An Introduction to Processing

Creating Static Drawings

Lecturer: Caio Fonseca

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

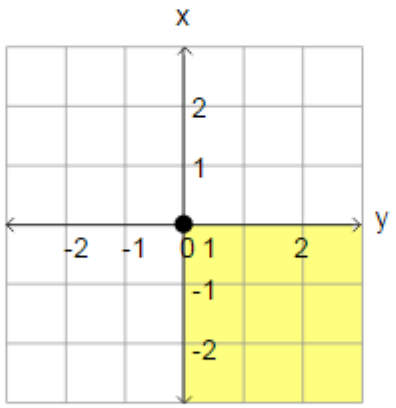
Department of Computing and Mathematics
http://www.wit.ie/

Topics list

- Coordinate System in Computing.
- Functions in Processing.
- Basic Shapes.
- Formatting the display window.
- Syntax Errors.
- Logic Errors.

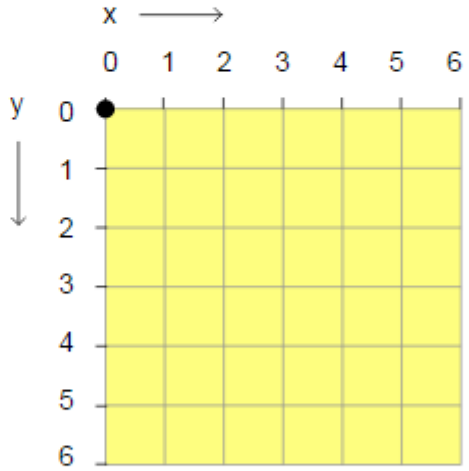
Coordinate System in Computing

In Geometry,
we use this type of
coordinate system:



point (0,0) is in the centre.

In Computing, we use this type of coordinate system to represent the screen:

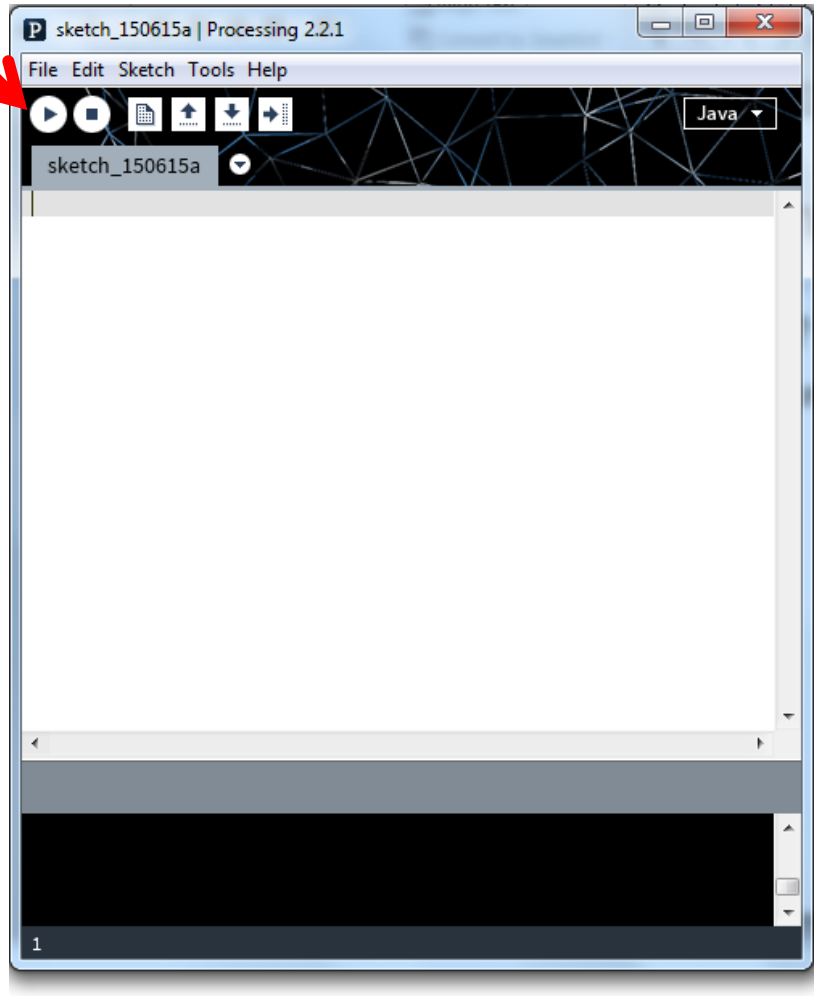


point (0,0) is in the top left hand corner. Each number is a pixel.

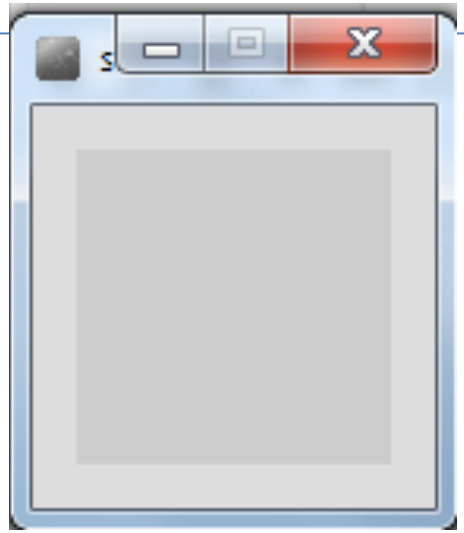
https://processing.org/

Coordinate System in Computing

Run button



- So how does this relate to Processing?
- When you open Processing and click on the run button, a display window pops up.



Display window

Coordinate System in Computing

- The display window is where your code is run/ displayed.
- It follows the rules of the Computing coordinate system i.e. the top left hand corner is (0,0).
- A point (10,20) is 10 pixels to the right of (0,0) and 20 pixels below (0,0).

Topics list

- Coordinate System in Computing.
- Functions in Processing.
- Basic Shapes.
- Formatting the display window.
- Syntax Errors.
- Logic Errors.

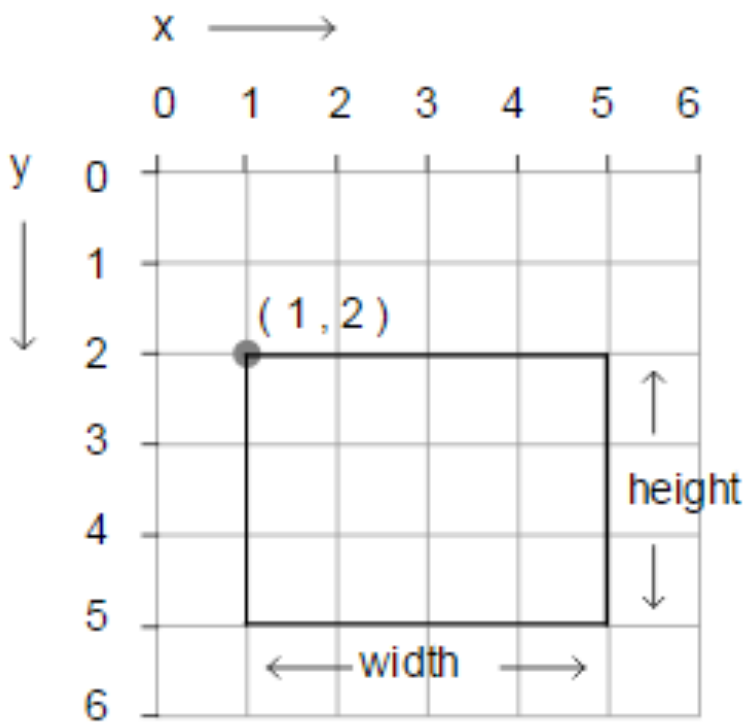
Functions in Processing

- Processing comes with several pre-written functions that we can use.
- A function comprises a set of instructions that performs some task.
- When you call the function, it performs the task.
- We will now look at functions that draw the following shapes:
  - Rectangle, square, line, oval and circle.

Topics List

- Coordinate System in Computing.
- Functions in Processing.
- Basic Shapes.
- Formatting the display window.
- Syntax Errors.
- Logic Errors.

rect()

