

NICHOLAS

slave2thealgod

nicholaswinter

winter

nicholas
winter

2



et

OC

mas

nicholaswinter
semester



ARTISTS STATEMENT

INSPIRATION

TRIAL & ERROR

CODE

REFLECTION

GITHUB LINKS

CONTENTS

slave2al

3-5

6-7

8-10

11-12

13

15

P

SLAVE2AL

n

L S
F C
E C

N

D V

statement

My personal design practice focuses on unconventional materials, and methods to achieving graphics. Heavily inspired by graphic artists such as David Carson, Skolos-Wedell, Saul Bass, even Rothko and or Twombly, I wanted to create a system that looks to further aid this unorthodox process. The nature of code was intimidating, challenging yet intriguing - I soon quickly found the process to be both illuminating and gruelling. 'Slave to the Algorithm' and or, 'Slave to your own generative system', each of which I've found to be very true as this non-linear route of design has demonstrated. Ending with this typographic experimentation I've learnt the fundamentals and basics of code; it being a form and combination of languages that mathematically dictate visual communications.

statement

STATEMENT

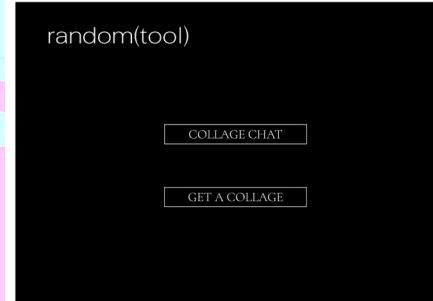
slave2al

artists
artists
artists
artists

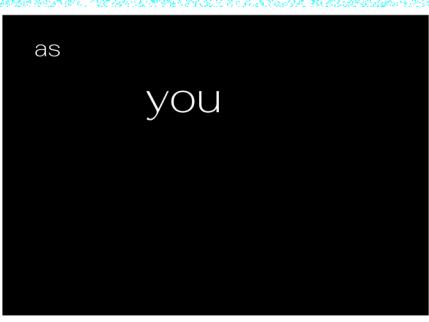
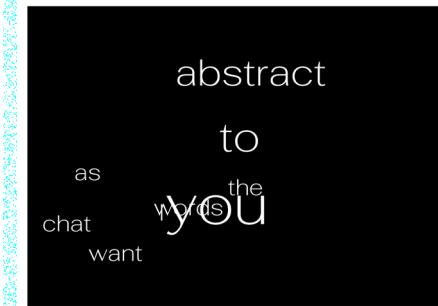
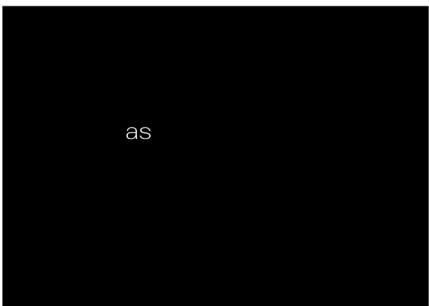
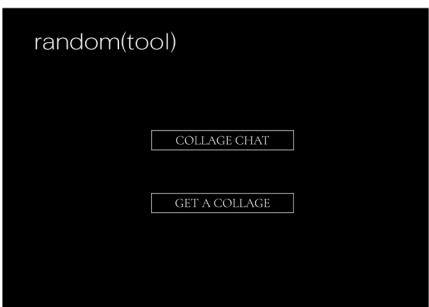
possible outcome

Have you ever struggled with breaking rigid type set structures and grids? Say no more. This design system allows for an abstraction of elements provided by the user. Inserting textures, pictures, type etc... gives any user the tools to create interesting randomised graphics and or type setting choices that they wouldn't usually use. It demonstrates a carefree unity between generative design, and the user and gives way to innovative, randomised expression. Personally, I struggle deterring from the safe side of structure, so this system can develop those gazes, and further my design practice.

Visually, the landing page, and selection process is straight forward and doesn't demonstrate 'expression' yet allows for legible navigation for initial use. With two options, both 'collage chat' and 'collage make', the intention is to seek graphically interesting shapes with type. Through either the typed option or the spoken, they both have their own 'voice' and 'purpose', however I'm still yet to figure the 'collage chat' function.



5



e

Visualising type as you speak forces the user to observe the real-time shapes, and organic symbols, unusual type can form when the user has no control... The idea is that you create a pile of words, that you have then shift like puzzle pieces, and slowly blend together words that catch your attention most; A live brainstorm of sorts.

