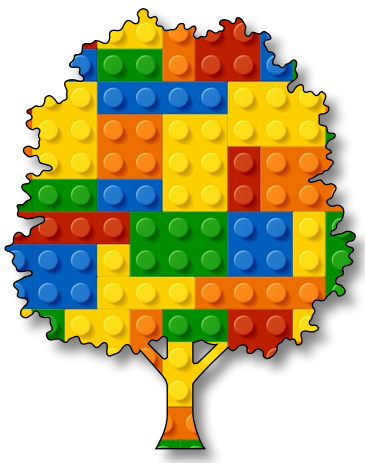
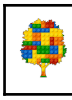
The Joy of Coding
Module D
Unit #8
3D arrays part 2





Content

Module D Unit #8 3D arrays part 2

Introduction to 3D arrays part 2

We are going to develop the concept of a 3D array and create some interesting shapes and patterns. First off is a cube of cubes and then we will have a spiral of bubbles!

Sketch D8.1 an array of cubes

Starting a new sketch with many of the previous elements. Making an array of cubes where we have 0 to 100 in steps of 20 for a cube size of 20 giving us 5 cubes. We do this with a simple for() loop.

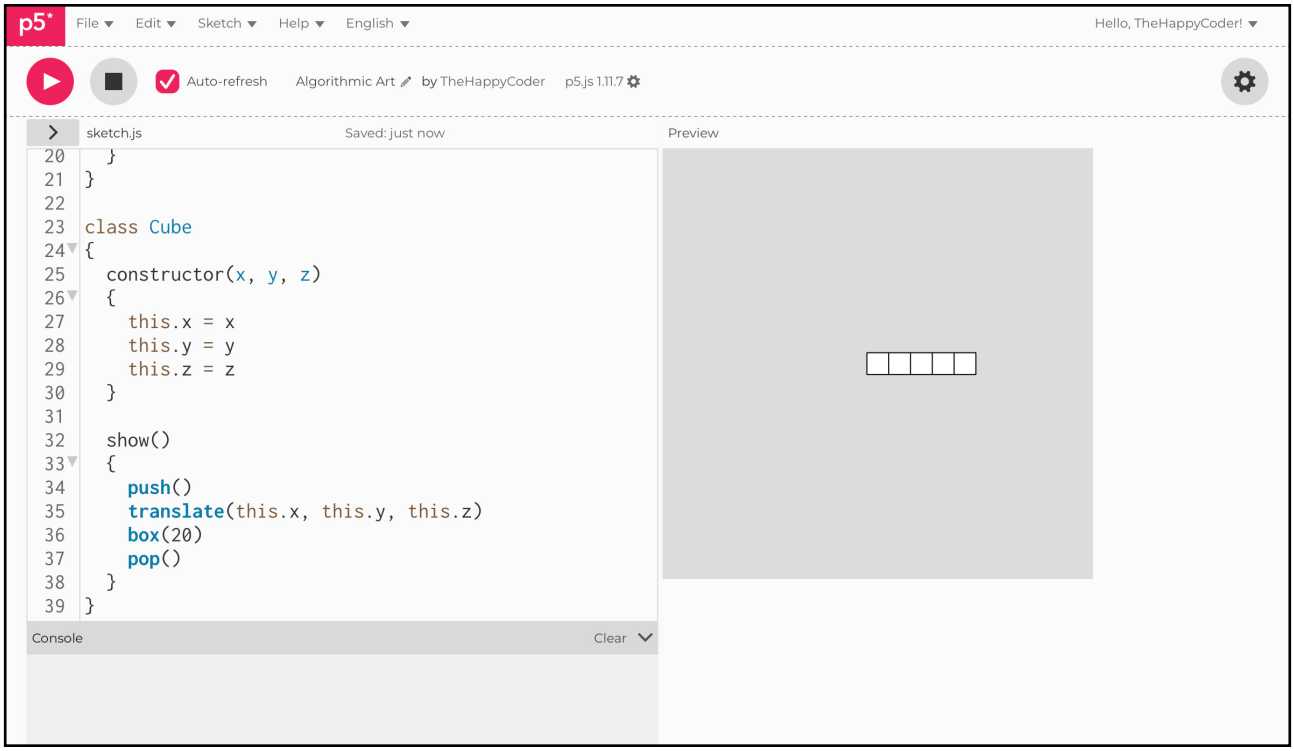
```
let cubes = []
let cube

function setup()
{
  createCanvas(400, 400, WEBGL)
  for (let i = 0; i < 100; i += 20)
  {
    cube = new Cube(i, 0, 0)
    cubes.push(cube)
  }
}

function draw()
{
  background(220)
  for (let i = 0; i < cubes.length; i++)
  {
    cubes[i].show()
  }
}

class Cube
{
  constructor(x, y, z)
  {
    this.x = x
```

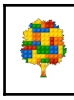
```
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    box(20)
    pop()
  }
}
```

🗒️ Notes

As you can see we have a problem. We have drawn the cubes from (0, 0) which is the centre of the canvas, we need to start at -100 rather than 0. Also we can only see from the front, it would be nice to rotate the array of cubes to see them properly.

