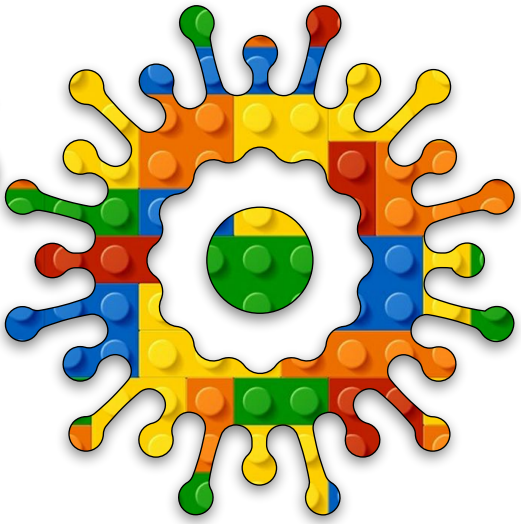
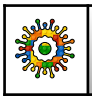
Creative Coding Module B Unit #6 colour charts and pickers

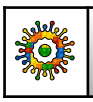




Content

Module B Unit #6 colour charts

- ☐ Pink
- ☐ Purple
- ☐ Yellow
- ☐ Red
- ☐ Orange
- ☐ Green
- ☐ Blue
- ☐ Brown
- ☐ Cyan
- ☐ Grey
- ☐ White



Introduction to colour charts and pickers

In this brief unit, we will look at two other colouring systems that use names and hexadecimal values. The hexadecimal system is quite common in coding circles. They are available across all browsers, and that is why they are often used for websites, etc.

Also, we will look at something you might find useful or interesting called lerpColour(). This interpolates between two colours, however identified, and calculates all the values in between in increments. So, if you want to see the colour change from, say, red to orange in increments.

Finally, I have included a comprehensive list of all the colour names, their hex and RGB corresponding values. A good resource if nothing else.

Sketch B6.1 using a name for the colour

We can, if we wish, just name the colours. There is a wide range of colours, too numerous to list them here (see end of unit for full list). The name must be in speech marks.

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

function draw()
{
  background('orange')
  fill('green')
  circle(100, 100, 100)
}
```

Notes

You will notice that it isn't the same exact colour. Because you put the name of the colour in speech marks, it gives you a square indicator to the colour. There are quite a few colour names. If you use `lightgreen` with no gap, you get light green. Some colours will take the word dark, e.g. `darkred`. Other names are ones like `teal`, `magenta`, and so on. You can use single or double quotes. It is useful if you just want a simple colour rather than trying to remember the values, especially if you want more than three colours.

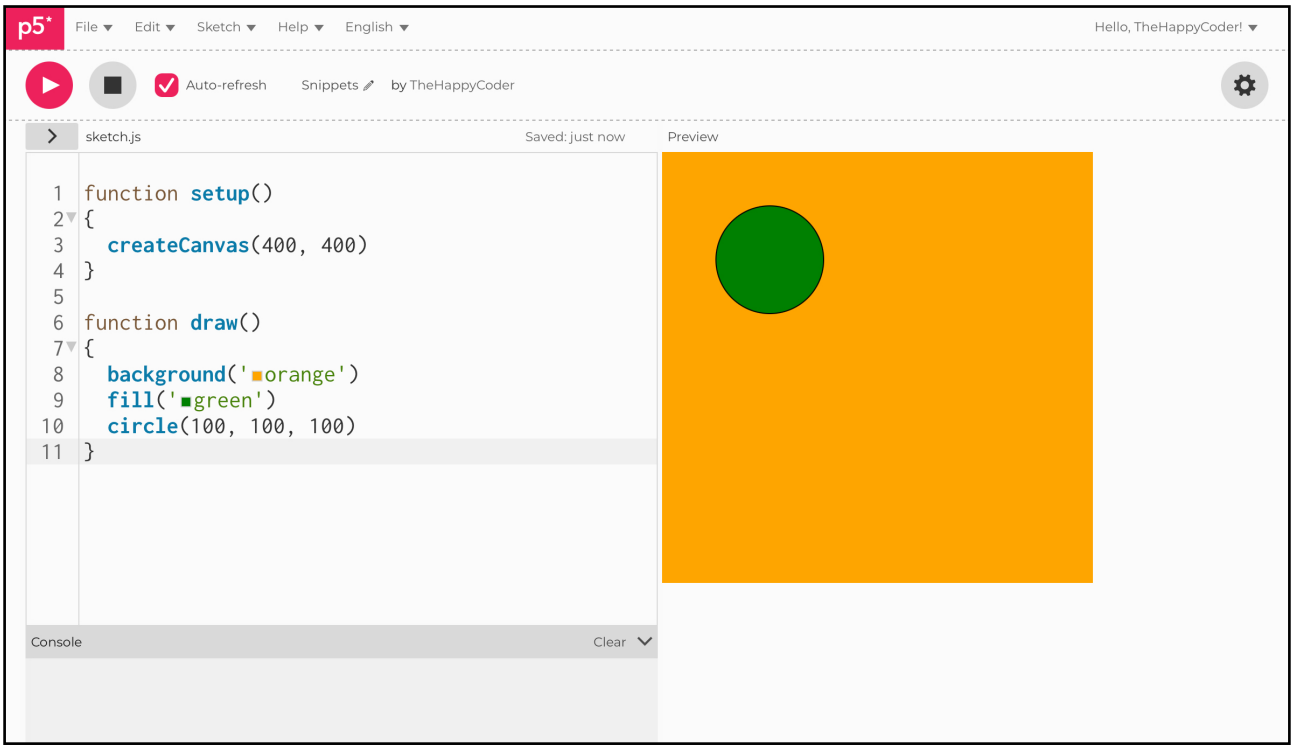
Challenges

1. Just experiment.
2. Search online for JavaScript colour names (or see the charts at the end of this unit).

🛠️ Code Explanation

`background('orange')`	Gives you an orange background
`fill('green')`	Fills the circle green

Figure B6.1


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
Snippets
by TheHappyCoder
sketch.js
Saved: just now
Preview
1 function setup()
2 {
3 createCanvas(400, 400)
4 }
5
6 function draw()
7 {
8 background('orange')
9 fill('green')
10 circle(100, 100, 100)
11 }
Console
Clear

Sketch B6.2 using a hex value for the colour

Hex, this is one colour format often used in website design, and it is a hexadecimal number (base 16) that starts with a # symbol. The hex number indicates a particular colour; you will notice that it uses letters as well as numbers. To explore the hex values, I will include a colour picker so that you can copy and paste the colour of your choice using the hex values.

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

function draw()
{
  background('#0000FF')
  fill('#FF0000')
  circle(100, 100, 100)
}
```

