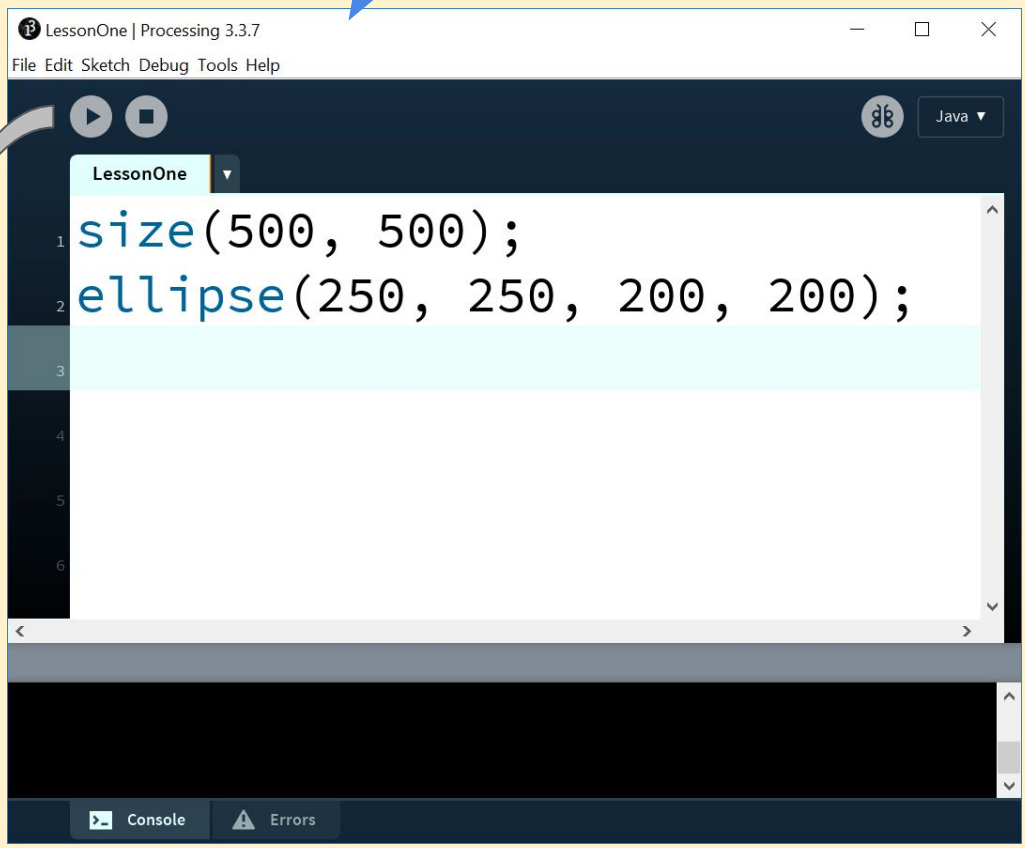
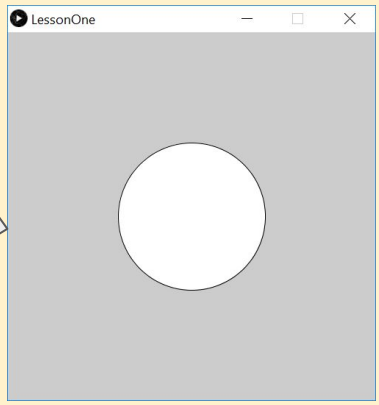
[the academy_of_code]

St. Fiachra’s Tech Club

1

Write the two lines of code below in the **text editor** then click on the ***run button*** ***(or try Ctrl + r)***





You should get a grey box with a white circle appearing on your screen, like the one to the left

Shortcut

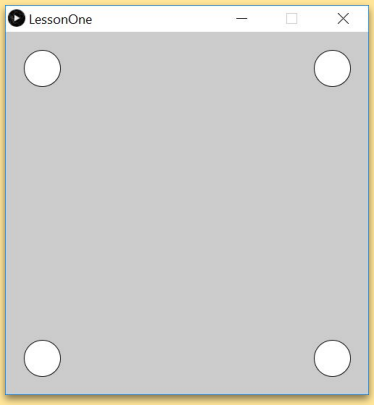
Run
Ctrl + R

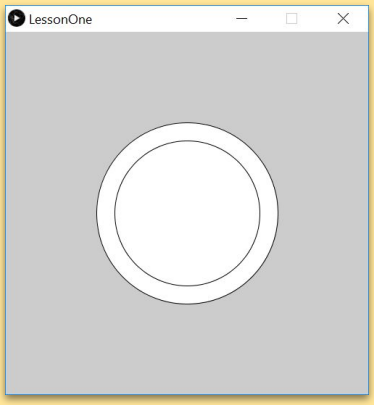
Lets get Coding!

Use the **ellipse() function** to complete the following tasks. If you are confused, ask a tutor.

Challenge 1

Draw a circle in each corner of the display window





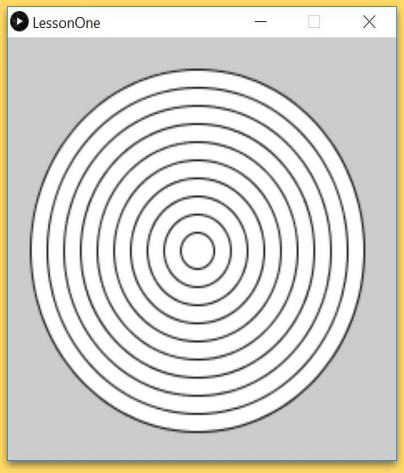
Challenge 2

Draw a circle inside a larger circle.
Hint: They'll have the same centres

Challenge 3

Building on from the last target, make a bigger **canvas** (window) that will take many circles from big to small. It'll look something like this. **Make sure to save your work.** We'll be doing something similar to this when we start learning about colour. Try following a pattern where your circle's diameter gets less making a number pattern (e.g. 300, 280, 260, 240 etc).

Remember: Only the size parameters will change, the position won't.

