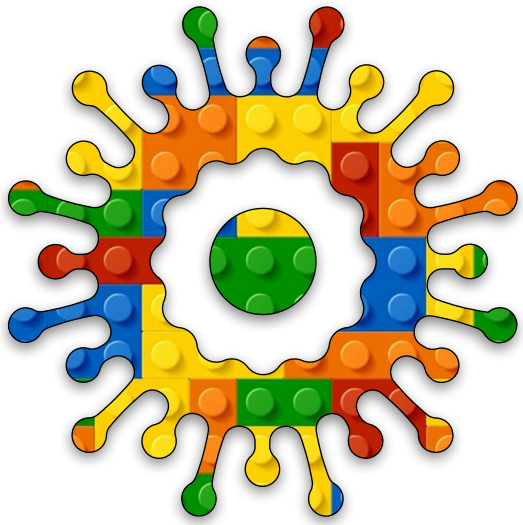
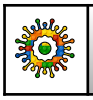
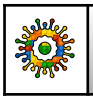
Creative Coding Module C Unit #5 graphics texture





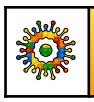
Content

Module C Unit #5 graphics texture



Introduction to graphics texture

This is a powerful function that is also extremely fun (in my opinion). It gives you the opportunity to put images, photos, videos and even draw on the faces of a shape while it is moving. For this unit, we are just keeping it simple to add shapes, text and colour for now.



Sketch C5.1 starting sketch

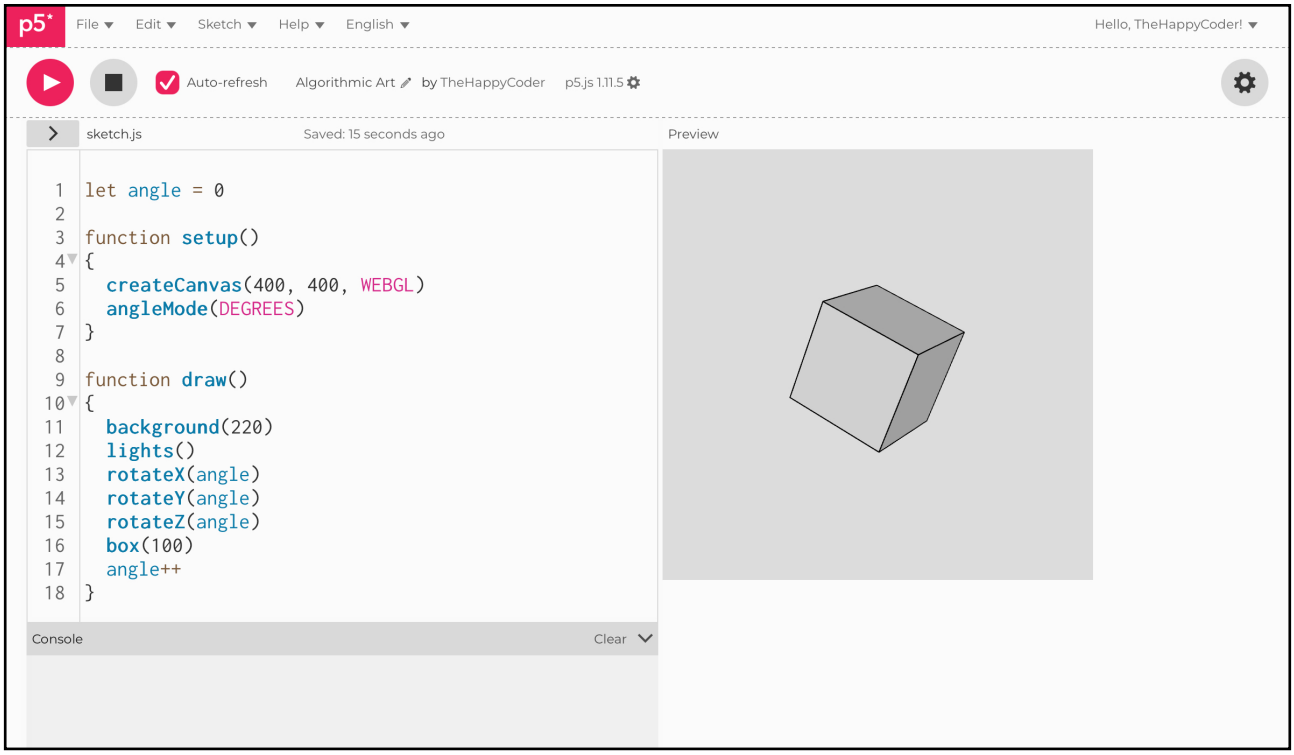
Our starting sketch gives us a nice rotating cube.