Figure D8.1


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
20 }
21 }
22
23 class Cube
24 {
25 constructor(x, y, z)
26 {
27 this.x = x
28 this.y = y
29 this.z = z
30 }
31
32 show()
33 {
34 push()
35 translate(this.x, this.y, this.z)
36 box(20)
37 pop()
38 }
39 }
Console
Clear



Sketch D8.2 rotating the array

Now we rotate the array about the centre of the canvas. Also we have now got 10 cubes

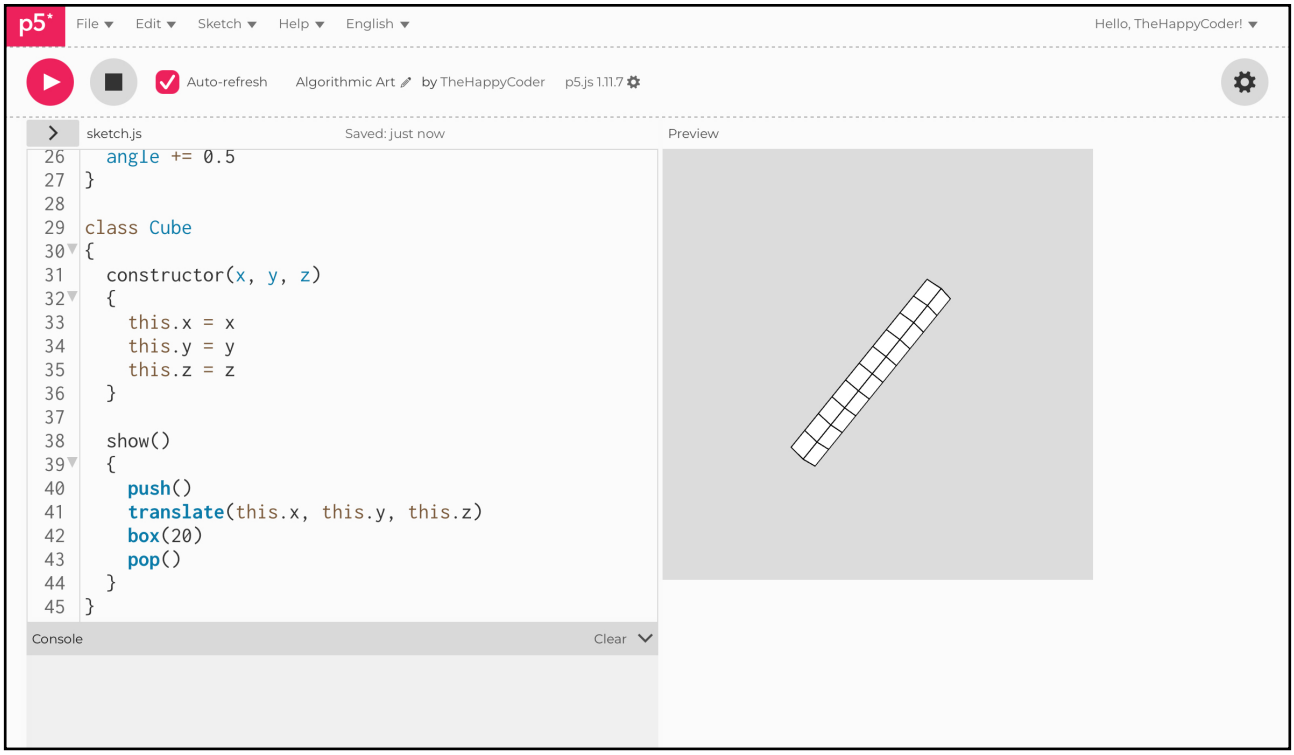
```
let cubes = []
let cube
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -100; i < 100; i += 20)
  {
    cube = new Cube(i, 0, 0)
    cubes.push(cube)
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < cubes.length; i++)
  {
    cubes[i].show()
  }
  angle += 0.5
}
```

```
class Cube
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    box(20)
    pop()
  }
}
```

🗒️ Notes

A long stick of cubes or a long cuboid

Figure D8.2


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
26 angle += 0.5
27 }
28
29 class Cube
30 {
31 constructor(x, y, z)
32 {
33 this.x = x
34 this.y = y
35 this.z = z
36 }
37
38 show()
39 {
40 push()
41 translate(this.x, this.y, this.z)
42 box(20)
43 pop()
44 }
45 }
Console
Clear

Sketch D8.3 spaced out

If we space them out a little bit more by increasing the steps from 20 to 25, this will also mean that we have only 8 cubes.

