com/KingSchlock) [Add files via upload](https://github.com/KingSchlock/UploadedCodeToPrintAgain/commit/d2f8a51882f3a5733ba113f072ab1963db790164)

|  |  |
| --- | --- |
| 66 lines (54 sloc) 1.49 KB | |
| 1 | using SplashKitSDK; |
| 2 | using System; |
| 3 |  |
| 4 | namespace \_4.\_2P |
| 5 | { |
| 6 | public class MyLine : Shape |
| 7 | { |
| 8 | //! Fields |
| 9 | private float \_length; |
| 10 |  |
| 11 | //! Constructors |
| 12 | public MyLine() |
| 13 | : this(Color.BlueViolet, 0, 0, false, 150) |
| 14 | { |
| 15 |  |
| 16 | } |
| 17 |  |
| 18 | public MyLine(Color color, float x, float y, bool selected, float length) : base(color |
| 19 | { |
| 20 | this.\_length = length; |
| 21 | } |
| 22 |  |
| 23 | //! Properties |
| 24 | public float Length |
| 25 | { |
| 26 | get { return \_length; } |
| 27 | set { \_length = value; } |
| 28 | } |
| 29 |  |
| 30 |  |
| 31 | //! Methods |
| 32 | public override bool IsAt(Point2D mouseLocation) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 33 | |  |  | { |  |
| 34 | |  |  | Point2D initialPoint = new() |  |
| 35 | |  |  | { |  |
| 36 | |  |  | X = X, |  |
| 37 | |  |  | Y = Y |  |
| 38 | |  |  | }; |  |
| 39 | |  |  |  |  |
| 40 | |  |  | Point2D finalPoint = new() |  |
| 41 | |  |  | { |  |
| 42 | |  |  | X = X + Length, |  |
| 43 | |  |  | Y = Y |  |
| 44 | |  |  | }; |  |
| 45 | |  |  |  |  |
| 46 | |  |  | Line line = SplashKit.LineFrom(initialPoint, finalPoint); |  |
| 47 | |  |  | return SplashKit.PointOnLine(mouseLocation, line); |  |
| 48 | |  |  | } |  |
| 49 | |  |  |  |  |
| 50 | |  |  | public override void DrawOutline() |  |
| 51 | |  |  | { |  |
| 52 | |  |  | SplashKit.DrawCircle(Color.GhostWhite, X, Y, 2); |  |
| 53 | |  |  | SplashKit.DrawCircle(Color.GhostWhite, X + Length, Y, 2); |  |
| 54 | |  |  | } |  |
| 55 | |  |  |  |  |
| 56 | |  |  | public override void Draw() |  |
| 57 | |  |  | { |  |
| 58 | |  |  | SplashKit.DrawLine(Color, X, Y, (X + Length), Y); |  |
| 59 | |  |  |  |  |
| 60 | |  |  | if (Selected) |  |
| 61 | |  |  | { |  |
| 62 | |  |  | DrawOutline(); |  |
| 63 | |  |  | } |  |
| 64 | |  |  | } |  |
| 65 | |  | } |  |  |
| 66 | | } |  |  |  |
|  |  | | | |  |

# [KingSchlock](https://github.com/KingSchlock) / [**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)

**Private**

[**Code**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)[Issues](https://github.com/KingSchlock/UploadedCodeToPrintAgain/issues) [Pull requests](https://github.com/KingSchlock/UploadedCodeToPrintAgain/pulls) [Actions](https://github.com/KingSchlock/UploadedCodeToPrintAgain/actions) [Projects](https://github.com/KingSchlock/UploadedCodeToPrintAgain/projects) [Security](https://github.com/KingSchlock/UploadedCodeToPrintAgain/security) [Insights](https://github.com/KingSchlock/UploadedCodeToPrintAgain/network/dependencies) [Settings](https://github.com/KingSchlock/UploadedCodeToPrintAgain/settings)



**main**

[**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain) / **Drawing.cs** / Jump to



**1** contributor

[**KingSchlock**](https://github.com/KingSchlock) [Add files via upload](https://github.com/KingSchlock/UploadedCodeToPrintAgain/commit/d2f8a51882f3a5733ba113f072ab1963db790164)

|  |  |
| --- | --- |
| 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 |  |
| 29 | //! Properties |
| 30 | public Color Background |
| 31 | { |
| 32 | get { return \_background; } |

|  |  |
| --- | --- |
| 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 | } |  |  |  |

# [KingSchlock](https://github.com/KingSchlock) / [**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)

**Private**

[**Code**](https://github.com/KingSchlock/UploadedCodeToPrintAgain)[Issues](https://github.com/KingSchlock/UploadedCodeToPrintAgain/issues) [Pull requests](https://github.com/KingSchlock/UploadedCodeToPrintAgain/pulls) [Actions](https://github.com/KingSchlock/UploadedCodeToPrintAgain/actions) [Projects](https://github.com/KingSchlock/UploadedCodeToPrintAgain/projects) [Security](https://github.com/KingSchlock/UploadedCodeToPrintAgain/security) [Insights](https://github.com/KingSchlock/UploadedCodeToPrintAgain/network/dependencies) [Settings](https://github.com/KingSchlock/UploadedCodeToPrintAgain/settings)



