Introduction to Processing

Formatting shapes

Produced by:

Department of Computing and Mathematics

Waterford Institute *of* Technology

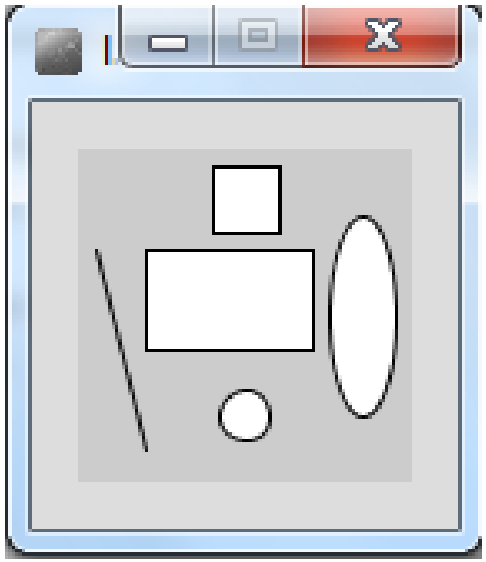
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Topics list

- Formatting the display window.
- Filling shapes with colour.
- Formatting the shape outline.

Formatting the display window

- Our display window is looking fairly cramped.
- The default size of your display window is 100x100 pixels, which is quite small.



Formatting the display window

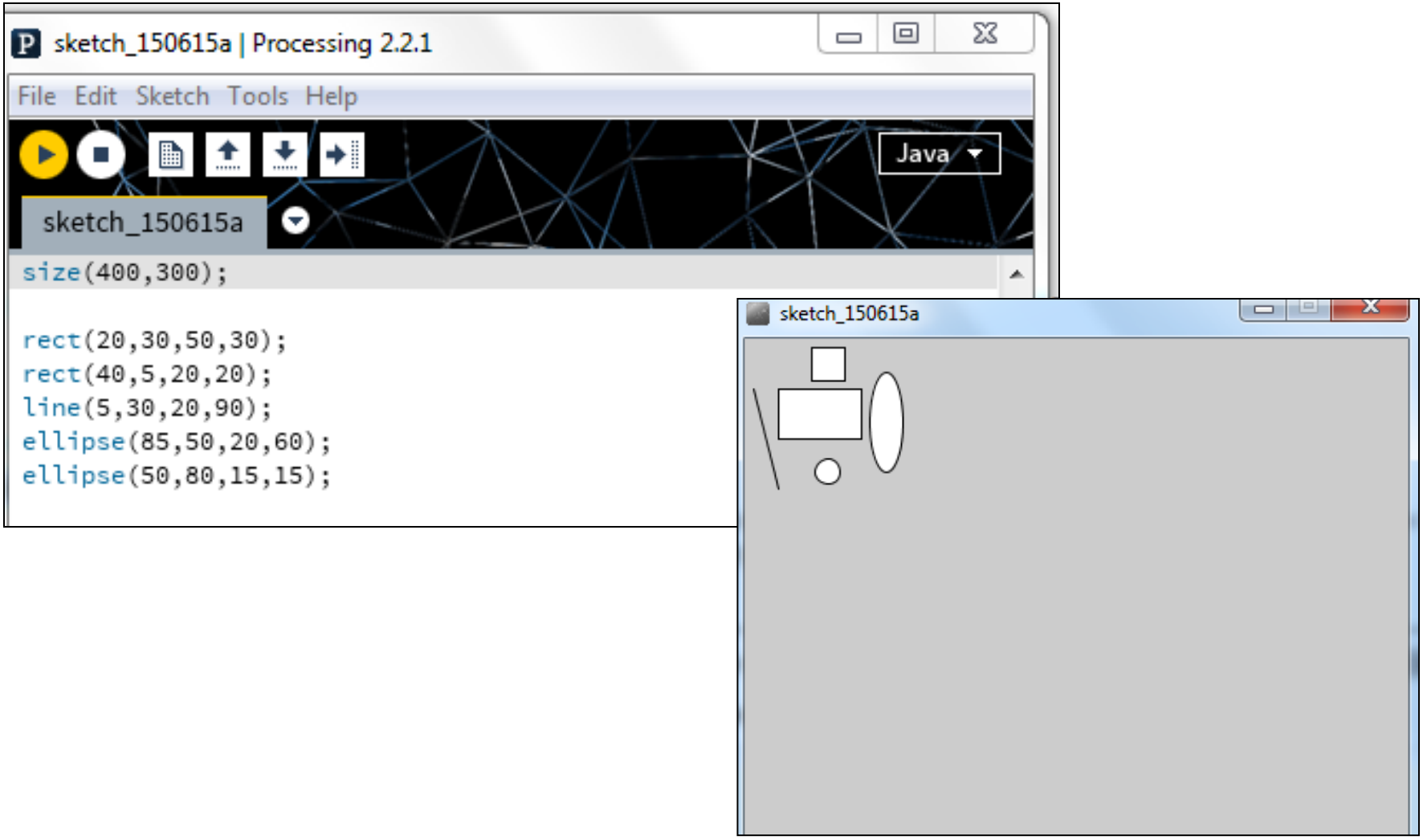
- We can change the size of the display window by calling the size function.
- When you use the size function in static drawings, it has to be the first line of code in your sketchbook.