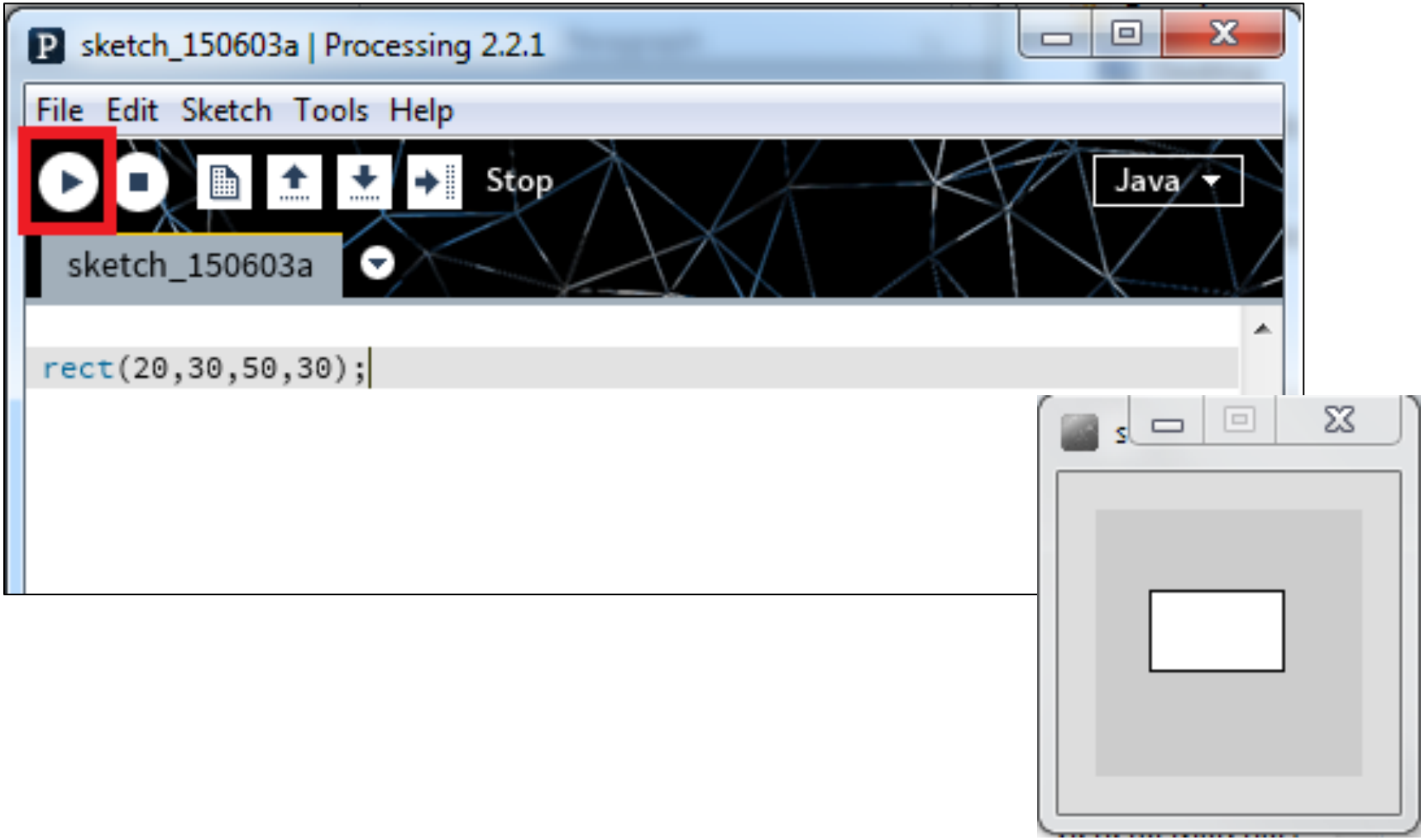


rect(x , y , width , height) ;

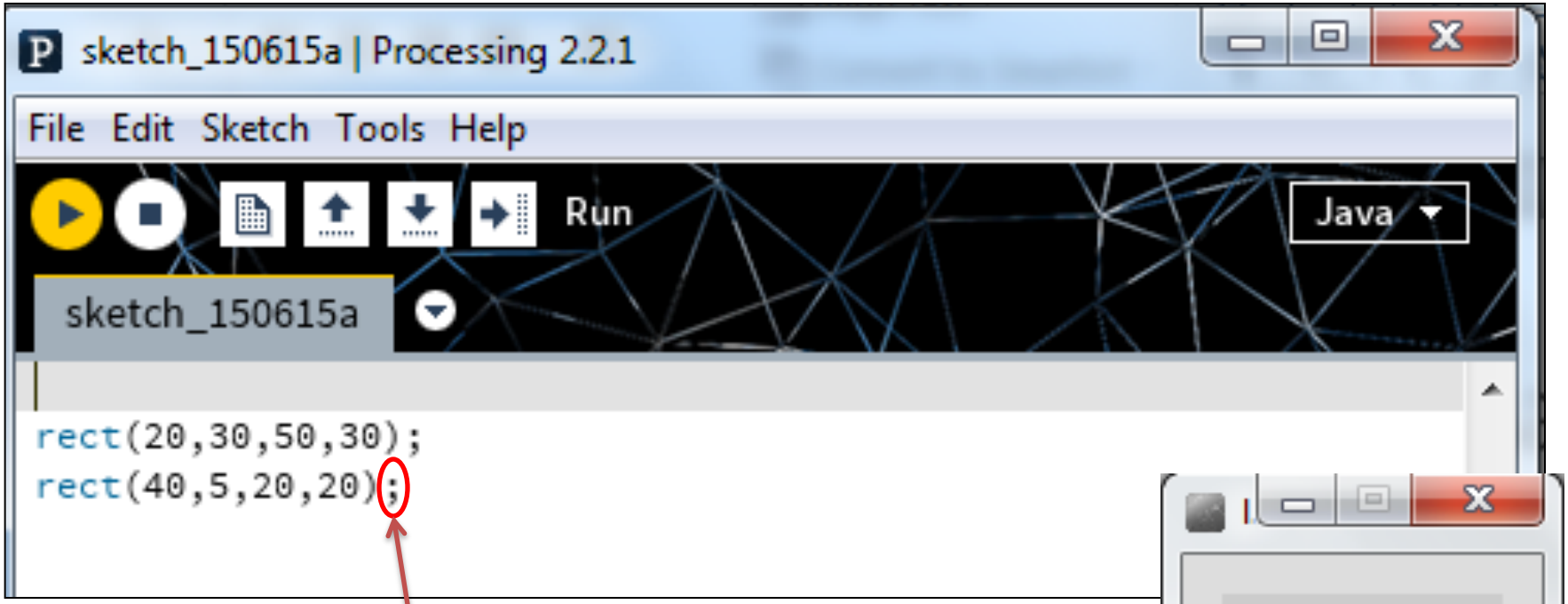
Example: rect (1 , 2 , 4 , 3) ;

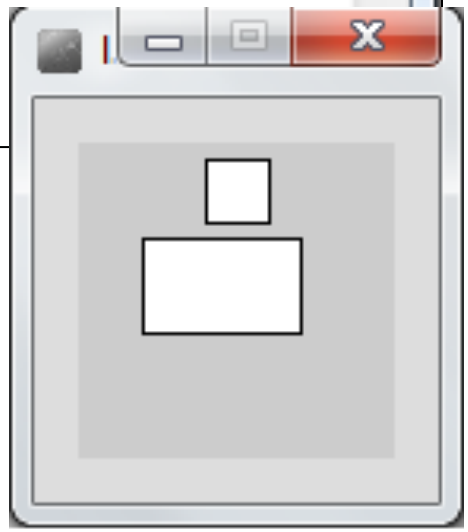
https://processing.org/

rect() – Drawing a Rectangle



rect() – Drawing a Square





Note how each line of code has a semi-colon (;) at the end of it. This is called a statement terminator and must be included.

rect() – Syntax

rect(x, y, w, h)

x = x-coordinate of the upper left corner of the rectangle

y = y-coordinate of the upper left corner of the rectangle

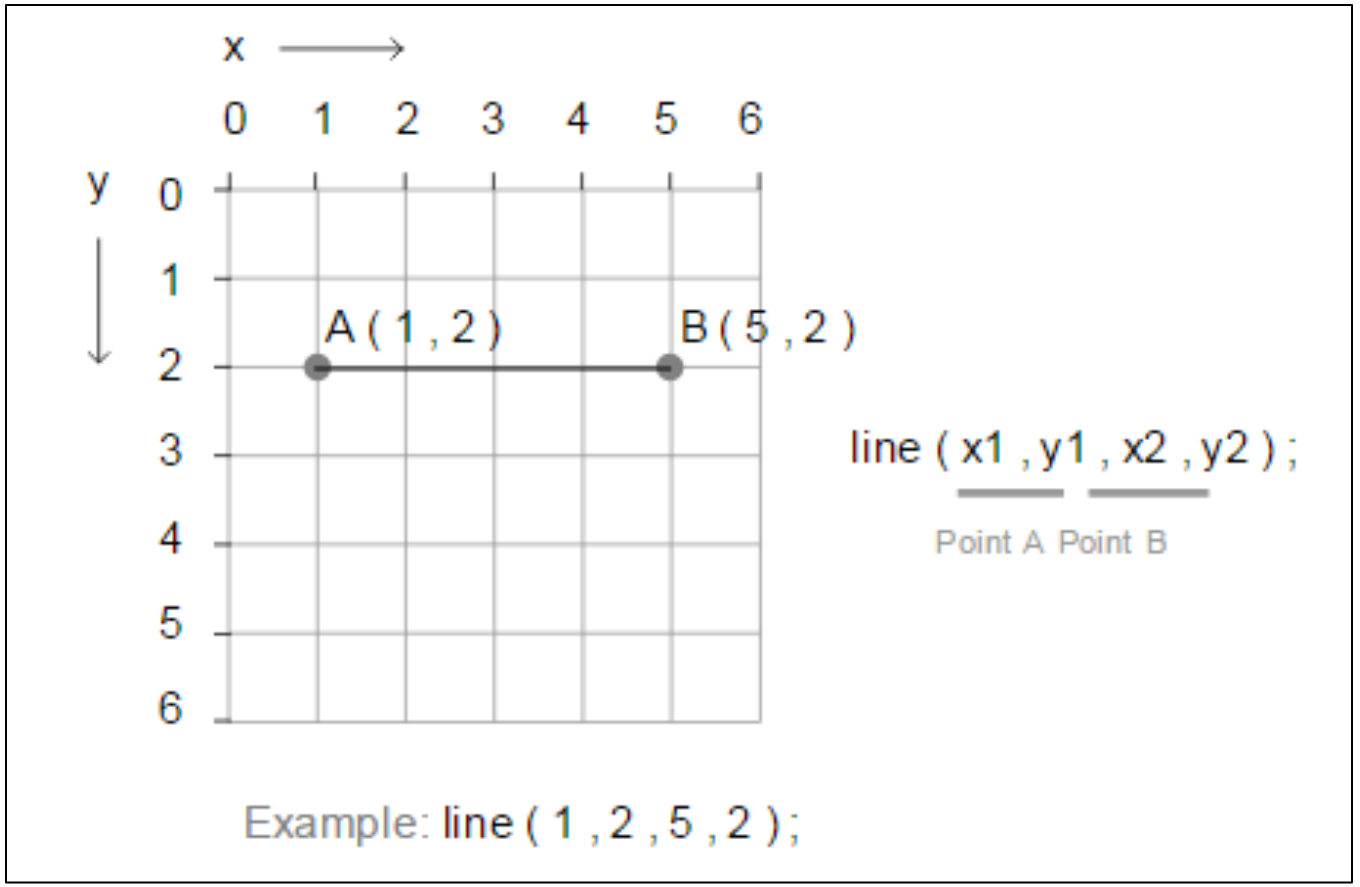
w = width of the rectangle

h = height of the rectangle

- The rect function above defines four parameters i.e. x, y, w, h.
- When you call rect, you are expected to pass four numbers to it. These actual numbers are called arguments.
- rect uses these four numbers to render the rectangle on the display window.