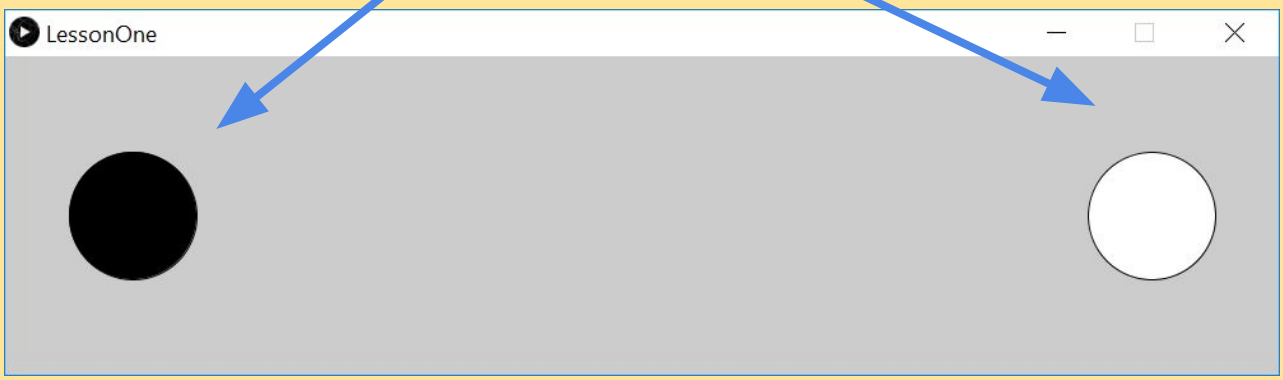


Numbers in fill():

In processing and other coding languages, colours go from **0** to **255**

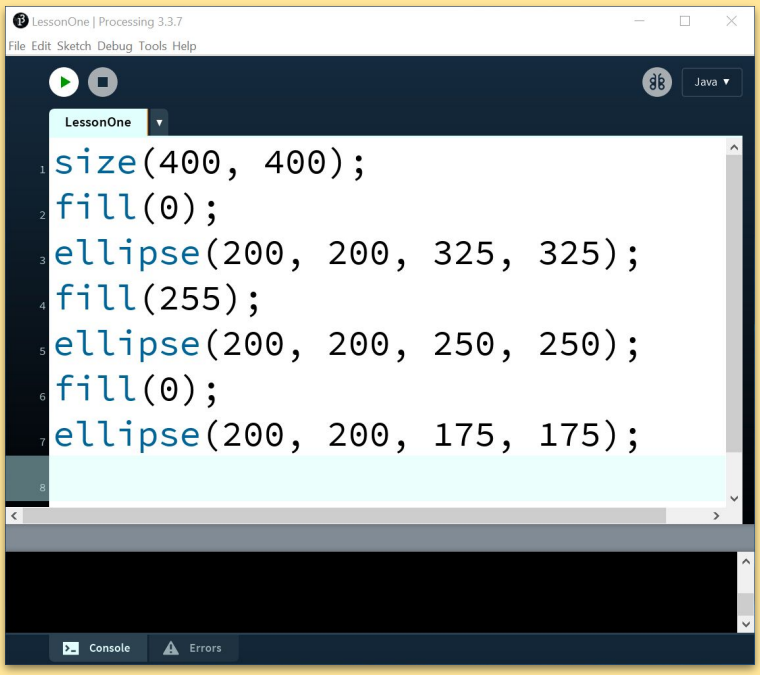
When dealing with greyscale

- **0 is black (no white):** **fill(0);**
- **255 is white (no black):** **fill(255);**



Lets Get Coding

We're going to look at making grayscale ellipses using **fill();**

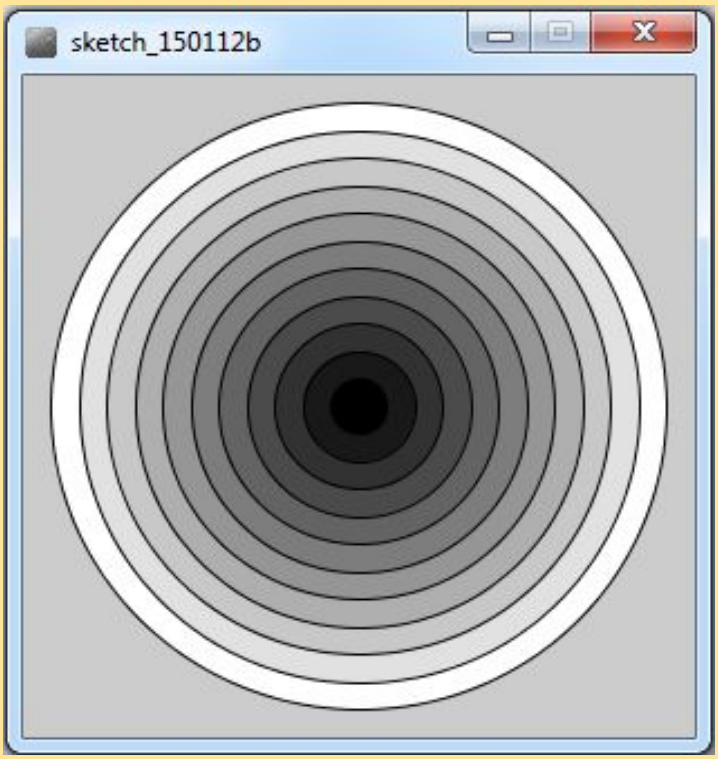


You should get this sketch when it runs

Now make this into a target

Try following a pattern where your circle's diameter gets less making a number pattern (e.g. 300, 280, 260, 240 etc). **Remember:** Only the size parameters will change, the position won't.

Use **fill()** to make a series of circles that fade from white on the outside to black in the middle. You want to make something like the Display Window below.