size(w, h)

w = width of the display window

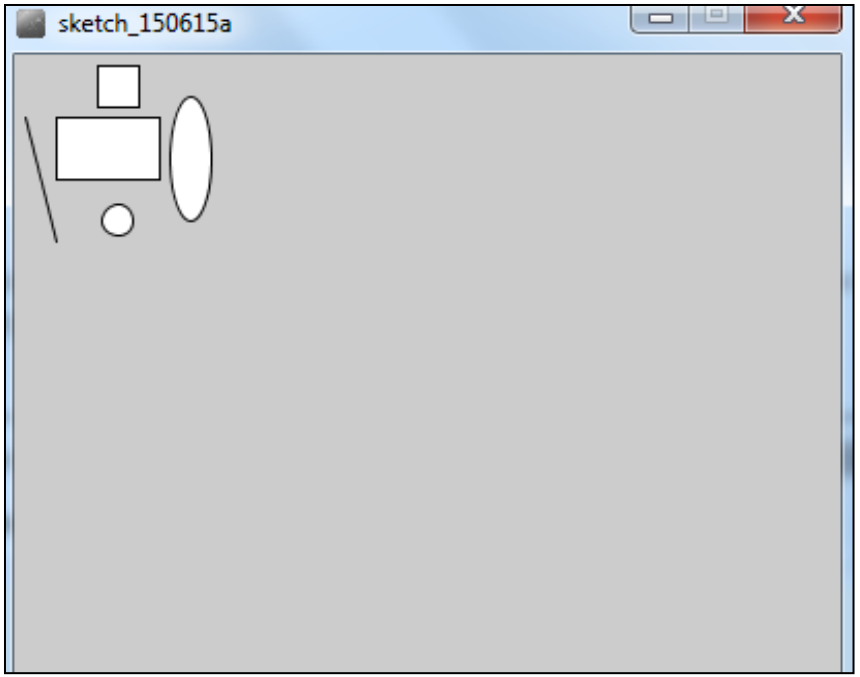
h = height of the display window

size()

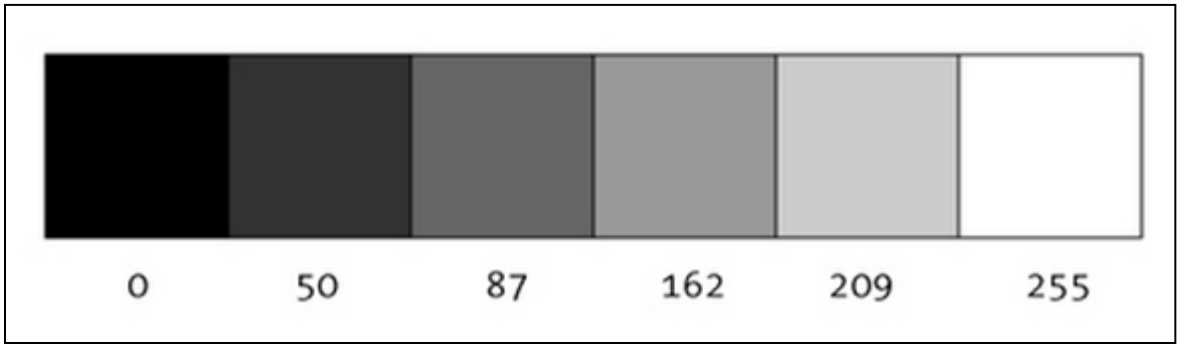

sketch_150615a | Processing 2.2.1
File Edit Sketch Tools Help
Java
sketch_150615a
size(400,300);
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
ellipse(85,50,20,60);
ellipse(50,80,15,15);
sketch_150615a

Formatting the display window

- Our display window looks less cramped now.
- But the default gray colour is not very appealing.
- We could use the background function to set the colour to something nicer.



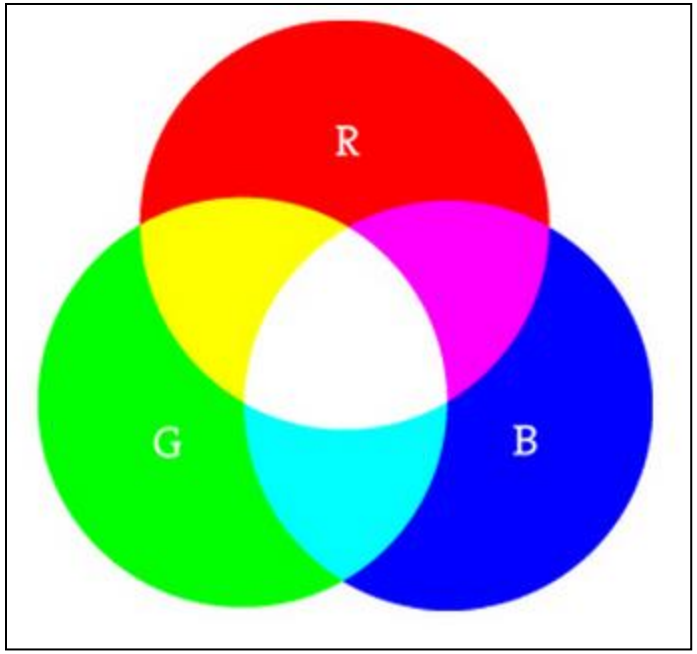
A note on colour first…Grayscale



"0 means black, 255 means white. In between, every other number - 50, 87, 162, 209, and so on - is a shade of gray ranging from black to white."

