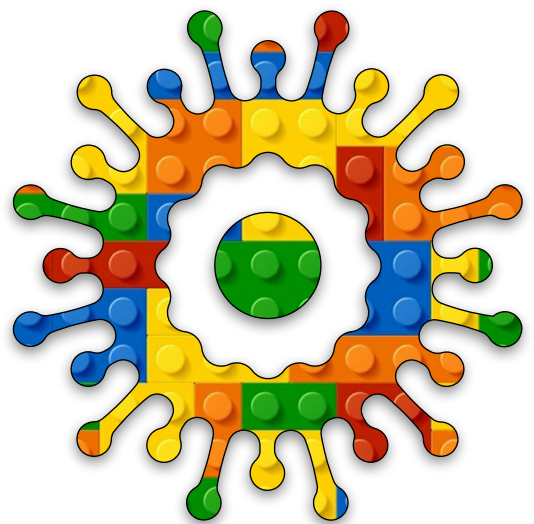


Creative  
Coding  
Module B  
Unit #4  
RGB colour  
slider





### Module B Unit #4 RGB colour slider

|              |                          |
|--------------|--------------------------|
| Sketch B4.1  | using RGB for the colour |
| Sketch B4.2  | colorMode()              |
| Sketch B4.3  | starting sketch          |
| Sketch B4.4  | create a slider          |
| Sketch B4.5  | slider value             |
| Sketch B4.6  | position slider          |
| Sketch B4.7  | showing the value        |
| Sketch B4.8  | text colour              |
| Sketch B4.9  | red value                |
| Sketch B4.10 | full RGB                 |



## Introducing RGB and the slider

So far, I have introduced colour as a name and as RGB values with red, green, and blue components. Before jumping into the other colour methods, I just want to expand on RGB.

### Key concepts



colorMode()



sliders



RGB slider colour change



## Sketch B4.1 using RGB for the colour

We have already introduced RGB. Here is a reminder: a green circle on an orange background.

```
function setup()
{
  createCanvas(400, 400)
}

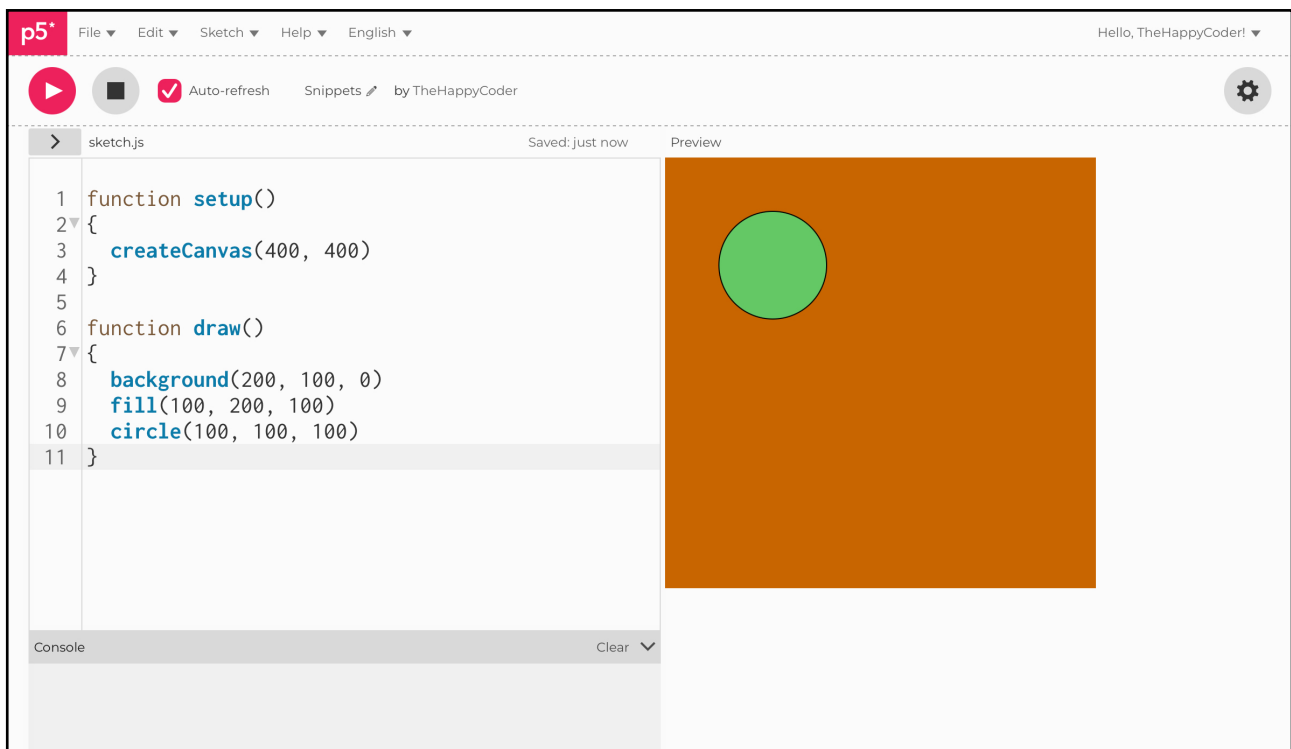
function draw()
{
  background(200, 100, 0)
  fill(100, 200, 100)
  circle(100, 100, 100)
}
```