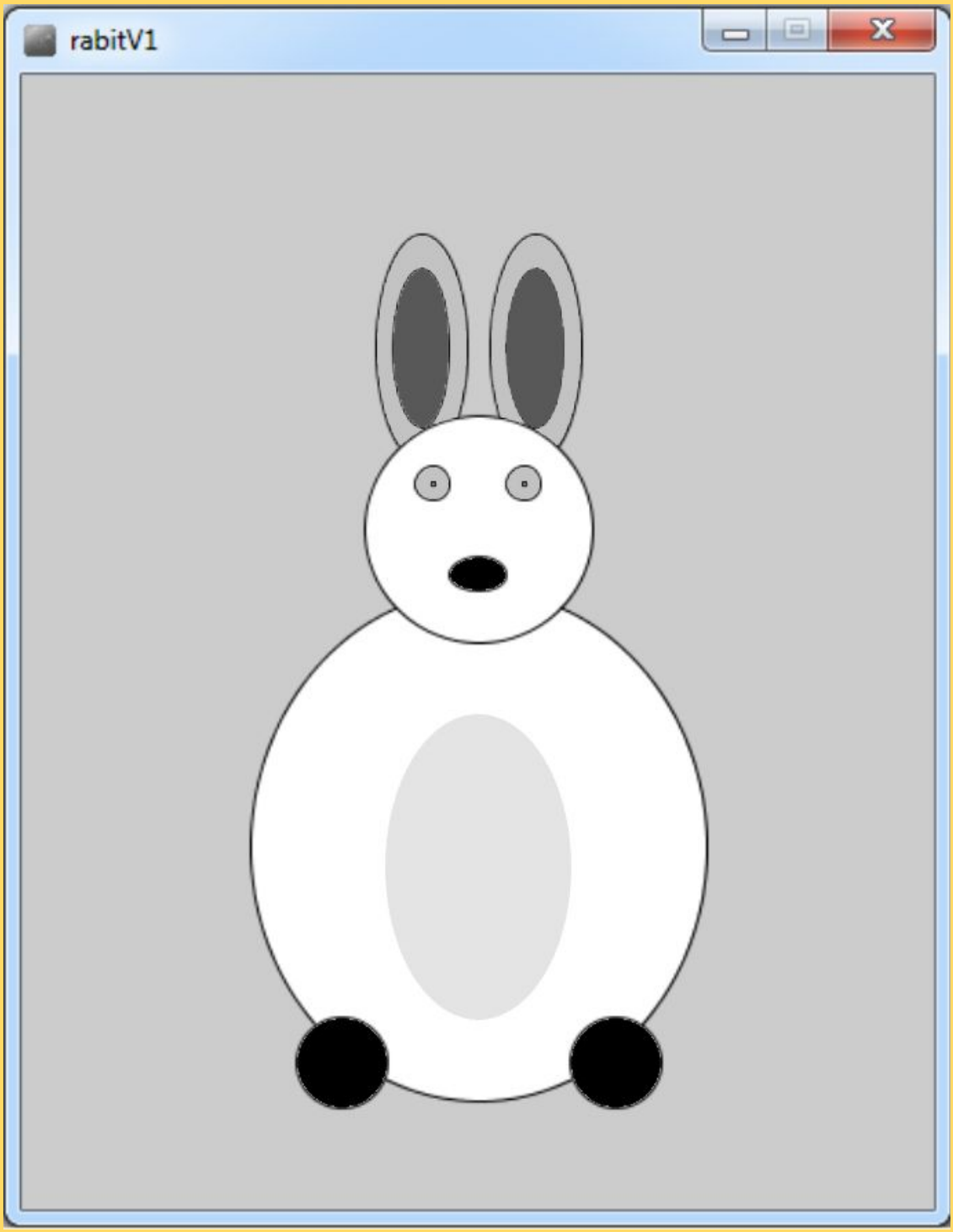
Hint: Each circle is going to have its own fill. This should get less in a pattern. (e.g. 255, 235, 215, 195, 175.. etc)

At this point, you should know how to:

- design **Display Windows** for our sketches using **size();**
- made ovals and circles appear using **ellipse();**
  - Some of you may have tried the rect() function
- used **fill();** to shade ellipses from black through grey to white

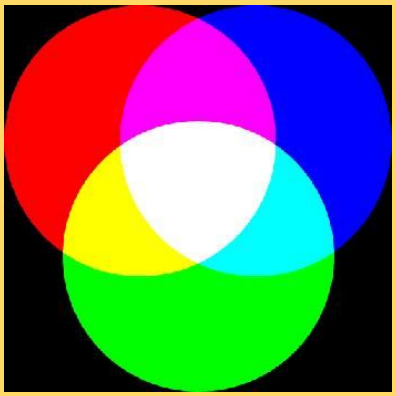
Making a Bunny - The Advanced Challenge

Here's one you might need to try at home, it should take a while but is great practice. Try making a bunny (like the one below) and colour it in in greyscale using fill(). Save it as “**ProcessingBunny**”



The RBG Colour space or RGB colour system

RGB Colours = **RED**,**GREEN**,**BLUE** colours. All colours on a computer's monitor/screen are made up by combining light from three colours.



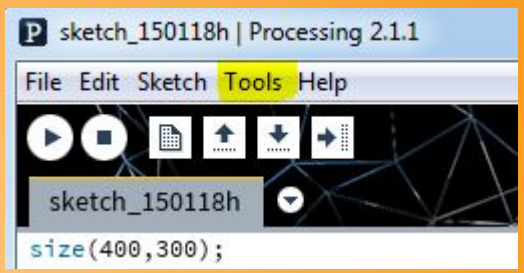
RED,**GREEN**,**BLUE**

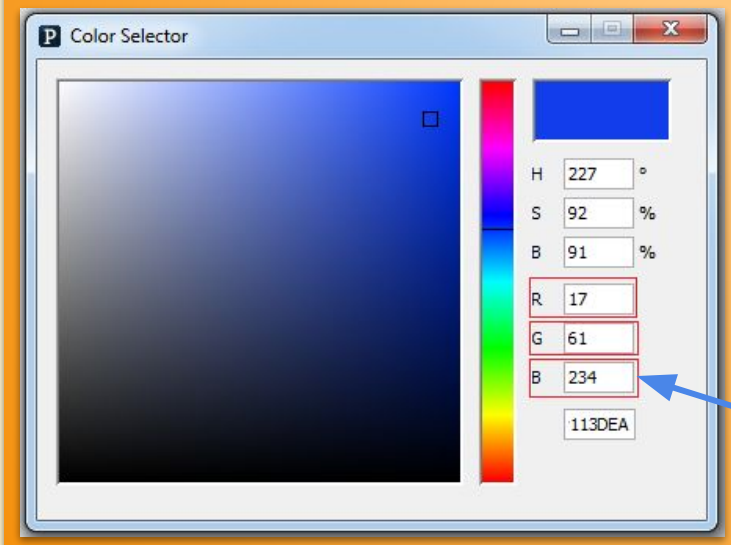
Black is **(0,0,0);** white is **(255,255,255);** and grey is any three colours between 0 and 255 where all three numbers are the same.