https://www.processing.org/tutorials/color/

A note on colour first…RGB

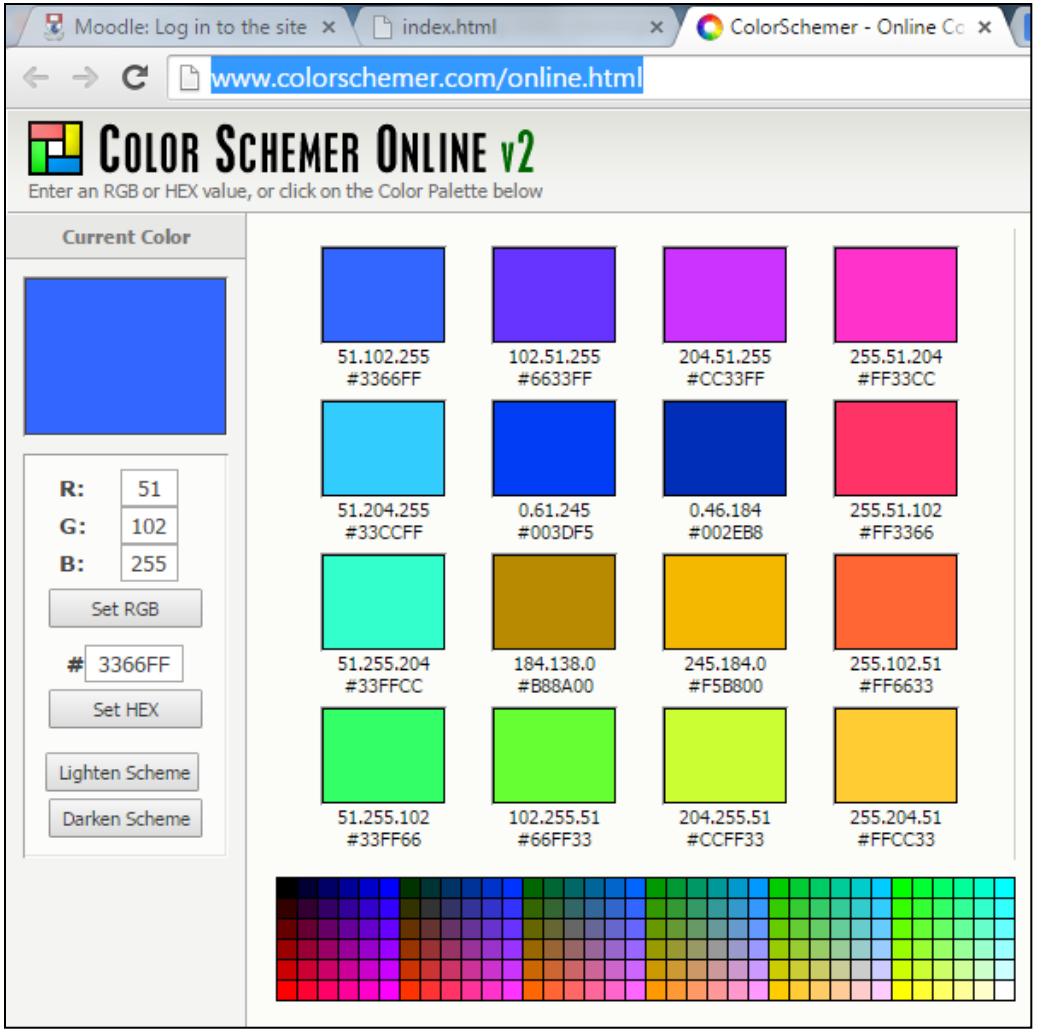


"As with grayscale, the individual color elements are expressed as ranges from 0 (none of that color) to 255 (as much as possible), and they are listed in the order R, G, and B."

Digital colours are made by mixing the three primary colours of light (red, green, and blue).

https://www.processing.org/tutorials/color/

A note on colour first…RGB



http://www.colorschemer.com/online.html

background() - syntax

background(grayscale)

grayscale = grayscale colour (a number between 0 [black] and 255 [white] inclusive)

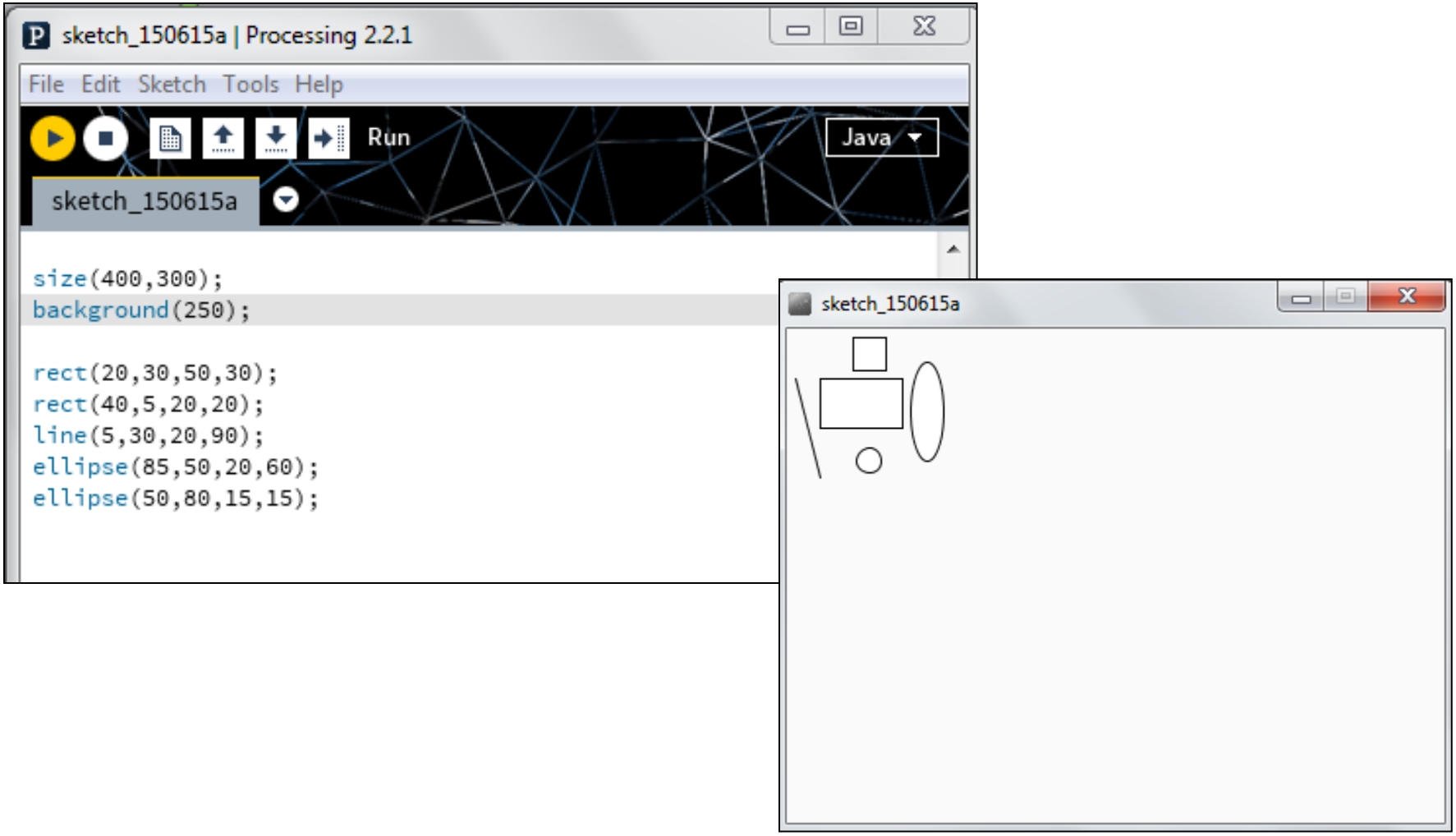
background(r, g, b)

