

Quick Explainer



Programming the InfoLab Lights



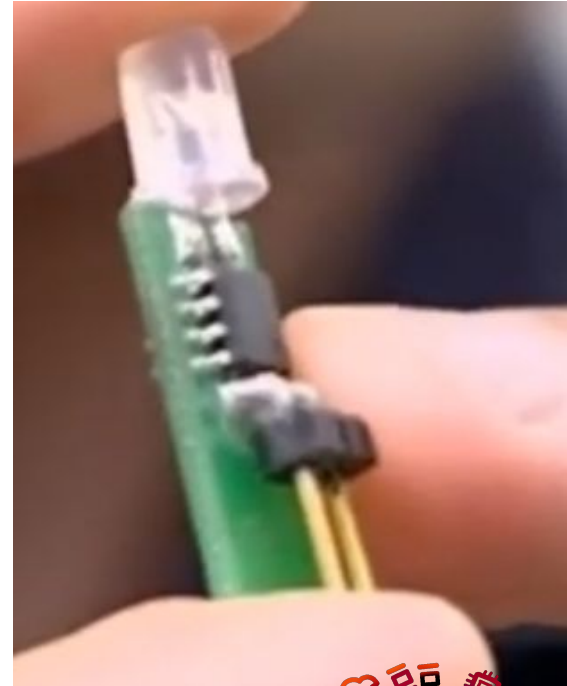
Background Information

- Dates back to **June 2020**
- Developed by **Professor Joe Finney**
- one of many light installations made by Finney and/or **Professor Alan Dix** (who left Lancaster Uni in 2018)
- Contains over 3000 LEDs

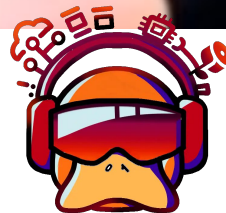


How do the LEDs work?

- Each LED contains a micro-controller
- Each LED is individually controllable
- The lights can be hung in any order
- Can be easily turned into 3D display



*From the 2007
video we watched
earlier*



Programming the Lights

We will be using the playground...
compsoc.io/w4 -> click 'Playground'
in the resources

We will return a class that contains
the our entire effect. This has a
single attribute, *display*.

*N.b. the # is used in JavaScript before a
method to mark it as private – you should
do this except for the constructor and
update()*

```
25 return class MyEffect {  
26   constructor(display) {  
27     this.display = display;  
28  
29     this.#clear();  
30   }  
31  
32   #clear() {  
33     for (let x = 0; x < this.display.width; x++) {  
34       for (let y = 0; y < this.display.height; y++) {  
35         this.display.setPixel(x, y, [255, 255, 255]);  
36       }  
37     }  
38  
39     this.display.flush();  
40   }  
41  
42   update() {  
43   }  
44 }
```

Programming the Lights

The display class...

Two methods:

- flush()
- setPixel(x, y, RGB)

e.g. `setPixel(0, 0, [255, 255, 255])`

It also has two **attributes**

- width
- height

You can access these methods and attributes using dot notation

e.g. `this.display.height`
or `this.display.setPixel()`

How does the display work?

- update is called once per frame

```
42 update() {
43     // YOUR CODE GOES HERE
44 }
```

General Advice

Use constants and private functions

```
1 // Greyscale value of each pixel, going left  
2 const LOGO_DATA = [255,255,255,255,255,255,255,  
3 const LOGO_W = 40;  
4 const LOGO_H = 18;  
5 const LOGO_SPEED = 2;  
6 const LOOG_COLORS = [  
7     [0, 1, 1],  
8     [0, 0, 1],  
9     [1, 0, 0],  
10    [1, 1, 0],  
11    [1, 0.5, 0],  
12    [1, 0, 0.5],  
13    [0.75, 0, 1],  
14 ]
```



A notice on acceptable patterns

Rules

If you're submitting an effect, please make sure it follows these rules:

1. Effects must not display explicit/ offensive content.
2. Effects must not fetch external resources.
3. Effects must avoid rapid flashing or intense strobe patterns. This display is public-facing, and we want to minimise risk for people with photosensitive epilepsy or similar sensitivities.

For obvious reasons, PRs containing effects that violate these restrictions will be rejected.

From the github repo README



Your turn...

Go the page on the website!

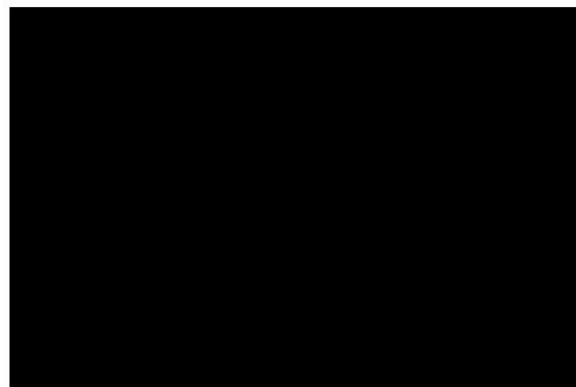
Click on **Playground**

You can write and test your code in here.

For tasks things you could try:

- Visualisations of algorithms / patterns
- Draw shapes
- Drawing something you like or a logo

<https://compsoc.io/w4>



Effects

This is the place to test out and develop effects for the display, when you have something you'd like to have added to the rotation, submit a Pull Request to [the repository](#). You can view some examples of effects in the repository [here](#).

Reload effect

Reset code



Please don't click reset code!