Programming Fundamentals

Starting to Code in Processing

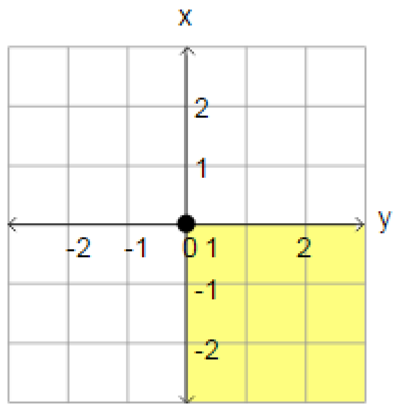
Produced by: Dr. Siobhán Drohan
Ms. Mairead Meagher

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

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

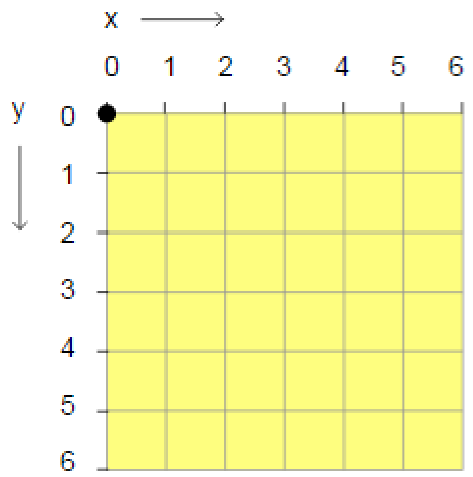
Coordinate System in Computing

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:



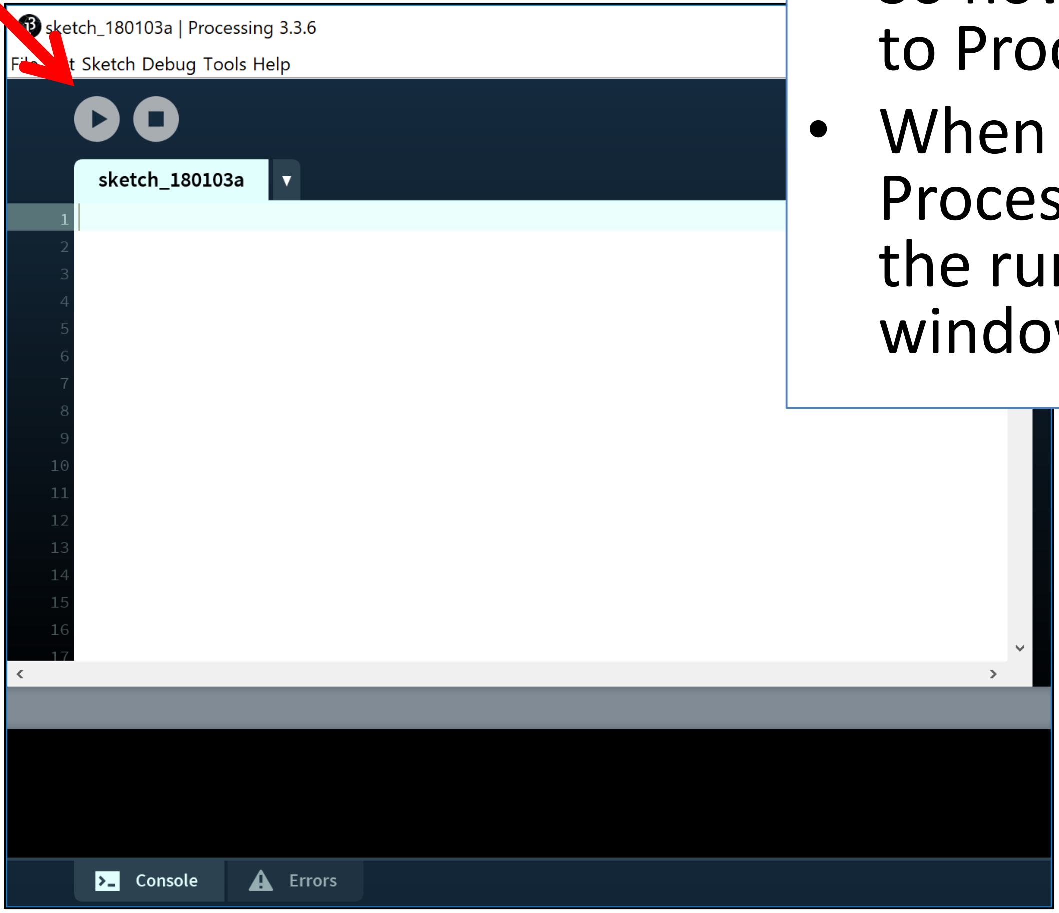
point (0,0) is in the top left hand corner. Each number is a pixel.