r = red colour (a number between 0 and 255 inclusive)

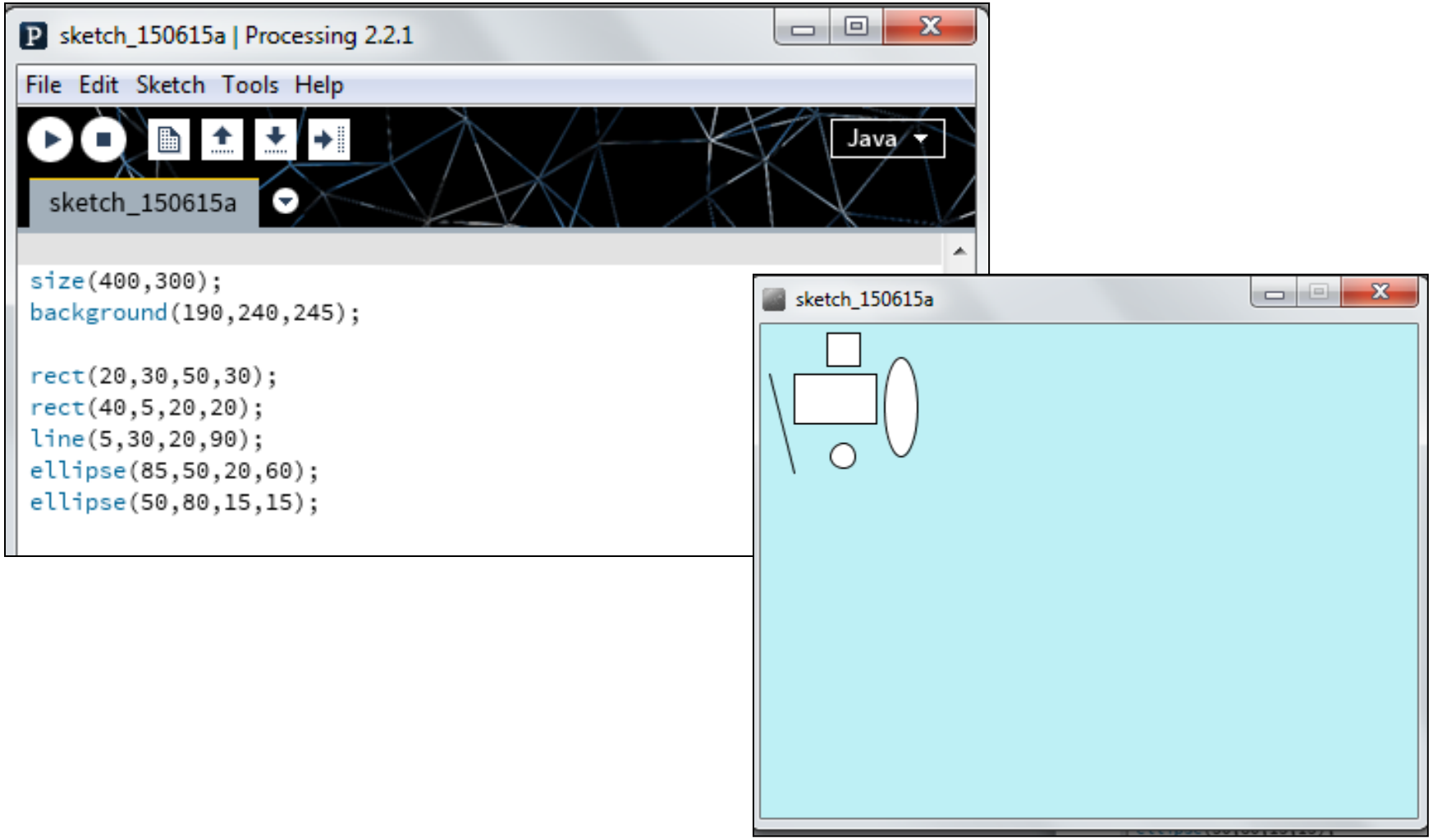
g = green colour (a number between 0 and 255 inclusive)