### Notes

By default, it assumes values (arguments) are for RGB, three for the red, blue, and green; the fourth is the amount of transparency.

Figure B4.1





## Sketch B4.2 colorMode()

We have even introduced `colorMode()` briefly. Here we can explore it more as we look at other colour modes. If we have `colorMode(RGB, 100)`, we can scale it to `100` (think of it as a percentage).

```
function setup()
{
  createCanvas(400, 400)
  colorMode(RGB, 100)
}

function draw()
{
  background(100, 100, 0)
  fill(0, 75, 75)
  circle(100, 100, 100)
}
```



### Notes

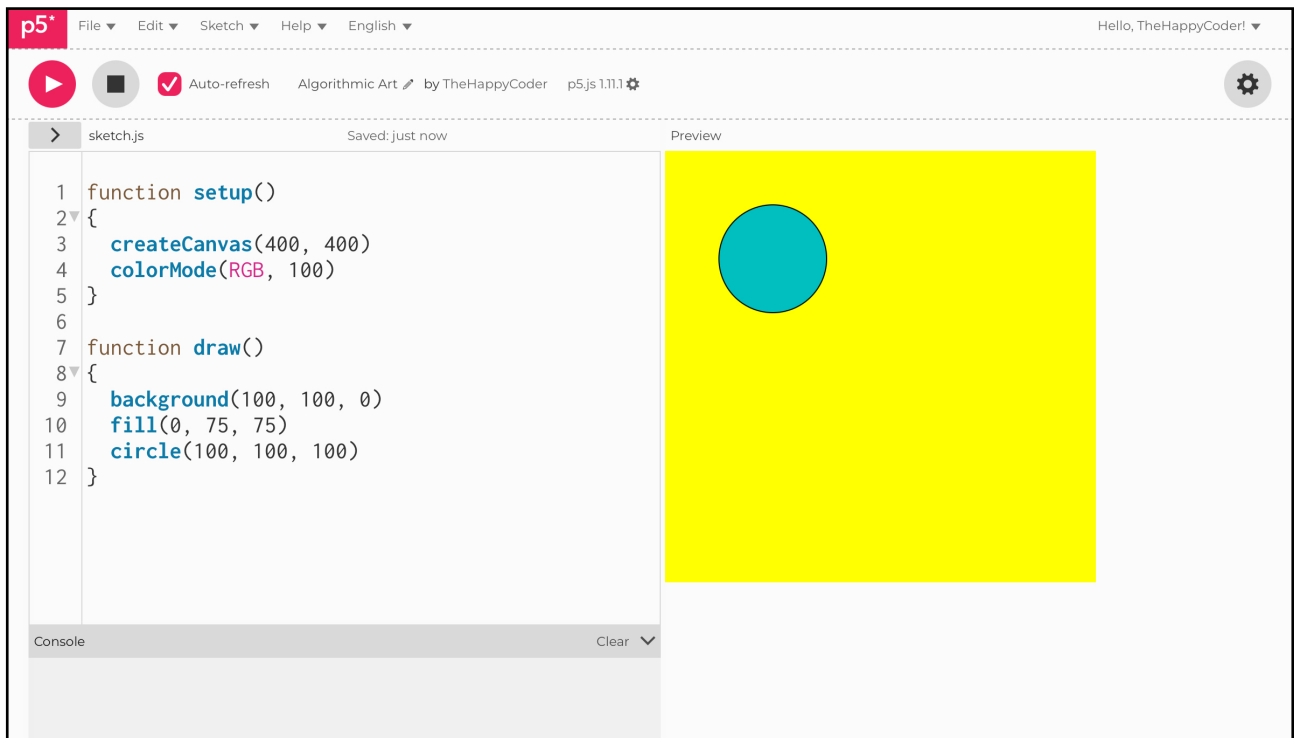
All the RGB values should be between `0` and `100`. If they exceed that value, then it will take the maximum, in this case `100`.



