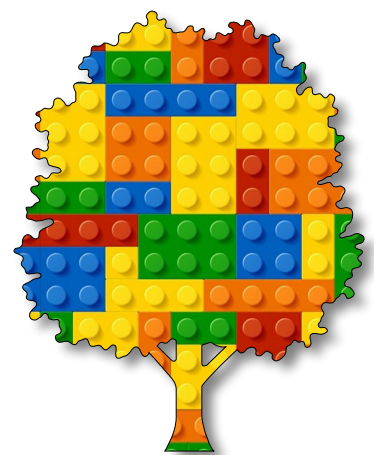


Algorithmic Art

Module F

Unit #6

microphone





Module F Unit #6 Visualisation

Sketch F6.1	the microphone
Sketch F6.2	the amplitude
Sketch F6.3	the volume
Sketch F6.4	a circle
Sketch F6.5	bouncing ball?



Introduction

This unit covers aspects such as visualising music and the microphone sound. Although we will be working with v2 of p5.js, we do need to add a library that will allow us some compatibility with p5.js version 1. This is easy to do; remember to do it, even especially if you start a new sketch.

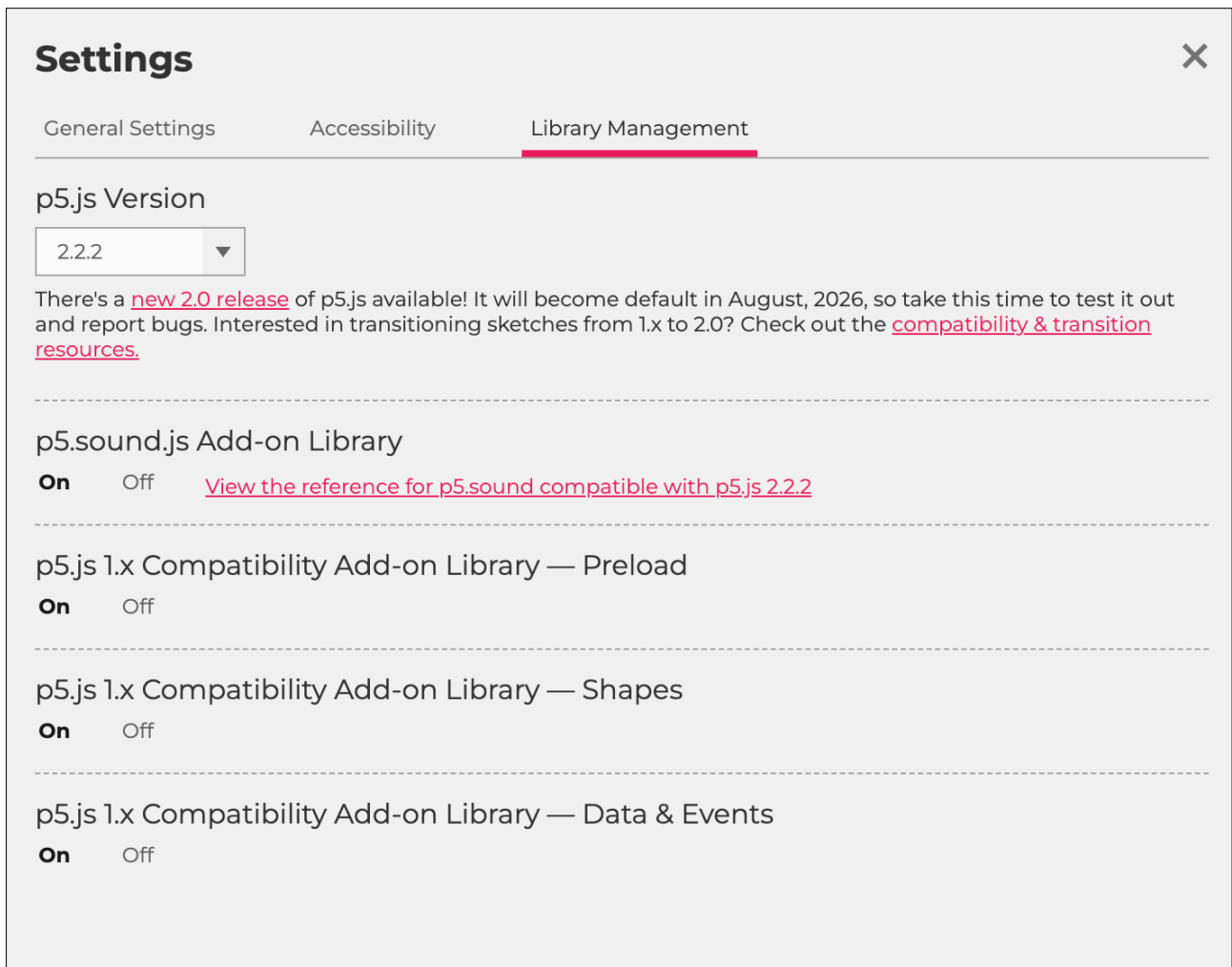
Click on the button that indicates the version you are using; yours might be 1.11.13 or something similar. See Fig. 1.