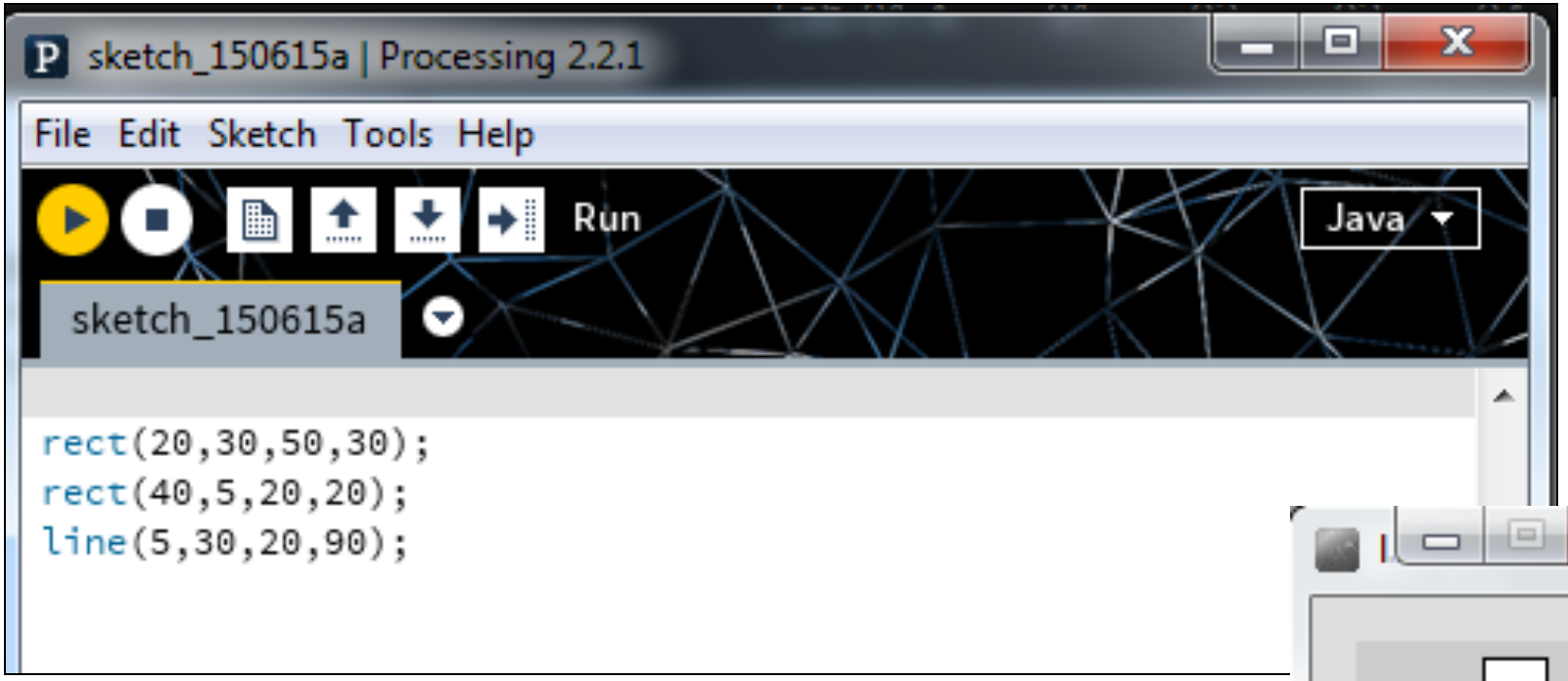
To draw a square, the width and height must be the same value.

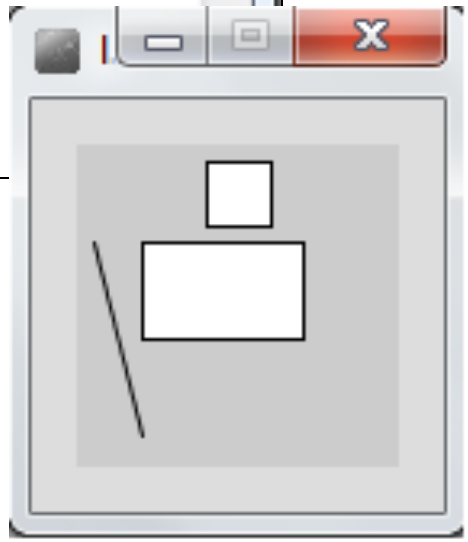
line()



https://processing.org/

line() – Drawing a Line





line() – Syntax

line(x1, y1, x2, y2)

