

```
1 // an unusual way to create shaped buttons with Processing
2
3 PImage button; //image of the buttons (visible)
4 PImage buttoncolmap; //colormap of the buttons (hidden)
5
6 String[] buttonstring={"quadrat","thumb up","smile","envelope",
7 "flower","rectangle","oval","fivestar",
8 "circle","four star","banner","boom",
9 "textfield",""}; //the names of the buttons
10 color[] buttoncolor={0xFF0000FF,0xFF00FFFF,0xFF00FF00,0xFFFFFFFF00,
11 0xFFFF0000,0xFFFF00FF,0xFF9900CC,0xFFFF6600,
12 0xFFFF99CC,0xFF663333,0xFF66FFCC,0xFF99FF00,
13 0xFFFF66,0xFFFFFFFF}; //the colors of the buttons
14
15 String textfield="";
16
17 PFont font;
18
19 void setup ()
20 {
21 size(800,600, JAVA2D);
22 smooth();
23 noLoop();
24 //noCursor();
25 font=loadFont("CourierNewPS-BoldMT-48.vlw");
26 textFont(font);
27 button=loadImage("button.png");
28 buttoncolmap=loadImage("colormap.png");
29 }
30
31 void draw()
32 {
33 background(button);
34 fill(0);
35 textAlign(CENTER);
36 text(textfield,400,205);
37 }
38
39 void mouseReleased()
40 {
41 color testcolor=0;
42 testcolor=buttoncolmap.get(mouseX,mouseY); //get the color in the hidden image
43 //println("0x"+hex(testcolor));
44
45 for(int i=0;i<buttonstring.length;++i) //check the color and copy the name of
the button
46 {
47 if(testcolor==buttoncolor[i])
48 textfield=buttonstring[i];
49 }
50
51 redraw();
52 }
53
54
```