The alter, 'collage make', is visualising type after it's input screen, where you're given more time to plan, and scheme. I've managed to add 4 font styles into my current code, however I was not able to figure an 'upload image', or a working enlarge 'slider'. These are elements I hope to figure in the near future.

carson

carson david

inspiration



slave

Specifically for this project, I sought to create work that could replicate the chaotic elements of David Carson, Cy Twombly, Skolos-Weddell and Jackson Pollocks work. Although far from these kinds of conceptual beauties, the system is unfinished and requires a fine tooth comb. Their abstract nature lends itself to the process this system undertakes. Another fault is the lack of angular type this system has, in its current horizontal profile, which is a prominent feature in David Carsons work.

pV

pc

isla

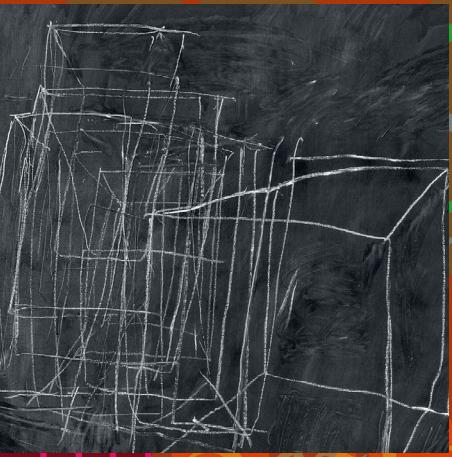
inspiration



Dutch Interior, 1962



Blooming, 2001-2008



Night watch, 1966

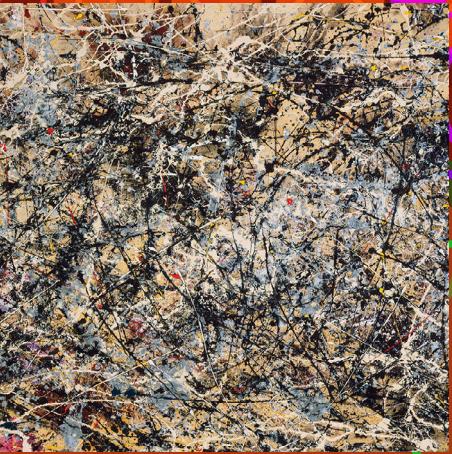
Cy Twombly



Mural, 1946



Red Composition, 1946



Number 1A, 1948

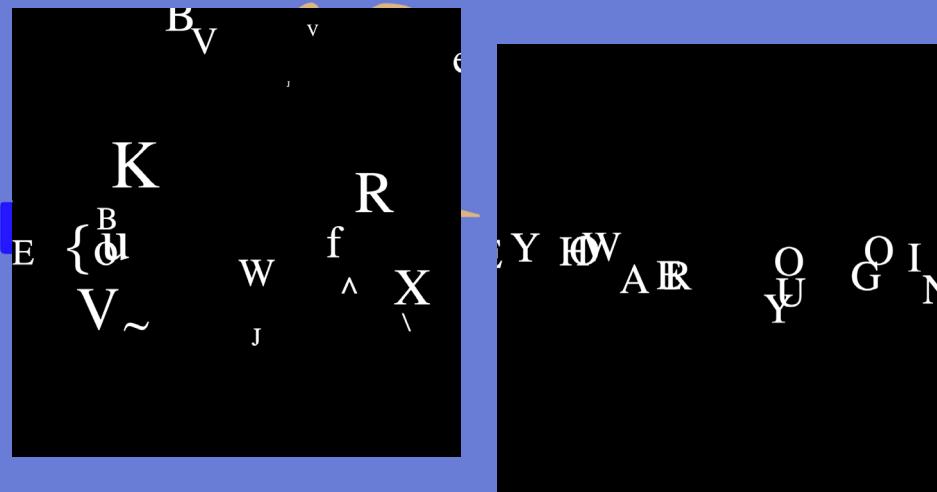
Jackson Pollock

This style of randomised, abstract art is what inspires my typographic decisions within the random(tool) application. Within my practice, I look to unconventional methods, that ultimately create conceptual pieces with intricate underlying messages. These pieces require the eye to look for symbols, shapes, images to engage with and relate too. Depending on whoever is interpreting I believe this is similar with type/ text, hence the inspiration for this system.

These pieces specifically speak to this idea of impulse, instinct and intuition. Specifically 'Night watch', where it has a generative feel in its layered, drawn over execution - This speaks to my practice as the legibility may be skewed, but can still be interpreted as 'art'.

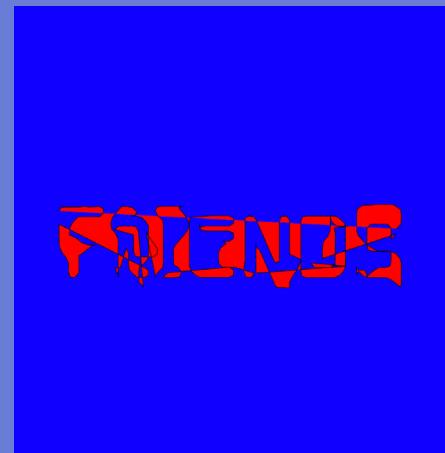
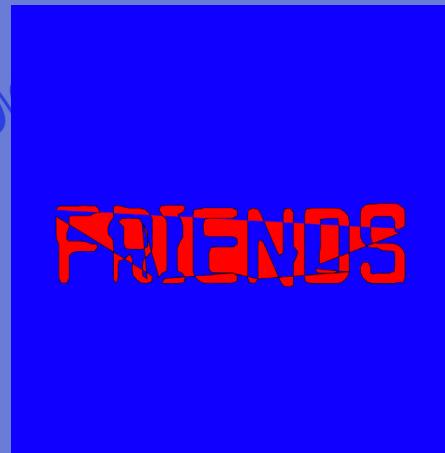
Wiggly text, combined with transparency settings demonstrated other creative possibilities that I was not able to incorporate into my end product. However, these processes brought to light those underlying possibilities, translating those into my current text I found to be a challenge. Utilising Developer tools, I was able to troubleshoot and create, but I found it most difficult.