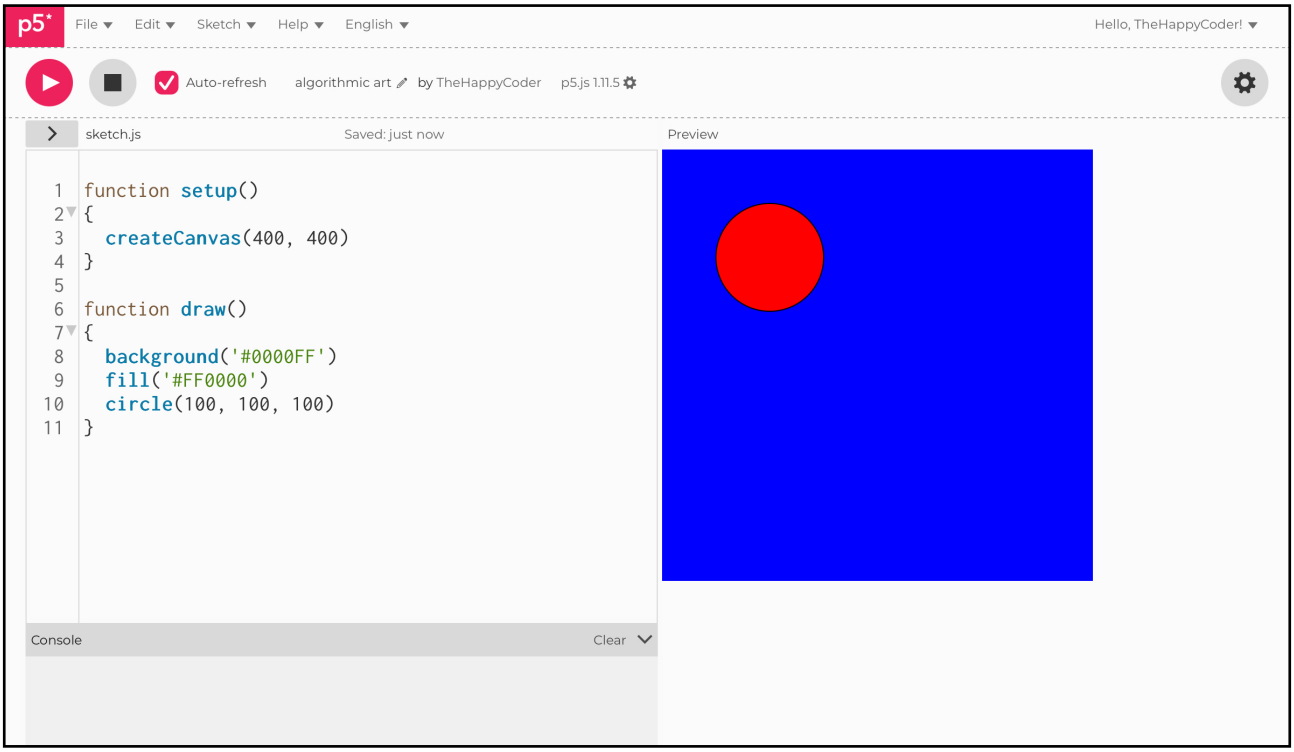
Notes

Here are some basic colours; notice that when you code it, you get the colour icon as well. See the colour charts at the end of this unit for more values.

Code Explanation

#FF0000	Red
#00FF00	Green
#0000FF	Blue
#000000	Black
#FFFFFF	White

Figure B6.2


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
1 function setup()
2 {
3 createCanvas(400, 400)
4 }
5
6 function draw()
7 {
8 background('#0000FF')
9 fill('#FF0000')
10 circle(100, 100, 100)
11 }
Console
Clear

Sketch B6.3 bench of ten

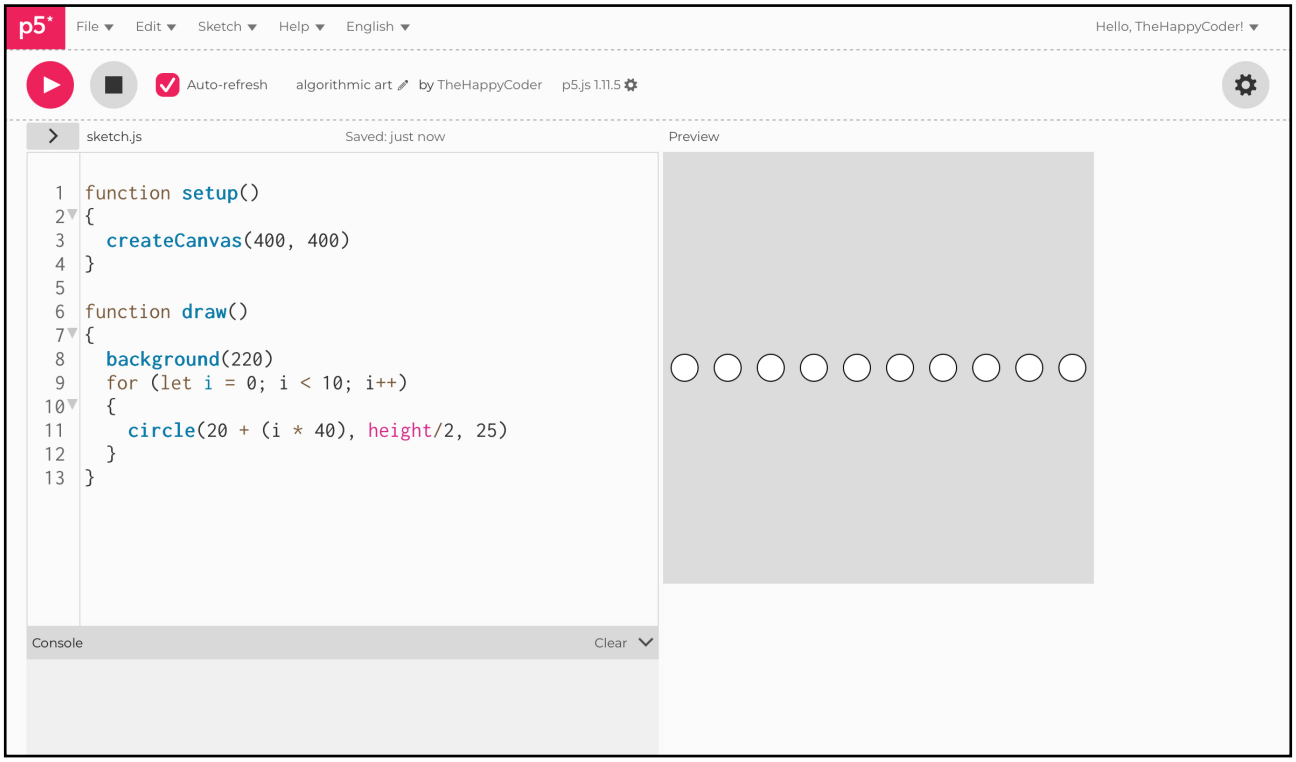
❗ Jumping in with a new sketch
We have drawn ten circles using a `for()` loop, evenly spaced out.

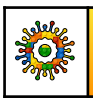
```
function setup()
{
  createCanvas(400, 400)
}

function draw()
{
  background(220)
  for (let i = 0; i < 10; i++)
  {
    circle(20 + (i * 40), height/2, 25)
  }
}
```

🗒️ Notes
We multiply by `40` to get the spacing.

🌻 Challenge
You can have more circles, smaller and closer together if you wish for this exercise.