### Challenge

You can add more arguments (any values you like), one for each element including alpha: `colorMode(RGB, 100, 100, 100, 100)`. Try it.

Figure B4.2





## Sketch B4.3 starting sketch

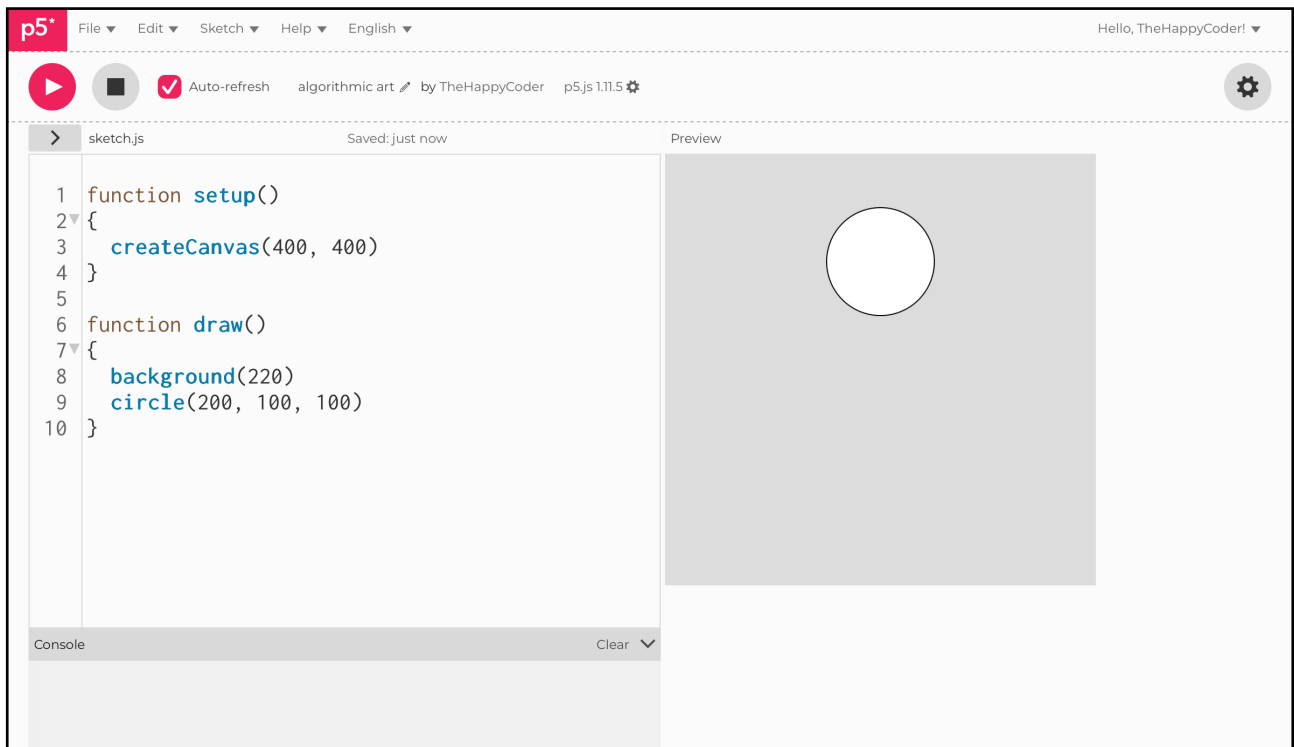
! new sketch

Circle on a blank canvas.

```
function setup()
{
  createCanvas(400, 400)
}

function draw()
{
  background(220)
  circle(200, 100, 100)
}
```

Figure B4.3





## Sketch B4.4 create a slider

Firstly, we give the slider a name, in this case the rather unimaginative name of **slider**. Secondly, we create the slider with three arguments:

- A. Begin value (**0**)
- B. End value (**255**)
- C. Default value (**100**)

The range is between **0** and **255**. We will set the slider to **100** initially when you run the sketch. You can give it any value you want. Just use your mouse to slide it backwards and forwards.

```
let slider

function setup()
{
  createCanvas(400, 400)
  slider = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  circle(200, 100, 100)
}
```