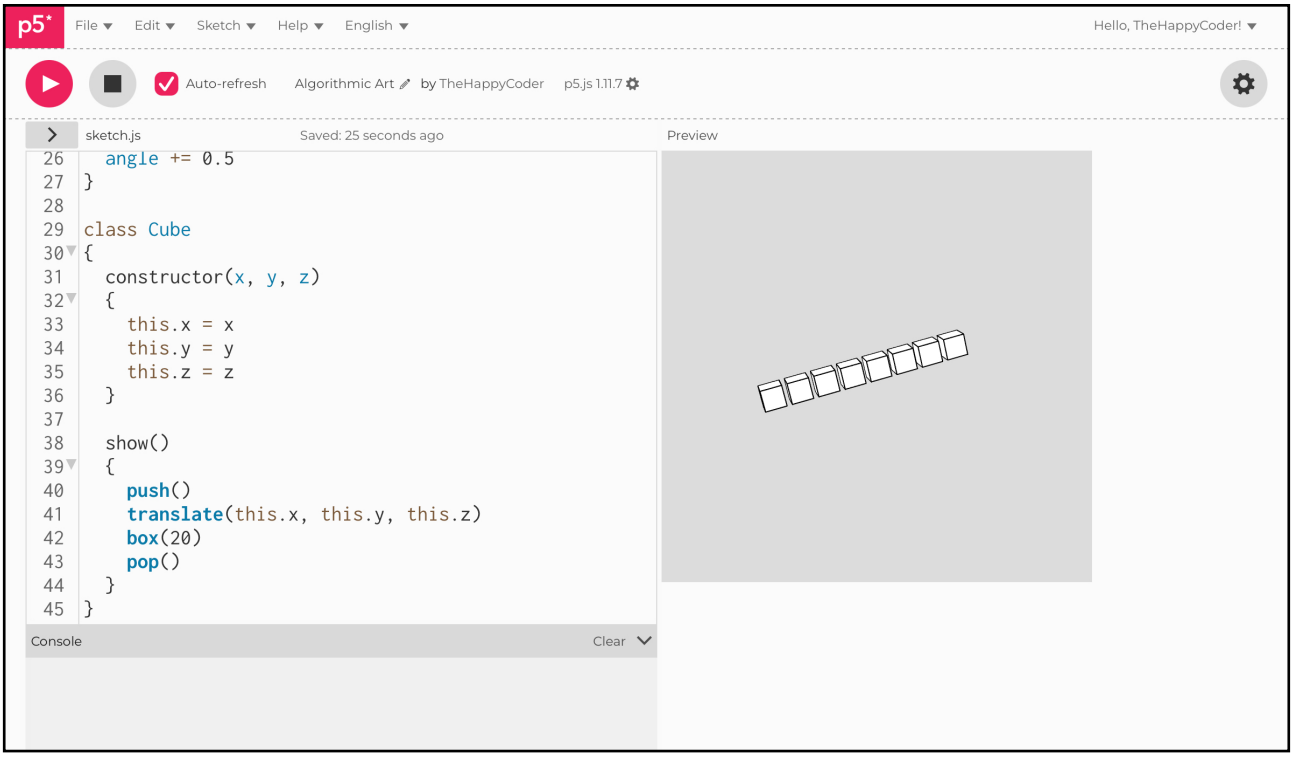
```
let cubes = []
let cube
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -100; i < 100; i += 25)
  {
    cube = new Cube(i, 0, 0)
    cubes.push(cube)
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < cubes.length; i++)
  {
    cubes[i].show()
  }
  angle += 0.5
}
```

```
class Cube
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    box(20)
    pop()
  }
}
```

🗒️ Notes

A bit more spaced out

Figure D8.3


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: 25 seconds ago
Preview
angle += 0.5
}
class Cube
{
constructor(x, y, z)
{
this.x = x
this.y = y
this.z = z
}
show()
{
push()
translate(this.x, this.y, this.z)
box(20)
pop()
}
}
Console
Clear

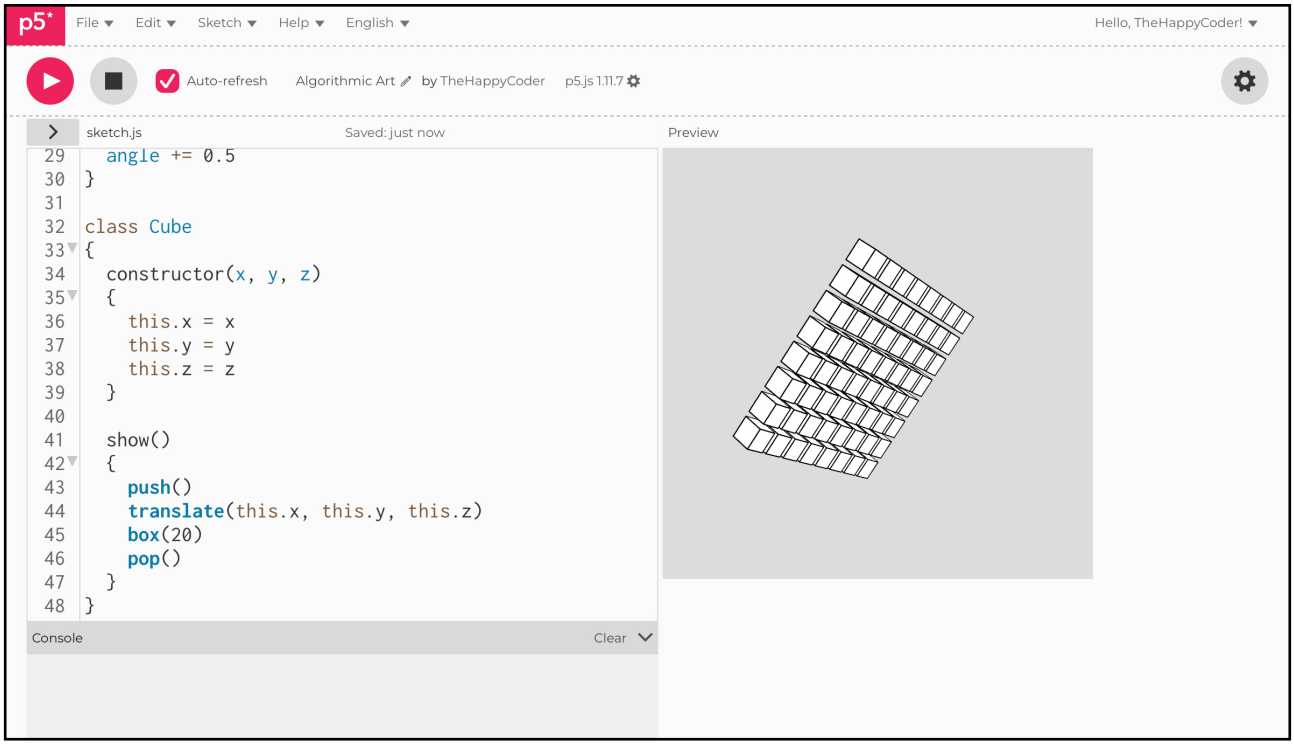
Sketch D8.4 an array in the y direction

Yet we can do even better because we can create an array in the y direction also. For that we need a nested loop.

```
let cubes = []
let cube
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -100; i < 100; i += 25)
  {
    for (let j = -100; j < 100; j += 25)
    {
      cube = new Cube(i, j, 0)
      cubes.push(cube)
    }
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < cubes.length; i++)
  {
    cubes[i].show()
  }
```

```
  angle += 0.5
}

class Cube
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    box(20)
    pop()
  }
}
```