Expert Tip - Need a specific colour?

Processing can help us find whatever colour you want using the **Color Selector**. Here's how to get to it:

- Select "**Tools**" in the Toolbar at the top
- Then select "**Color Selector**"



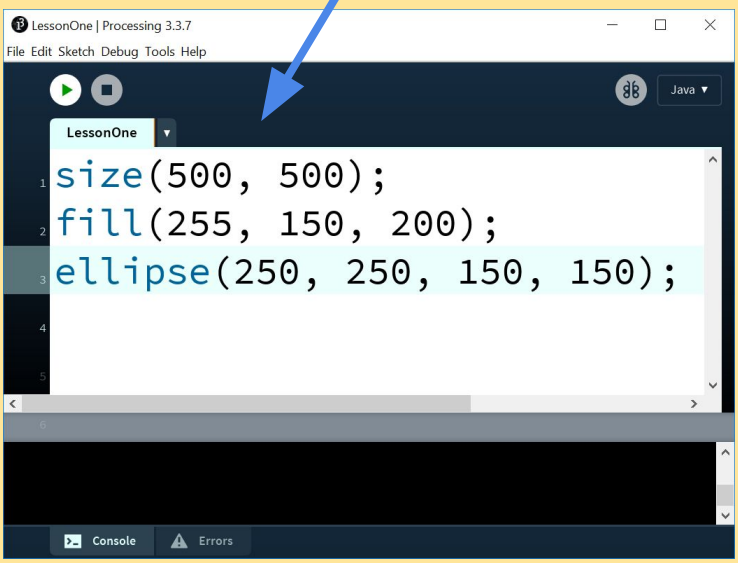


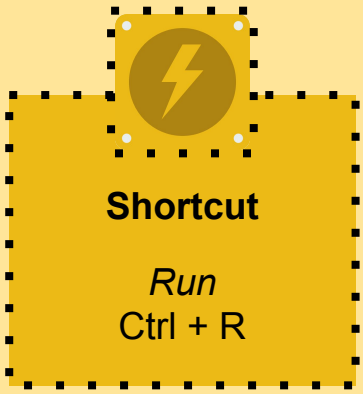
This is the colour selector, click on the colour you want and copy the **R,G** and **B** values and copy them into the `fill()` function.

For this colour we'd take these three numbers and write: `fill`**(17,61,234);**

Multicolour Sketches

Write the code below in the **text editor**
then click on the ***run button***
(or try Ctrl + r)

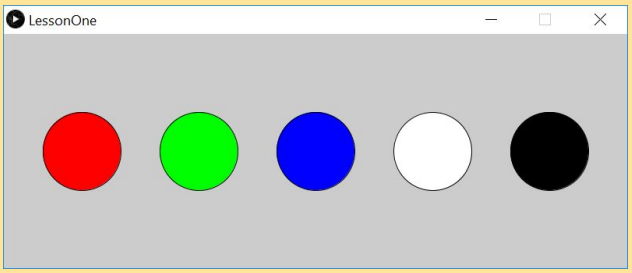




What colour does this make?

Thinking back to your previous lessons, see if you can make whats in this display window, five circles with each of these colours:

- Red
- Green
- Blue
- Black
- White

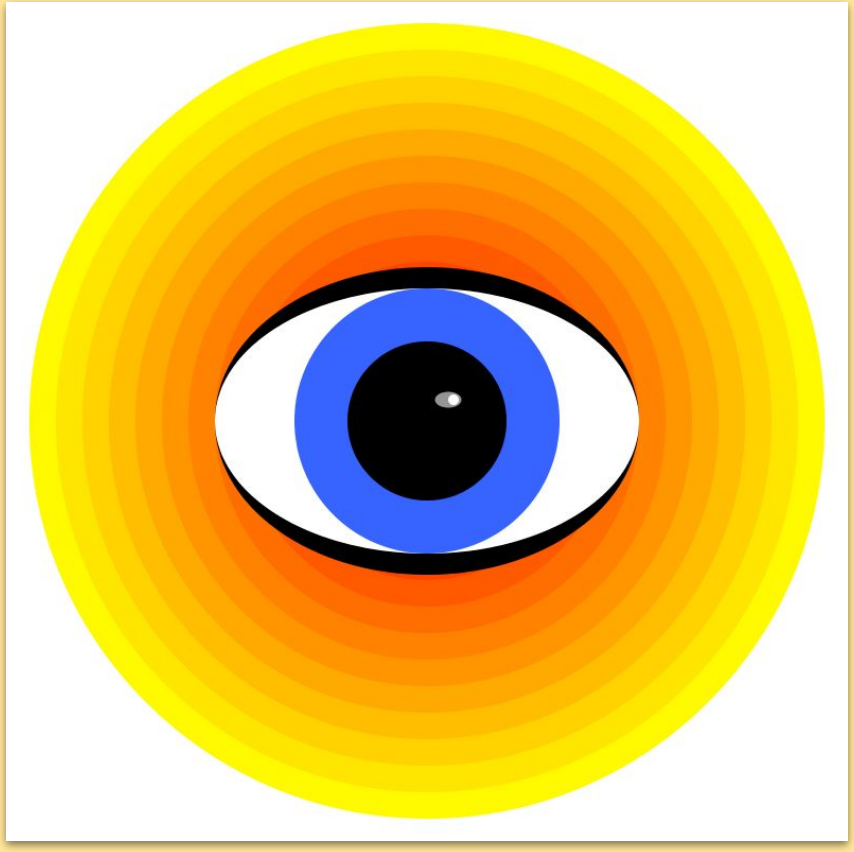


Try adding extra circles in of each of these colours

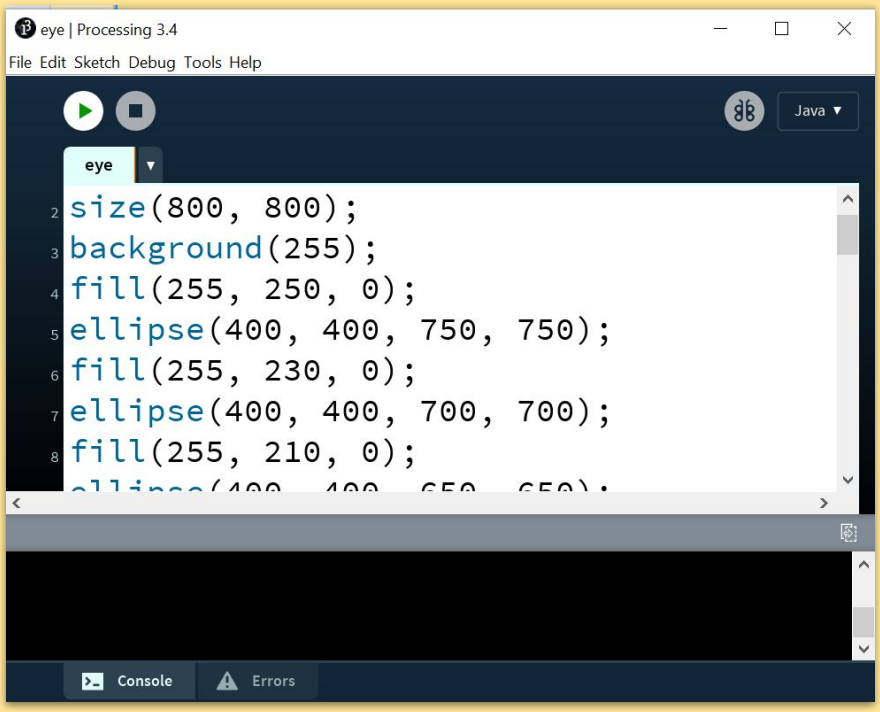
- Yellow (a combination of red and green)
- Magenta (a combination of red and blue)
- Cyan (a combination of green and blue)

Eye Challenge

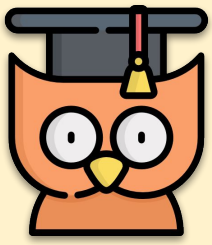
Imagine an indie/electronic band have asked you to design cover art for a release on Spotify. Using *Processing* remake the idea below



Here's some code to get you started.

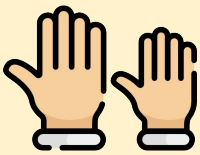


Lesson 2 - User Input



Learning Outcomes:

- We will learn how to use **mouseX** and **mouseY** to programme user-controlled mouse input.
- We will learn about **void setup()** and **void draw()**



REMEMBER: Put up your hand and stay in your seat. We love to help!

First - create a new file called **LessonTwoUserInput**

Thinking Time

Before we go on let's see if You can remember what all of these words and numbers mean.

```
size(200,200);

ellipse(100,100,150,150);
```

Testing setup() and draw()

So far we've been working with images that, when run is pressed, a single image appears. In the real world we need interactive apps and software. This is where **setup()** and **draw()** come in.

Let's give it a go. Type the following code:

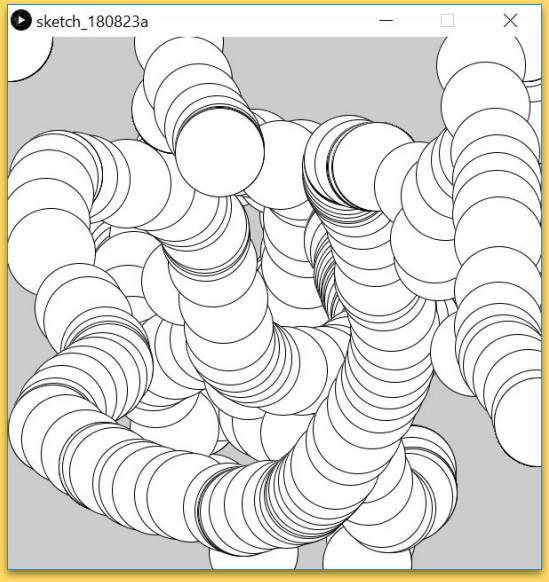
```
void setup(){
 size(600,600);
}

void draw(){
 ellipse(mouseX, mouseY, 100, 100);
}
```

What is setup() and draw()

```
void setup(){
 size(600,600);
}

void draw(){
 ellipse(mouseX, mouseY, 100, 100);
}
```



This code gave us an **interactive** window. It's not the kind of application we could make much money from, but having **User input** makes it interactive, it's fun to mess around with!

void setup() and **void draw()** are what we call **code blocks**. They open and close with **curly brackets**. That is the actual technical definition for them!

Expert Tip

As soon as you use an opening bracket eg **"("**, **"{"** or **"["** close it straight away with its pair **")"**, **"}"** or **"]"**. Not doing this is the source of most problems for beginners.

- Whatever is inside **setup()** will run once when run is clicked.
- Whatever is inside **draw()** will keep running and running over and over again (60 times per second) until you close the application. This is why you made so many circles.

Tasks

Have a look at the code we wrote (above)

1. Replace **mouseX** with a number (say, 200). What happens now? Why?

2. Put **mouseX** back where it was, and now replace **mouseY** with a number. What happens? Is it different to what happened in Task 1?

3

Just for fun, switch `mouseX` and `mouseY`

eg `ellipse(mouseY, mouseX, 100, 100);`

What happens now? Why?

4

Now try having `mouseX` twice:

eg `ellipse(mouseX, mouseX, 100, 100);`

How does the circle move? Why? Try with two `mouseY`'s.

5

Try some new random variations yourself, for example:

```
ellipse(mouseX, mouseY, mouseX, mouseY);
ellipse(100, 100, mouseX, mouseY);
ellipse(mouseY, mouseX, mouseY, mouseX);
ellipse(mouseX, 100, 100, mouseY);
```

Etc.

6

Your circles leave a trail, a mark where they've been.

Change the background colour by insert **fill()** and **background()** into your code. See what happens when you put it before and after the text.