b = blue colour (a number between 0 and 255 inclusive)

background()


sketch_150615a | Processing 2.2.1
File Edit Sketch Tools Help
Run
Java
sketch_150615a
size(400,300);
background(250);
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
ellipse(85,50,20,60);
ellipse(50,80,15,15);
sketch_150615a

background()


sketch_150615a | Processing 2.2.1
File Edit Sketch Tools Help
Java
sketch_150615a
size(400,300);
background(190,240,245);
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
ellipse(85,50,20,60);
ellipse(50,80,15,15);
sketch_150615a

Topics list

- Formatting the display window.
- Filling shapes with colour.
- Formatting the shape outline.

fill() - syntax

fill (r, g, b)

r = red colour (a number between 0 and 255 inclusive)

g = green colour (a number between 0 and 255 inclusive)

b = blue colour (a number between 0 and 255 inclusive)

- fills shapes with a chosen colour.
- can use the RGB colours to select a colour.
- all shapes drawn after the **fill** function is called, will be filled with the chosen colour.

fill()

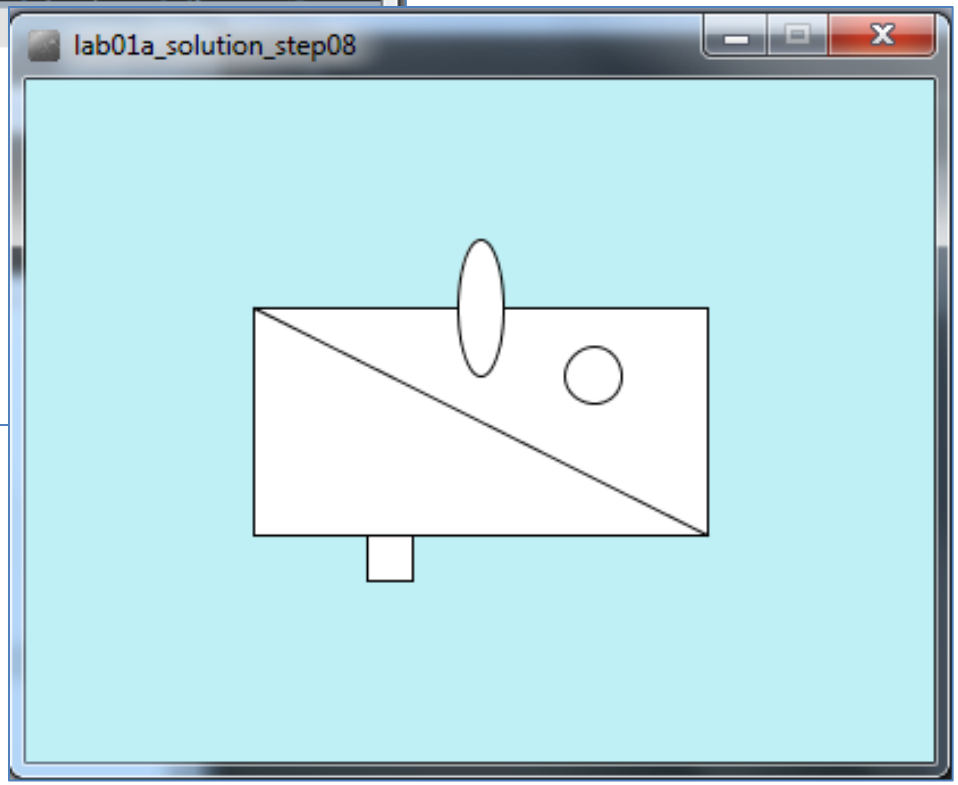
lab01a_solution_step08 | Processing 2.2.1

File Edit Sketch Tools Help

Java

lab01a_solution_step08

```
size(400,300);
background(190,240,245);

rect(100,100,200,100);
rect(150,200,20,20);
line(100,100,300,200);
ellipse(200,100,20,60);
ellipse(250,130,25,25);
```



Starting code…

fill()

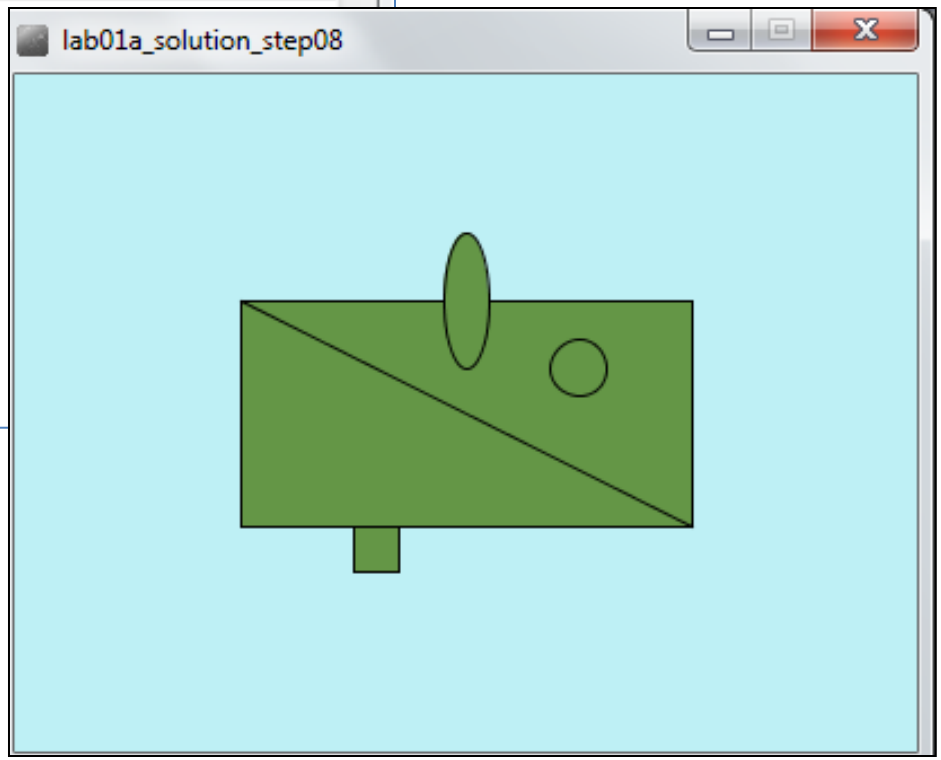
lab01a_solution_step08 | Processing 2.2.1