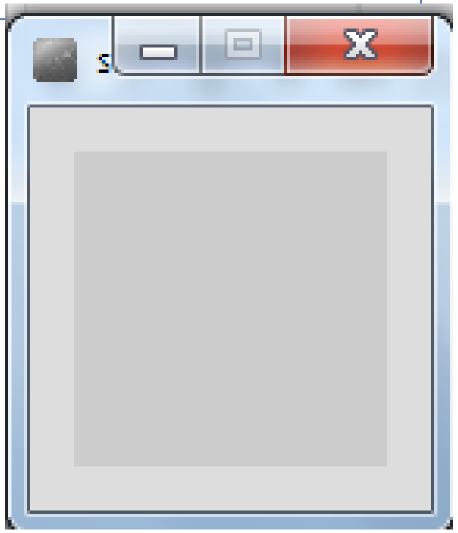
https://processing.org/

Coordinate System in Computing

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

Run button

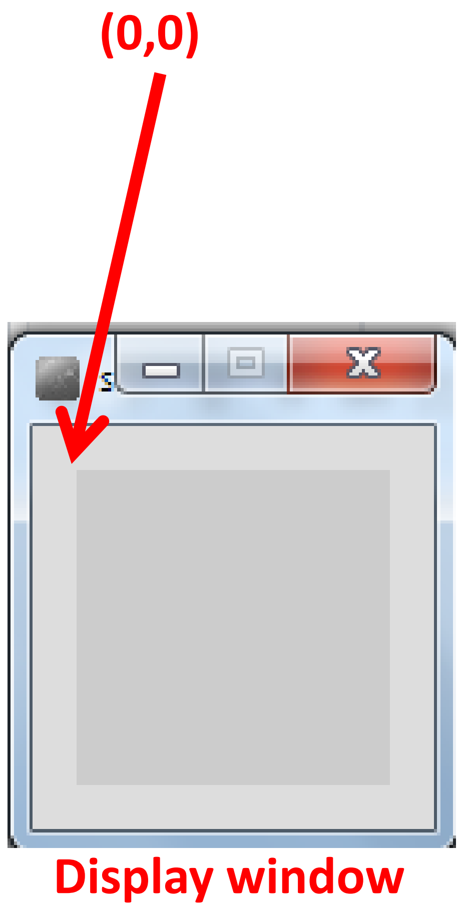

sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
sketch_180103a
Console
Errors



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).