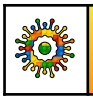
```
let angle = 0

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
}

function draw()
{
  background(220)
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  box(100)
  angle++
}
```

Figure C5.1


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: 15 seconds ago
Preview
let angle = 0
function setup()
{
createCanvas(400, 400, WEBGL)
angleMode(DEGREES)
}
function draw()
{
background(220)
lights()
rotateX(angle)
rotateY(angle)
rotateZ(angle)
box(100)
angle++
}
Console
Clear



Sketch C5.2 applying texture

We create a square object 400 by 400 and apply it as a texture to the box. The texture() function wraps an image onto a regular, primitive shape. Effectively, we have created and added another canvas onto the sides of the cube.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
}

function draw()
{
  background(220)
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  box(100)
  angle++
}
```

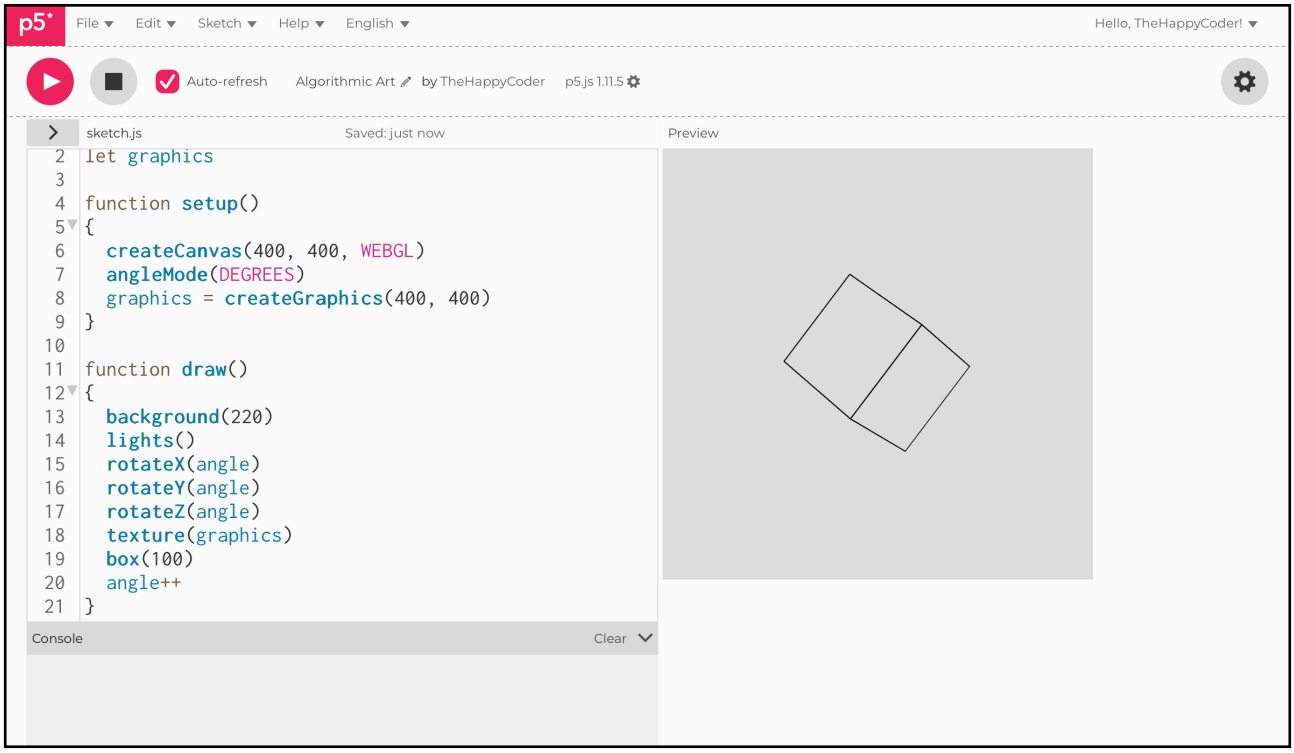
🗒️ Notes

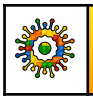
We haven't done anything with the texture as such, so it is a blank canvas. It is difficult to try to explain in words what is happening; that is why it is better to do it and see what it can produce. It makes it more intuitive than academic.

🛠️ Code Explanation

`let graphics`	Variable to hold the graphics object
`graphics = createGraphics(400, 400)`	Creating the graphics object and giving it a size of 400, 400
`texture(graphics)`	Putting the texture onto the cube with the graphics object

Figure C5.2


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: just now
Preview
let graphics
function setup()
{
createCanvas(400, 400, WEBGL)
angleMode(DEGREES)
graphics = createGraphics(400, 400)
}
function draw()
{
background(220)
lights()
rotateX(angle)
rotateY(angle)
rotateZ(angle)
texture(graphics)
box(100)
angle++
}
Console
Clear



Sketch C5.3 separate canvas

Now we can manipulate the texture as a separate canvas. Starting simple, we give it a yellow background.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
}

function draw()
{
  background(220)
  graphics.background(200, 200, 0)
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  box(100)
  angle++
}
```

Notes

We now have a yellow cube; we haven't filled it with a colour but treated each side as a canvas and given it a background colour. We call the new `background()` function onto the `graphics` object.