File Edit Sketch Tools Help

Java

lab01a_solution_step08

```
size(400,300);
background(190,240,245);

fill(100,150,70);

rect(100,100,200,100);
rect(150,200,20,20);
line(100,100,300,200);
ellipse(200,100,20,60);
ellipse(250,130,25,25);
```



All shapes filled with dark green…

fill()

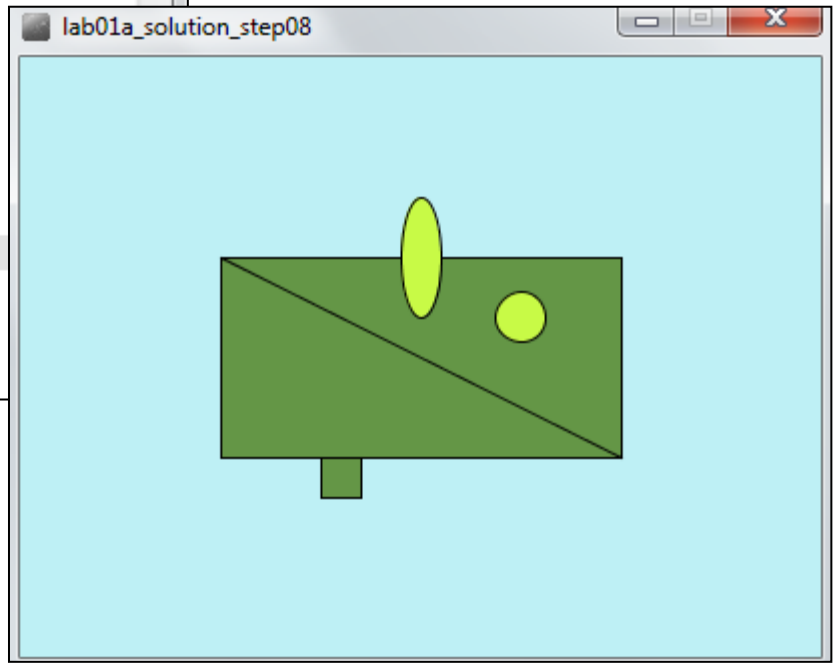
lab01a_solution_step08 | Processing 2.2.1

File Edit Sketch Tools Help

Java

lab01a_solution_step08

```
size(400,300);
background(190,240,245);

fill(100,150,70);

rect(100,100,200,100);
rect(150,200,20,20);
line(100,100,300,200);

fill(200,250,70);

ellipse(200,100,20,60);
ellipse(250,130,25,25);
```



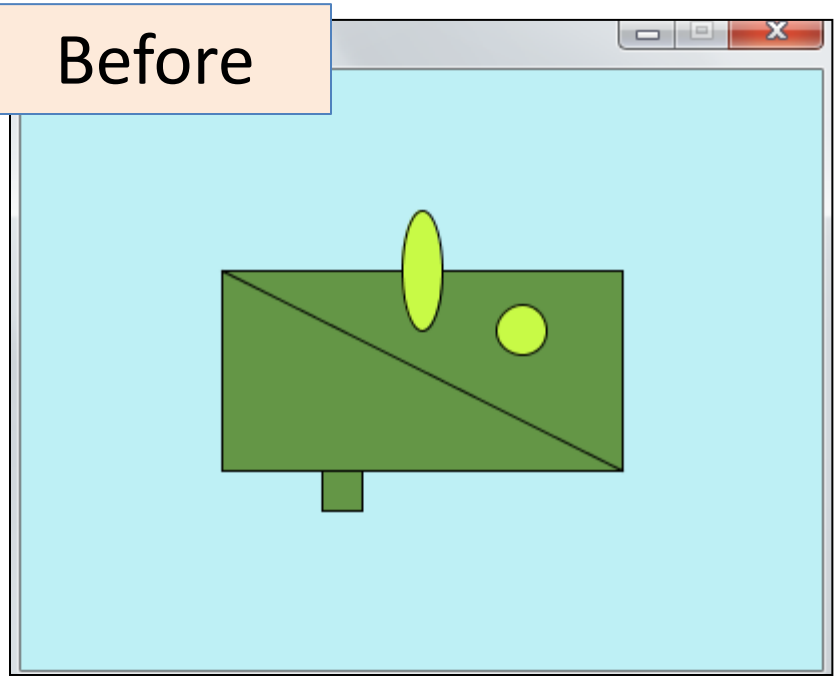
Rectangles filled with dark green…
Ellipses filled with light green…
Order of statements matter!!!