FRIENDS



DEVELOPMENT

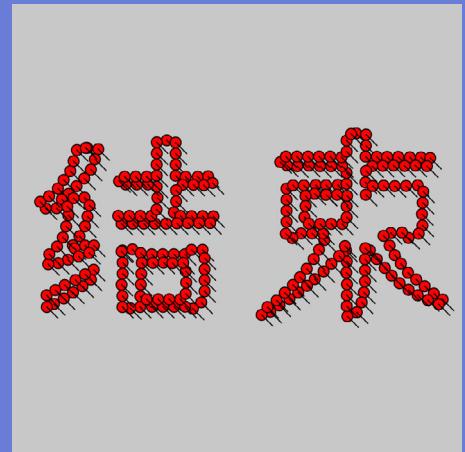
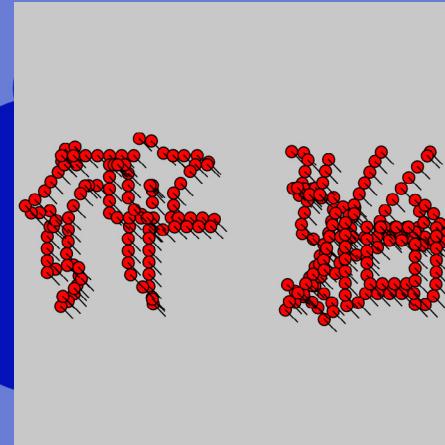
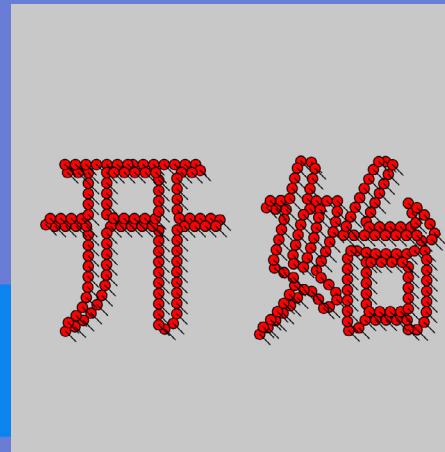
Friends/ Fiends TexttoPoint transformation allowed me to experiment with changing type, and 'morphing' as a technique that could combine elements, to then have them animate - also another element I couldn't adapt into my present code.