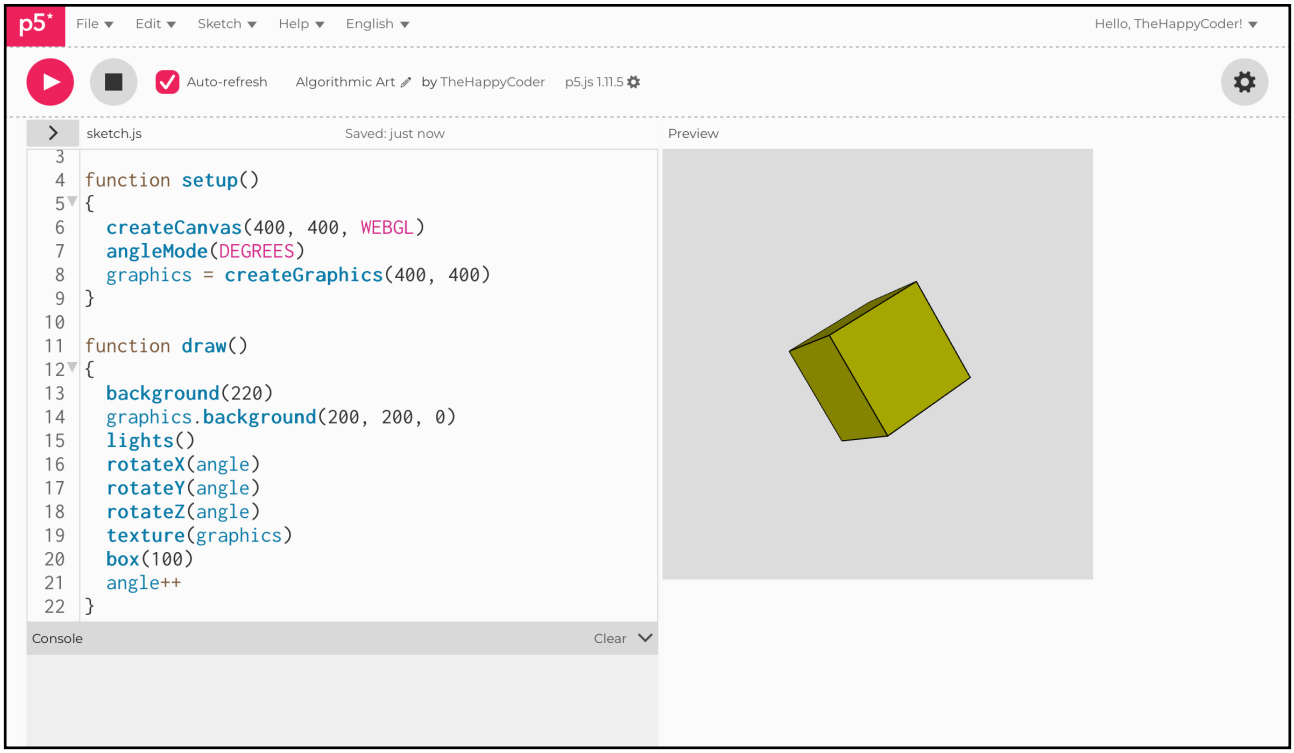
🌻 Challenges

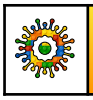
1. Change the colour.
2. Try `graphics.fill()` to see if that does anything.

🛠️ Code Explanation

`graphics.background(200, 200, 0)`	Gives the graphics canvas a background

Figure C5.3


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: just now
Preview
function setup()
{
createCanvas(400, 400, WEBGL)
angleMode(DEGREES)
graphics = createGraphics(400, 400)
}
function draw()
{
background(220)
graphics.background(200, 200, 0)
lights()
rotateX(angle)
rotateY(angle)
rotateZ(angle)
texture(graphics)
box(100)
angle++
}
Console
Clear



Sketch C5.4 adding graphics

We can draw shapes on the side of the cube; in this case, we can draw a circle which appears on all the surfaces of the box.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
}