Figure 1: the version button (top left)



Now select the p5.sound.js Add-on Library in **Settings**, as shown in Fig 2.

Figure 2: settings





Sketch F6.1 the microphone

! Our starting sketch.

You may need to give your computer permission to use the microphone for `p5.js`.

```
let mic

function setup()
{
  createCanvas(400, 400)
  mic = new p5.AudioIn()
  mic.start()
}

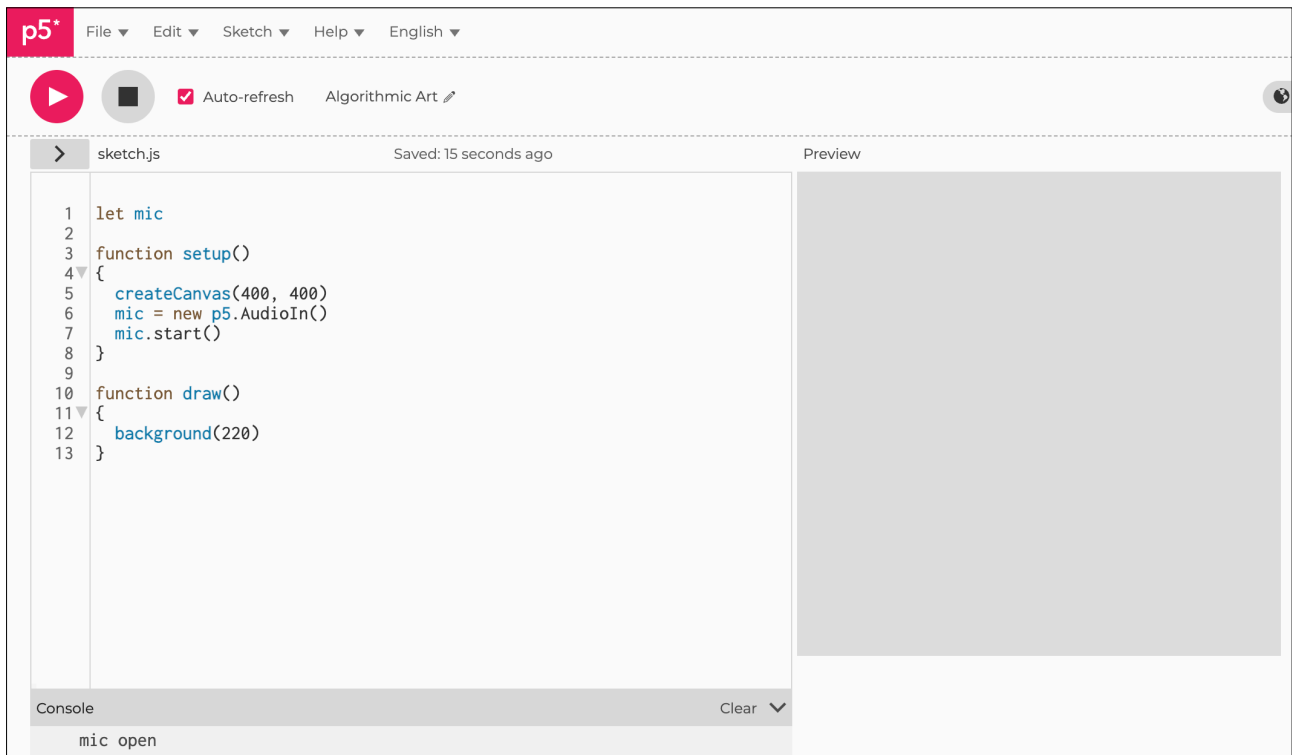
function draw()
{
  background(220)
}
```



Notes

We now have the microphone working, hopefully.

Figure F6.1





Sketch F6.2 the amplitude

We need to use the `p5.Amplitude()` function to get the volume, and connect it to the microphone.

```
let mic
let amp

function setup()
{
  createCanvas(400, 400)
  mic = new p5.AudioIn()
  amp = new p5.Amplitude()
  mic.connect(amp)
  mic.start()
}

function draw()
{
  background(220)
}
```



Notes

This connects the microphone to the built-in amplitude function.