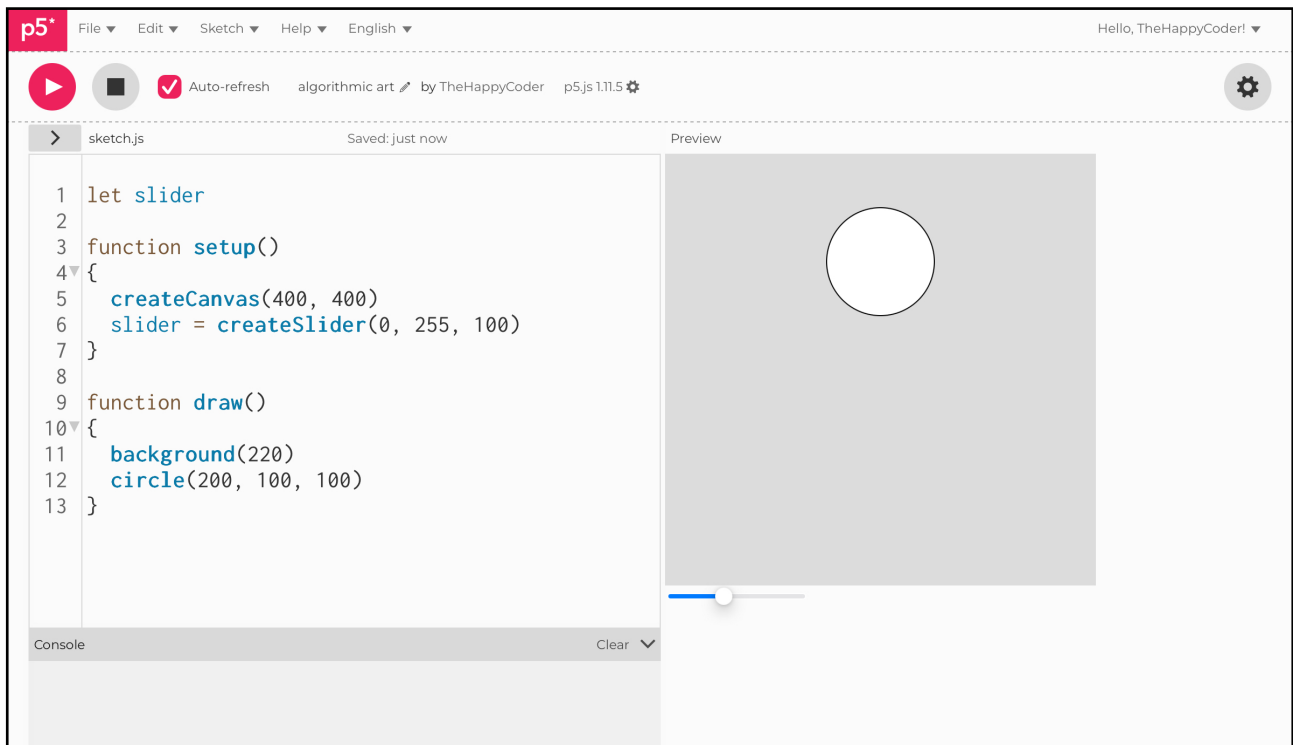
### Notes

You will notice that the slider is off the canvas; this is because we have not given it any coordinates, so by default it puts it at the bottom, below the canvas. Also, the slider doesn't do anything except slide!

## Code Explanation

|                                                 |                                                        |
|-------------------------------------------------|--------------------------------------------------------|
| <code>let slider</code>                         | Slider object                                          |
| <code>slider = createSlider(0, 255, 100)</code> | Create the slider object with the corresponding values |

Figure B4.4





## Sketch B4.5 slider value

We can take the value of the slider using `slider.value()`, the `value()` function returns the value. We can fill the circle with that value. Now when you move the slider, it changes the colour of the circle.

```
let slider

function setup()
{
  createCanvas(400, 400)
  slider = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  fill(slider.value())
  circle(200, 100, 100)
}
```



### Notes

You should see it change as you move the slider. We have put it straight into the `fill()` function.