Figure D8.4


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
angle += 0.5
}
class Cube
{
constructor(x, y, z)
{
this.x = x
this.y = y
this.z = z
}
show()
{
push()
translate(this.x, this.y, this.z)
box(20)
pop()
}
}
Console
Clear

Sketch D8.5 a cube of cubes

But why stop there. Let's add the third dimension the z direction, we just add another nested loop.

```
let cubes = []
let cube
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -100; i < 100; i += 25)
  {
    for (let j = -100; j < 100; j += 25)
    {
      for (let k = -100; k < 100; k += 25)
      {
        cube = new Cube(i, j, k)
        cubes.push(cube)
      }
    }
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < cubes.length; i++)
```

```
  {
    cubes[i].show()
  }
  angle += 0.5
}

class Cube
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    box(20)
    pop()
  }
}
```

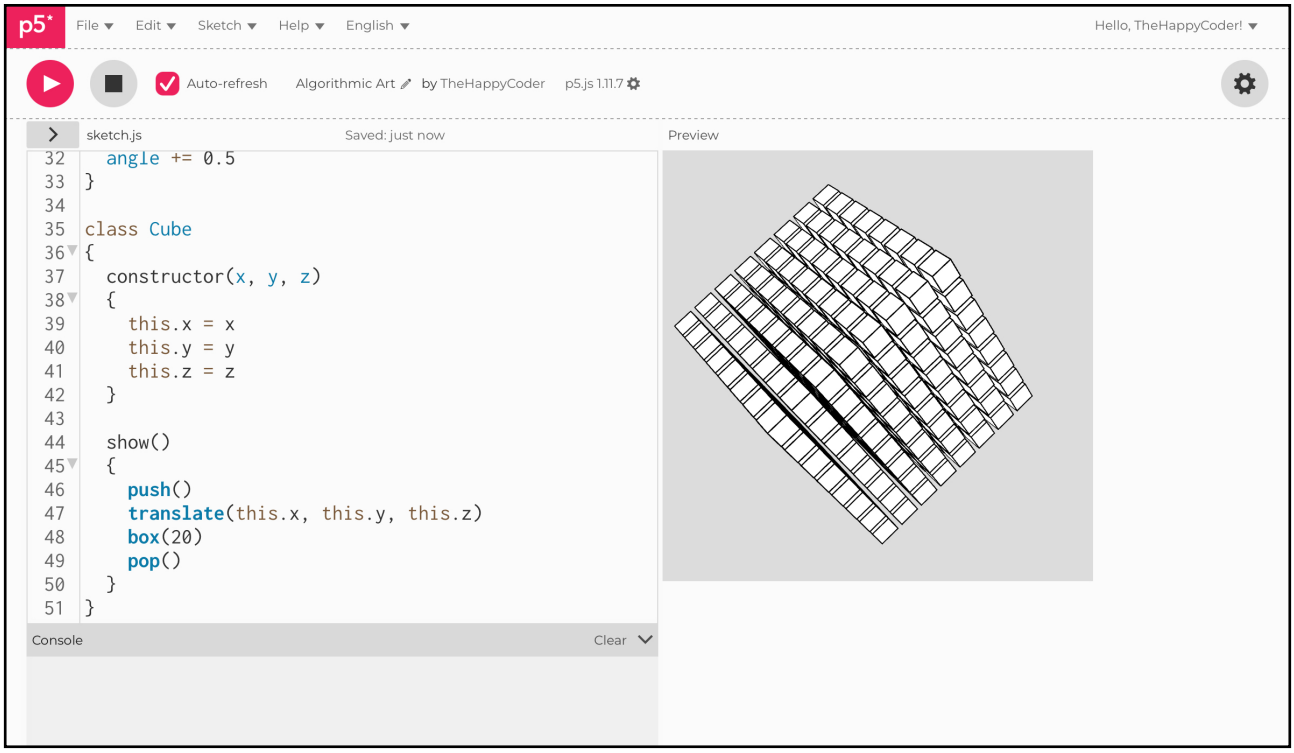
🗒️ Notes

Inside the cubes[] array we have a 3D array

🌻 Challenge

Have a look inside the `cubes[]` array with `console.log()`

Figure D8.5


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
angle += 0.5
}
class Cube
{
constructor(x, y, z)
{
this.x = x
this.y = y
this.z = z
}
show()
{
push()
translate(this.x, this.y, this.z)
box(20)
pop()
}
}
Console
Clear

Sketch D8.6 adding a splash of colour

We can add individual colours (random) to each cube. We use a function called `color(a, b, c)` with three arguments for the red, green and blue. We add a variable (`c`) to carry this information to the new cube being created.

```
let cubes = []
let cube
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -100; i < 100; i += 25)
  {
    for (let j = -100; j < 100; j += 25)
    {
      for (let k = -100; k < 100; k += 25)
      {
        let c = color(random(255), random(255), random(255))
        cube = new Cube(i, j, k, c)
        cubes.push(cube)
      }
    }
  }
}

function draw()
{
  background(220)
  rotateX(angle)
```

```
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < cubes.length; i++)
  {
    cubes[i].show()
  }
  angle += 0.5
}

class Cube
{
  constructor(x, y, z, c)
  {
    this.x = x
    this.y = y
    this.z = z
    this.c = c
  }

  show()
  {
    push()
    fill(this.c)
    translate(this.x, this.y, this.z)
    box(20)
    pop()
  }
}
```