Sketch F6.3 the volume

Using the amplitude value, `amp`, we can get the volume levels.

```
let mic
let amp
let vol

function setup()
{
  createCanvas(400, 400)
  mic = new p5.AudioIn()
  amp = new p5.Amplitude()
  mic.connect(amp)
  mic.start()
}

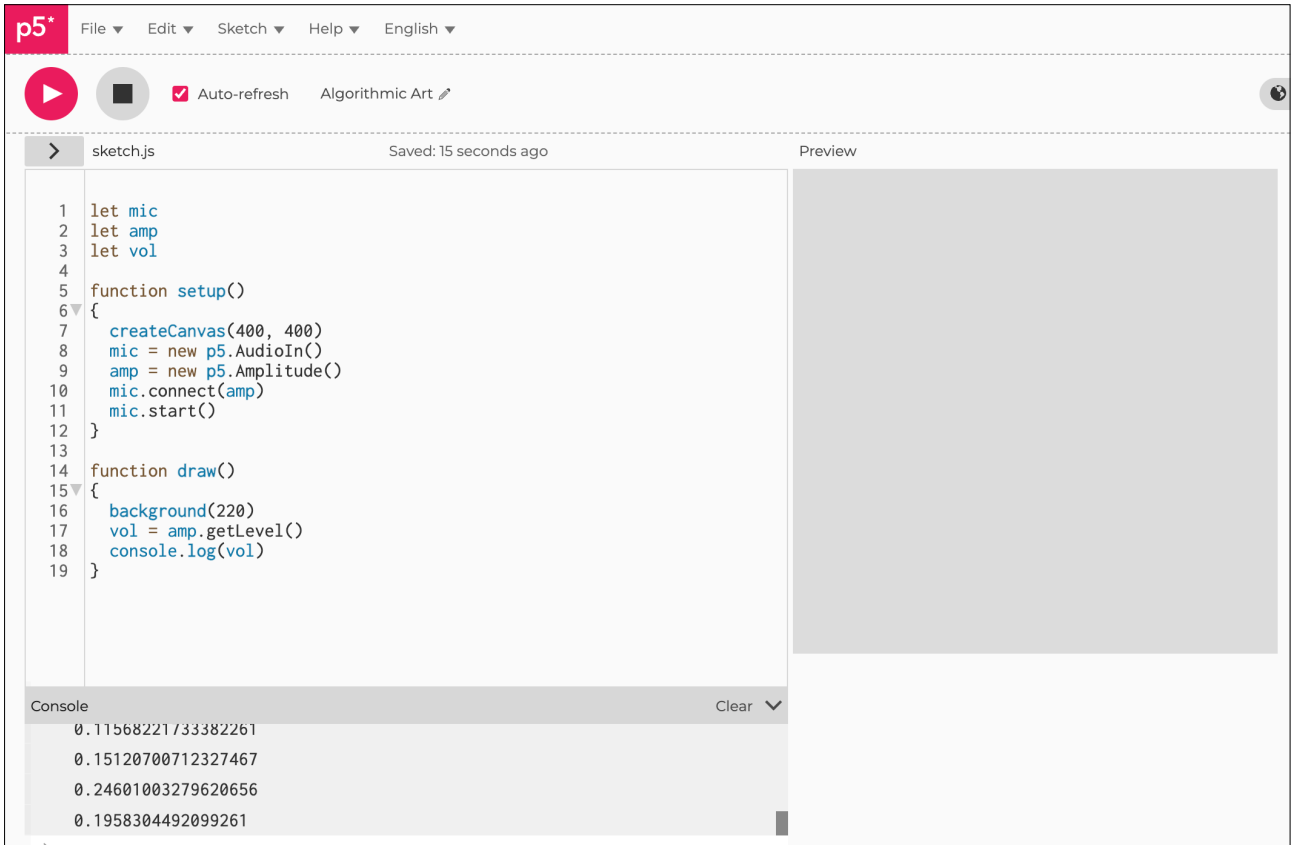
function draw()
{
  background(220)
  vol = amp.getLevel()
  console.log(vol)
}
```



Notes

The `console.log()` will give us the volume between `0` and `1`.

Figure F6.3





Sketch F6.4 a circle

! Remove the `console.log()`.