7

Use the rect() function that to write a code that has a moving coloured ellipse, that doesn't leave a trail, that has a different background colour to the rect.

Challenge 1

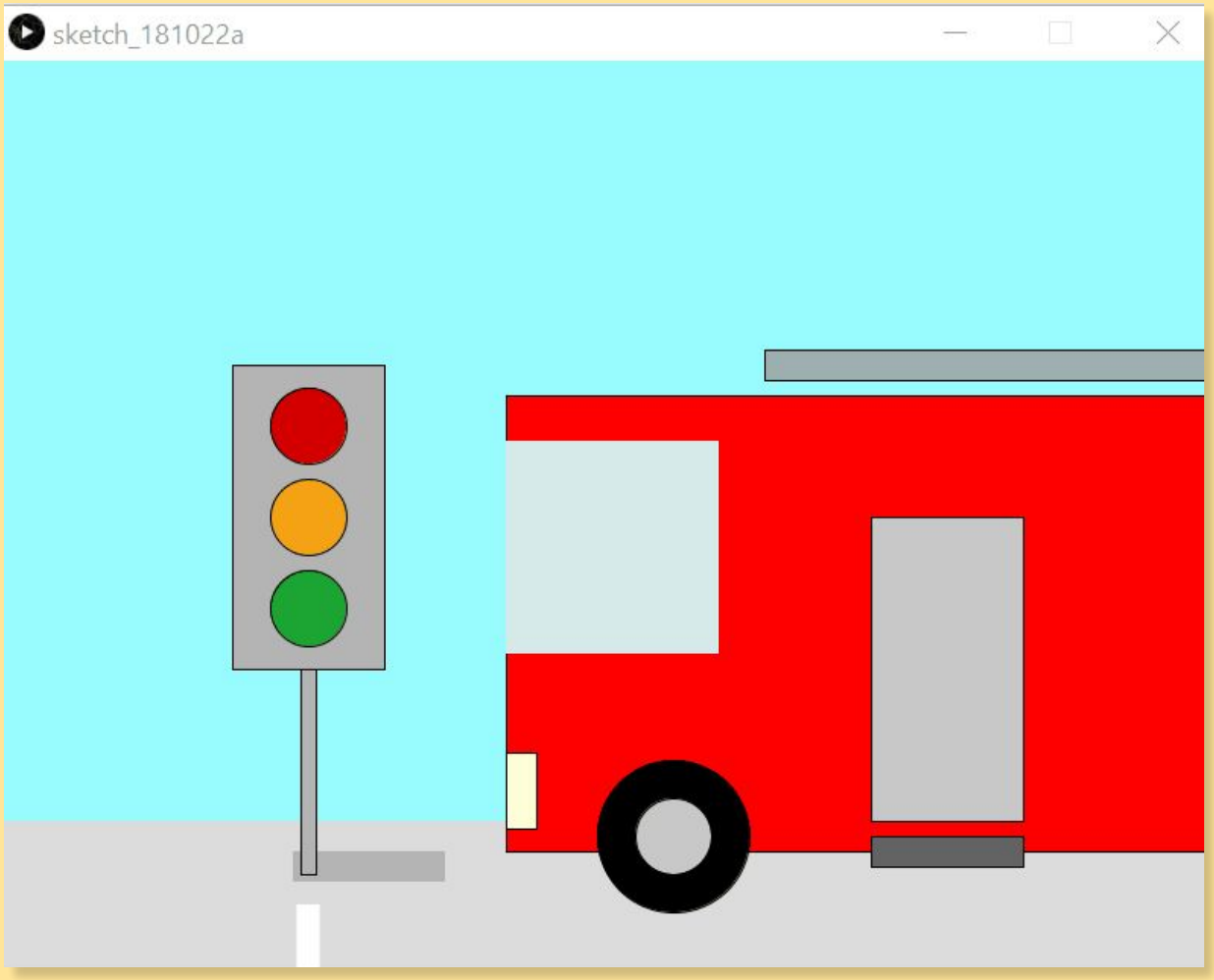
Write a code which features more than four shapes that move in different ways.

Challenge 2

Copy the code of a target, face or bunny that you made before and make this move around the screen!

Eye Challenge

Using *Processing* make a traffic scene like the one below:



Here's some code to get you started.

File Edit Sketch Debug Tools Help

Java

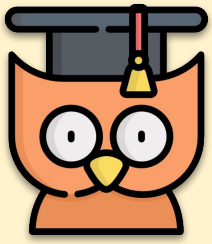
eye

```
void setup() {
  size (800, 600);
}

void draw() {
  background(0, 255, 255); //blue
  rectMode(CENTER);
  noStroke();
  fill(255, random(150, 200), 0); //sun colour
  ellipse(mouseX, mouseY, 100, 100); //sun
  fill(200); //road colour
  rect(400, 550, 800, 150); //road
  fill(255, 0, 0);
  rect(600, 350, 400, 300);
}
```

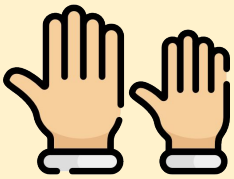
Console Errors

Lesson 3 - HTML/CSS - Making a Story



Learning Outcomes

- How to create CSS within HTML Pages
- Creating a comic Strip with HTML and CSS



REMEMBER: Put up your hand. We love to help!

Introduction

In this project, you'll learn how to create your own webpage to tell a story, joke or poem.

Decide on a story

Before you get coding, you'll need to decide on a story to tell. Think about the story/joke that you want to tell. It could be a famous story, a story you have made up, something that has happened to you or someone you know.

It doesn't even have to be a story. It could be a joke, a poem, or anything else you want. But make sure you have a plan.

Editing your story

Let's start by editing the HTML content and CSS style of the story webpage.