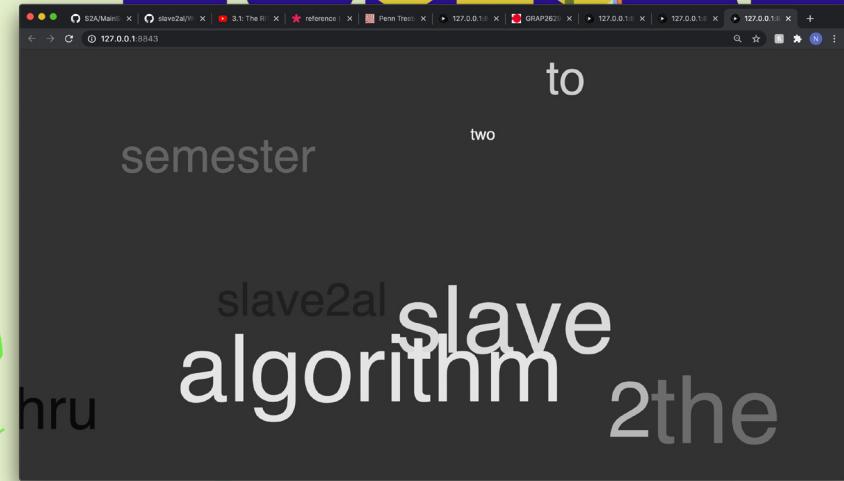
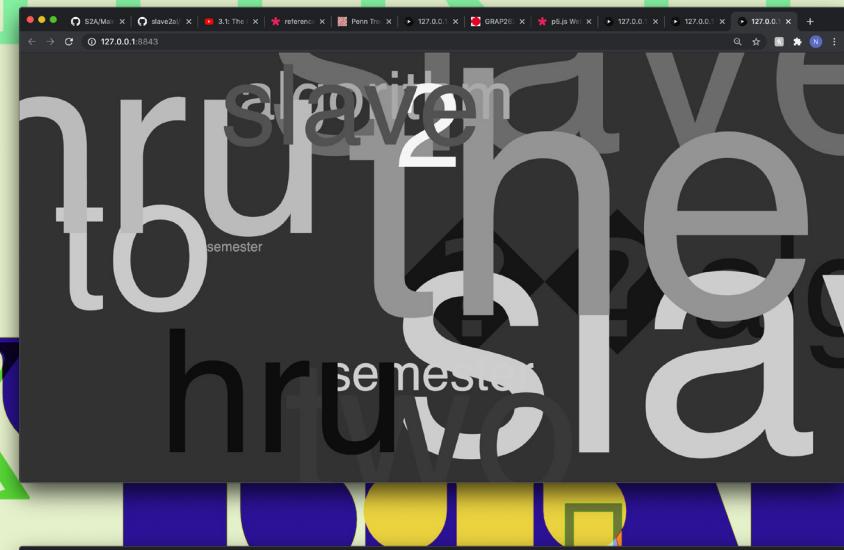
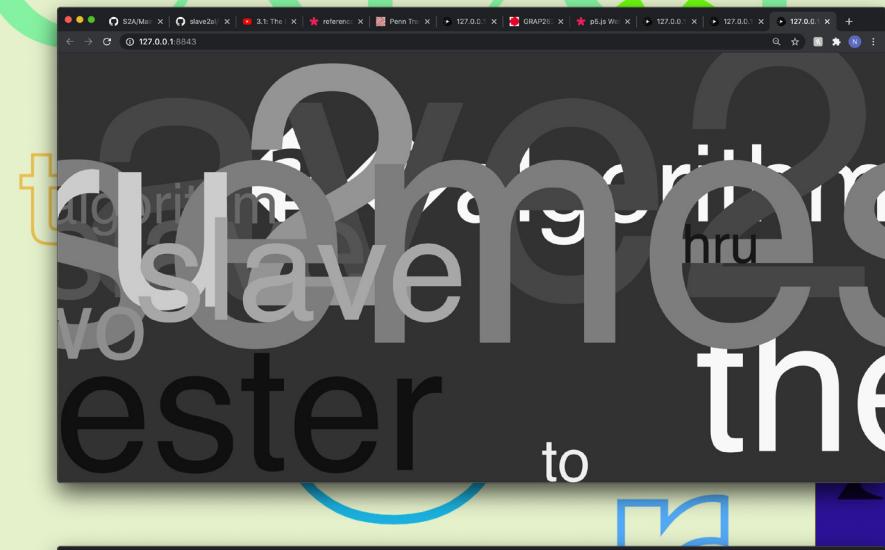
development

结束 (end) to 开始 (start) was another TexttoPoint experiment as I originally wished to experiment with Chinese typography. However, I was unable to figure the syntax to generate both english and chinese language.

I further looked into an API for Google Translate, yet this still wasn't able to connect through my program.