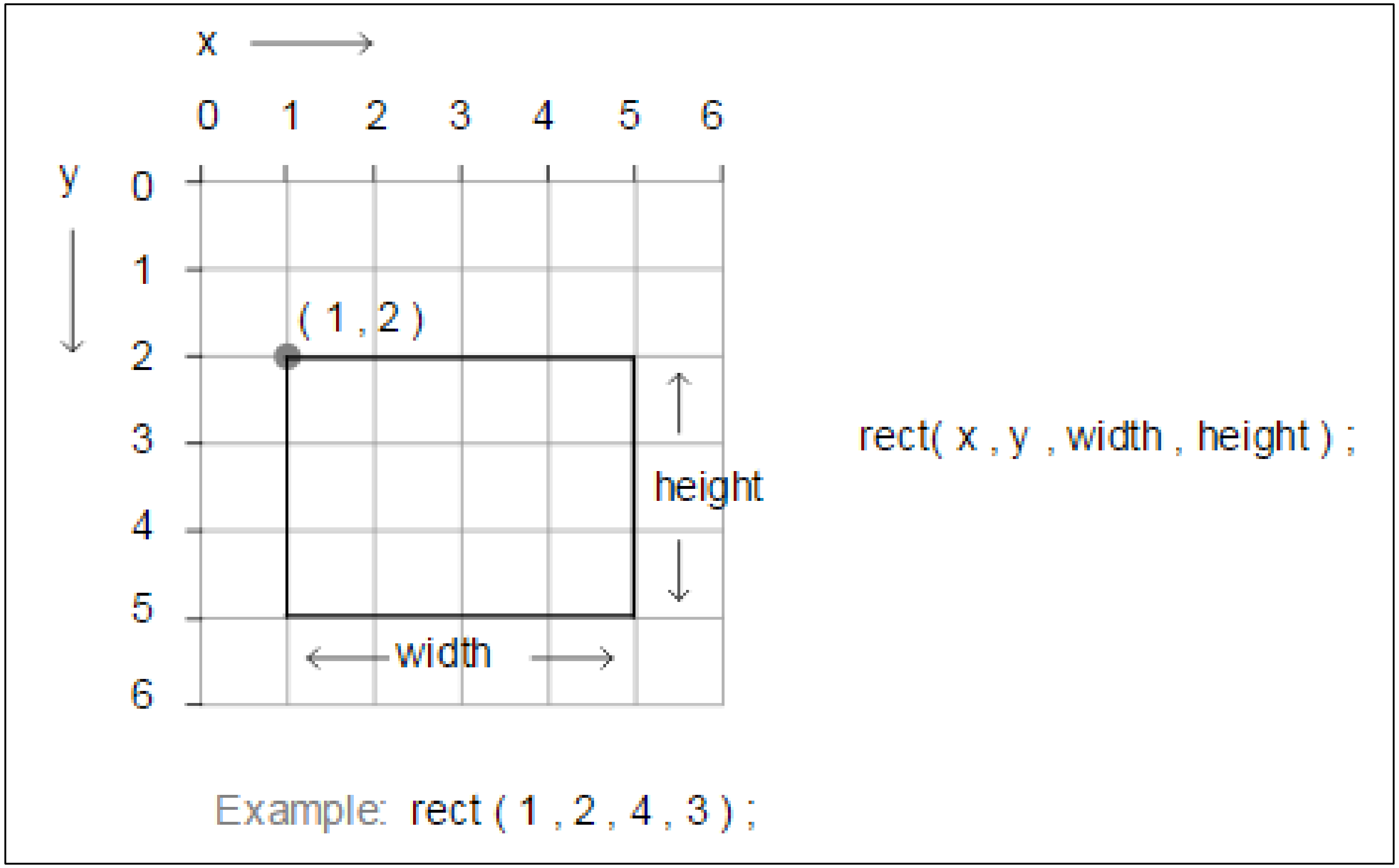


Drawing Shapes

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.

rect()