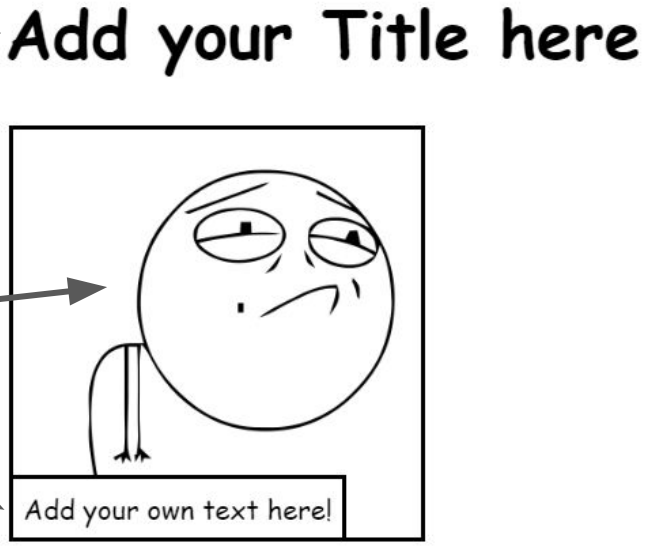
Open this trinket: www.bit.ly/trinketcomic.

Let's Get Coding

Go to line 7 of the code, and you will see the webpage content, which is inside **`<body>`** and **`</body>`** tags. *Can you tell which tags are used to create the different parts of the webpage?*

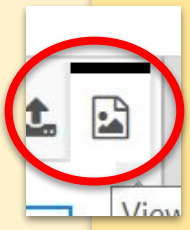
```
7   <body>
8
9     <h1>Add your title here!</h1>
10
11    <div>
12      <img src="sun.png">
13      <p>Add your own text here!</p>
14    </div>
15
16  </body>
```

```
1  <html>
2
3    <head>
4      <link rel="stylesheet" href="style.css">
5    </head>
6
7    <body>
8
9      <h1>Add your Title here</h1>
10
11     <div>
12       <img src="meme (483).jpg">
13       <p>Add your own text here!</p>
14     </div>
15
16   </body>
17
18 </html>
19
20
21
22
```



- **`<h1>`** is a heading. You can use the numbers 1 to 6 to create headings of different sizes. It needs a **`</h1>`** closing tag.
- **`<div>`** is short for division, and is a way of grouping stuff together. In this webpage, you'll use it to group together all the stuff for each part of your story. It needs a closing **`</div>`** tag.
- **`<img>`** is an image. It does **not** need a closing tag.
- **`<p>`** is a paragraph of text. It needs a closing **`</p>`** tag.

Changing the Images



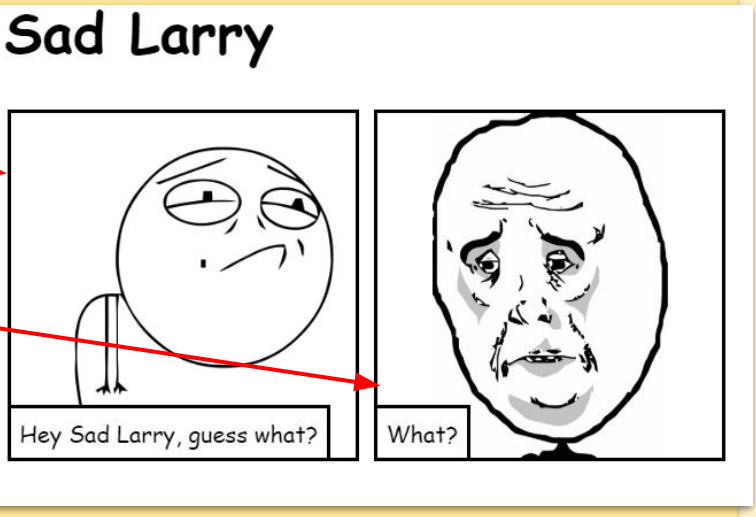
Can you find the HTML for the image (Hint: it's on line 12)? Click on the **image library** icon to find a different image title. Change the name from "meme (483).jpg" to another one and you'll see the image change! You can use any of the images in the library. We've loaded these in already.

Adding Another Frame

1 Use **CTRL + C** to copy these lines, the first frame and it's text. Paste (**CTRL + V**) a copy including the ** ** and ** ** tags to create a second picture.

```
<div>
  <img src="meme (483).jpg">
  <p>Hey Sad Larry, guess what?</p>
</div>
```

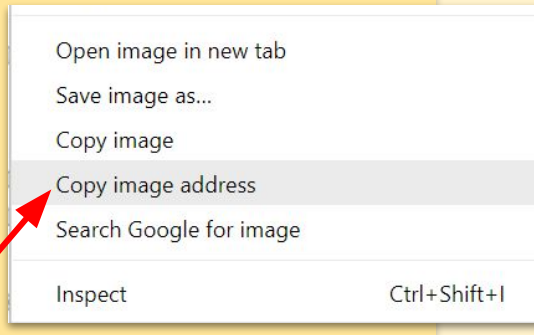
```
7   <body>
8
9     <h1>Sad Larry</h1>
10
11    <div>
12      <img src="meme (483).jpg">
13      <p>Hey Sad Larry, guess what?</p>
14    </div>
15
16    <div>
17      <img src="1.jpg">
18      <p>What?</p>
19    </div>
20
21  </body>
22
23 </html>
24
25
26
```



2 Add a new line of text inside your new ** ** tag.

3 Add an image to your new box, by adding this code inside your **`<div>`** tag, **`<img src="The name of the image.png">`**. For HTML images, you need to add the source of the image, inside the speech marks.

4 With *Trinket.io,* you can upload images from the web if you have an account but if you don't, don't worry. You can insert images links/URLs directly in your **`<img>`** tag.



Right click on an image online, copy the image address and Paste the URL between the speech marks in your **`<img>`** tag. You should see your image appear.