Topics list

- Formatting the display window.
- Filling shapes with colour.
- Formatting the shape outline.

Changing the outline (i.e. stroke)

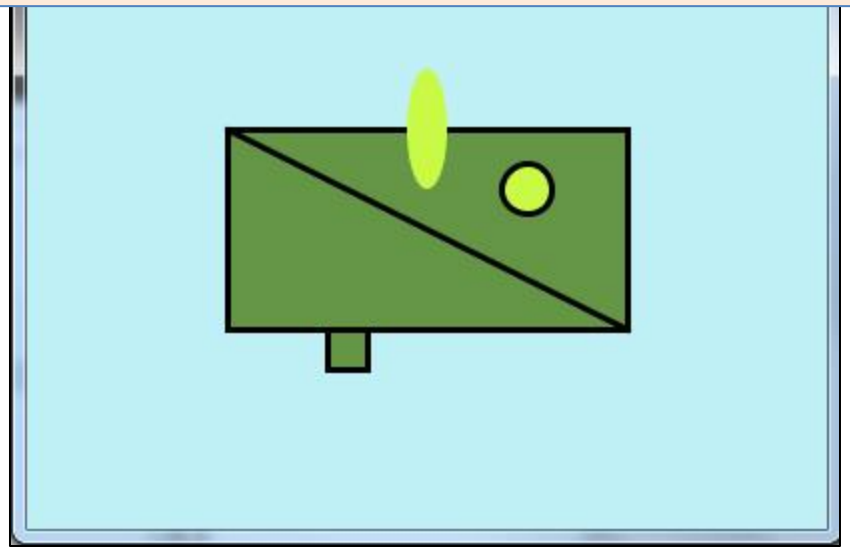
Before



After (changes):

- The oval has no border; all other shapes do.
- The outline is heavier.

We will now make those changes

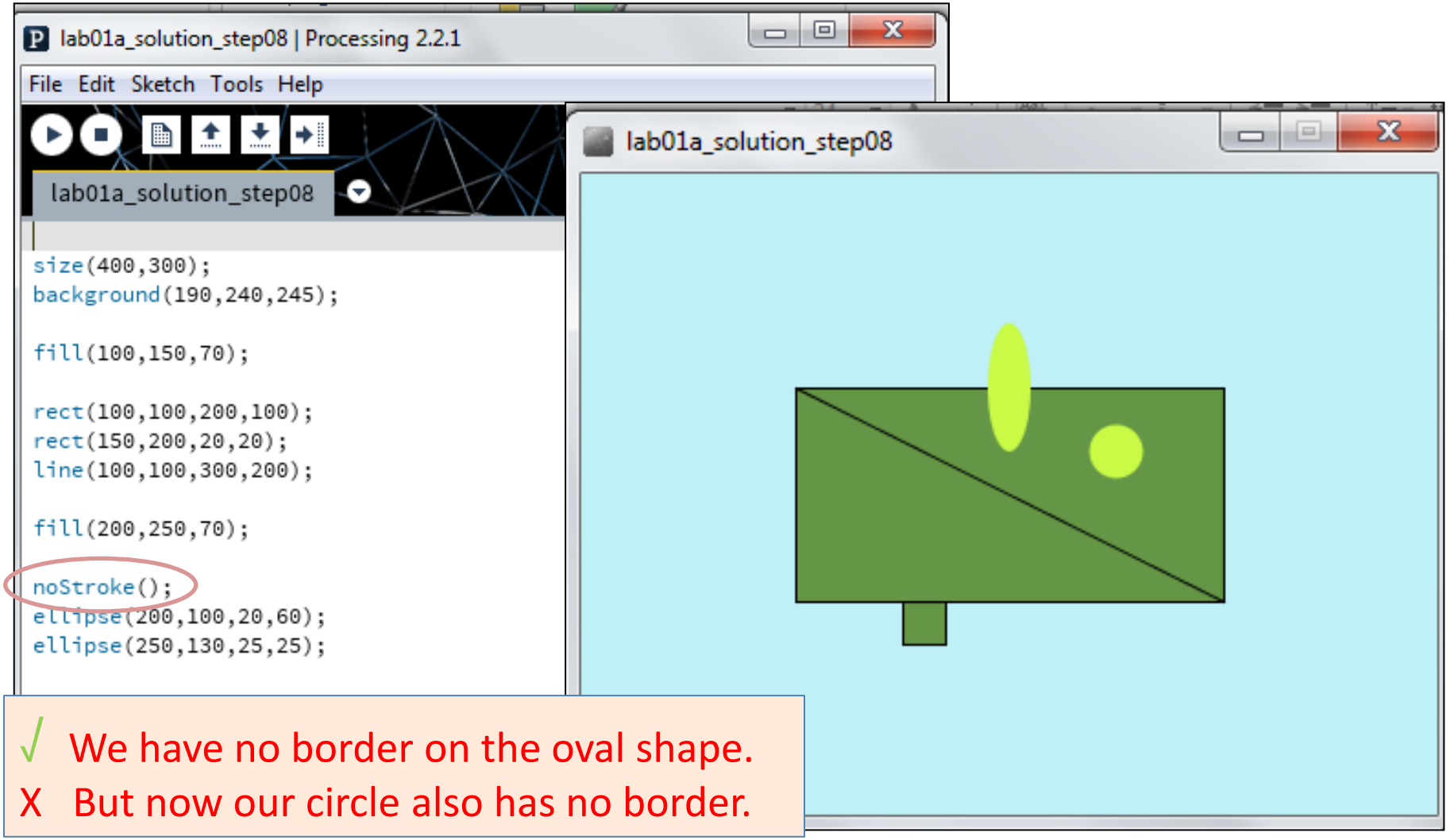


noStroke() - syntax

```
noStroke();
    //no parameters defined for this function.
```

- A **stroke** is the outline of a shape.
- The noStroke() function disables the outline on shapes that are drawn after the function is called.
- All shapes drawn after the **noStroke** function is called, will have no outline.

noStroke()



√ We have no border on the oval shape.