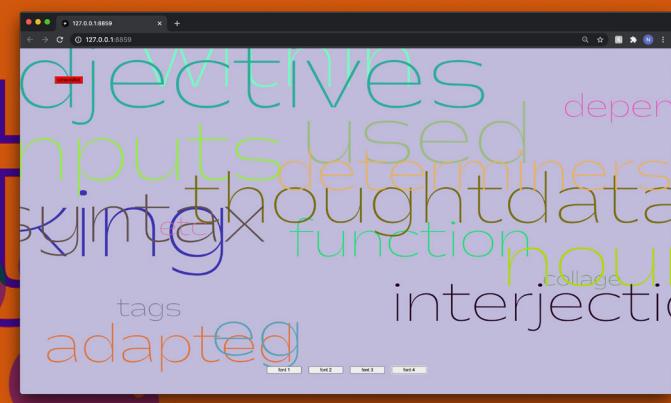
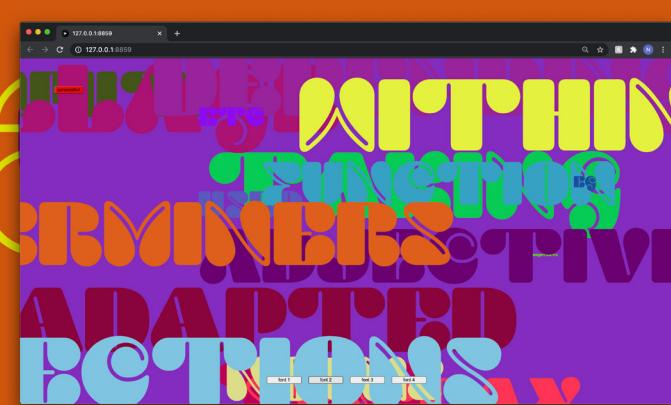
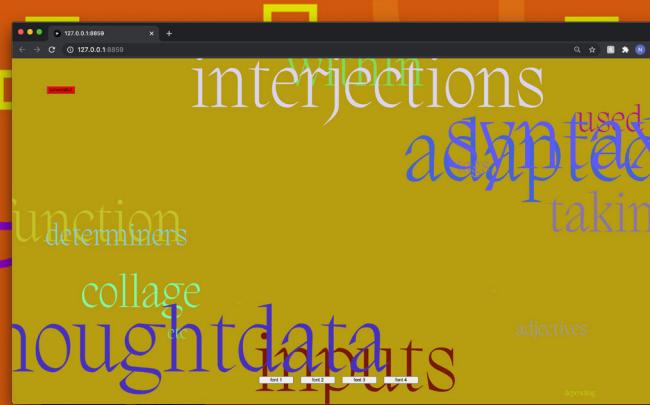
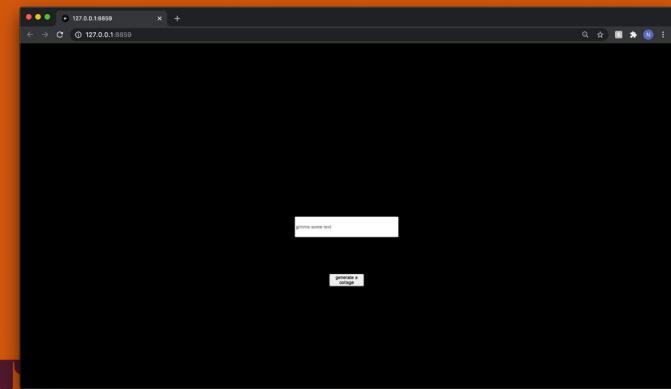
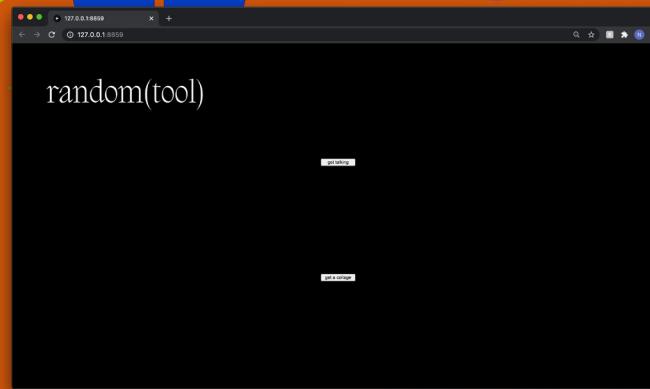


experiment



18

Having considered the eclectic compilation of projects I desired to create, my final coding system is riddled with possibilities of something far more interesting than its current capability. Beginning with its p5.SpeechRec function, within my 'collage chat' (not yet working), with each spoken word the graphic changes to create a pile of type. This pile can then be screenshot and used at the users digression.



METHODOLOGY

The inputs used are not limited to personal data, but can be scrapped from anywhere. The use is to create interesting, interactive, generative graphics utilising the human language. Further, the process translating the natural language into counted word objects, then abstracted and applied to your own designs.

eleven
com
eleven
pos
eleven

Within the 'collage make' function, I've utilised the RiTa library which recognises generative language which the makers have entitled 'computational literature'. This capability recognises Part-of-Speech functions, that I've eliminated to allow for complete random control. Through the input, in random(tool)'s 2nd window (L90, function type), these inputs each are processed within the RiTa.concordance (L157, counts = RiTa.concordance(boxdata, params)), creating a final 'word count object' (an alphabetical sequence). In these word count objects you retrieve a key/ value that is your product to be manipulated (L109, function getText, box data = box.value). Within the syntax however, 'k' represents the number word counts acquired from the initial input. However, my use of the RiTa library within this project is limit and hasn't equipped the full power of it's libraries and or, lexicon. To embrace the ironically unsystematic generated type, I opted for 'params' to be true - those being, ignoreStopWords, ignoreCase, and ignore Punctuation.

Further for capability and user interested, creating layers and interest within landing pages, and interactivity is important, such as p5.js's 'button' or 'slider', functions. Although I didn't manage functioning sliders, these elements would create dynamic differences in user experience and interaction; it would also contribute to largely diverse outcomes.

```
1 // speech variables
2 var speechRec = new p5.SpeechRec();
3 speechRec.continuous = true;
4 speechRec.interimResults = true;
5 speechRec.onResult = showResult;
6
7 let angle = 0;
8 let x = 400;
9 let y = 400;
10 var textInput;
11 var part;
12 var F1, F2, F3, F4;
13 var textEntry, RTA;
14 var textDisplay = [];
15 var box, button1, button2, button3;
16 var resultValue, resultWord, result;
17 var slider;
18 var boxdata = [];
19 var selectedFont;
20 var imgCount = 0;
21 //colours
22 var color1;
23 var color2;
24 function preload() {
25   //F1=loadFont('data/CormorantGaramond-Bold.ttf');
26   //F1=loadFont('data/Campfire.ttf');
27   //F1=loadFont('data/âš¢s .ttf');
28   F1=loadFont('data/Mirra-Extralight.otf');
29   //F2=loadFont('data/Fahkwang-Extralight.ttf');
30   F2=loadFont('data/Campfire.ttf');
31   //F3=loadFont('data/Osgard-Pro.otf');
32   //F3=loadFont('data/Neucha-Light.otf');
33   //F3=loadFont('data/irisone-Light.otf');
34   F3=loadFont('data/HelveticaNeueTC-BdObu.ttf');
35   F4=loadFont('data/Gatwick-Ultralight.ttf');
36   //F2 = loadFont('data/Fuji-Light.otf');
37 }
38 function setup() {
39   mode = 0;
40   createCanvas(windowWidth, windowHeight);
41   background(0);
42   fill (255);
43   textSize(90);
44   textFont(F1);
45   text("random(100)", 100, 170);
46   text("Press ESC to return to homepage", color1);
47   button1=createButton('get talking');
48   button1.style('background-color', color2);
49 }
```