### Challenges

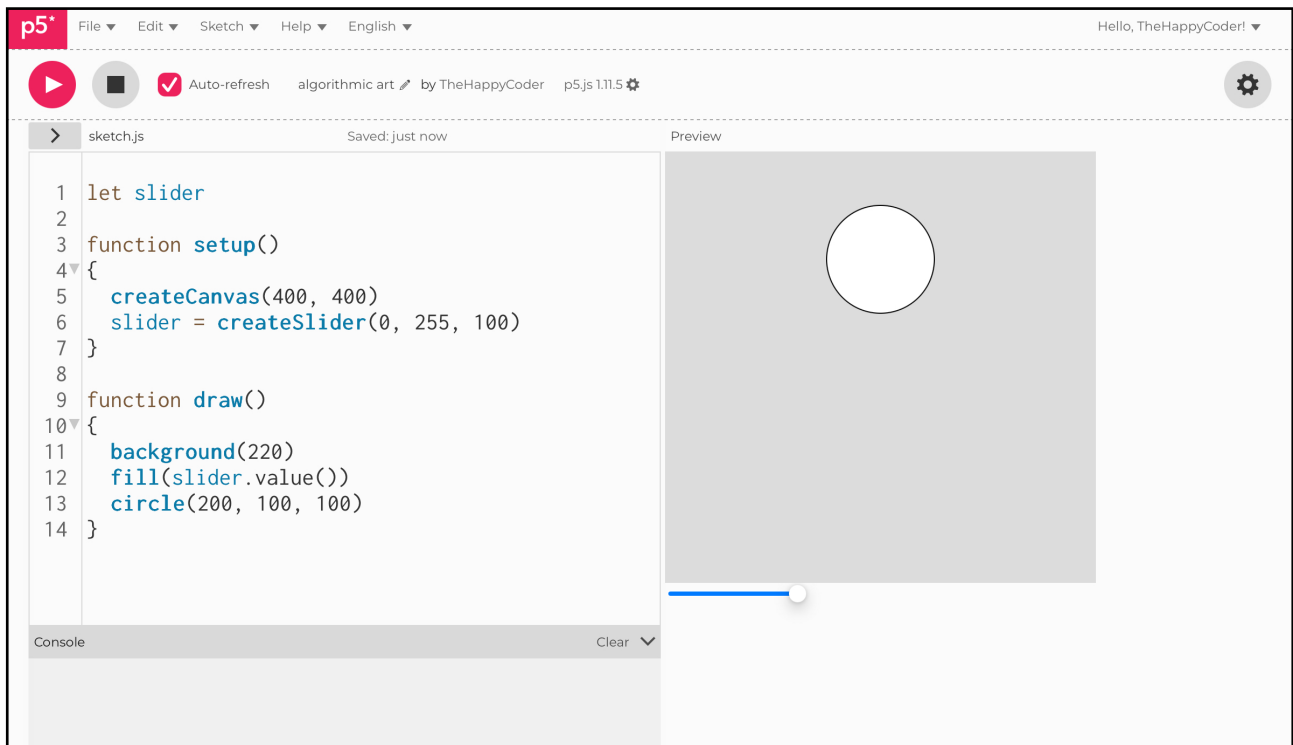
1. Try different start, final, and initial values for the slider.
2. You could create a variable for the slider value.



### Code Explanation

|                                   |                                 |
|-----------------------------------|---------------------------------|
| <code>fill(slider.value())</code> | Returns the value of the slider |
|-----------------------------------|---------------------------------|

Figure B4.5





## Sketch B4.6 position slider

We will now move it to the canvas; we simply give it the coordinates we want to display it at.

```
let slider

function setup()
{
  createCanvas(400, 400)
  slider = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  slider.position(50, 200)
  fill(slider.value())
  circle(200, 100, 100)
}
```



### Notes

Slider now on the canvas.



### Challenge

Try different positions.