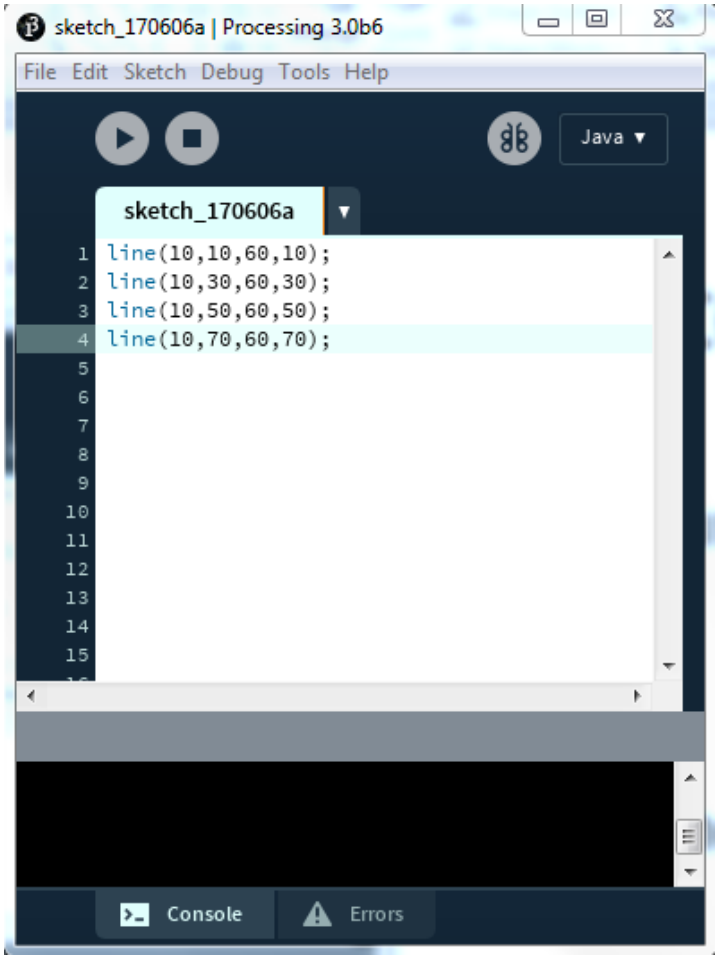
x1 = x-coordinate of first point

y1 = y-coordinate of first point

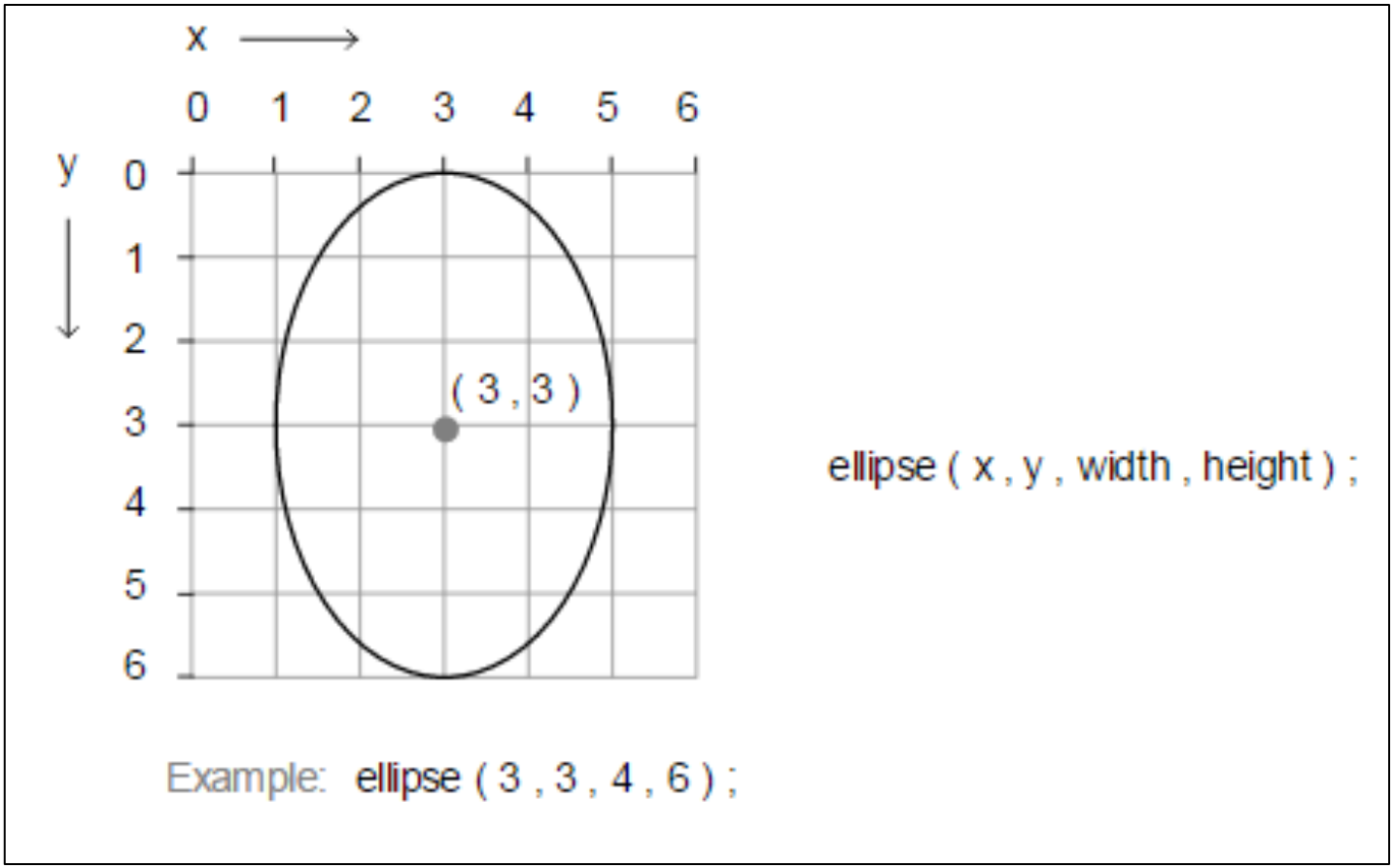
x2 = x-coordinate of second point

y2 = y-coordinate of second point

line() – More Examples

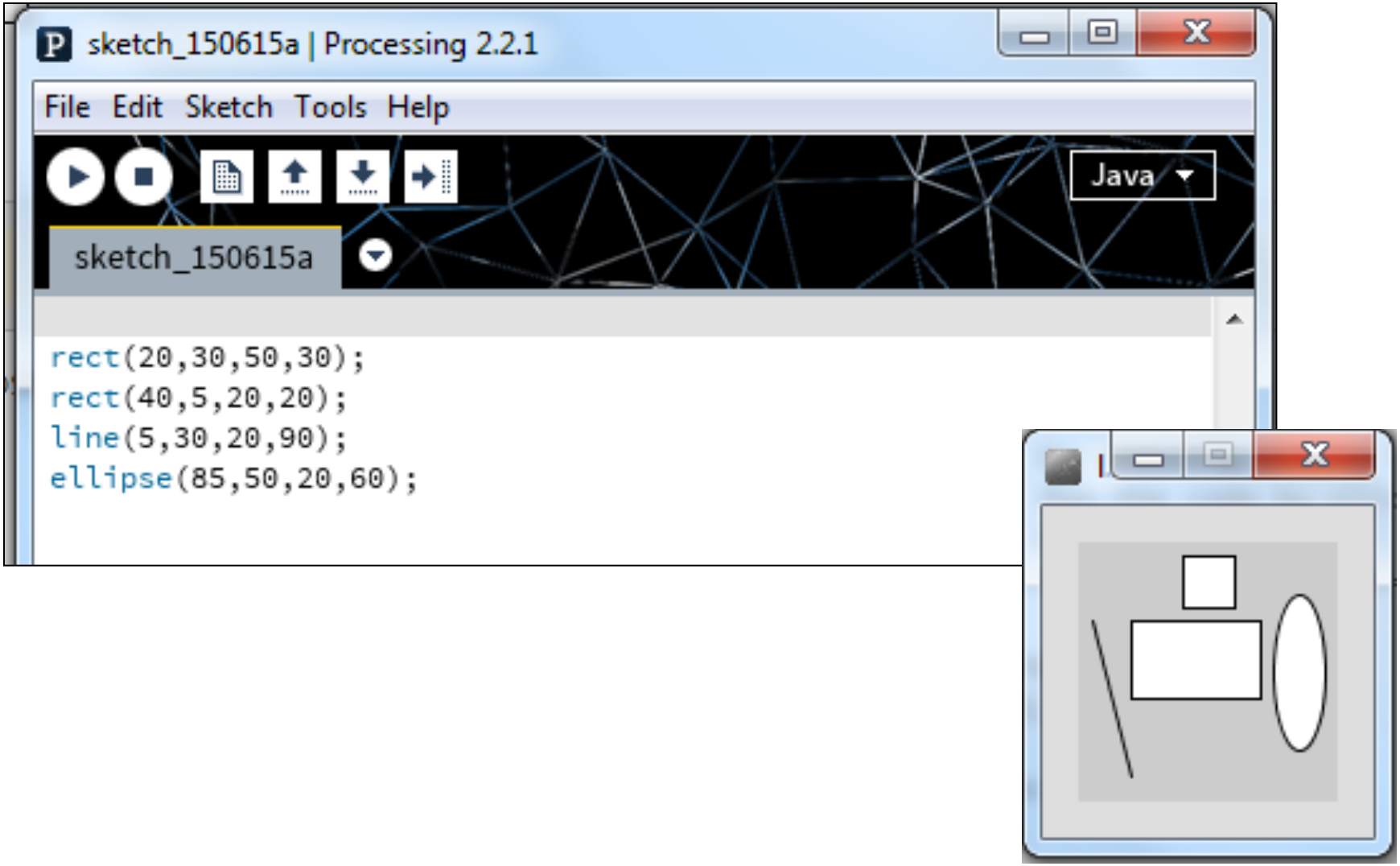


ellipse()

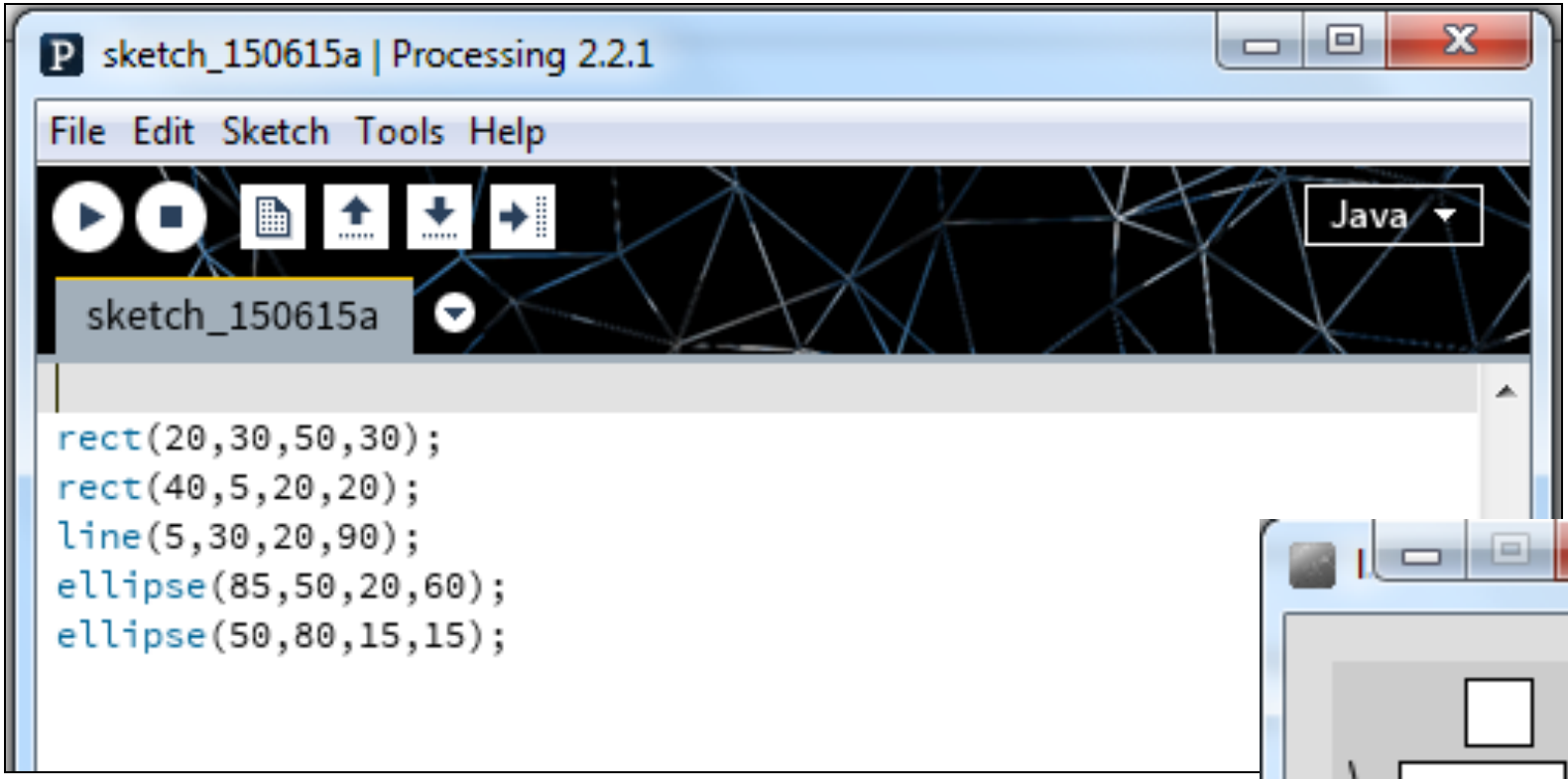


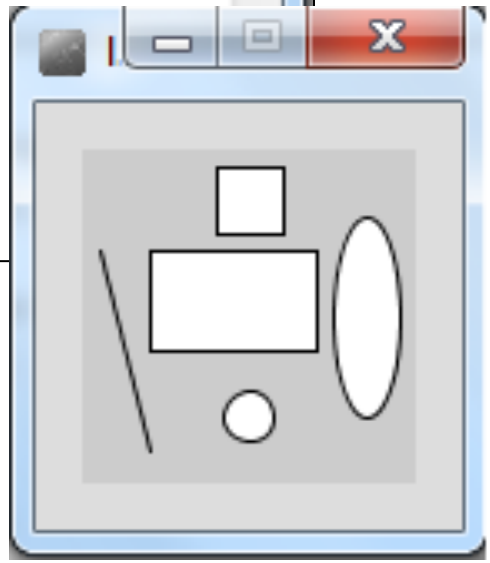
https://processing.org/

ellipse()