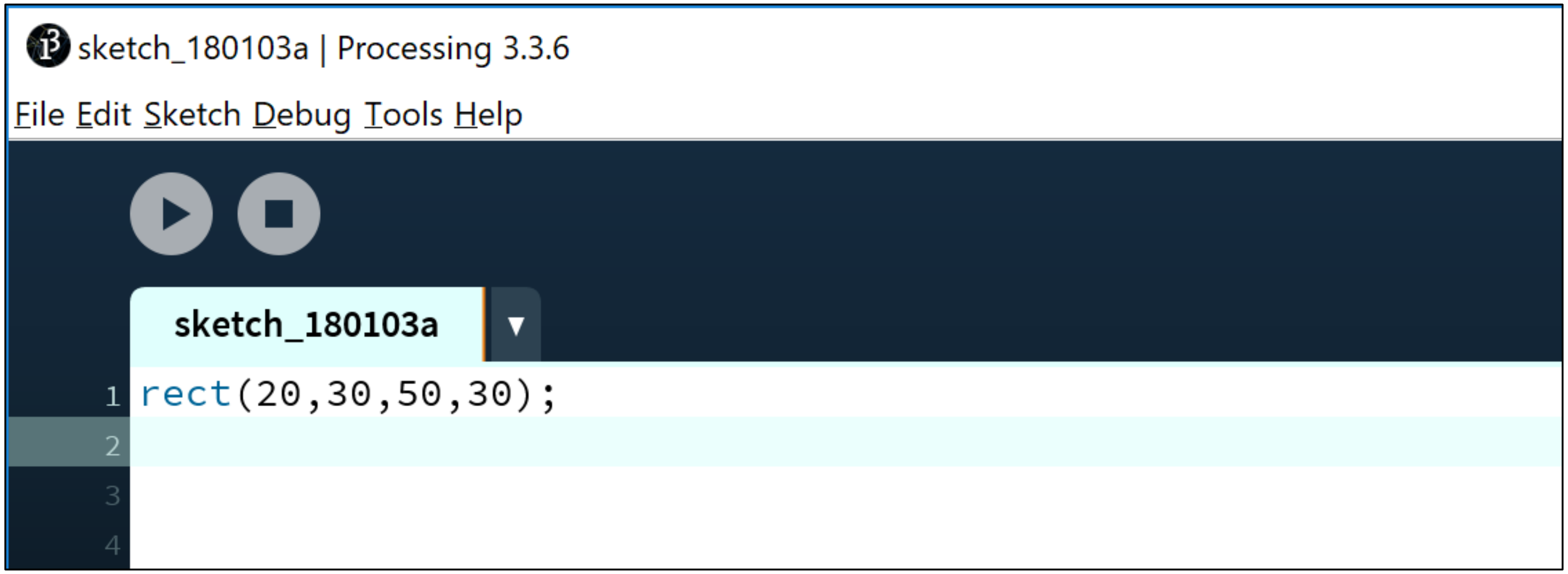


rect(x , y , width , height) ;

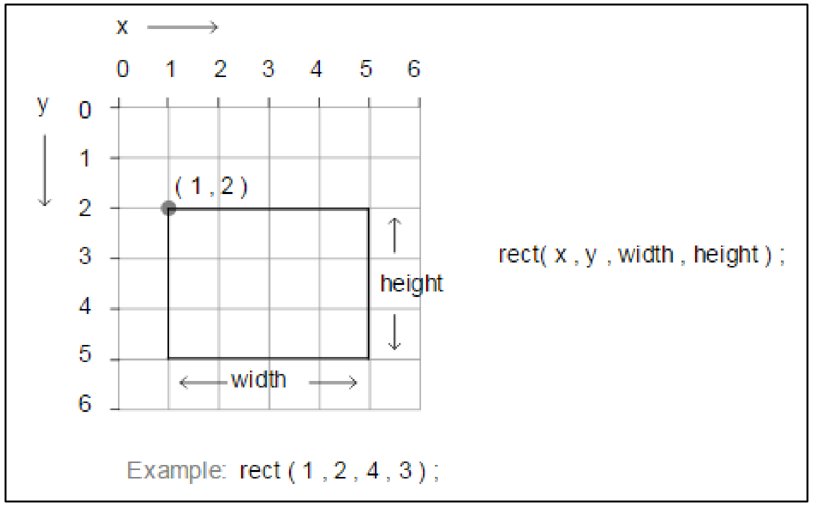
Example: rect (1 , 2 , 4 , 3) ;

https://processing.org/

rect() – drawing a rectangle


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
sketch_180103a

```
rect(20,30,50,30);
```