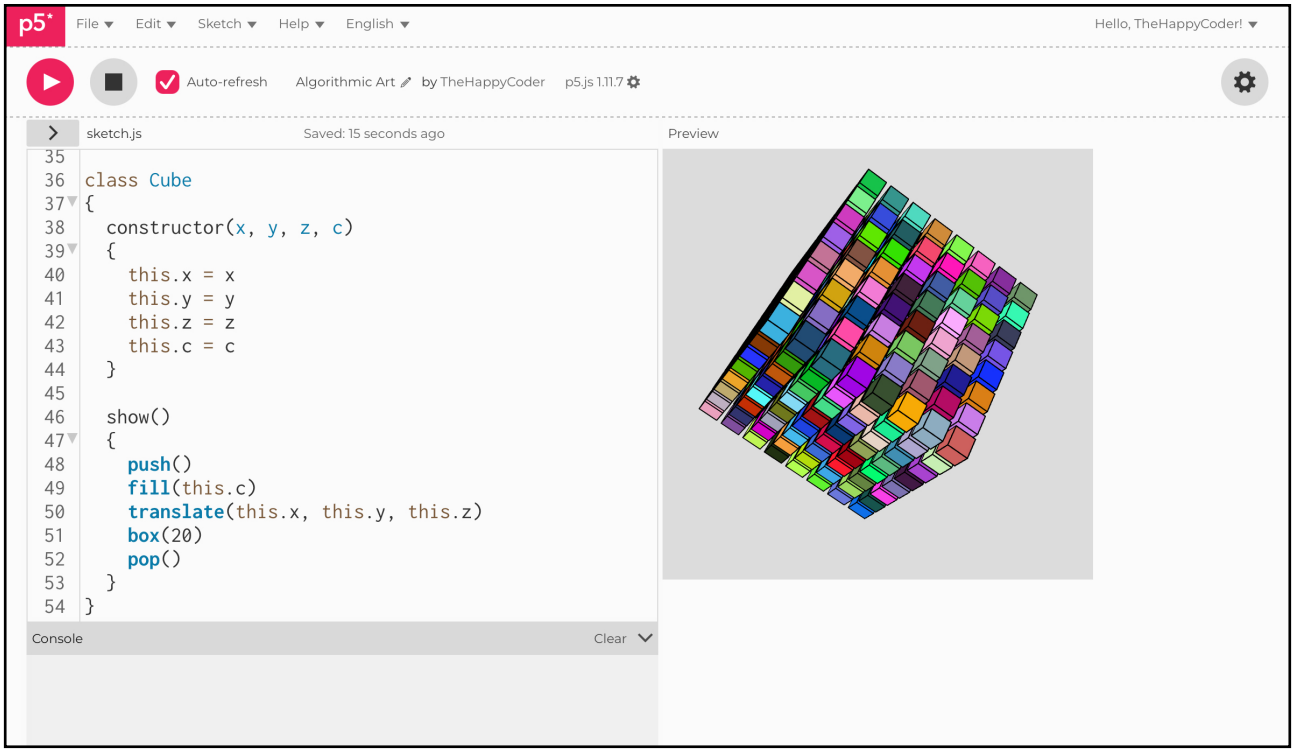
🗒️ Notes

We give each individual cube its own colour

🌻 Challenge

1. Add some alpha (fourth argument) to the colour to make it transparent and remove the stroke.
2. Change the shape to a sphere or cylinder

Figure D8.6


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: 15 seconds ago
Preview
35
36 class Cube
37 {
38 constructor(x, y, z, c)
39 {
40 this.x = x
41 this.y = y
42 this.z = z
43 this.c = c
44 }
45
46 show()
47 {
48 push()
49 fill(this.c)
50 translate(this.x, this.y, this.z)
51 box(20)
52 pop()
53 }
54 }
Console
Clear

Sketch D8.7 a bubble factory

A bit of refactoring. We are going to use a sphere so we will change the name of the class, variable and array to something more appropriate and familiar: `bubble`. Also added `sphere(10)`, `lights()` and `noStroke()`

```
let bubbles = []
let bubble
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -100; i < 100; i += 25)
  {
    for (let j = -100; j < 100; j += 25)
    {
      for (let k = -100; k < 100; k += 25)
      {
        let c = color(random(255), random(255), random(255))
        bubble = new Bubble(i, j, k, c)
        bubbles.push(bubble)
      }
    }
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
```

```
  rotateZ(angle)
  for (let i = 0; i < bubbles.length; i++)
  {
    bubbles[i].show()
  }
  angle += 0.5
}

class Bubble
{
  constructor(x, y, z, c)
  {
    this.x = x
    this.y = y
    this.z = z
    this.c = c
  }

  show()
  {
    push()
    fill(this.c)
    translate(this.x, this.y, this.z)
    lights()
    noStroke()
    sphere(10)
    pop()
  }
}
```