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

ellipse()


sketch_150615a | Processing 2.2.1
File Edit Sketch Tools Help
Java
sketch_150615a
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);



ellipse() – Syntax

ellipse(x, y, w, h)

x = x-coordinate of the centre of the ellipse

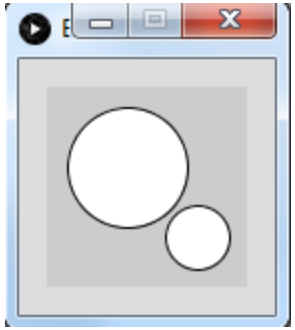
y = y-coordinate of the centre of the ellipse

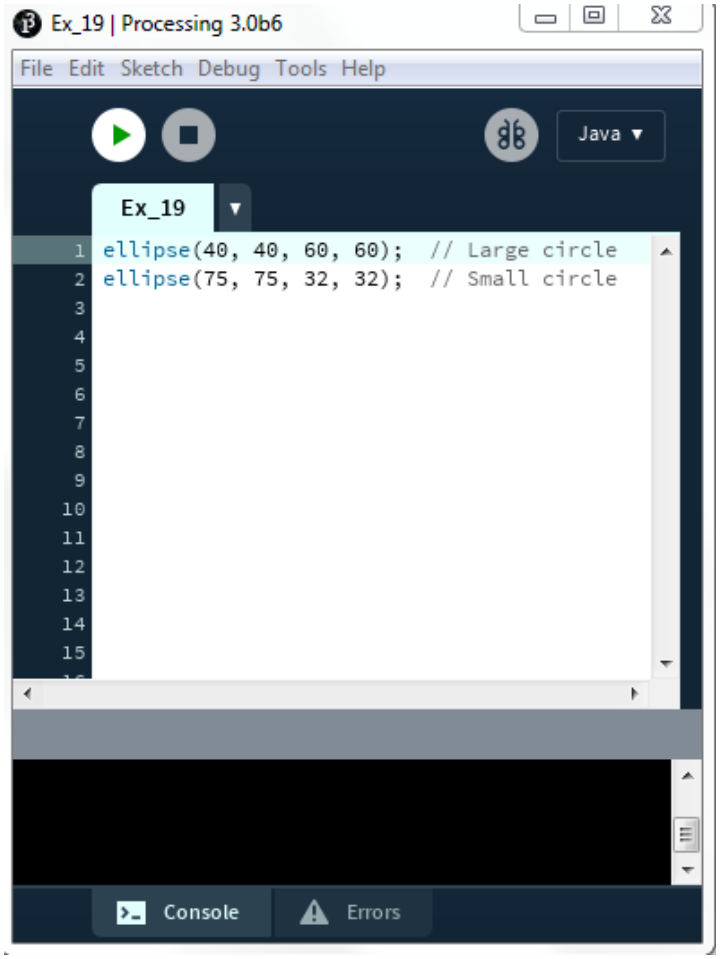
w = width of the ellipse

h = height of the ellipse

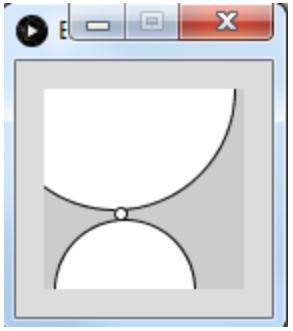
To draw a circle, the width and height must be the same value.

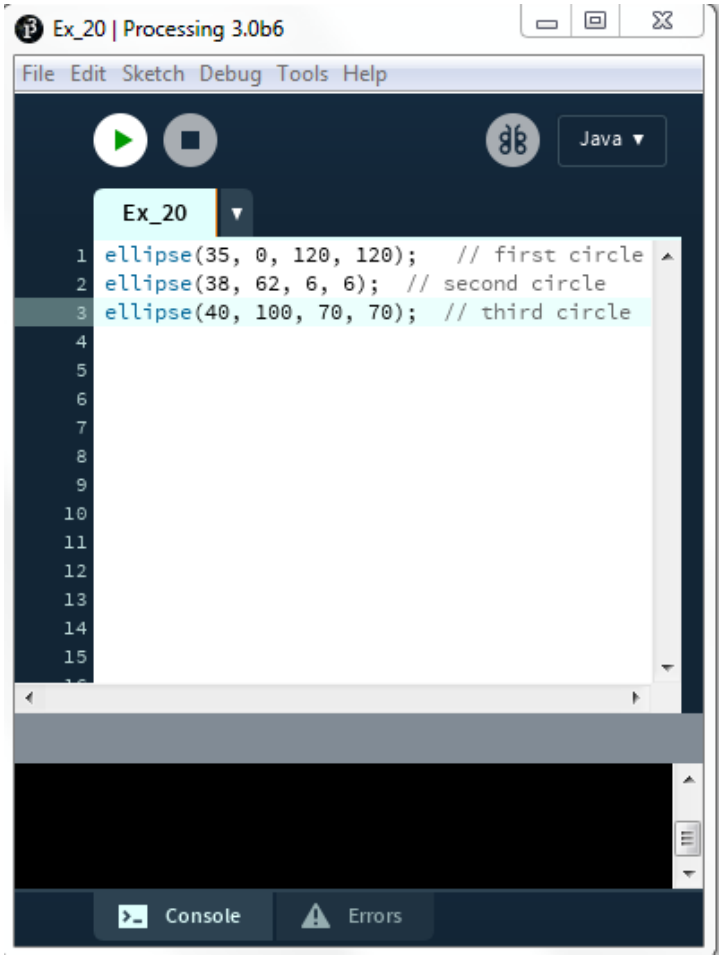
ellipse() - More Examples





ellipse() - More Examples





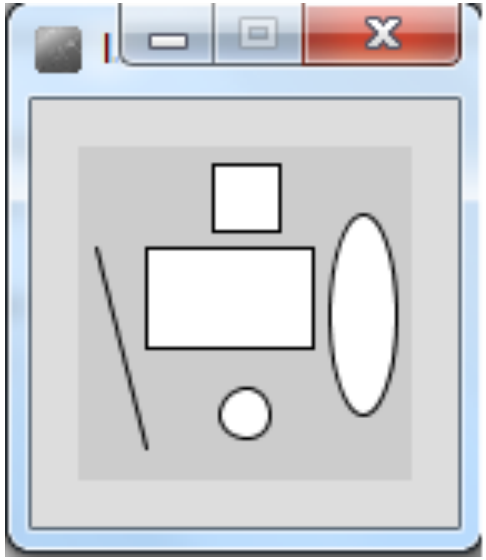
What's wrong with the ellipse here??

Topics List

- Coordinate System in Computing.
- Functions in Processing.
- Basic Shapes.
- Formatting the display window.
- Syntax Errors.
- Logic Errors.

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

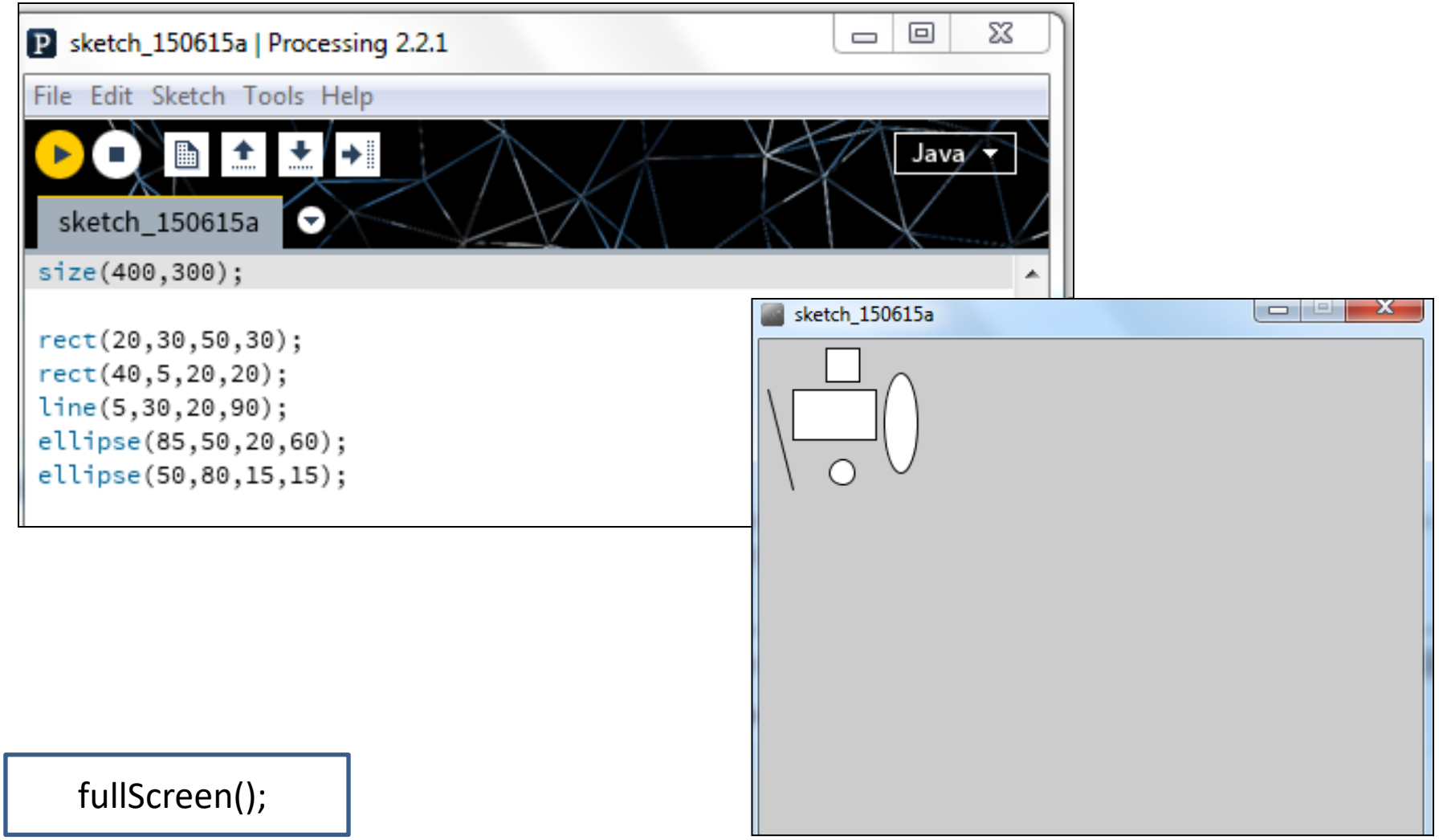
- We can change the size of the display window by calling the size function.
- When you use the size function in static drawings, it **HAS TO BE** the first line of code in your sketchbook.