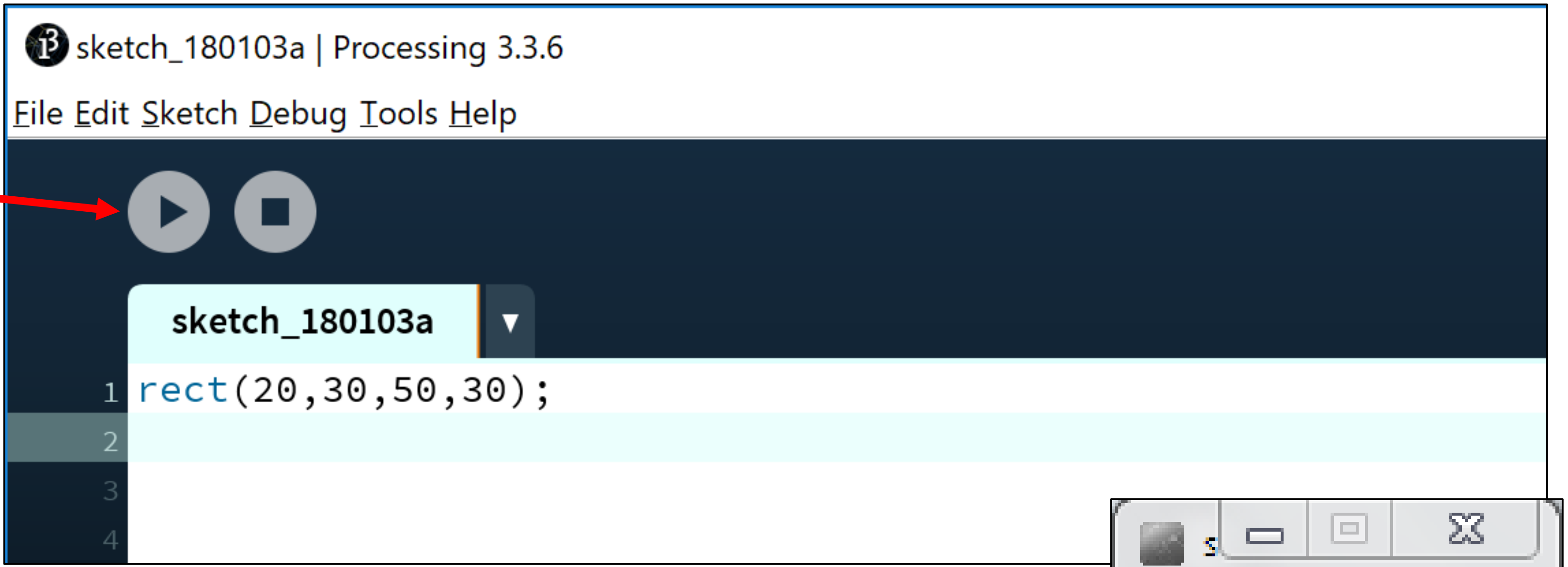


rect(x , y , width , height) ;

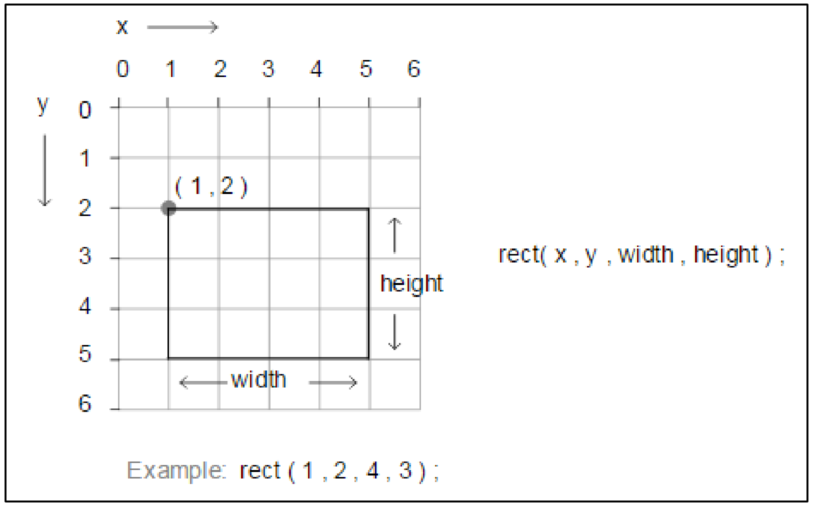
Example: rect (1 , 2 , 4 , 3) ;