X But now our circle also has no border.

stroke() - syntax

stroke (r, g, b)

r = red colour (a number between 0 and 255 inclusive)

g = green colour (a number between 0 and 255 inclusive)

b = blue colour (a number between 0 and 255 inclusive)

- The stroke() function enables the outline on all shapes that are drawn after the function is called.
- When you call stroke(), you need to specify a colour.

stroke()

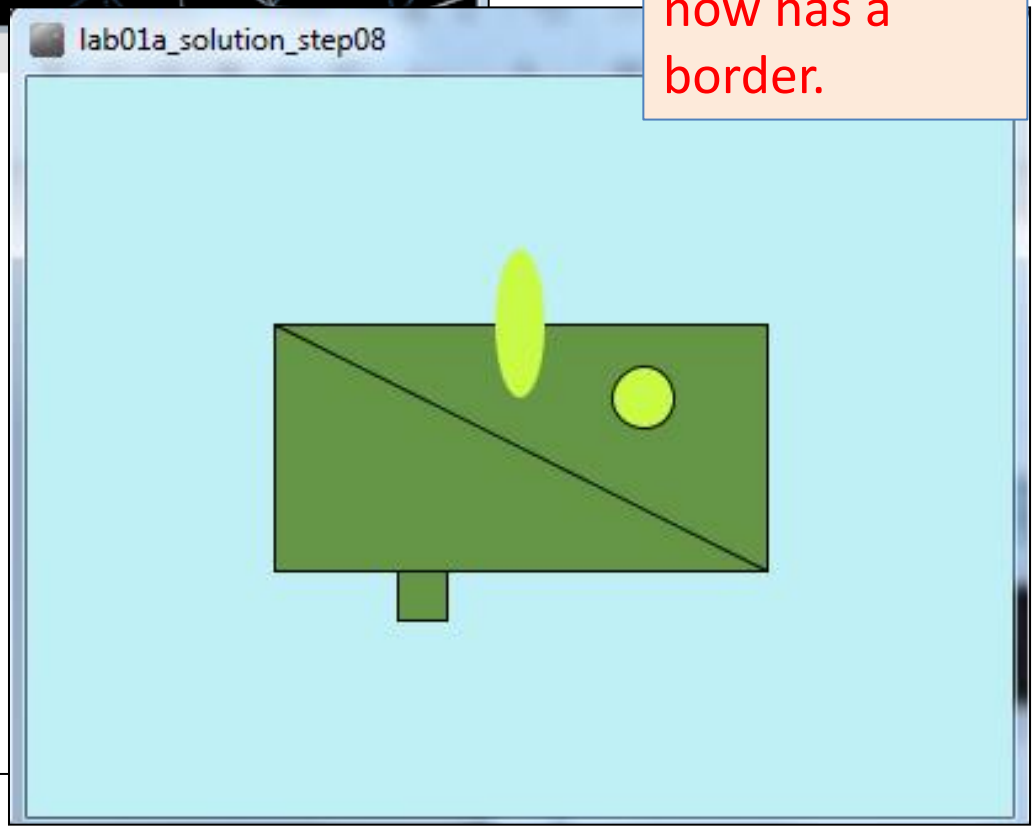
lab01a_solution_step08 | Processing 2.2.1

File Edit Sketch Tools Help

Java

lab01a_solution_step08

```
size(400,300);
background(190,240,245);

fill(100,150,70);

rect(100,100,200,100);
rect(150,200,20,20);
line(100,100,300,200);

fill(200,250,70);

noStroke();
ellipse(200,100,20,60);

stroke(0,0,0);
ellipse(250,130,25,25);
```



√ Our circle now has a border.

strokeWeight() - syntax

strokeWeight (pixels)

pixels = thickness of the outline measured in pixels.

- The strokeWeight() function allows you to choose the thickness of a line/outline on shapes.
- The chosen thickness will apply to all lines/shapes that are drawn after the function is called.
- The thickness is specified in pixels.
- The default thickness is 1 pixel.

strokeWeight()

lab01a_solution_step08 | Processing 2.2.1

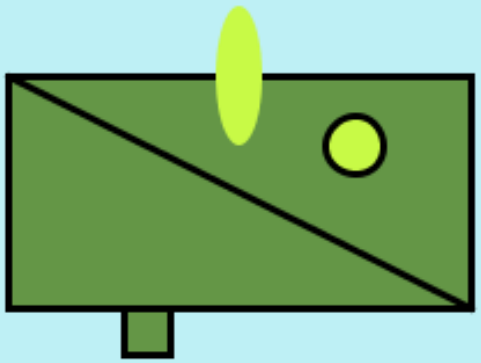
File Edit Sketch Tools Help

Stop Java

lab01a_solution_step08

```
size(400,300);
background(190,240,245);
strokeWeight(3);

fill(100,150,70);

rect(100,100,200,100);
rect(150,200,20,20);
line(100,100,300,200);

fill(200,250,70);

noStroke();
ellipse(200,100,20,60);

stroke(0,0,0);
ellipse(250,130,25,25);
```

lab01a_solution_step08



√ Our outline is now heavier

Questions?

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
http://www.wit.ie/