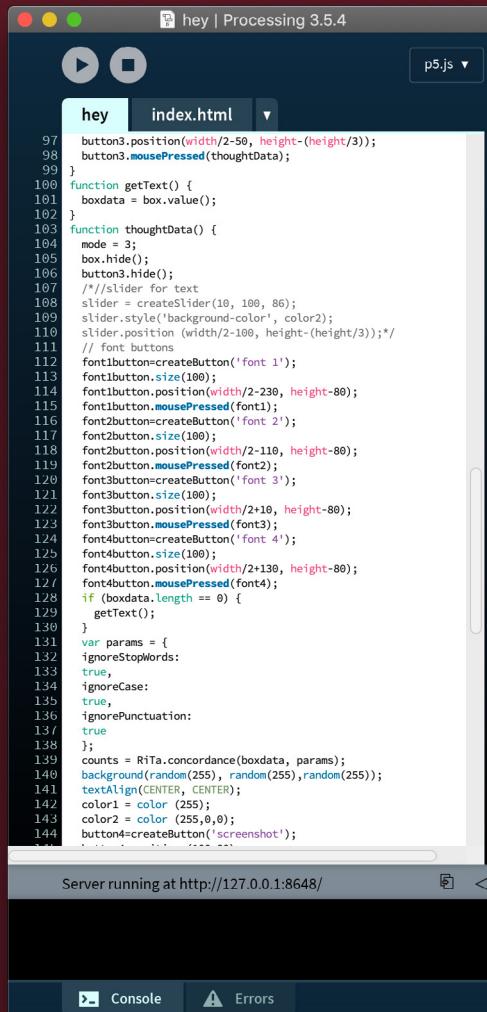
Server running at http://127.0.0.1:8648/

```
50 button1.size(100);
51 button1.position(width/2-50, height-(2*(height/3)));
52 button1.mousePressed(shout);
53 box=createButton('get a collage');
54 button2.style('background-color', color2);
55 button2.style('line-color', color2);
56 button2.size(100);
57 button2.position(width/2-50, height-(height/3));
58 button2.mousePressed(type);
59 selectedFont = F1;
60 }
61
62 function shout() {
63   mode = 1;
64   button1.hide();
65   button2.hide();
66   clear();
67   background(random(255));
68   textSize(72);
69   textFont(F2);
70   speechRec.start();
71 }
72 function showResult() {
73   if (speechRec.resultValue === true) {
74     clear();
75     background(0);
76     if (match(speechRec.resultString, "hello")) {
77       fill(255, 0, 0);
78     } else {
79       fill(255);
80     }
81     text(speechRec.resultString, 70, 140, width-100);
82   }
83 }
84 function type() {
85   mode = 2;
86   button1.hide();
87   button2.hide();
88   clear();
89   background(0);
90   box = createInput("");
91   box.attribute("placeholder", "gimme some text");
92   box.size(300, 60);
93   box.position(width/2-150, height/2);
94   button3=createButton('generate a collage');
95   button3.style('background-color', color2);
96   button3.size(100);
97 }
```