function draw()
{
  background(220)
  graphics.background(200, 200, 0)
  graphics.circle(100, 100, 150)
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  box(100)
  angle++
}
```

Notes

Notice that the circle really does appear on each side of the cube; this shows that the texture is applied separately to each side of the cube. The size of the canvas (and circle) is scaled to fit.

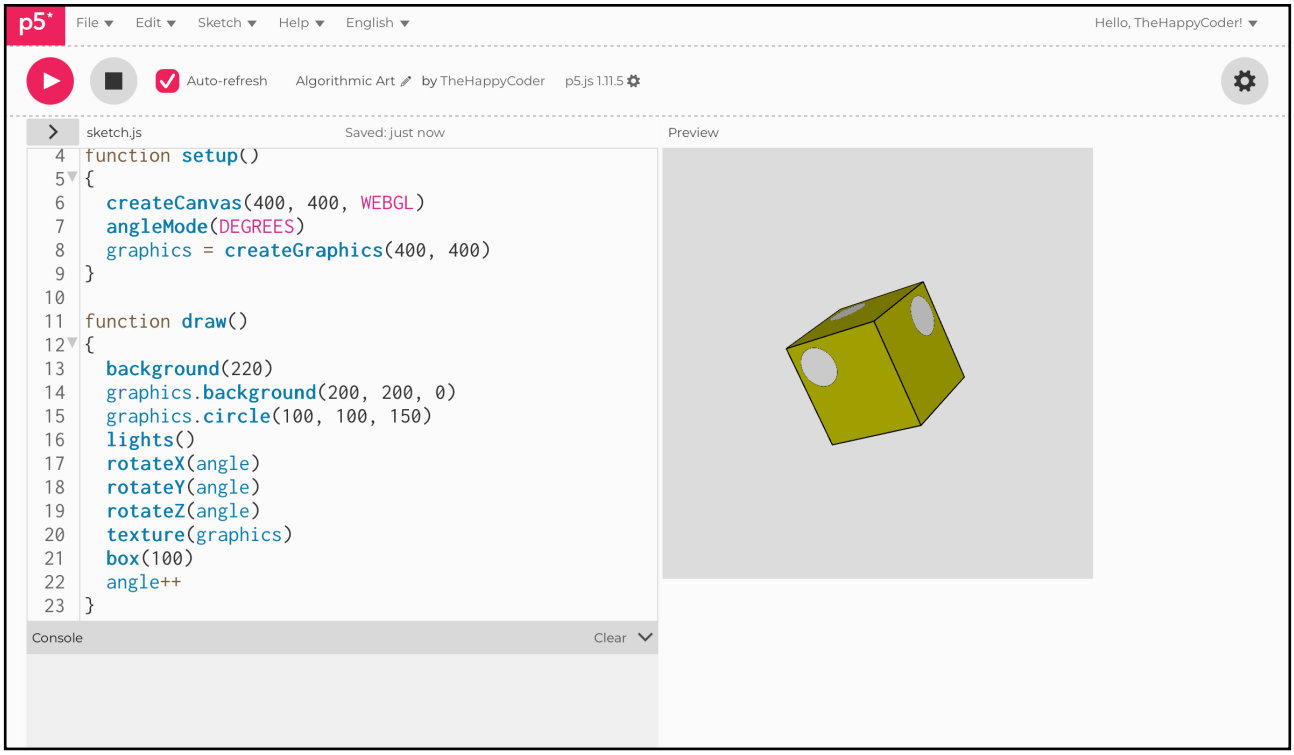
🌻 Challenges

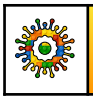
1. Change the size of the graphics to (100, 400).
2. Try other 2D shapes.

🛠️ Code Explanation

`graphics.circle(100, 100, 150)`	Creating a circle on the graphics canvas, 100 in, 100 down and radius 150

Figure C5.4


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: just now
Preview
4 function setup()
5 {
6 createCanvas(400, 400, WEBGL)
7 angleMode(DEGREES)
8 graphics = createGraphics(400, 400)
9 }
10
11 function draw()
12 {
13 background(220)
14 graphics.background(200, 200, 0)
15 graphics.circle(100, 100, 150)
16 lights()
17 rotateX(angle)
18 rotateY(angle)
19 rotateZ(angle)
20 texture(graphics)
21 box(100)
22 angle++
23 }
Console
Clear



Sketch C5.5 drawing on the sides

We can fill the circle with a colour (red in this case), and we can also have the circle move across the sides of the box as you move your mouse across the main canvas.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
}

function draw()
{
  background(220)
  graphics.background(200, 200, 0)
  graphics.fill(255, 0, 0)
  graphics.circle(mouseX, mouseY, 150)
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  box(100)
  angle++
}
```

🗒️ Notes

The circle follows the mouse across the main canvas.