Figure B6.3


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
1 function setup()
2 {
3 createCanvas(400, 400)
4 }
5
6 function draw()
7 {
8 background(220)
9 for (let i = 0; i < 10; i++)
10 {
11 circle(20 + (i * 40), height/2, 25)
12 }
13 }
Console
Clear



Sketch B6.4 new colour variable

We add a new variable called `newColour` and give it the colour `orange`.

```
let newColour = 'orange'


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


function draw()
{
  background(220)
  for (let i = 0; i < 10; i++)
  {
    fill(newColour)
    circle(20 + (i * 40), height/2, 25)
  }
}
```

🗒️ Notes

A nice row of orange circles. We are using a variable for a good reason, you will see.

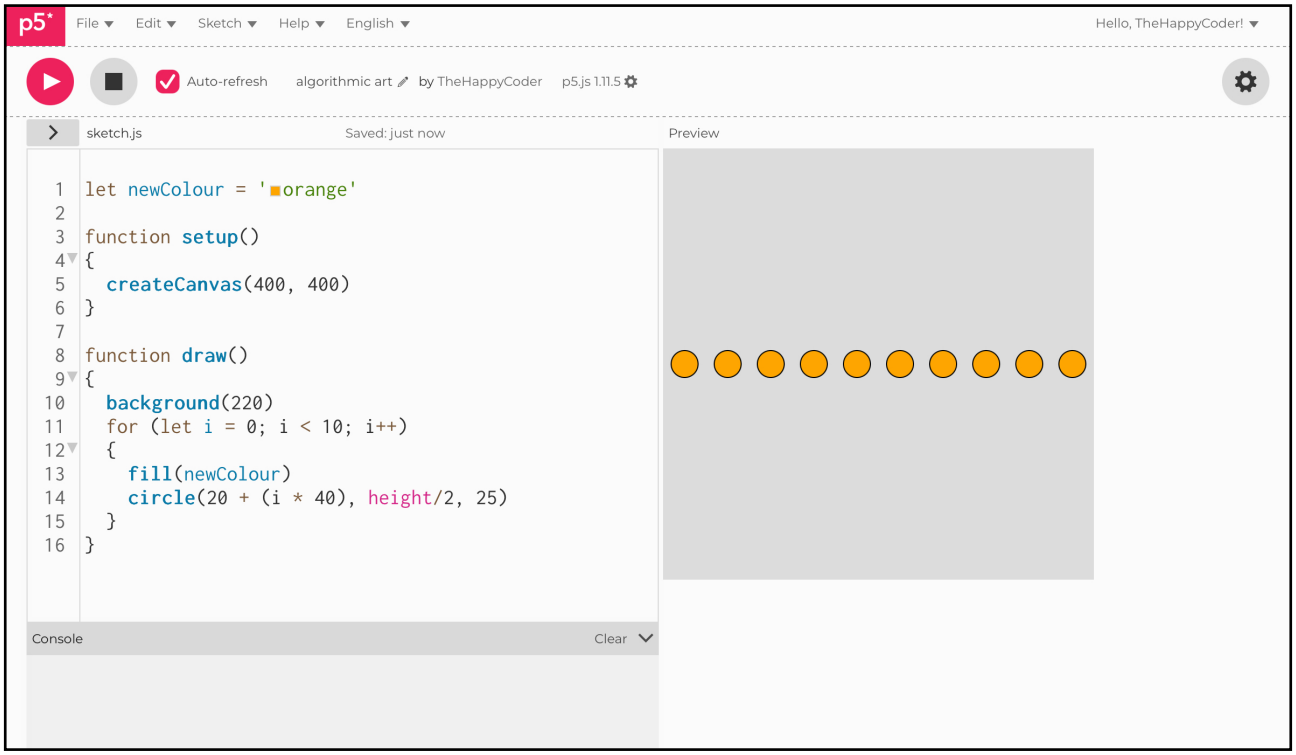
🌻 Challenge

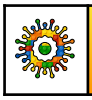
You can choose a different colour if you wish.

🛠️ Code Explanation

`let newColour = 'orange'`	The variable is initiated and defined being given the colour string 'orange'
`fill(newColour)`	Fill the circles with 'orange'
`circle(20 + (i * 40), height/2, 25)`	Circle is spaced out every 40 pixels and starts at 20 from the left, evenly spaced

Figure B6.4


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
let newColour = 'orange'
function setup()
{
createCanvas(400, 400)
}
function draw()
{
background(220)
for (let i = 0; i < 10; i++)
{
fill(newColour)
circle(20 + (i * 40), height/2, 25)
}
}
Console
Clear



Sketch B6.5 the lerp effect

We can interpolate between two colours with the `lerpColor()` function, which takes three arguments:

1. The `first` colour (lightyellow)
2. The `last` colour (darkred)
3. The `incremental` change (increment)

Lerp is short for interpolation.

❗ We add the `noLoop()` otherwise we just get all the circles filled with the final colour.

```
let newColour = 'orange'
let increment = 0

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

function draw()
{
  background(220)
  for (let i = 0; i < 10; i++)
  {
    newColour = lerpColor(color('lightyellow'), color('darkred'), increment)
    fill(newColour)
    circle(20 + (i * 40), height/2, 25)
    increment += 0.1
  }
}
```

🗒️ Notes

This is all very hard-coded. We can work out the increments, that they increase by `0.1` because it has to be between `0` and `1`, and `0.1` is a `tenth` of `1`, we have `10` circles. The next sketch will show a slightly better way with more circles.