**main**

[**UploadedCodeToPrintAgain**](https://github.com/KingSchlock/UploadedCodeToPrintAgain) / **Program.cs** / Jump to



**1** contributor

[**KingSchlock**](https://github.com/KingSchlock) [Add files via upload](https://github.com/KingSchlock/UploadedCodeToPrintAgain/commit/d2f8a51882f3a5733ba113f072ab1963db790164)

|  |  |
| --- | --- |
| 107 lines (89 sloc) 3.53 KB | |
| 1 | using SplashKitSDK; |
| 2 | using System; |
| 3 |  |
| 4 | namespace \_4.\_2P |
| 5 | { |
| 6 | public class Program |
| 7 | { |
| 8 | private enum ShapeKind |
| 9 | { |
| 10 | Rectangle, |
| 11 | Circle, |
| 12 | Line |
| 13 | } |
| 14 |  |
| 15 | public static void Main() |
| 16 | { |
| 17 | Window window = new("4.2P: Thomas Horsley - 103071494", 800, 600); |
| 18 | Drawing drawing = new(); |
| 19 |  |
| 20 | ShapeKind kindToAdd = ShapeKind.Circle; |
| 21 |  |
| 22 | do |
| 23 | { |
| 24 | SplashKit.ProcessEvents(); |
| 25 | SplashKit.ClearScreen(); |
| 26 |  |
| 27 | Point2D mouseLocation = SplashKit.MousePosition(); |
| 28 |  |
| 29 | //? Draws a shape at mouse position and adds it to a shapes array |
| 30 | if (SplashKit.MouseClicked(MouseButton.LeftButton)) |
| 31 | { |
| 32 | if (kindToAdd == ShapeKind.Rectangle) |

|  |  |
| --- | --- |
| 33 | { |
| 34 | Shape rectangleShape = new MyRectangle(Color.Green, 0, 0, false, 100, |
| 35 | { |
| 36 | X = (float)mouseLocation.X, |
| 37 | Y = (float)mouseLocation.Y |
| 38 | }; |
| 39 |  |
| 40 | drawing.AddShape(rectangleShape); |
| 41 | } |
| 42 | if (kindToAdd == ShapeKind.Circle) |
| 43 | { |
| 44 | Shape circleShape = new MyCircle(Color.Red, 0, 0, false, 50) |
| 45 | { |
| 46 | X = (float)mouseLocation.X, |
| 47 | Y = (float)mouseLocation.Y |
| 48 | }; |
| 49 |  |
| 50 | drawing.AddShape(circleShape); |
| 51 | } |
| 52 | if(kindToAdd == ShapeKind.Line) |
| 53 | { |
| 54 | Shape lineShape = new MyLine(Color.GreenYellow, 0, 0, false, 50) |
| 55 | { |
| 56 | X = (float)mouseLocation.X, |
| 57 | Y = (float)mouseLocation.Y, |
| 58 | }; |
| 59 |  |
| 60 | drawing.AddShape(lineShape); |
| 61 | } |
| 62 | } |
| 63 |  |
| 64 | //? Relates keystrokes to shape kinds |
| 65 | if (SplashKit.KeyReleased(KeyCode.RKey)) |
| 66 | { |
| 67 | kindToAdd = ShapeKind.Rectangle; |
| 68 | } |
| 69 | else if (SplashKit.KeyReleased(KeyCode.CKey)) |
| 70 | { |
| 71 | kindToAdd = ShapeKind.Circle; |
| 72 | } |
| 73 | else if (SplashKit.KeyReleased(KeyCode.LKey)) |
| 74 | { |
| 75 | kindToAdd = ShapeKind.Line; |
| 76 | } //TODO <--- can i use cases instead? |
| 77 |  |
| 78 |  |
| 79 | //? Checks if shape is selected |
| 80 | if (SplashKit.MouseClicked(MouseButton.RightButton)) |
| 81 | { |
| 82 | drawing.SelectShapesAt(mouseLocation); |
| 83 | } |
| 84 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 85 | |  |  |  |  | |
| 86 | |  |  |  | //? Changes background color when user presses space | |
| 87 | |  |  |  | if (SplashKit.KeyReleased(KeyCode.SpaceKey)) | |
| 88 | |  |  |  | { | |
| 89 | |  |  |  | drawing.Background = SplashKit.RandomRGBColor(255); | |
| 90 | |  |  |  | } | |
| 91 | |  |  |  |  | |
| 92 | |  |  |  | if (SplashKit.KeyReleased(KeyCode.DeleteKey) || SplashKit.KeyReleased(KeyCode. | |
| 93 | |  |  |  | { | |
| 94 | |  |  |  | foreach(Shape genericShape in drawing.SelectedShapes) | |
| 95 | |  |  |  | { | |
| 96 | |  |  |  | drawing.RemoveShape(genericShape); | |
| 97 | |  |  |  | } | |
| 98 | |  |  |  | } | |
| 99 | |  |  |  |  | |
| 100 | |  |  |  | drawing.Draw(); | |
| 101 | |  |  |  | SplashKit.RefreshScreen(60); | |
| 102 | |  |  |  | } while (!window.CloseRequested); | |
| 103 | |  |  | } |  | |
| 104 | |  | } |  |  | |
| 105 | | } |  |  |  | |
| 106 | |  |  |  |  | |
| 107 | |  |  |  |  | |
|  |  | | | | |  |