size(w, h)

w = width of the display window

h = height of the display window

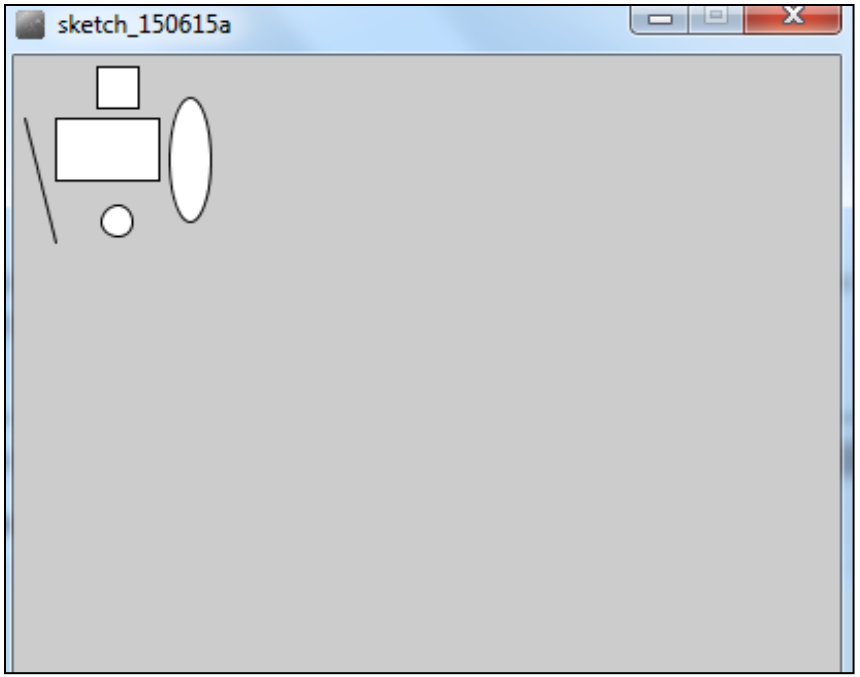
size()



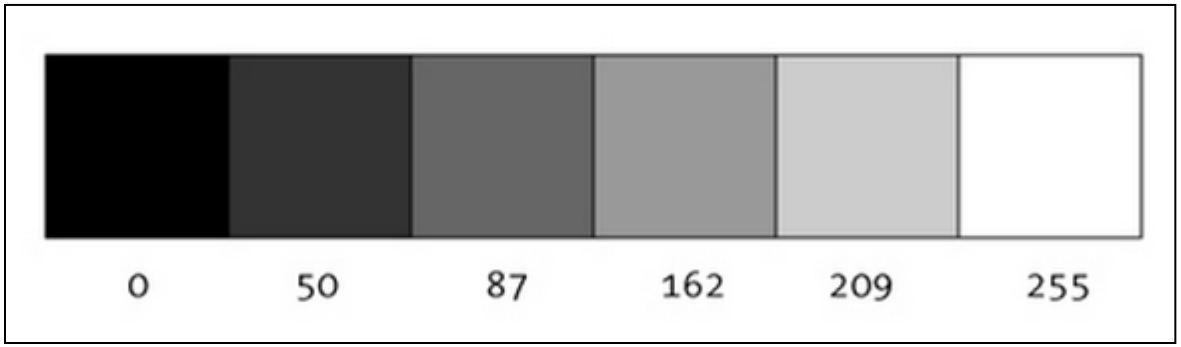
fullScreen();

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.



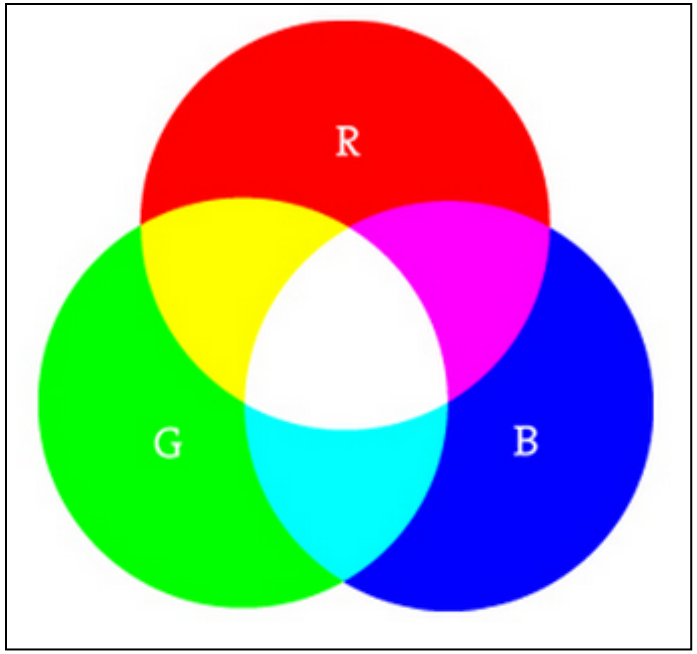
A Note on Colour First…Grayscale



"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

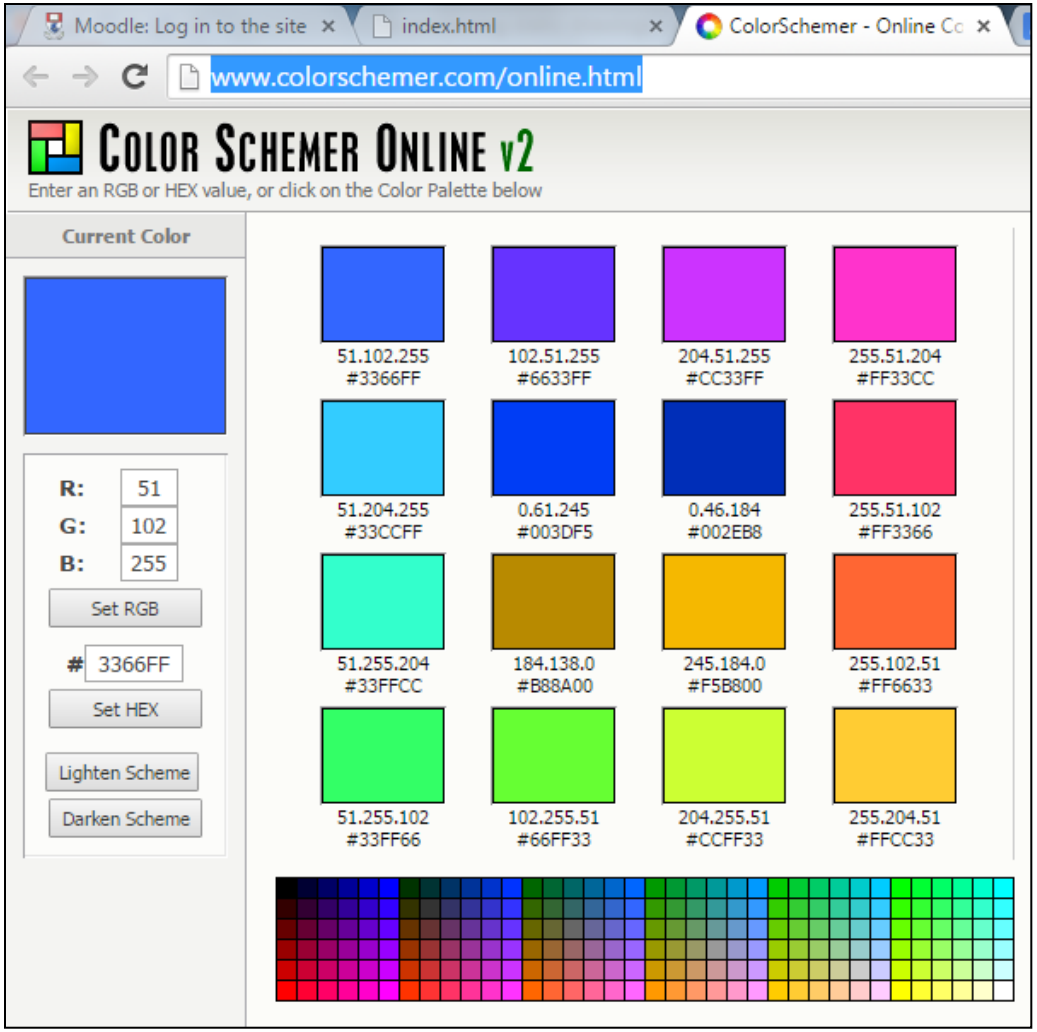


"As with grayscale, the individual color elements are expressed as ranges from 0 (none of that color) to 255 (as much as possible), and they are listed in the order R, G, and B."