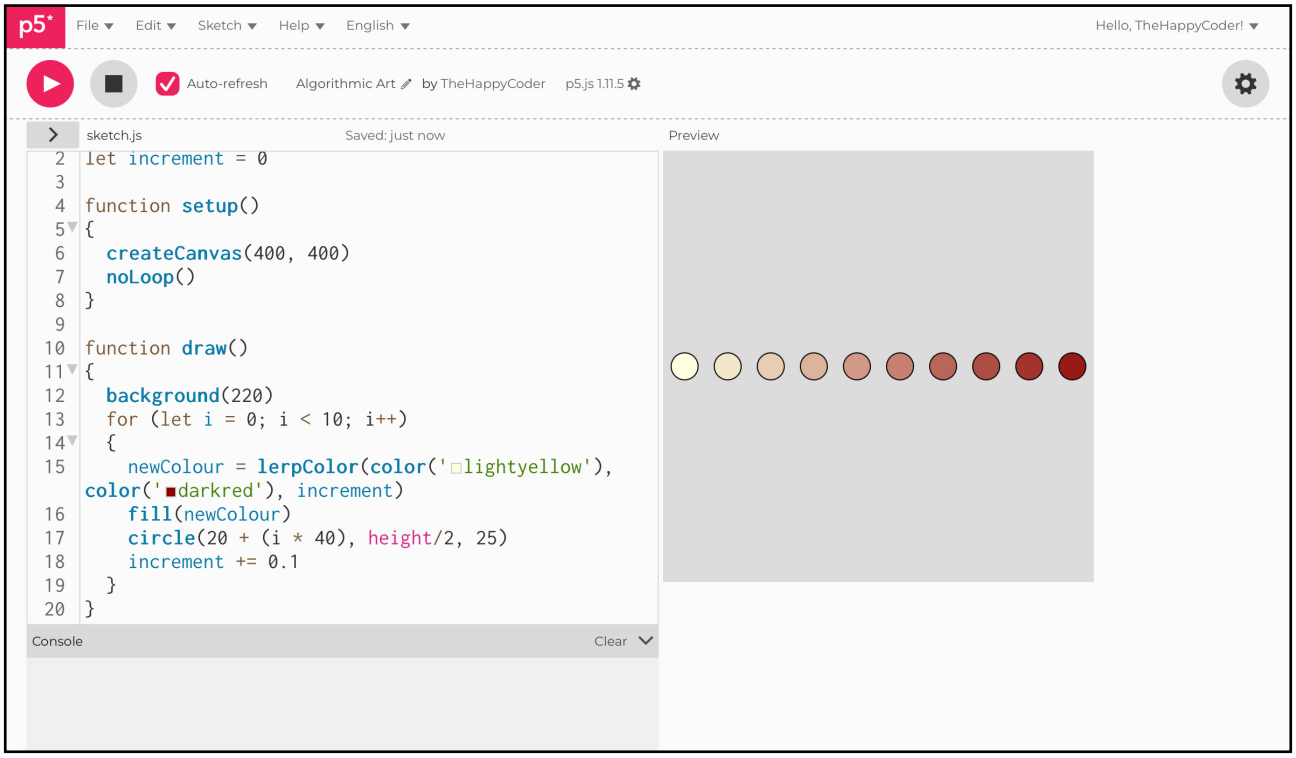
🌻 Challenge

Try combinations of different colours.

🛠️ Code Explanation

`lerpColor(color('lightyellow'), color('darkred'), increment)`	The new colour is something between the lightyellow and the darkred depending on the increment.

Figure B6.5


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
let increment = 0
function setup()
{
createCanvas(400, 400)
noLoop()
}
function draw()
{
background(220)
for (let i = 0; i < 10; i++)
{
newColour = lerpColor(color('lightyellow'), color('darkred'), increment)
fill(newColour)
circle(20 + (i * 40), height/2, 25)
increment += 0.1
}
}
Console
Clear

Sketch B6.6 refining lerp

Using a variable (num) to calculate the incremental steps based on the number of circles.

```
let newColour = 'orange'
let increment = 0
let num = 10

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

function draw()
{
  background(220)
  for (let i = 0; i < num; i++)
  {
    newColour = lerpColor(color('yellow'), color('purple'), increment)
    fill(newColour)
    circle(20 + (i * 40), height/2, 25)
    increment += 1/num
  }
}
```

Notes

Changing the colours as well.

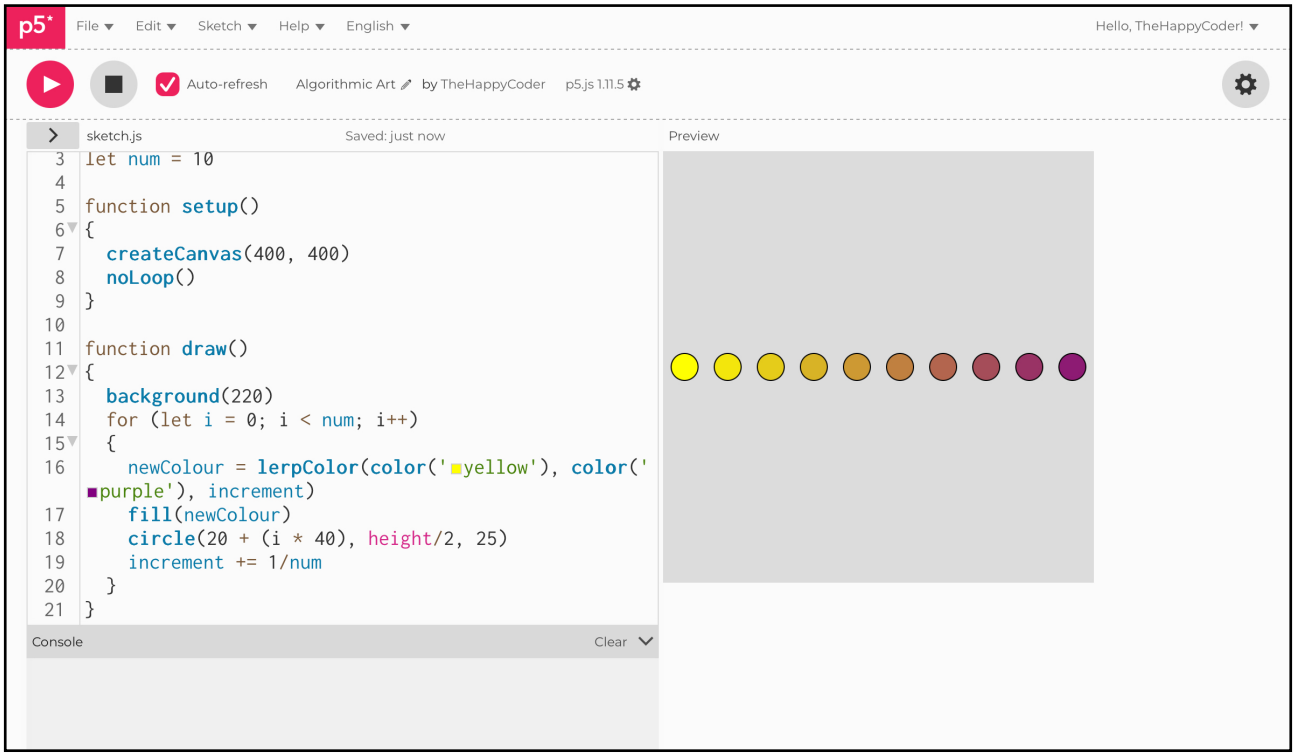
🌻 Challenge

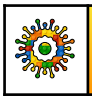
Have 20 or more circles (or other shapes) to see the more gradual change in the migration or interpolation.

🛠️ Code Explanation

`for (let i = 0; i < num; i++)`	Uses this variable to determine how many circles to draw
`increment += 1/num`	Gives you the fraction for interpolation increment

Figure B6.6


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 num = 10
function setup()
{
createCanvas(400, 400)
noLoop()
}
function draw()
{
background(220)
for (let i = 0; i < num; i++)
{
newColour = lerpColor(color('yellow'), color('purple'), increment)
fill(newColour)
circle(20 + (i * 40), height/2, 25)
increment += 1/num
}
}
Console
Clear