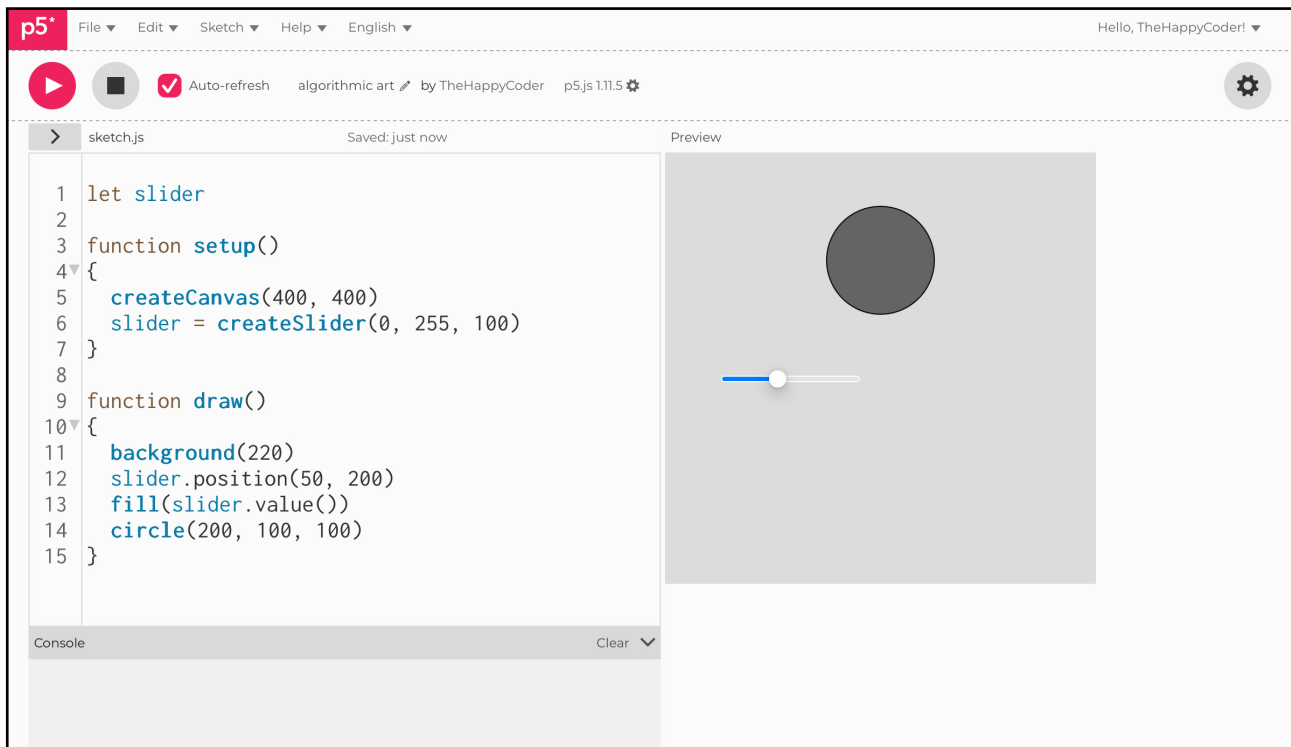


### Code Explanation

```
slider.position(50, 200)
```

Positioned the slider at 50 from the left and 200 down from the top

Figure B4.6





## Sketch B4.7 showing the value

We can display the value of the slider.

```
let slider

function setup()
{
  createCanvas(400, 400)
  slider = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  slider.position(50, 200)
  fill(slider.value())
  circle(200, 100, 100)
  textSize(20)
  text(slider.value(), 200, 215)
}
```



### Notes

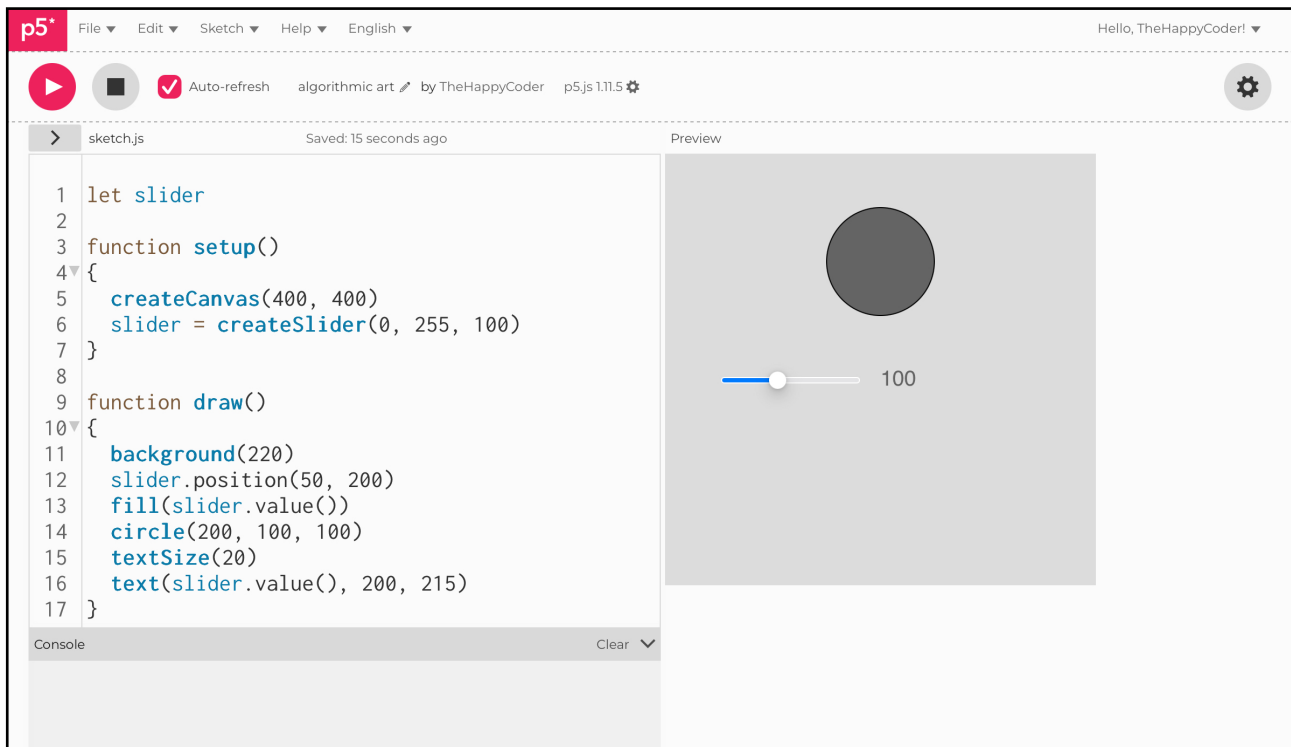
One issue, as you will see when you slide the slider, is that the text changes colour as well.