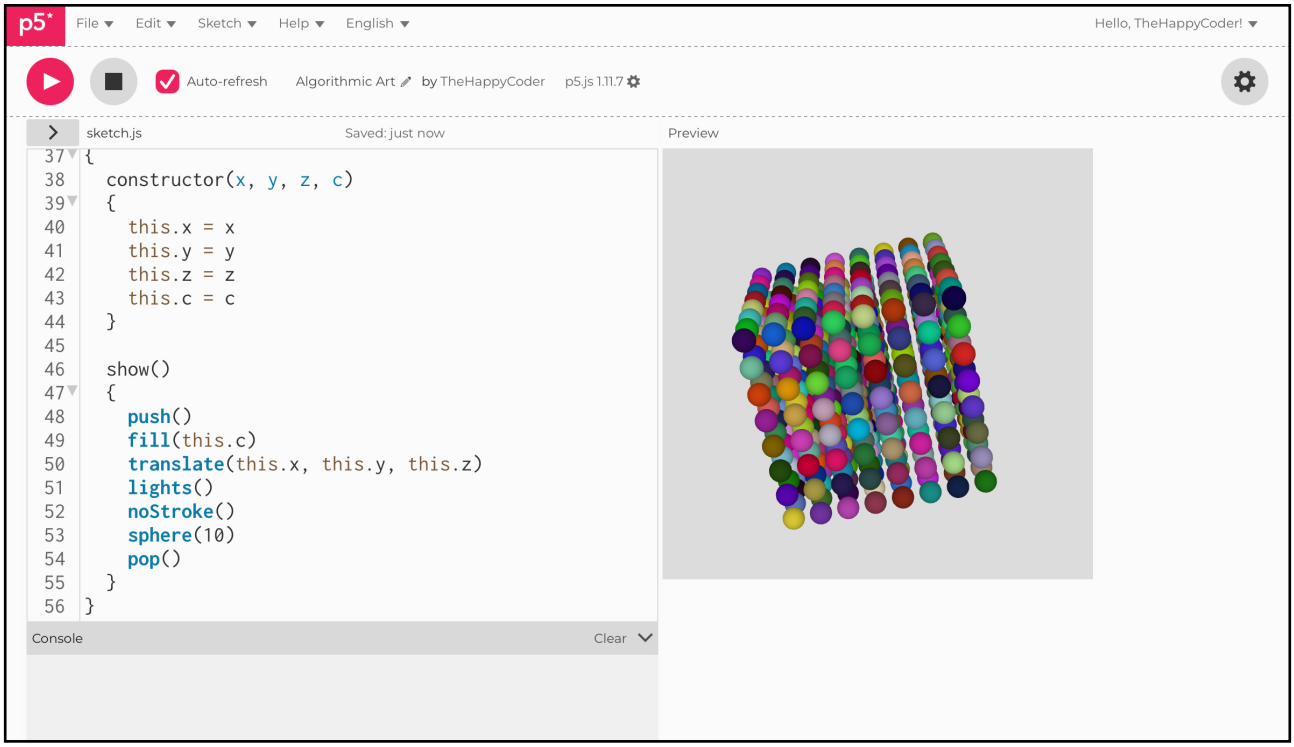
🗒️ Notes

The noStroke() is just so that you can see the colours

🌻 Challenge

Call the variable, class and array anything you want

Figure D8.7


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
{
constructor(x, y, z, c)
{
this.x = x
this.y = y
this.z = z
this.c = c
}
show()
{
push()
fill(this.c)
translate(this.x, this.y, this.z)
lights()
noStroke()
sphere(10)
pop()
}
}
Console
Clear

Sketch D8.8 not a cube

Our next step is putting all those bubbles in a circle rather than a cube, but first remove all the code highlighted and commented, then change the constructor() function

```
let bubbles = []
let bubble
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  // for (let i = -100; i < 100; i += 25)
  // {
  //   for (let j = -100; j < 100; j += 25)
  //   {
  //     for (let k = -100; k < 100; k += 25)
  //     {
  //       let c = color(random(255), random(255), random(255))
  //       bubble = new Bubble(i, j, k, c)
  //       bubbles.push(bubble)
  //     }
  //   }
  // }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
```

```
  rotateZ(angle)
  for (let i = 0; i < bubbles.length; i++)
  {
    bubbles[i].show()
  }
  angle += 0.5
}

class Bubble
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
    // this.c = c
  }

  show()
  {
    push()
   // fill(this.c)
    translate(this.x, this.y, this.z)
    lights()
    noStroke()
    sphere(10)
    pop()
  }
}
```

🗒️ Notes

Don't try running this just yet

Sketch D8.9 now a circle

Creating a circle of bubbles

```
let bubbles = []
let bubble
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = 0; i < 360; i += 30)
  {
    let x = 100 * sin(i)
    let y = 100 * cos(i)
    bubble = new Bubble(x, y, 0)
    bubbles.push(bubble)
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < bubbles.length; i++)
  {
    bubbles[i].show()
  }
  angle += 0.5
}
```

```
class Bubble
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    lights()
    fill('yellow')
    noStroke()
    sphere(10)
    pop()
  }
}
```