We can draw a circle that responds to the noise level the microphone picks up.

```
let mic
let amp
let vol

function setup()
{
  createCanvas(400, 400)
  mic = new p5.AudioIn()
  amp = new p5.Amplitude()
  mic.connect(amp)
  mic.start()
}

function draw()
{
  background(220)
  vol = amp.getLevel()
  vol = map(vol, 0, 1, 0, width)
  circle(200, 200, vol * 2)
}
```

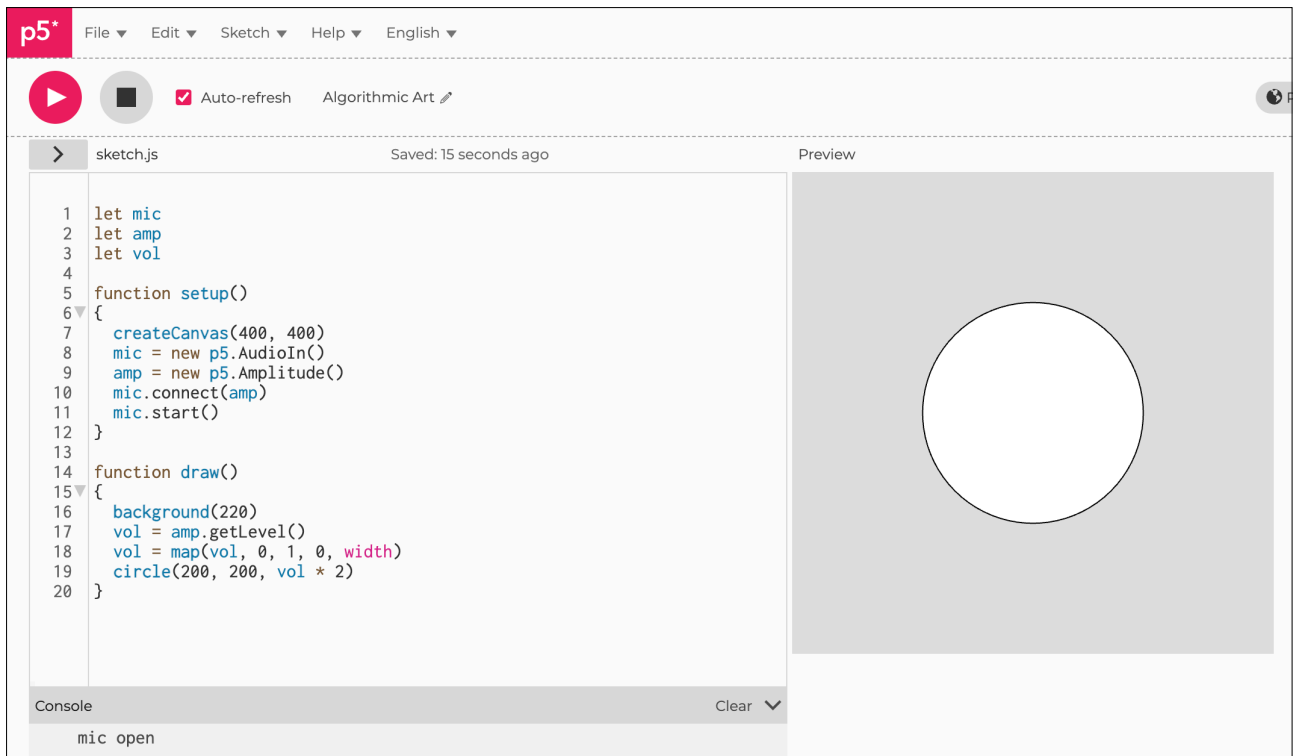
Notes

You can play around with the value of the diameter (`vol`) of the circle if there appears nothing or too much happening.

Challenges

1. Could you connect the `vol` to the colour of a circle, e.g. white to black, or other colours?
2. Constrain the circle between two limits. For instance: `vol = constrain(vol, 50, 200)`.

Figure F6.4





Sketch F6.5 bouncing ball?

! Remove mapping the volume.

We can move the ball with our voice, sort of.

```
let mic
let amp
let vol
let y

function setup()
{
  createCanvas(400, 400)
  mic = new p5.AudioIn()
  amp = new p5.Amplitude()
  mic.connect(amp)
  mic.start()
}

function draw()
{
  background('darkred')
  fill('yellow')
  noStroke()

  vol = amp.getLevel()
  y = height - (vol * height * 5)
  circle(200, y - 50, 100)
}
```



Notes

Every time you make a sound, the ball bounces, or at least moves.



Challenge

Can you create other designs that respond to your voice?

Figure F6.5