### Challenge

Do you know how to stop that?

Figure B4.7





## Sketch B4.8 text colour

Just to keep the text from changing colour.

```
let slider

function setup()
{
  createCanvas(400, 400)
  slider = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  slider.position(50, 200)
  fill(slider.value())
  circle(200, 100, 100)
  fill(0)
  textSize(20)
  text(slider.value(), 200, 215)
}
```



### Notes

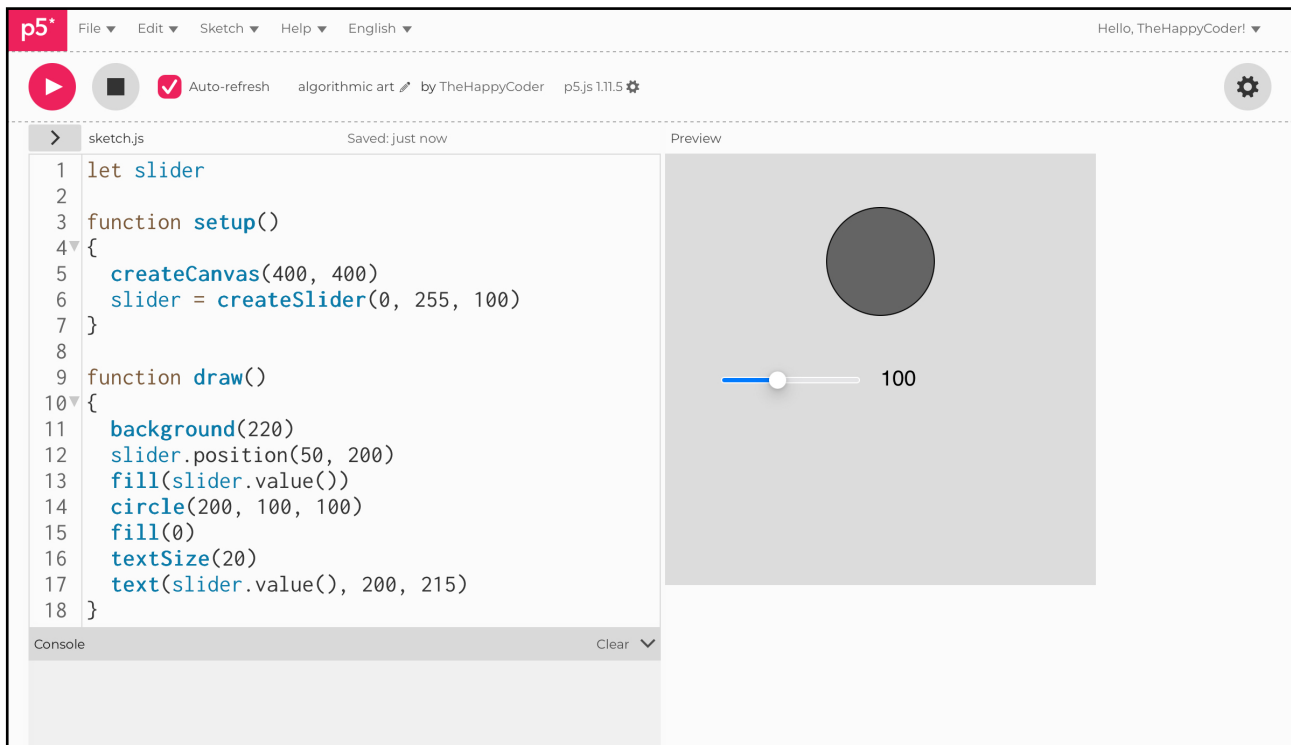
We have taken this bit nicely and slowly; next, we will look at red, green, and blue.



### Challenge

Any ideas how to do red, green, and blue?

