

```

1 /**
2  * Clickable Button
3  * Includes all functions from interactive object
4  * @param {number} x x
5  * @param {number} y y
6  * @param {number} w width
7  * @param {number} h height
8  * @param {string} fill fill colour
9  * @param {string} over hover over colour
10 * @param {string} selected button has been clicked colour
11 * @param {string} stroke stroke colour
12 * @param {string} text button text
13 * @param {string} text button text colour
14 */
15 class ButtonBase extends InteractiveObject{
16     constructor(x,y, w, h, fill, over, selected, stroke, text, textColour){
17         super()
18         this.x = x;
19         this.y = y;
20         this.w = w;
21         this.h = h;
22         this.fill = fill;
23         this.over=over;
24         this.selected = selected;
25         this.currentFill = fill
26         this.stroke=stroke;
27         this.text = text;
28         this.textColour = textColour
29         this.inBounds = false
30     }
31     mUp(e) {
32         super.mUp(e);
33         console.log("up call from button")
34     }
35
36     update(){
37         this.inBounds = this.getBoundary(this.x, this.y, this.w, this.h, this.
xMouse, this.yMouse)
38         let fill = this.currentFill
39         this.draw(this.x, this.y, this.w, this.h, fill, this.stroke, this.text,
this.textColour )
40
41         if(this.inBounds){
42             this.currentFill = this.over
43         }else{
44             this.currentFill = this.fill
45         }

```

```

46
47     }
48     mClick() {}
49
50     draw(x,y, w,h,fillColour,strokeColour, txt, txtCol){
51         this.strokeFillRect(x,y,w,h,fillColour, strokeColour, 0.5)
52         this.centredText(txt, x+w/2, y+h/2, txtCol)
53
54     }
55 }
56 ButtonBase.prototype.strokeFillRect = strokeFillRect
57 ButtonBase.prototype.centredText = centredText
58
59
60 /**
61  * Basic, selectable options button
62  * Includes all functions from interactive object and button base
63  * @param {number} x x
64  * @param {number} y y
65  * @param {number} w width
66  * @param {number} h height
67  * @param {string} fill fill colour
68  * @param {string} over hover over colour
69  * @param {string} selected button has been clicked colour
70  * @param {string} stroke stroke colour
71  * @param {string} text button text
72  * @param {string} text button text colour
73  */
74 class OptionsButton extends ButtonBase{
75     constructor(x,y, w, h, fill, over, selected, stroke, text, textColour) {
76         super(x,y, w, h, fill, over, selected, stroke, text, textColour);
77     }
78     mClick() {
79
80         if(this.inBounds){
81             console.log("Click call from button:" + this.text)
82             OptionsButton.selected = this;
83         }
84     }
85     update(){
86         super.update()
87         if(OptionsButton.selected === this){
88             this.currentFill = this.selected
89         }
90     }
91 }
92 OptionsButton.selected = null;

```

```

93
94 // noinspection DuplicatedCode
95 class SwatchButton extends InteractiveObject{
96     constructor(x,y,r,fillColour, strokeColour, strokeOverColour, strokeWidth) {
97         super()
98         this.x = x;
99         this.y = y;
100        this.r = r;
101        this.fillColour = fillColour;
102        this.strokeColour = strokeColour;
103        this.strokeOverColour = strokeOverColour;
104        this.strokeWidth = strokeWidth;
105        this.inBounds = false
106    }
107    mClick(e) {
108        super.mClick(e);
109        if(this.inBounds){
110            SwatchButton.colour = this.fillColour
111        }
112    }
113 }
114
115 update(){
116     this.draw()
117     this.inBounds=this.getDistance(this.x, this.y, this.xMouse, this.yMouse,
118 this.r)
119 }
120 draw(){
121     let currentStroke = this.strokeColour
122     if(this.inBounds){
123         currentStroke = this.strokeOverColour
124     }
125     if(this.fillColour === SwatchButton.colour){
126         currentStroke = this.fillColour
127     }
128     this.filledStrokedCircle(this.x,this.y,this.r, this.fillColour,
129 currentStroke, this.strokeWidth)
130 }
131 SwatchButton.prototype.filledStrokedCircle = drawFilledStrokeCircle
132 SwatchButton.colour = "rgb(0,0,0)"

```

Listing 1: Buttons