Server running at http://127.0.0.1:8648/

CODE

twelve

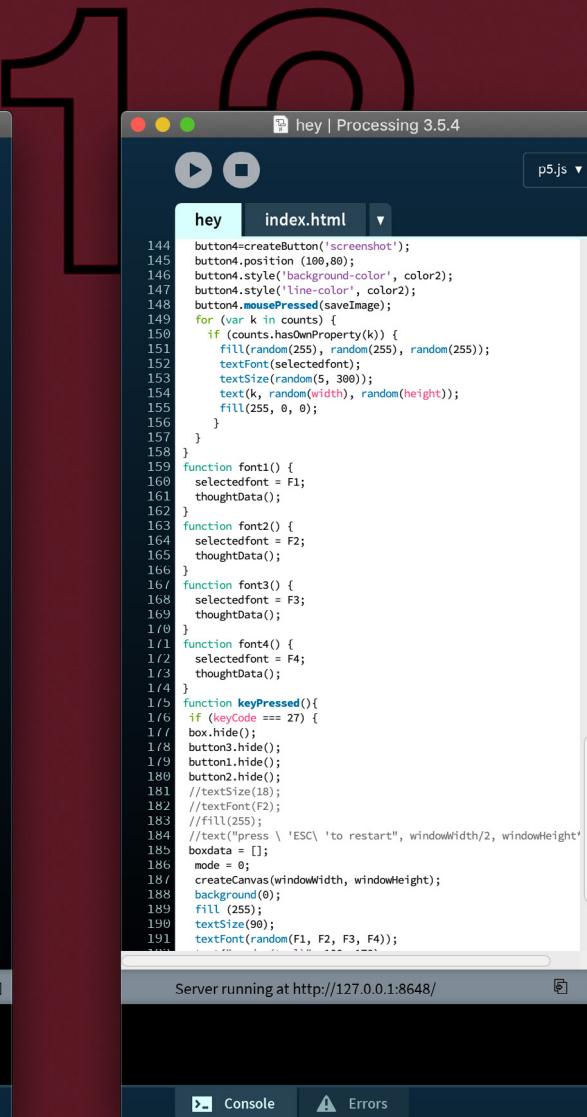


```
hey | index.html | p5.js ▾

97 button3.position(width/2-50, height-(height/3));
98 button3.mousePressed(thoughtData);
99 }
100 function getText() {
101   boxdata = box.value();
102 }
103 function thoughtData() {
104   mode = 3;
105   box.hide();
106   button3.hide();
107   /*//slider for text
108   slider = createSlider(10, 100, 86);
109   slider.style('background-color', color2);
110   slider.position (width/2-100, height-(height/3));*/
111   // font buttons
112   fontbutton=createButton('font 1');
113   fontbutton.size(100);
114   fontbutton.position(width/2-230, height-80);
115   fontbutton.mousePressed(font1);
116   font2button=createButton('font 2');
117   font2button.size(100);
118   font2button.position(width/2-110, height-80);
119   font2button.mousePressed(font2);
120   font3button=createButton('font 3');
121   font3button.size(100);
122   font3button.position(width/2+10, height-80);
123   font3button.mousePressed(font3);
124   font4button=createButton('font 4');
125   font4button.size(100);
126   font4button.position(width/2+130, height-80);
127   font4button.mousePressed(font4);
128   if (boxdata.length == 0) {
129     getText();
130   }
131   var params = {
132     ignoreStopWords:
133     true,
134     ignoreCase:
135     true,
136     ignorePunctuation:
137     true
138   };
139   counts = RiTa.concordance(boxdata, params);
140   background(random(255), random(255),random(255));
141   textAlign(CENTER, CENTER);
142   color1 = color (255);
143   color2 = color (255,0,0);
144   button=createElement('Screenshot');
```

Server running at http://127.0.0.1:8648/

Console Errors

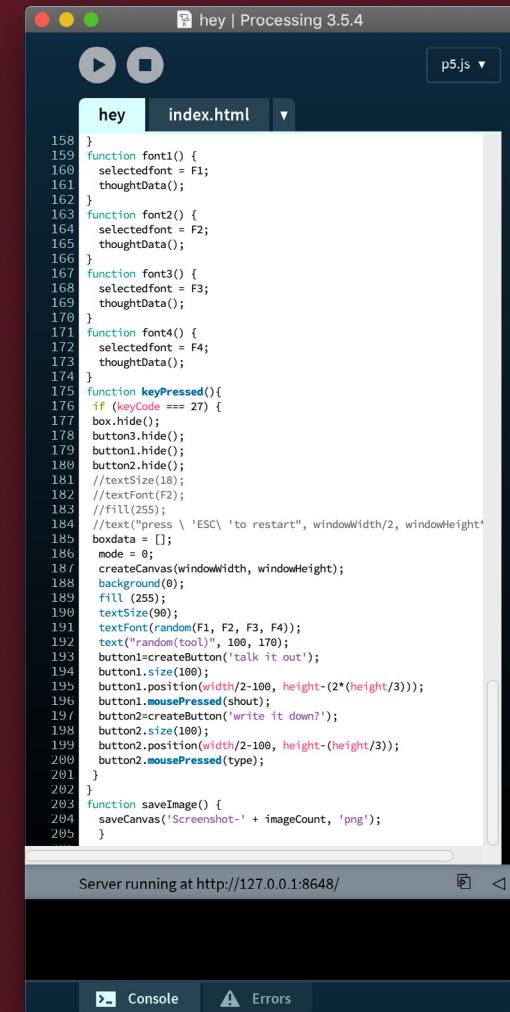


```
hey | index.html | p5.js ▾

144 button4=createButton('Screenshot');
145 button4.position (100,80);
146 button4.style('background-color', color2);
147 button4.style('line-color', color2);
148 button4.mousePressed(saveImage);
149 for (var k in counts) {
150   if (counts.hasOwnProperty(k)) {
151     fill(random(255), random(255), random(255));
152     textFont(selectedfont);
153     textSize(random(5, 300));
154     text(k, random(width), random(height));
155     fill(255, 0, 0);
156   }
157 }
158 function font1() {
159   selectedfont = F1;
160   thoughtData();
161 }
162 function font2() {
163   selectedfont = F2;
164   thoughtData();
165 }
166 function font3() {
167   selectedfont = F3;
168   thoughtData();
169 }
170 function font4() {
171   selectedfont = F4;
172   thoughtData();
173 }
174 function keyPressed(){
175   if (keyCode === 27) {
176     box.hide();
177     button3.hide();
178     button1.hide();
179     button1.hide();
180     button2.hide();
181     //textSize(18);
182     //textFont(F2);
183     //fill(255);
184     //text("press \\'ESC\\' to restart", windowWidth/2, windowHeight);
185     boxdata = [];
186     mode = 0;
187     createCanvas(windowWidth, windowHeight);
188     background(0);
189     fill (255);
190     textSize(90);
191     textFont(random(F1, F2, F3, F4));
192     text("random(too)", 100, 170);
193     button1=createButton('talk it out');
194     button1.size(100);
195     button1.position(width/2-100, height-(2*(height/3)));
196     button1.mousePressed(shout);
197     button2=createButton('write it down?');
198     button2.size(100);
199     button2.position(width/2-100, height-(height/3));
200     button2.mousePressed(type);
201   }
202 }
203 function saveImage() {
204   saveCanvas('Screenshot-' + imageCount, 'png');
205 }
```