Figure B4.8





## Sketch B4.9 red value

We need to change the name of the slider to `sliderRed()` everywhere, and also change the `fill()` to include the values of the green and blue (`0`).

```
let sliderRed

function setup()
{
  createCanvas(400, 400)
  sliderRed = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  sliderRed.position(50, 200)
  fill(sliderRed.value(), 0, 0)
  circle(200, 100, 100)
  fill(0)
  textSize(20)
  text(sliderRed.value(), 200, 215)
}
```



### Notes

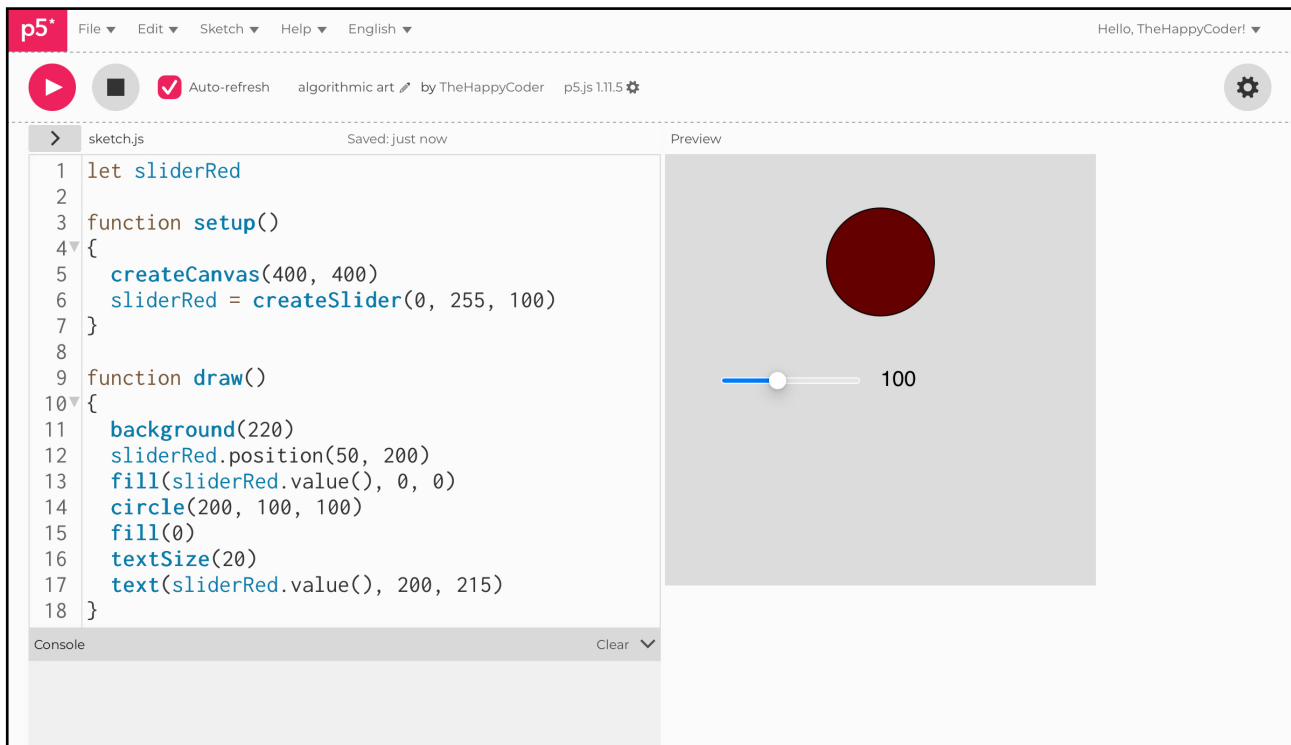
The circle will change from black to bright red.



### Challenge

How about adding the green and the blue?

Figure B4.9





## Sketch B4.10 full RGB

Adding the other colours.

```
let sliderRed
let sliderGreen
let sliderBlue

function setup()
{
  createCanvas(400, 400)
  sliderRed = createSlider(0, 255, 100)
  sliderGreen = createSlider(0, 255, 100)
  sliderBlue = createSlider(0, 255, 100)
}

function draw()
{
  background(220)
  sliderRed.position(50, 200)
  sliderGreen.position(50, 250)
  sliderBlue.position(50, 300)
  fill(sliderRed.value(), sliderGreen.value(),
sliderBlue.value())
  circle(200, 100, 100)
  fill(0)
  textSize(20)
  text(sliderRed.value(), 200, 215)
  text(sliderGreen.value(), 200, 265)
  text(sliderBlue.value(), 200, 315)
}
```



## Notes

A lot of copying and pasting.



## Challenge

Have it change the background instead.

Figure B4.10