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)

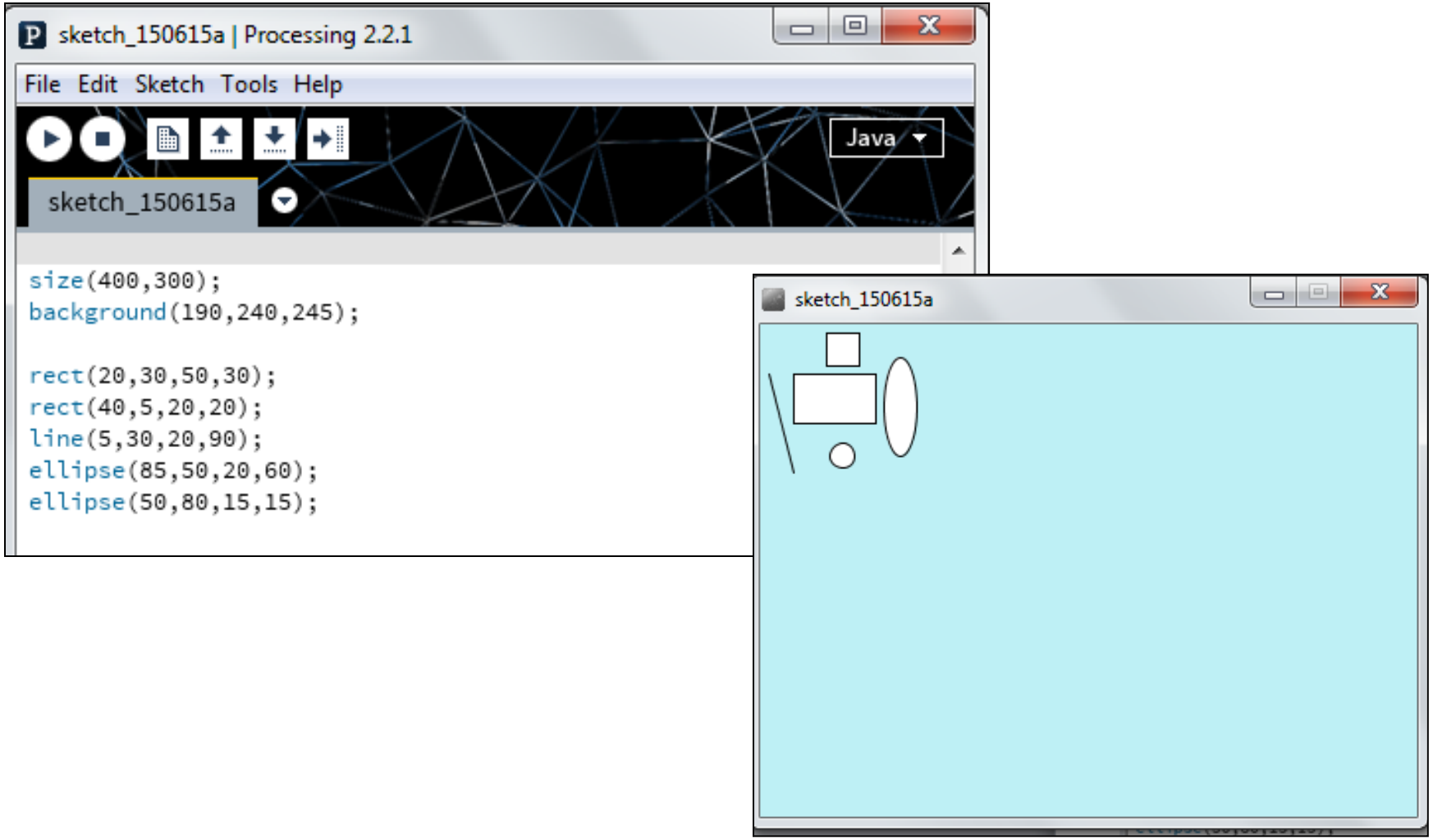
g = green colour (a number between 0 and 255 inclusive)

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

background()

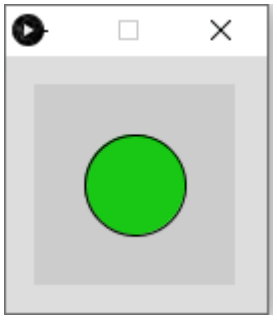

sketch_150615a | Processing 2.2.1
File Edit Sketch Tools Help
Run
Java
sketch_150615a
size(400,300);
background(250);
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
ellipse(85,50,20,60);
ellipse(50,80,15,15);
sketch_150615a

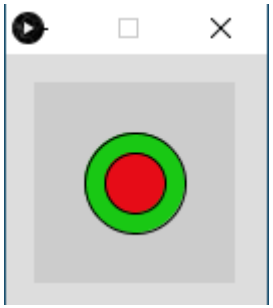
background()


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

Order

- Order of drawing is important





Topics List

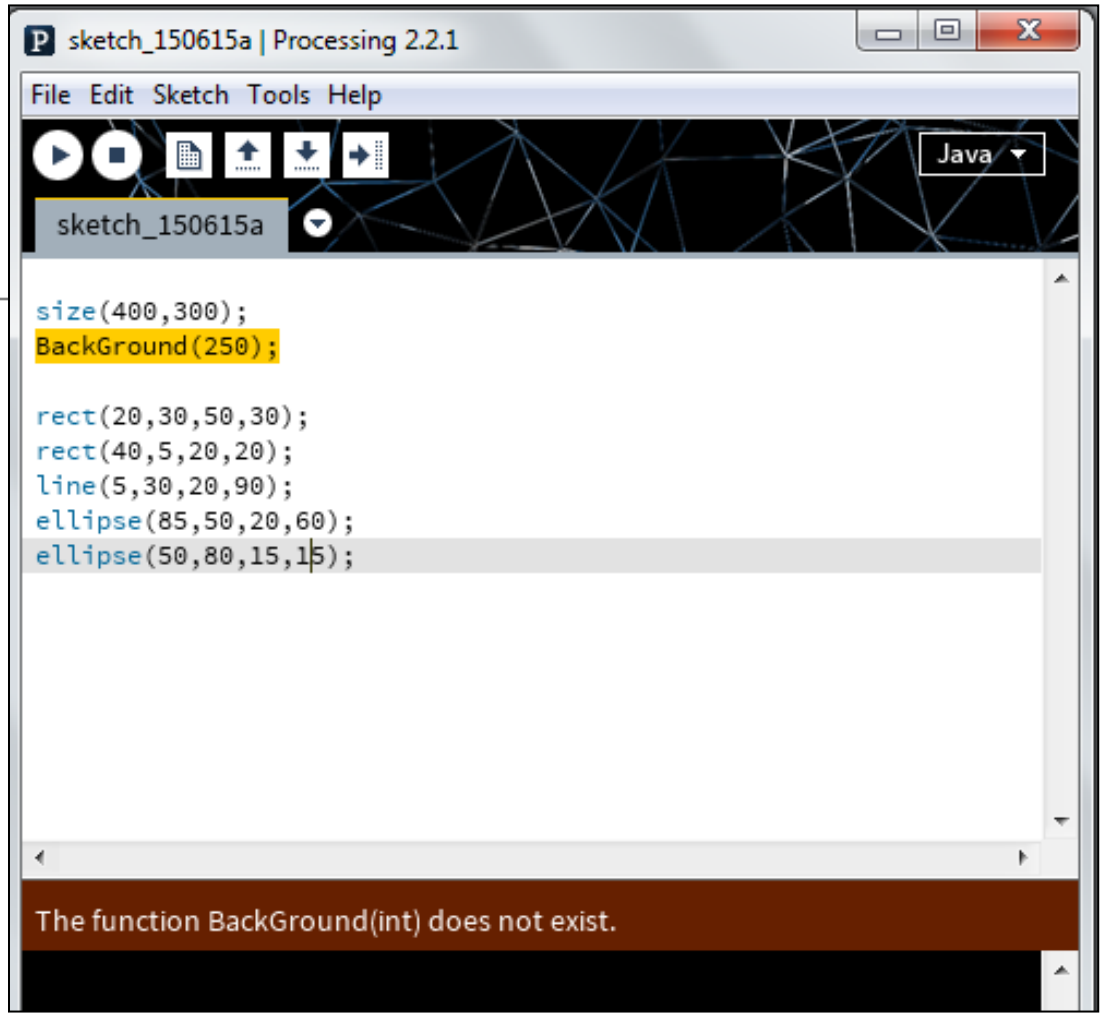
- Coordinate System in Computing.
- Functions in Processing.
- Basic Shapes.
- Formatting the display window.
- Syntax Errors.
- Logic Errors.

Syntax and Syntax Errors

- You will have seen the term Syntax mentioned above.
- Syntax are the rules you must follow when writing well-formed statements in a programming language.
- When you don’t follow the rules, Processing will not run your code; instead you will get an error.
- Some syntax error examples are on the upcoming slides.