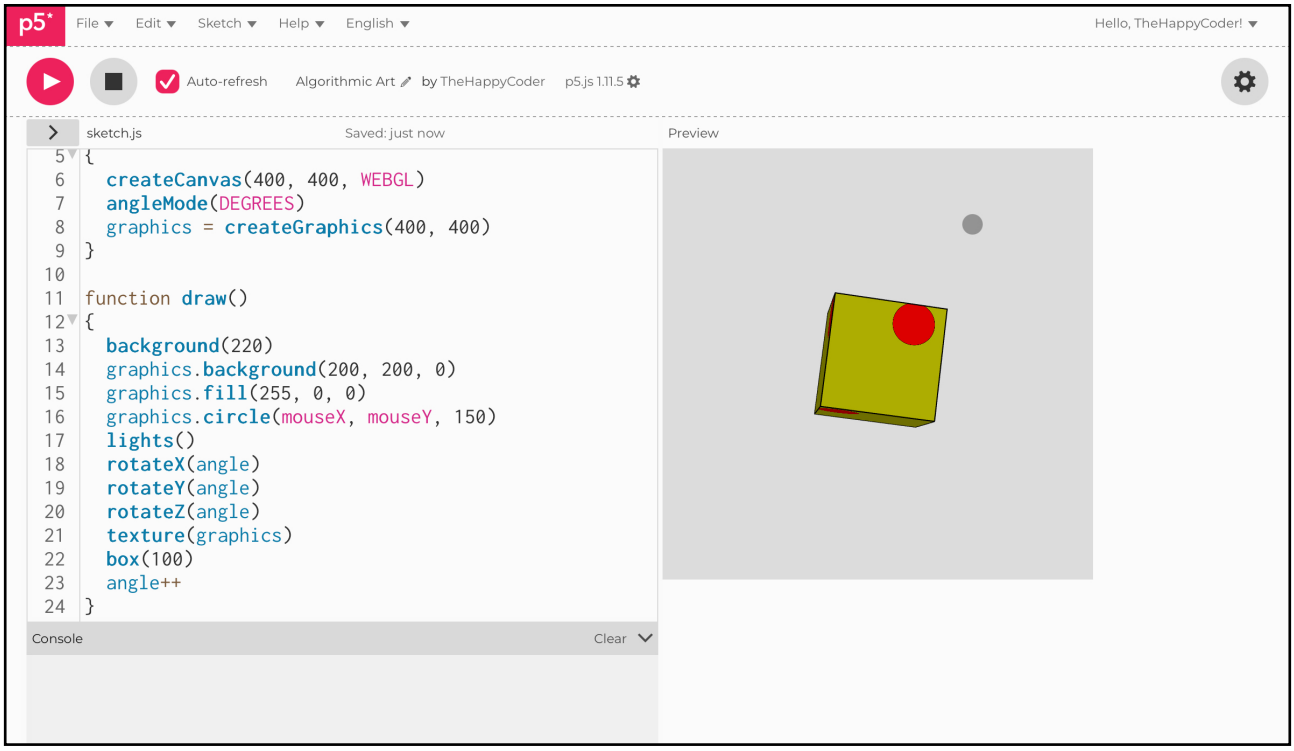
🌻 Challenge

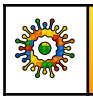
Could you get the circle to bounce off the edges?

🛠️ Code Explanation

`graphics.fill(255, 0, 0)`	Fills the circle with red colour
`graphics.circle(mouseX, mouseY, 150)`	Gives the x and y coordinates of the mouse

Figure C5.5


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: just now
Preview
5 {
6 createCanvas(400, 400, WEBGL)
7 angleMode(DEGREES)
8 graphics = createGraphics(400, 400)
9 }
10
11 function draw()
12 {
13 background(220)
14 graphics.background(200, 200, 0)
15 graphics.fill(255, 0, 0)
16 graphics.circle(mouseX, mouseY, 150)
17 lights()
18 rotateX(angle)
19 rotateY(angle)
20 rotateZ(angle)
21 texture(graphics)
22 box(100)
23 angle++
24 }
Console
Clear



Sketch C5.6 a bit of fun tweaking

Watch what happens when we comment out the graphics background and add in `noStroke()`.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
}

function draw()
{
  background(220)
  // graphics.background(200, 200, 0)
  graphics.fill(255, 0, 0)
  graphics.circle(mouseX, mouseY, 150)
  noStroke()
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  box(100)
  angle++
}
```

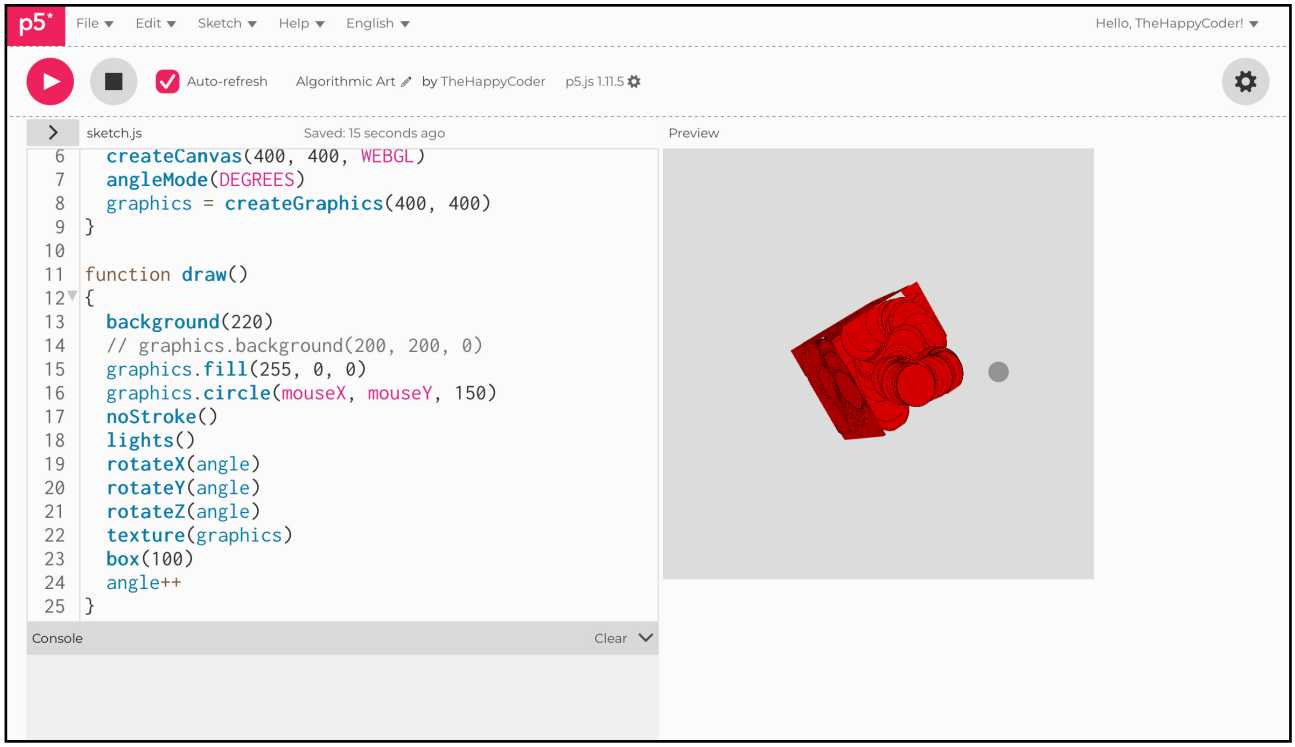
🗒️ Notes

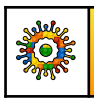
We have an invisible cube to paint on. The `noStroke()` only works on the cube, not the circle.

🌻 Challenge

How would you use `noStroke()` on the circle?

Figure C5.6


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: 15 seconds ago
Preview
createCanvas(400, 400, WEBGL)
angleMode(DEGREES)
graphics = createGraphics(400, 400)
}
function draw()
{
background(220)
// graphics.background(200, 200, 0)
graphics.fill(255, 0, 0)
graphics.circle(mouseX, mouseY, 150)
noStroke()
lights()
rotateX(angle)
rotateY(angle)
rotateZ(angle)
texture(graphics)
box(100)
angle++
}
Console
Clear



Sketch C5.7 creating text

We can also create text on the cube.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
  graphics.textAlign(CENTER, CENTER)
}

function draw()
{
  background(220)
  // graphics.background(200, 200, 0)
  graphics.textSize(75)
  graphics.text('HAPPY', 200, 200)
  noStroke()
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  box(100)
  angle++
}
```