Sketch B6.7 the colour c

❗ starting sketch

Creating a variable `c` to hold any new colour. We will be using all three colour modes in this example.

```
let c = 220

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

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

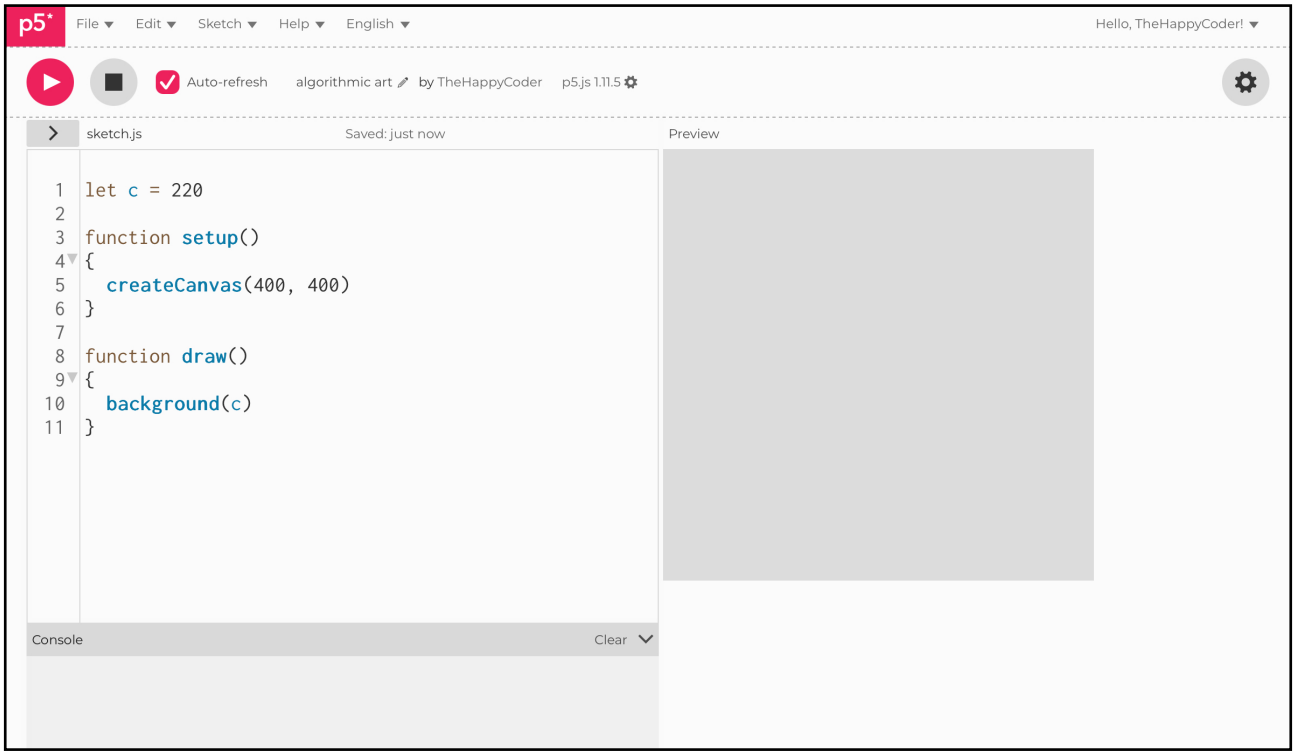
🗒️ Notes

We are just creating a background variable and giving it a grey colour.

🛠️ Code Explanation

`background(c)`	Giving the background the value of the variable c

Figure B6.7


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 c = 220
function setup()
{
createCanvas(400, 400)
}
function draw()
{
background(c)
}
Console
Clear

Sketch B6.8 blue circle

Let us make a nice blue circle using a hex colour value.

```
let c = 220

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

function draw()
{
  background(c)
  fill('#CBF0FF')
  circle(200, 200, 200)
}
```

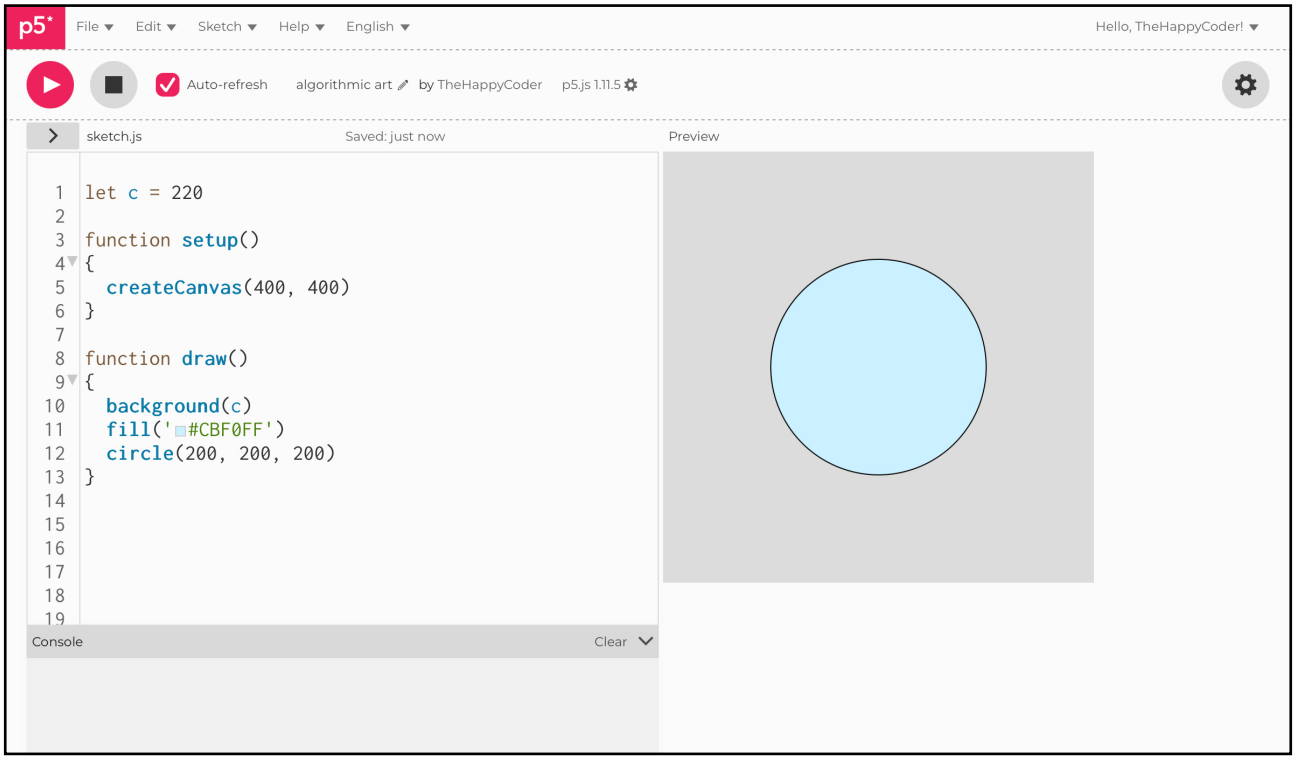
🗒️ Notes

This gives us a colour to compare the background to.

🛠️ Code Explanation

`fill('#CBF0FF')`	This is the hex value for light blue

Figure B6.8


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
let c = 220
function setup()
{
createCanvas(400, 400)
}
function draw()
{
background(c)
fill('#CBF0FF')
circle(200, 200, 200)
}
Console
Clear

Sketch B6.9 some text

We will put the value of the `c` variable (background colour) on the canvas; you will see why shortly.