Syntax Errors

The spelling of the background function must be identical to the spelling below (case sensitive!).



background(r, g, b)

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

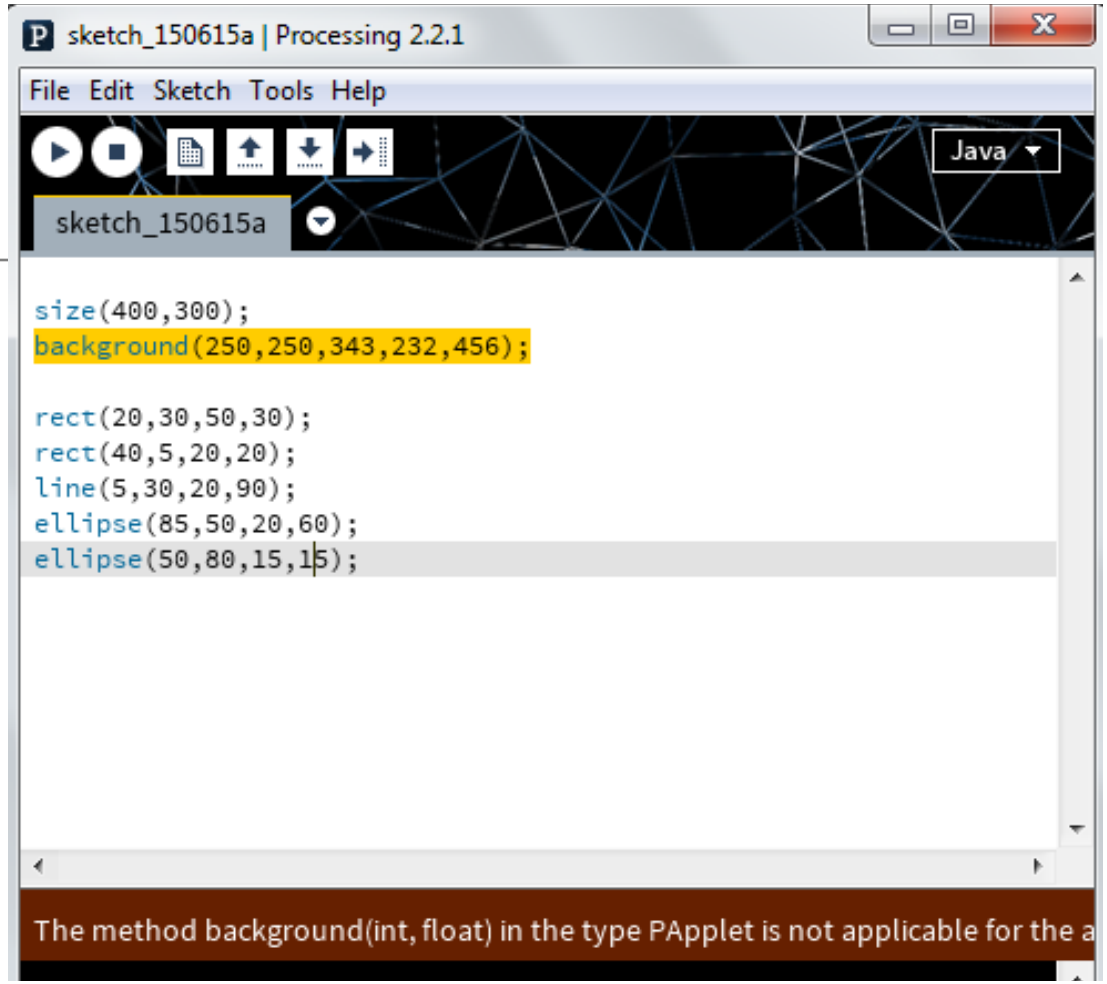
g = green colour (a number between 0 and 255 inclusive)

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

Syntax Errors

The background function has too many arguments passed to it i.e.

- *RGB version is defined with 3 parameters.*
- *Grayscale version is defined with 1 parameter.*



background(r, g, b)

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

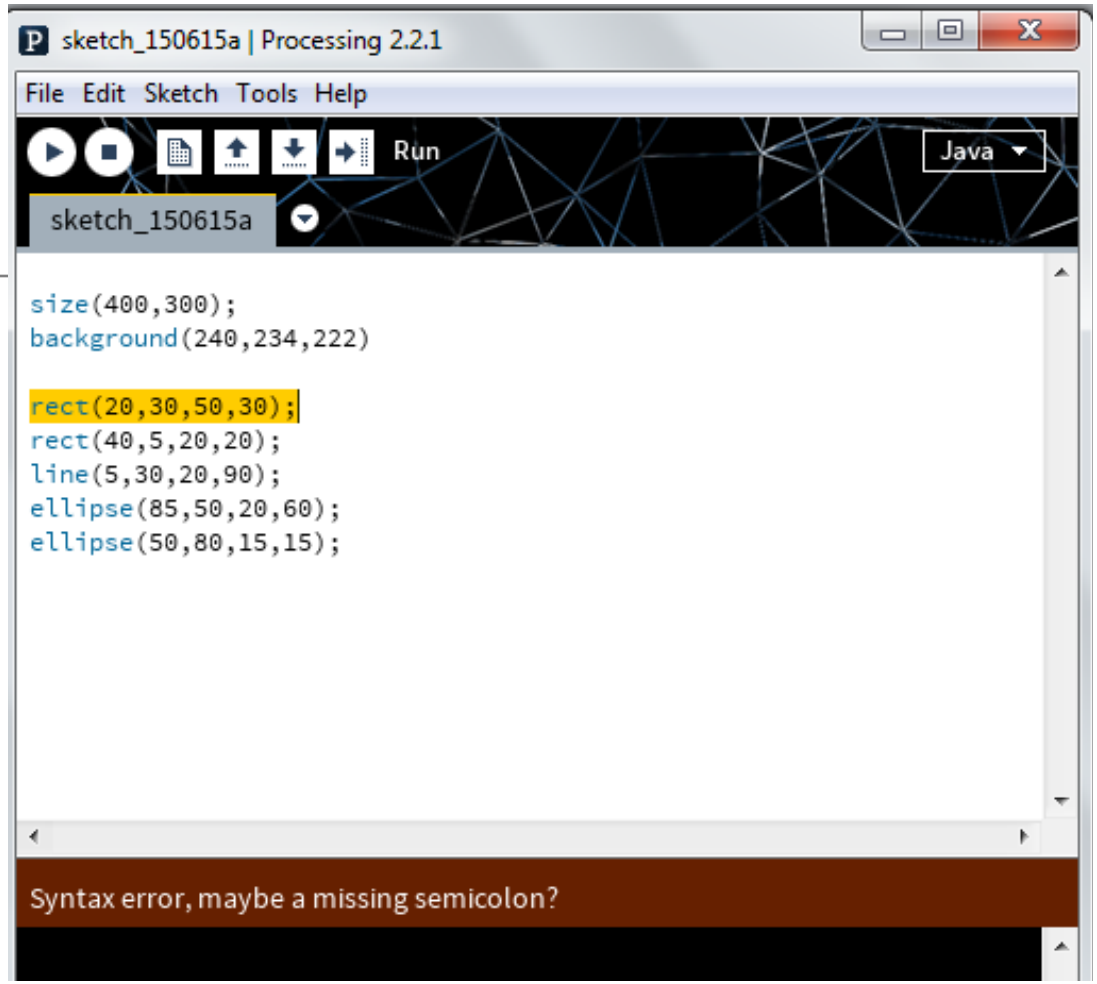
g = green colour (a number between 0 and 255 inclusive)

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

Syntax Errors

The semi-colon (;) is missing at the end of the statement.

Java needs a statement terminator for each line!



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)

Topics List

- Coordinate System in Computing.
- Functions in Processing.
- Basic Shapes.
- Formatting the display window.
- Syntax Errors.
- Logic Errors.

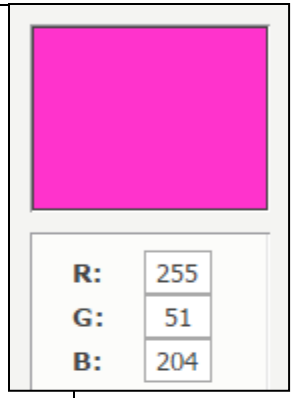
Logic Errors

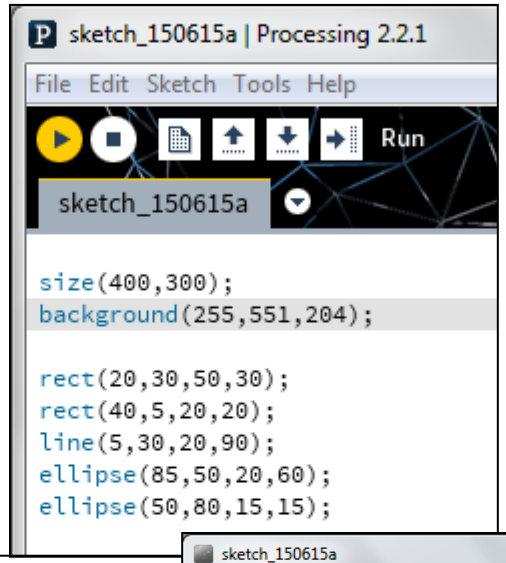
In computer programming, a **logic error** is a bug in a program that causes it to operate incorrectly, but not to terminate abnormally (or crash). A **logic error** produces unintended or undesired output or other behaviour, although it may not immediately be recognised as such.

Logic error - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Logic_error**

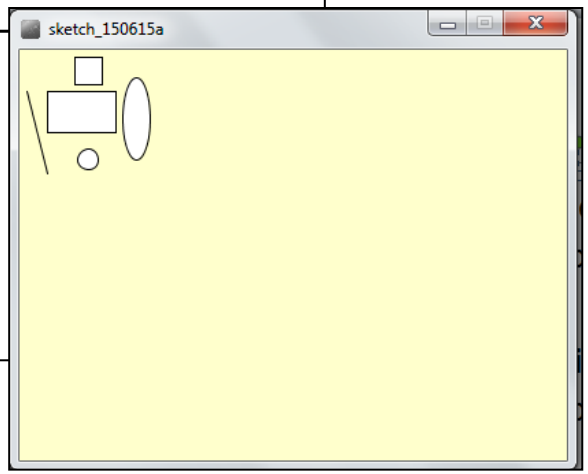
Logic Errors

Say we wanted a pink background for our display window.





- However, we incorrectly enter the **G** colour as 551 instead of 51.
- We now have a yellowish background.
- This is an example of a simple logic error.



Questions?

Produced by:

Dr. Siobhán Drohan
Mairead Meagher
Sinéad Walsh

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

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