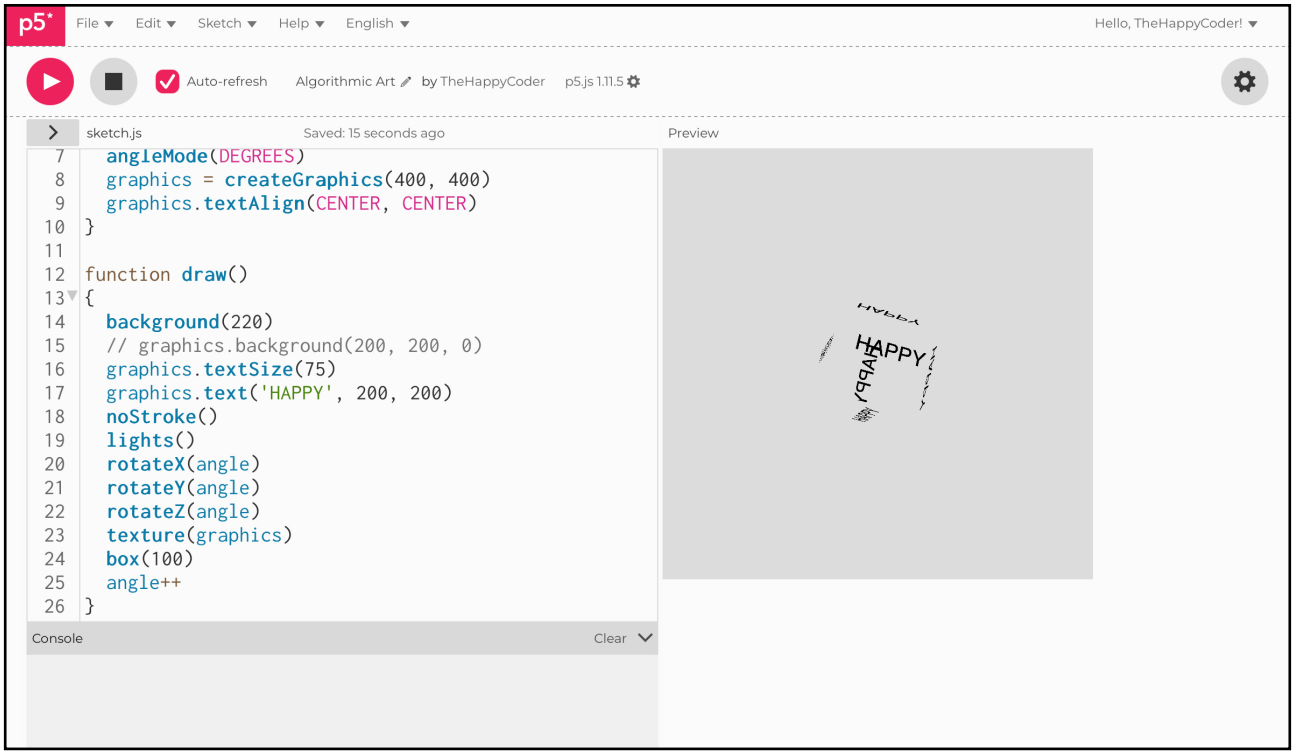
🗒️ Notes

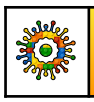
Notice that we had to prefix the `textAlign()` with the `graphics` reference.