```
let c = 220

function setup()
{
  createCanvas(400, 400)
  textSize(50)
}

function draw()
{
  background(c)
  fill('#CBF0FF')
  circle(200, 200, 200)
  fill(0)
  stroke(255)
  text(c, 150, 75)
}
```

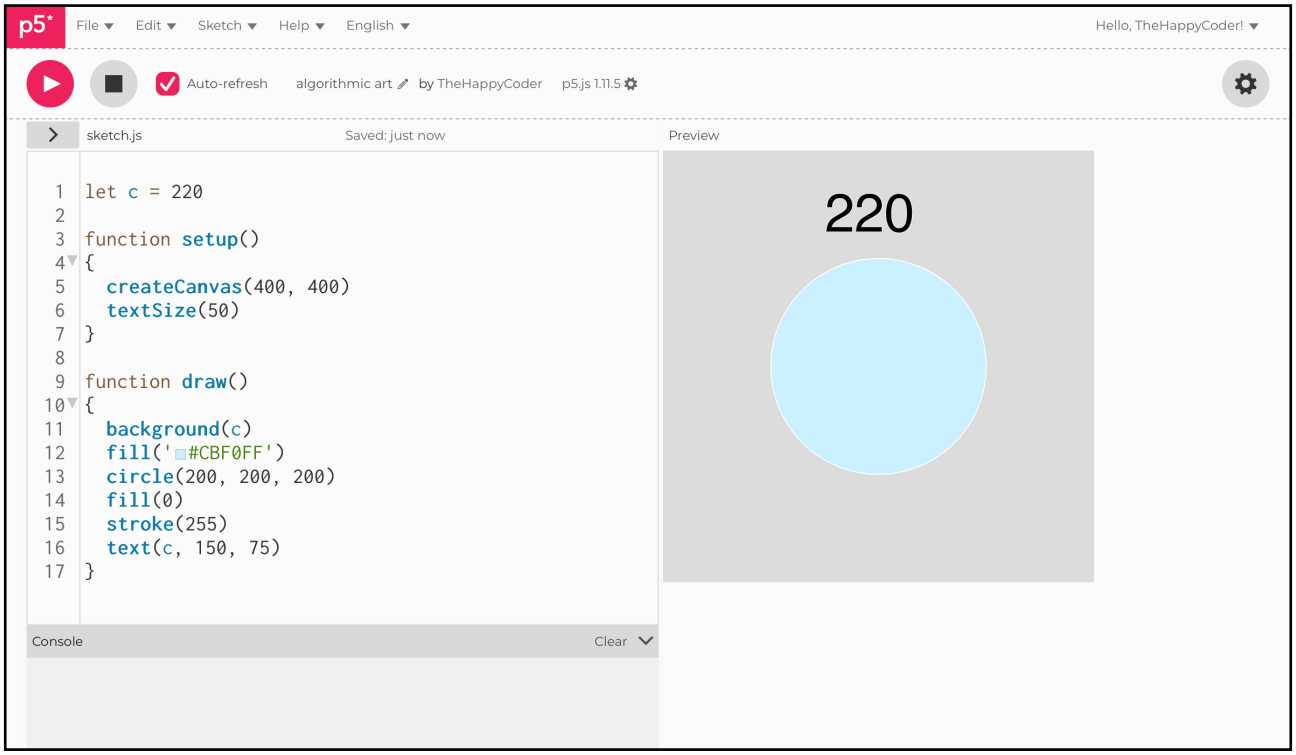
🗒️ Notes

We want to see what the hex value will be; for now, it shows us the value of the background (the `c` value).

🛠️ Code Explanation

`text(c, 150, 75)`	Puts the value of the variable as text on the canvas

Figure B6.9


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
let c = 220
function setup()
{
createCanvas(400, 400)
textSize(50)
}
function draw()
{
background(c)
fill('#CBF0FF')
circle(200, 200, 200)
fill(0)
stroke(255)
text(c, 150, 75)
}
220
Console
Clear

Sketch B6.10 colour picker

There is a function we can use that is built in, which allows us to pick colours. The `colorPicker()` function is now included. This produces a convenient table of useful colour information. You can even copy and paste the hex value you have selected.

```
let c = 220
let myPicker

function setup()
{
  createCanvas(400, 400)
  textSize(50)
  myPicker = createColorPicker('yellow')
}

function draw()
{
  c = myPicker.value()
  background(c)
  fill('#CBF0FF')
  circle(200, 200, 200)
  fill(0)
  stroke(255)
  text(c, 150, 75)
}
```

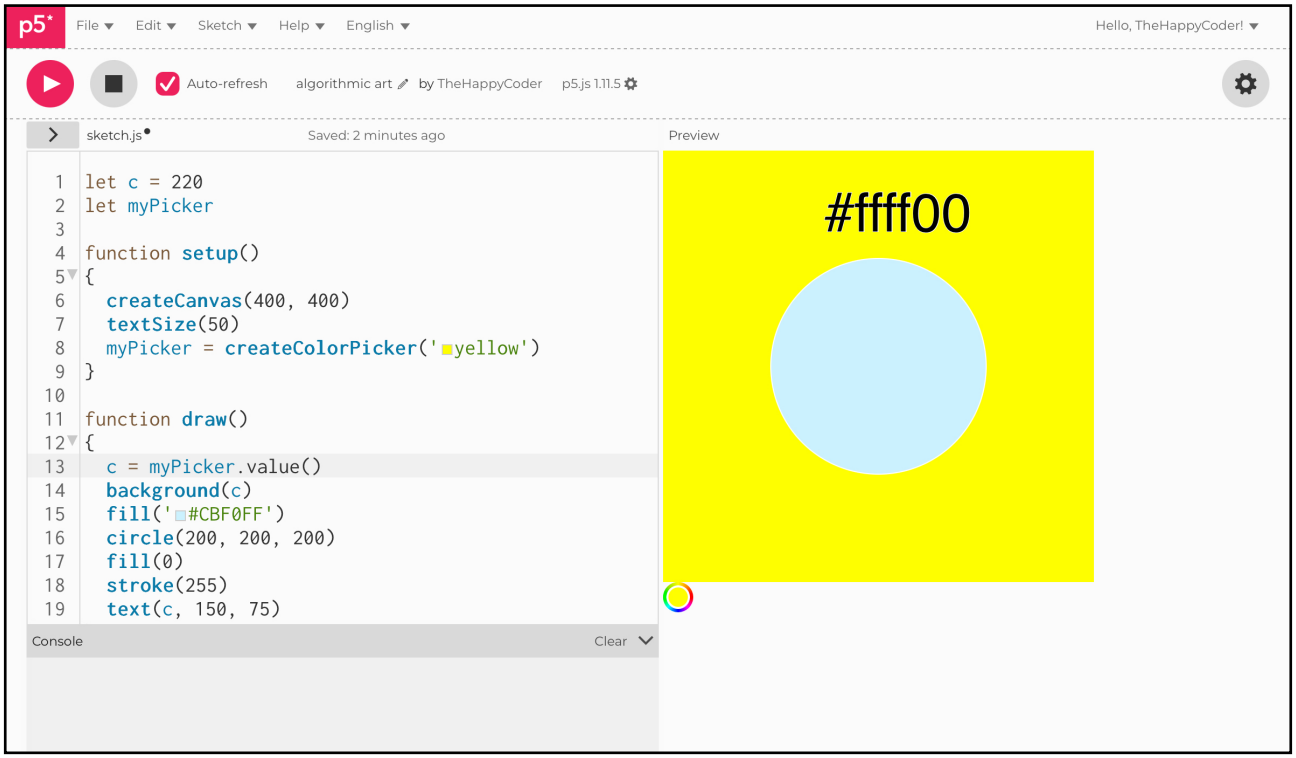
Notes

If you click on the icon at the bottom, you get presented with the option of three windows; each one is a different way to select a colour. The third one gives you the hex value as text, and you can copy that and paste it into your code to change the colour of the circle fill.

🛠️ Code Explanation

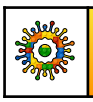
`createColorPicker('yellow')`	We give the colorPicker() function an initial colour (yellow)
`c = myPicker.value()`	The variable c now pulls the hex value from the colour
`text(c, 150, 75)`	This value is now displayed on the canvas

Figure B6.10



Notes

If in the future you are using version `2.x` of `p5.js` the colour picker will look a bit different.



Sketch B6.11 a bit of a tweak

We will add a `noStroke()` and move the colour picker. If you slide the canvas to the left, then you aren't obscuring the canvas when you click on the icon.