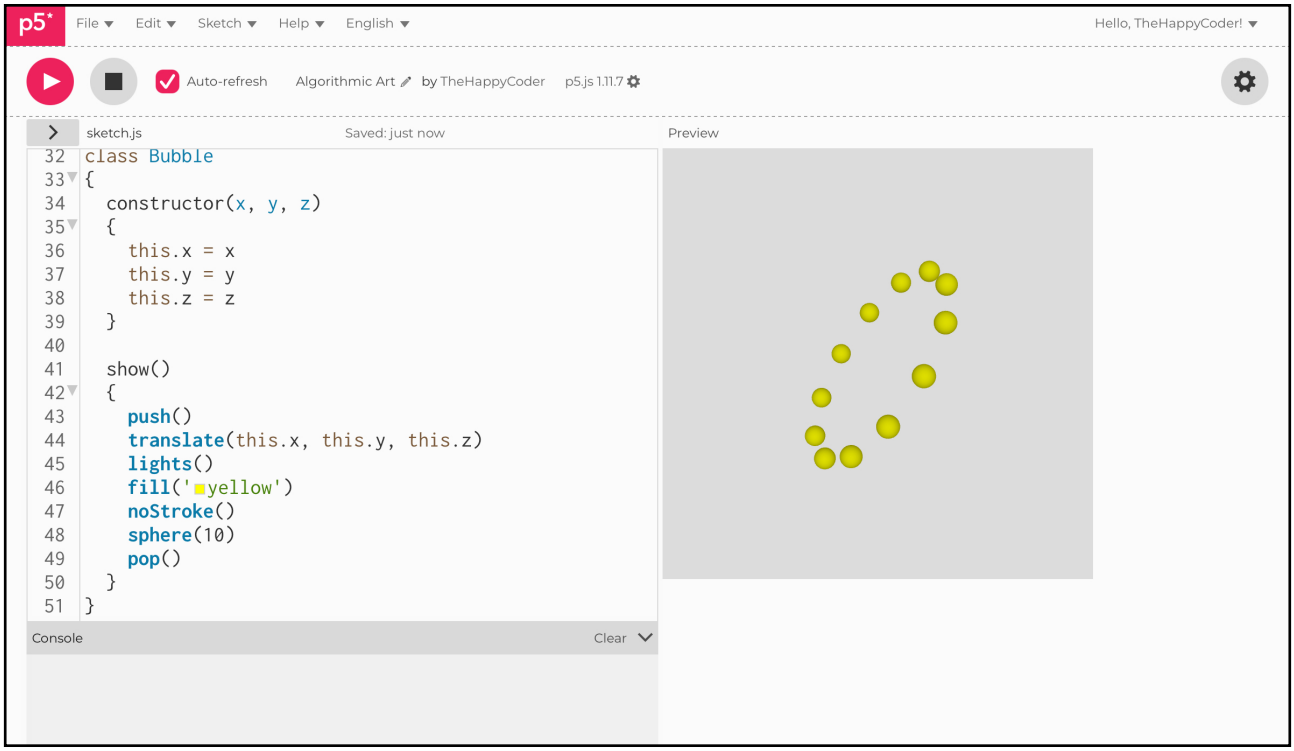
🗒️ Notes

We have a nice ring of spheres

🌻 Challenge

Add other lights or materials

Figure D8.9


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
class Bubble
{
constructor(x, y, z)
{
this.x = x
this.y = y
this.z = z
}
show()
{
push()
translate(this.x, this.y, this.z)
lights()
fill('yellow')
noStroke()
sphere(10)
pop()
}
}
Console
Clear

Sketch D8.10 creating a spiral

Next let's try to make a spiral so that the array of bubbles also has a `z` component.

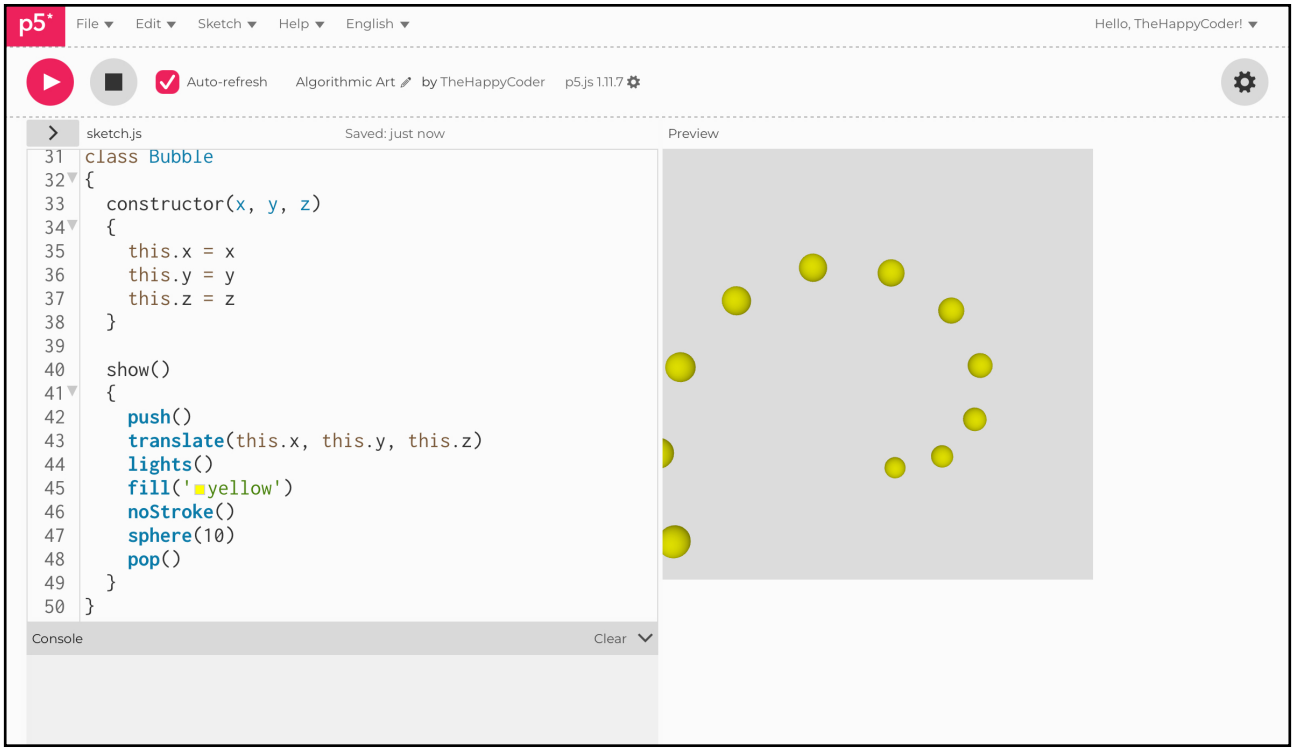
```
let bubbles = []
let bubble
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = 0; i < 360; i += 30)
  {
    let x = 100 * sin(i)
    let y = 100 * cos(i)
    bubble = new Bubble(x, y, i)
    bubbles.push(bubble)
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < bubbles.length; i++)
  {
    bubbles[i].show()
  }
  angle += 0.5
```

```
}

class Bubble
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    lights()
    fill('yellow')
    noStroke()
    sphere(10)
    pop()
  }
}
```