🌻 Challenge

Could you rotate the text at the same time?

Figure C5.7


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: 15 seconds ago
Preview
7 angleMode(DEGREES)
8 graphics = createGraphics(400, 400)
9 graphics.textAlign(CENTER, CENTER)
10 }
11
12 function draw()
13 {
14 background(220)
15 // graphics.background(200, 200, 0)
16 graphics.textSize(75)
17 graphics.text('HAPPY', 200, 200)
18 noStroke()
19 lights()
20 rotateX(angle)
21 rotateY(angle)
22 rotateZ(angle)
23 texture(graphics)
24 box(100)
25 angle++
26 }
Console
Clear



Sketch C5.8 words on a plane

If we put it on a plane...

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
  graphics.textAlign(CENTER, CENTER)
}

function draw()
{
  background(220)
  // graphics.background(200, 200, 0)
  graphics.textSize(75)
  graphics.text('HAPPY', 200, 200)
  noStroke()
  lights()
  rotateX(angle)
  rotateY(angle)
  rotateZ(angle)
  texture(graphics)
  plane(100)
  angle++
}
```

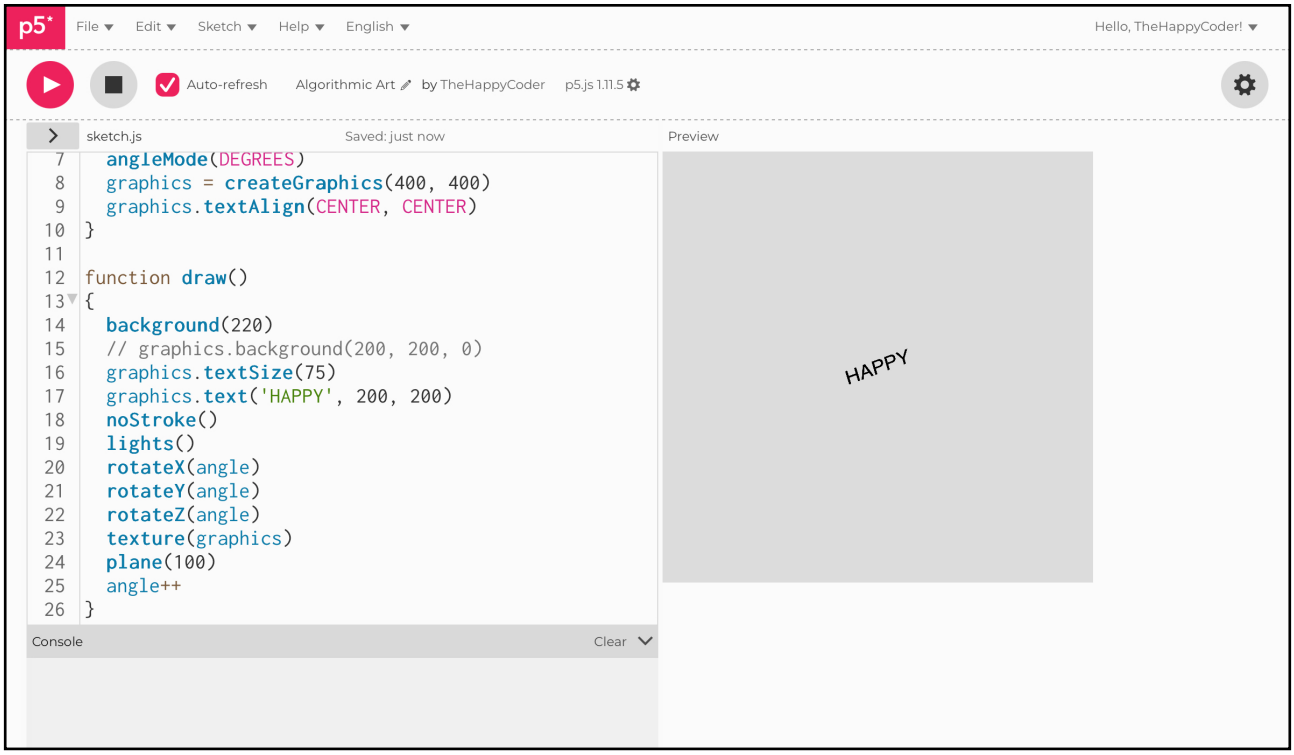
🗒️ Notes

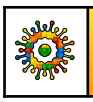
What is interesting is that the text always ends up the right way (not upside down or reversed) when you use all three axes.

🌻 Challenge

See what happens when you only have the x or y axis of rotation.

Figure C5.8


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: just now
Preview
angleMode(DEGREES)
graphics = createGraphics(400, 400)
graphics.textAlign(CENTER, CENTER)
}
function draw()
{
background(220)
// graphics.background(200, 200, 0)
graphics.textSize(75)
graphics.text('HAPPY', 200, 200)
noStroke()
lights()
rotateX(angle)
rotateY(angle)
rotateZ(angle)
texture(graphics)
plane(100)
angle++
}
HAPPY
Console
Clear



Sketch C5.9 text on a cylinder

What about round the side of a cylinder? We need to make some adaptations to get this to work reasonably. You can play with everything later.