```
let c = 220
let myPicker

function setup()
{
  createCanvas(400, 400)
  textSize(50)
  myPicker = createColorPicker('yellow')
  myPicker.position(width, height)
}

function draw()
{
  c = myPicker.value()
  background(c)
  noStroke()
  fill('#CBF0FF')
  circle(200, 200, 200)
  fill(0)
  stroke(255)
  text(c, 150, 75)
}
```

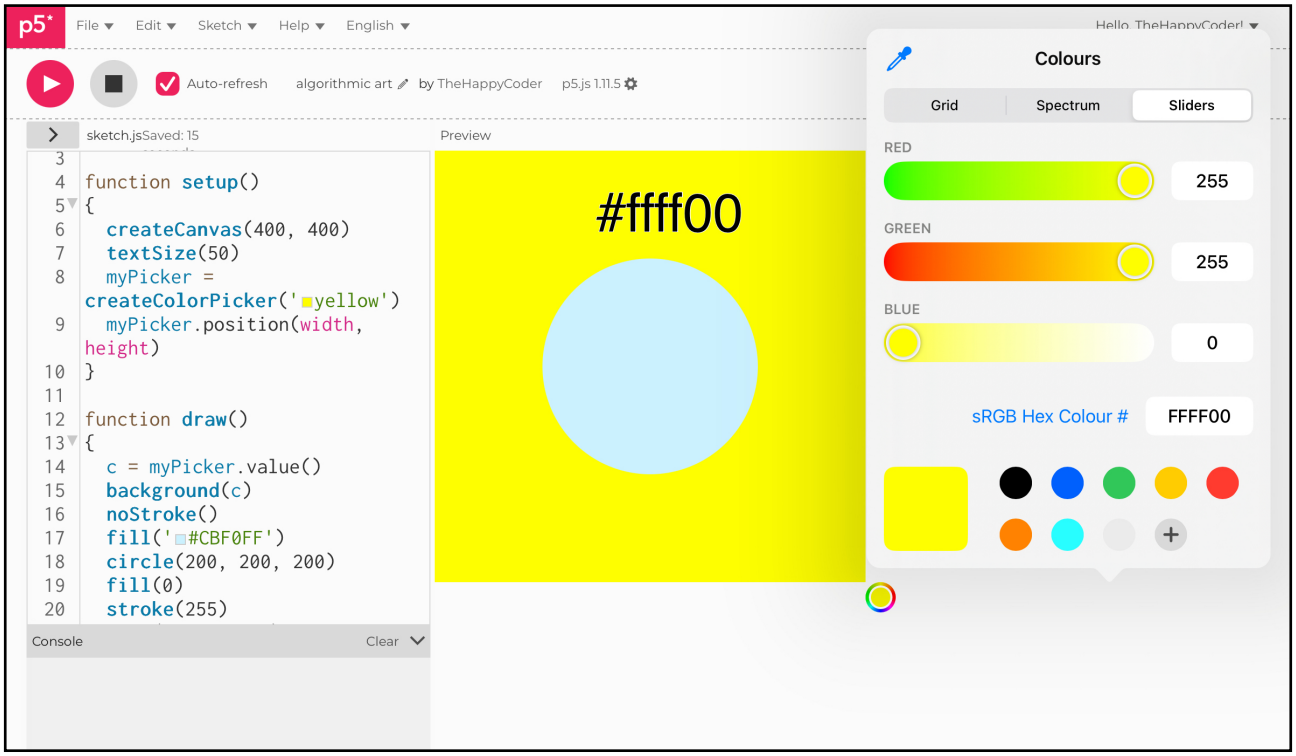
Notes

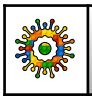
Works for me.

🛠️ Code Explanation

`myPicker.position(width, height)`	How to position the colour picker icon

Figure B6.11


p5*
File
Edit
Sketch
Help
English
Hello, TheHappyCoder!
Auto-refresh
algorithmic art
by TheHappyCoder
p5.js 1.11.5
sketch.jsSaved: 15
Preview
3
4 function setup()
5 {
6 createCanvas(400, 400)
7 textSize(50)
8 myPicker =
createColorPicker('yellow')
9 myPicker.position(width,
height)
10 }
11
12 function draw()
13 {
14 c = myPicker.value()
15 background(c)
16 noStroke()
17 fill('#CBF0FF')
18 circle(200, 200, 200)
19 fill(0)
20 stroke(255)
Console
Clear
#ffff00
Colours
Grid
Spectrum
Sliders
RED
255
GREEN
255
BLUE
0
sRGB Hex Colour #
FFFF00



Introduction to the colour charts

All the colour names, their HEX and RGB values. The groupings of colours are: Pink, Purple, Yellow, Red, Orange, Green, Blue, Brown, Cyan, Grey, and White.

Pink Colours

Pink	FFC0CB	255,192,203
LightPink	FFB6C1	255,182,193
HotPink	FF69B4	255,105,180
DeepPink	FF1493	255,20,147
PaleVioletRed	DB7093	219,112,147
MediumVioletRed	C71585	199,21,133

Purple Colours

Lavender	E6E6FA	230,230,250
Thistle	D8BFD8	216,191,216
Plum	DDA0DD	221,160,221
Orchid	DA70D6	218,112,214
Violet	EE82EE	238,130,238
Fuchsia	FF00FF	255,0,255
Magenta	FF00FF	255,0,255
MediumOrchid	BA55D3	186,85,211
DarkOrchid	9932CC	153,50,204
DarkViolet	9400D3	148,0,211
BlueViolet	8A2BE2	138,43,226
DarkMagenta	8B008B	139,0,139
Purple	800080	128,0,128
MediumPurple	9370DB	147,112,219
MediumSlateBlue	7B68EE	123,104,238
SlateBlue	6A5ACD	106,90,205
DarkSlateBlue	483D8B	72,61,139
RebeccaPurple	663399	102,51,153
Indigo	4B0082	75,0,130

Yellow Colours

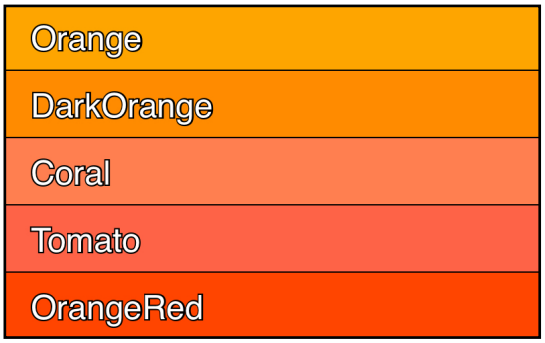
Gold	FFD700	255,215,0
Yellow	FFFF00	255,255,0
LightYellow	FFFFE0	255,255,224
LemonChiffon	FFFACD	255,250,205
LightGoldenRodYellow	FAFAD2	250,250,210
PapayaWhip	FFEFD5	255,239,213
Moccasin	FFE4B5	255,228,181
PeachPuff	FFDAB9	255,218,185
PaleGoldenRod	EEE8AA	238,232,170
Khaki	F0E68C	240,230,140
DarkKhaki	BDB76B	189,183,107

Red Colours

LightSalmon	FFA07A	255,160,122
Salmon	FA8072	250,128,114
DarkSalmon	E9967A	233,150,122
LightCoral	F08080	240,128,128
IndianRed	CD5C5C	205,92,92
Crimson	DC143C	220,20,60
Red	FF0000	255,0,0
FireBrick	B22222	178,34,34
DarkRed	8B0000	139,0,0

Orange Colours

Orange	FFA500	255,165,0
DarkOrange	FF8C00	255,140,0
Coral	FF7F50	255,127,80
Tomato	FF6347	255,99,71
OrangeRed	FF4500	255,69,0



Green Colours