Server running at http://127.0.0.1:8648/

Console Errors



```
hey | index.html | p5.js ▾

158 }
159 function font1() {
160   selectedfont = F1;
161   thoughtData();
162 }
163 function font2() {
164   selectedfont = F2;
165   thoughtData();
166 }
167 function font3() {
168   selectedfont = F3;
169   thoughtData();
170 }
171 function font4() {
172   selectedfont = F4;
173   thoughtData();
174 }
175 function keyPressed(){
176   if (keyCode === 27) {
177     box.hide();
178     button3.hide();
179     button1.hide();
180     button2.hide();
181     //textSize(18);
182     //textFont(F2);
183     //fill(255);
184     //text("press \\'ESC\\' to restart", windowWidth/2, windowHeight);
185     boxdata = [];
186     mode = 0;
187     createCanvas(windowWidth, windowHeight);
188     background(0);
189     fill (255);
190     textSize(90);
191     textFont(random(F1, F2, F3, F4));
192     text("random(too)", 100, 170);
193     button1=createButton('talk it out');
194     button1.size(100);
195     button1.position(width/2-100, height-(2*(height/3)));
196     button1.mousePressed(shout);
197     button2=createButton('write it down?');
198     button2.size(100);
199     button2.position(width/2-100, height-(height/3));
200     button2.mousePressed(type);
201   }
202 }
203 function saveImage() {
204   saveCanvas('Screenshot-' + imageCount, 'png');
205 }
```

Server running at http://127.0.0.1:8648/

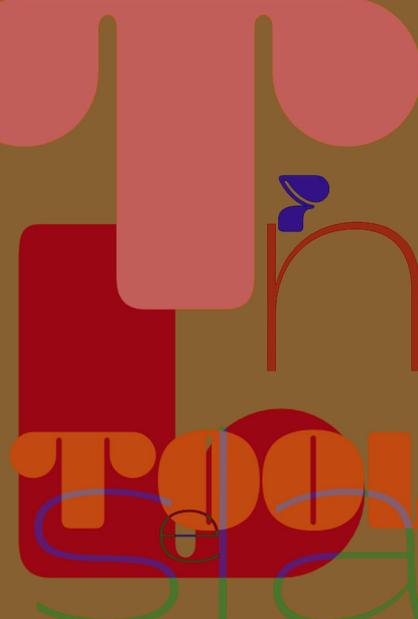
Console Errors

ve

RANDOM

final

Having developed this piece further, I would have liked a deeper sense of concept throughout, however due to situation I changed the projects overall outcome several times (eg. Hypnotic sleep device, Hypnotic study system, PVP (Practical Vs. Play), Expressive Translator, HRU (How are you? A diary system for expression/ exhaust)), til now a random(tool) crafted, speculated for designers that struggle with 'breaking the rules' as it were. Working through YouTube tutorials, Daniel Shiffman was an inspiration as he made the world of 'code' entertaining and joyous.





s g t

u links

s

n
1
K

links

github

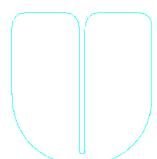
LINKS TO SYSTEM.

<https://nicholaswinter.github.io/slave2al/FinalSubmission/hey/index.html>
<https://github.com/nicholaswinter/slave2al/tree/master/FinalSubmission/hey>

g b

t

1



S
V

e

THANKS TO

David Shiffman
The Coding Train
RiTa Library
OpenProcessing

TYPOGRAPHY USED

Migra
Helvetica Neue LT Com
Campfire
Gatwick

TYPOGRAPHERS

Valerio Monopoli
Max Miedinger & Eduard Hoffmann
UNKNOWN
Thomas Thiemich

1 slave2algorithm

SLAVE2ALGORITHM

end



end