```
let angle = 0
let graphics

function setup()
{
  createCanvas(400, 400, WEBGL)
  angleMode(DEGREES)
  graphics = createGraphics(400, 400)
  graphics.textAlign(CENTER, CENTER)
}

function draw()
{
  background(220)
  graphics.background(220)
  graphics.textSize(50)
  graphics.text('HAPPY CODER', 200, 200)
  noStroke()
  // lights()
  // rotateX(angle)
  rotateY(-angle)
  // rotateZ(angle)
  texture(graphics)
  cylinder(50, 200)
  angle++
}
```

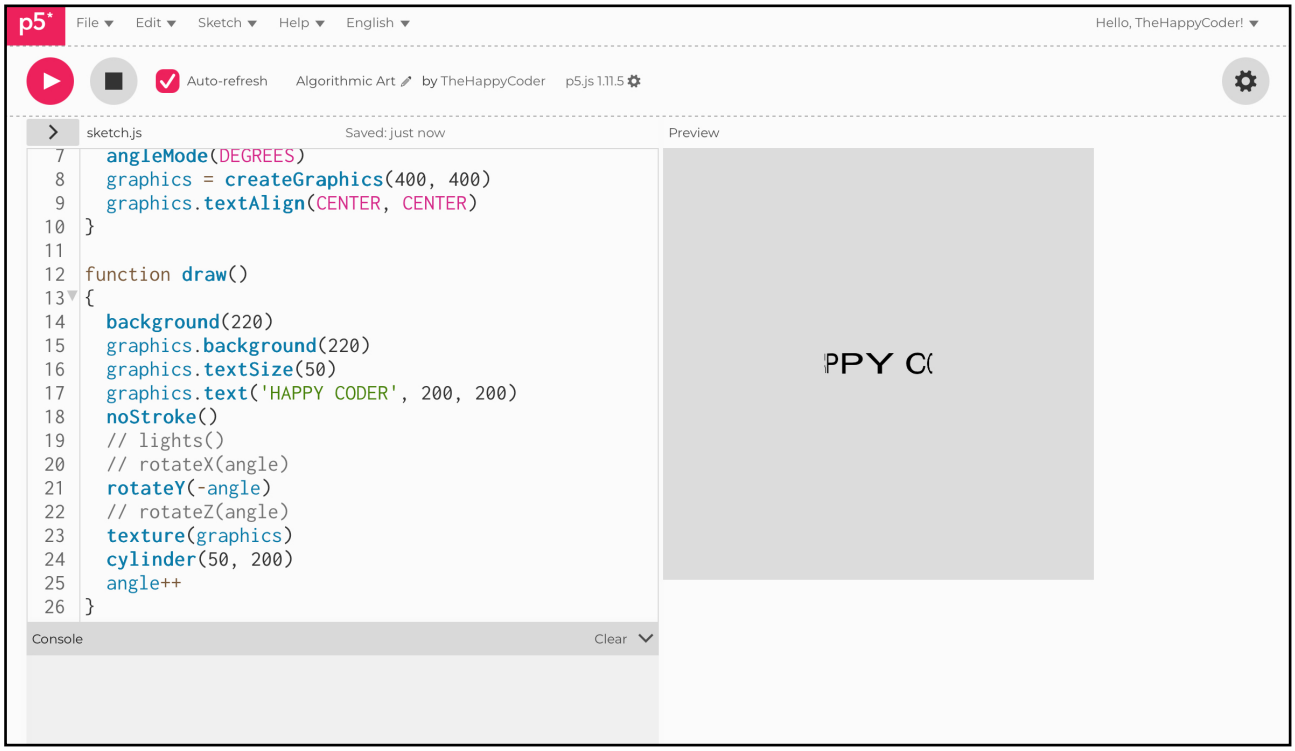
🗒️ Notes

Quite a bit of refactoring. I won't go into all of it, but I will allow you to experiment with what works or doesn't work.

🌻 Challenges

1. Try other shapes.
2. Add some colour.
3. Could you get the text to move up and down?

Figure C5.9


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Algorithmic Art
by TheHappyCoder
p5.js 1.11.5
sketch.js
Saved: just now
Preview
angleMode(DEGREES)
graphics = createGraphics(400, 400)
graphics.textAlign(CENTER, CENTER)
}
function draw()
{
background(220)
graphics.background(220)
graphics.textSize(50)
graphics.text('HAPPY CODER', 200, 200)
noStroke()
// lights()
// rotateX(angle)
rotateY(-angle)
// rotateZ(angle)
texture(graphics)
cylinder(50, 200)
angle++
}
Console
Clear