🗒️ Notes

It's a spiral Jim but not as we know it. Let's see if we can improve

Figure D8.10


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
class Bubble
{
constructor(x, y, z)
{
this.x = x
this.y = y
this.z = z
}
show()
{
push()
translate(this.x, this.y, this.z)
lights()
fill('yellow')
noStroke()
sphere(10)
pop()
}
}
Console
Clear

Sketch D8.11 more spirally

We can make some improvements to this to make it more spirally.

```
let bubbles = []
let bubble
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  for (let i = -360; i < 360; i += 10)
  {
    let x = 100 * sin(i)
    let y = 100 * cos(i)
    bubble = new Bubble(x, y, i/4)
    bubbles.push(bubble)
  }
}

function draw()
{
  background(220)
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  for (let i = 0; i < bubbles.length; i++)
  {
    bubbles[i].show()
  }
  angle += 0.5
}
```

```
class Bubble
{
  constructor(x, y, z)
  {
    this.x = x
    this.y = y
    this.z = z
  }

  show()
  {
    push()
    translate(this.x, this.y, this.z)
    lights()
    fill('yellow')
    noStroke()
    sphere(10)
    pop()
  }
}
```

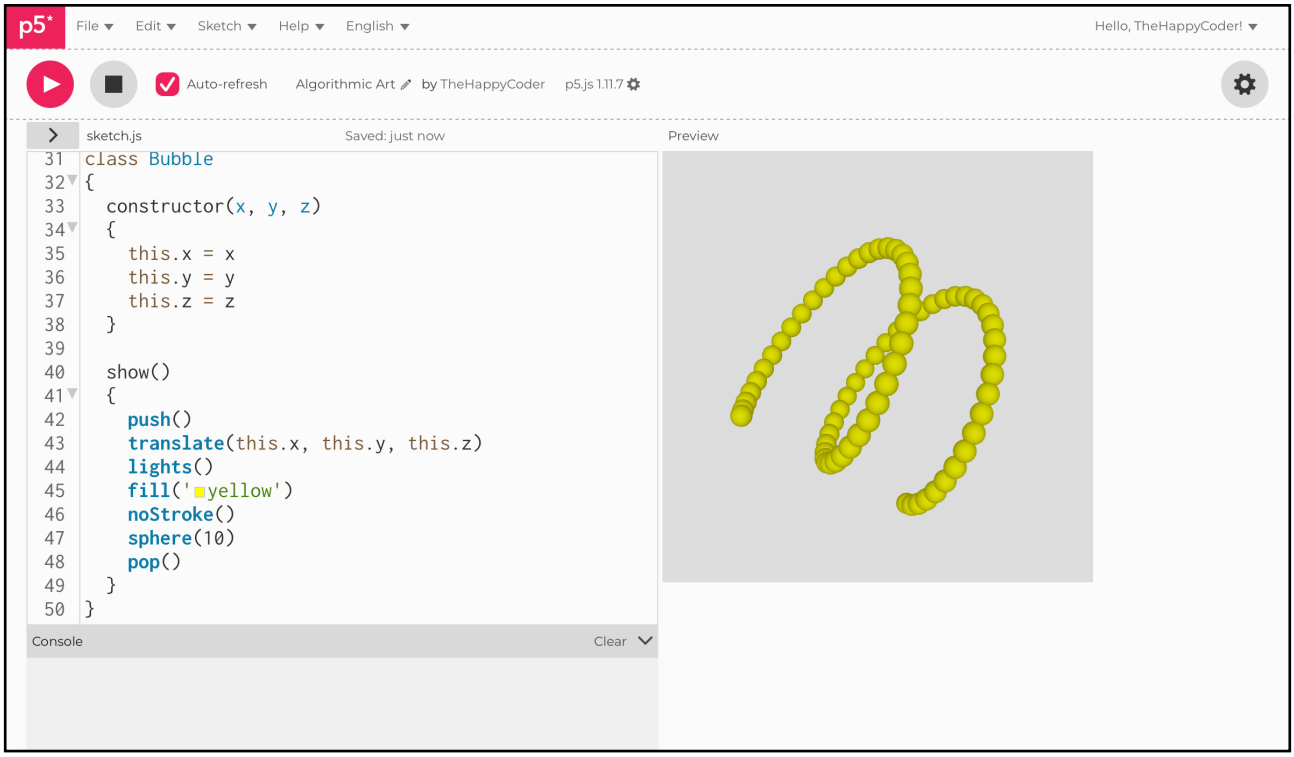
🗒️ Notes

Much better

🌻 Challenges

1. Can you randomise the diameter of the sphere
2. Add random colours to the spheres
3. Change the radius of the spiral so that it starts small and increases
4. How would extend it
5. How would you rotate it about an axis

Figure D8.11


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.7
sketch.js
Saved: just now
Preview
class Bubble
{
constructor(x, y, z)
{
this.x = x
this.y = y
this.z = z
}
show()
{
push()
translate(this.x, this.y, this.z)
lights()
fill('yellow')
noStroke()
sphere(10)
pop()
}
}
Console
Clear