GreenYellow	ADFF2F	173,255,47
Chartreuse	7FFF00	127,255,0
LawnGreen	7CFC00	124,252,0
Lime	00FF00	0,255,0
LimeGreen	32CD32	50,205,50
PaleGreen	98FB98	152,251,152
LightGreen	90EE90	144,238,144
MediumSpringGreen	00FA9A	0,250,154
SpringGreen	00FF7F	0,255,127
MediumSeaGreen	3CB371	60,179,113
SeaGreen	2E8B57	46,139,87
ForestGreen	228B22	34,139,34
Green	008000	0,128,0
DarkGreen	006400	0,100,0
YellowGreen	9ACD32	154,205,50
OliveDrab	6B8E23	107,142,35
DarkOliveGreen	556B2F	85,107,47
MediumAquaMarine	66CDAA	102,205,170
DarkSeaGreen	8FBC8F	143,188,143
LightSeaGreen	20B2AA	32,178,170
DarkCyan	008B8B	0,139,139
Teal	008080	0,128,128

Blue Colours

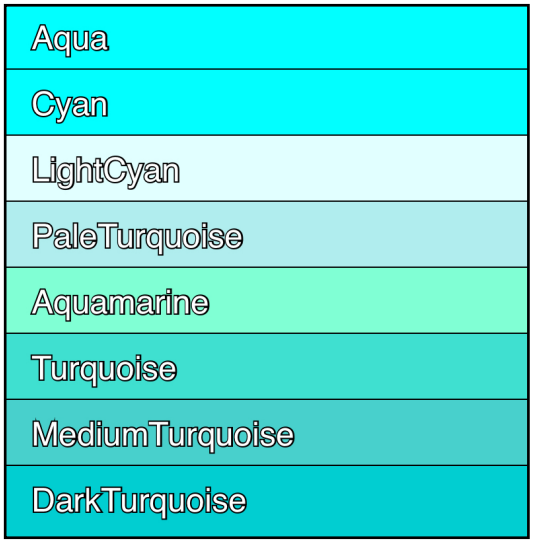
CadetBlue	5F9EA0	95,158,160
SteelBlue	4682B4	70,130,180
LightSteelBlue	B0C4DE	176,196,222
LightBlue	ADD8E6	173,216,230
PowderBlue	B0E0E6	176,224,230
LightSkyBlue	87CEFA1	35,206,250
SkyBlue	87CEEB	135,206,235
CornflowerBlue	6495ED	100,149,237
DeepSkyBlue	00BFFF	0,191,255
DodgerBlue	1E90FF	30,144,255
RoyalBlue	4169E1	65,105,225
Blue	0000FF	0,0,255
MediumBlue	0000CD	0,0,205
DarkBlue	00008B	0,0,139
Navy	000080	0,0,128
MidnightBlue	191970	25,25,112

Brown Colours

Cornsilk	FFF8DC	255,248,220
BlanchedAlmond	FFEBCD	255,235,205
Bisque	FFE4C4	255,228,196
NavajoWhite	FFDEAD	255,222,173
Wheat	F5DEB3	245,222,179
BurlyWood	DEB887	222,184,135
Tan	D2B48C	210,180,140
RosyBrown	BC8F8F	188,143,143
SandyBrown	F4A460	244,164,96
GoldenRod	DAA520	218,165,32
DarkGoldenRod	B8860B	184,134,11
Peru	CD853F	205,133,63
Chocolate	D2691E	210,105,30
Olive	808000	128,128,0
SaddleBrown	8B4513	139,69,19
Sienna	A0522D	160,82,45
Brown	A52A2A	165,42,42
Maroon	800000	128,0,0

Cyan Colours

Aqua	00FFFF	0,255,255
Cyan	00FFFF	0,255,255
LightCyan	E0FFFF2	24,255,255
PaleTurquoise	AFEEEE	175,238,238
Aquamarine	7FFFD4	127,255,212
Turquoise	40E0D0	64,224,208
MediumTurquoise	48D1CC	72,209,204
DarkTurquoise	00CED1	0,206,209



Grey Colours

Gainsboro	DCDCDC	220,220,220
LightGrey	D3D3D3	211,211,211
Silver	C0C0C0	192,192,192
DarkGrey	A9A9A9	169,169,169
DimGrey	696969	105,105,105
Grey	808080	128,128,128
LightSlateGrey	778899	119,136,153
SlateGrey	708090	112,128,144
DarkSlateGrey	2F4F4F	47,79,79
Black	000000	0,0,0

White Colours

White	FFFFFF	255,255,255
Snow	FFFAFA	255,250,250
HoneyDew	F0FFF0	240,255,240
MintCream	F5FFFA	245,255,250
Azure	F0FFFF	240,255,255
AliceBlue	F0F8FF	240,248,255
GhostWhite	F8F8FF	248,248,255
WhiteSmoke	F5F5F5	245,245,245
SeaShell	FFF5EE	255,245,238
Beige	F5F5DC	245,245,220
OldLace	FDF5E6	253,245,230
FloralWhite	FFFAF0	255,250,240
Ivory	FFFFF0	255,255,240
AntiqueWhite	FAEBD7	250,235,215
Linen	FAF0E6	250,240,230
LavenderBlush	FFF0F5	255,240,245
MistyRose	FFE4E1	255,228,225

White
Snow
HoneyDew
MintCream
Azure
AliceBlue
GhostWhite
WhiteSmoke
SeaShell
Beige
OldLace
FloralWhite
Ivory
AntiqueWhite
Linen
LavenderBlush
MistyRose