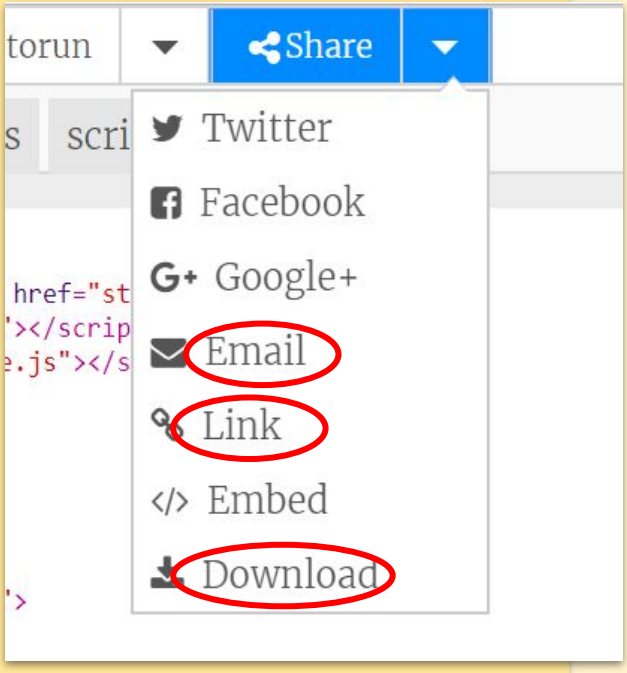
```
16    <div>
17      <img src="https://s3-eu-west-1.amazonaws.com/codeclub-h
18      <p>More text here!</p>
19    </div>
20
```

Finish your story

Use what you have learned to create a finished project telling your story.

Saving your work

You have a few saving options accessible via the **Share** menu at the top of the screen. If it is actually your friends birthday, save the **link** or you can send the link in an **email**.



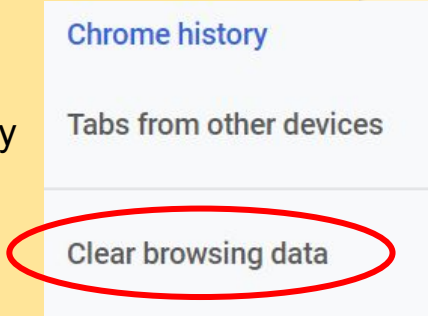
Alternatively you can hit **download** to save all of your files (including pictures) in one zipped file. This can then be saved on your USB or in a specific folder on your computer.

Using Bit.ly

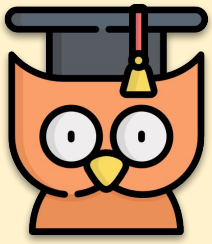
Your link may be a bit long. www.bit.ly is a URL shortening tool so that you can easily and quickly share your long link with your tutor, friends or family.

Help us out - Clear your History

Once you've saved your work, you'll need to delete the history so that the next student can start fresh. Press **Ctrl+H,** Click **Clear browsing data** and clear your history from the last 24 hours.

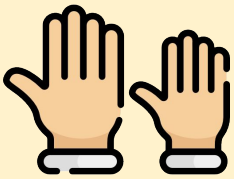


Lesson 4 - HTML/CSS - Wanted Poster

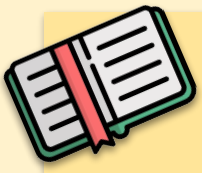


Learning Outcomes

- How to use CSS within HTML Pages
- How to edit HTML and CSS to create our own Wanted Poster



REMEMBER: Put up your hand. We love to help!



Making a Wanted Poster

Today we're going to make a Wanted Poster with HTML and CSS.

Access the beginning code at the following link. Remember to **hold CTRL** when clicking as you'll need to keep the notes open. www.bit.ly/wantedcoder

Click on the "**style.css**" tab. You'll notice that there are already CSS properties for anything called **`<div>`**

What do you think these do?


```
div {
    text-align: center;
    overflow: hidden;
    border: 2px solid black;
    width: 300px;
}
```

Editing the CSS

1 Let's start by changing the `text-align` property:

```
text-align: center;
```

What happens when you change the word `center` to `left` or `right`?

2 How about the `border` property?

```
border: 2px solid black;
```

`2px` in the code above means 2 pixels. What happens when you change `2px solid black` to `4px dotted red`?

3 Change the `width` of the poster to `400px`. What happens to the poster?

4 Let's add some CSS to set the background colour of the poster. Go to the end of line 5 of your code and press return, so that you have a new blank line.

```
1 div {
2   text-align: center;
3   overflow: hidden;
4   border: 4px dotted red;
5   width: 400px;
6   background: yellow;
7 }
```

5 Add the following CSS property to your div style.

```
border-radius: 40px;
```

What does this property do? What happens if you change the number in the code above?

Styling images

Let's improve the style of the image in the poster. At the moment, there aren't any CSS properties for your `<img>` tag, so let's add some!

1. Firstly, add the following code underneath the **CSS** for your **div** to set the width of the image. You'll see that the size of the image changes, so that its width is 100 pixels.

index.html | style.css

```
div {
  text-align: center;
  overflow: hidden;
  border: 4px dotted red;
  width: 400px;
  background: yellow;
}

img {
  width: 100px;
}
```

2. You can also add a border around the image with this code

```
border: 1px solid black;
```

3. There's not much space between the image and the border. You can fix this by adding some **padding** around the image.

```
padding: 10px;
```

What do you think would happen if you changed the padding to `50px`?

Can you give your image a background colour? Or a rounded border?

You can find more CSS colour names at

www.trinket.io/docs/colors

Styling headings

1 Add the following code underneath your image's CSS:

```
h1 {

}
```

This is where you'll add CSS properties for your main `<h1>` heading.

2 To change the font of your `<h1>` headings, add the following code between the curly brackets:

```
font-family: Impact;
```

3 You can also change the size of the heading:

```
font-size: 50pt;
```

Have you noticed that there's a big space between the `<h1>` heading and the stuff around it? This is because there's a margin around the heading. A margin is the space between the element (in this case a heading) and the other stuff around it.

4 Make the margin smaller with this code:

```
margin: 10px;
```

5 Underline your heading:

```
text-decoration: underline;
```

Challenge: Finish your Poster

Use what you have learned to create a finished project of your Poster.

Experiment with the different CSS properties below to see what they do

```
color: black;
background: white;
font-family: Arial / Comic Sans MS / Courier / Impact / Tahoma;
font-size: 12pt;
font-weight: bold;
text-decoration: underline overline line-through;
margin: 10px;
padding: 10px;
width: 100px;
height: 100px;
```

Extra Task

www.bit.ly/wantedcoder

Using everything you've learned in this and previous lessons, make a poster for a **lost dog or cat.**

Flick back to **page 12** if you can't remember how to import an image from somewhere on the web.