rect() – drawing a rectangle

Click to Run

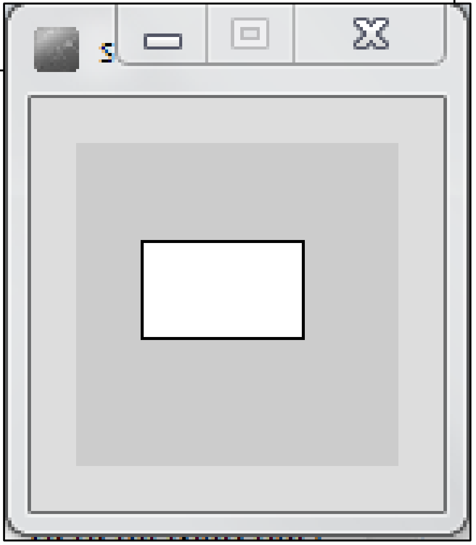

sketch_180103a | Processing 3.3.6

```
rect(20,30,50,30);
```

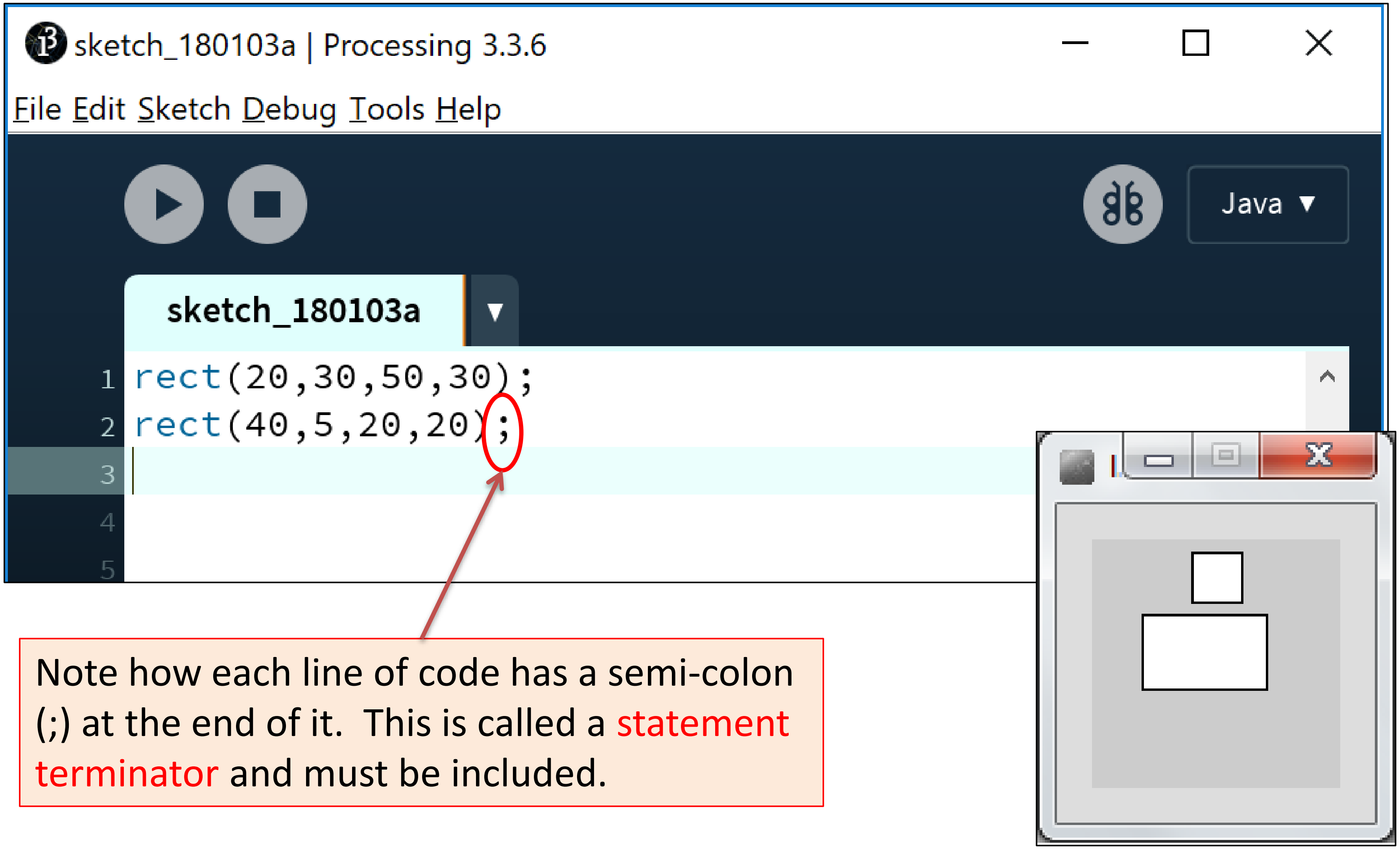


rect(x , y , width , height) ;

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



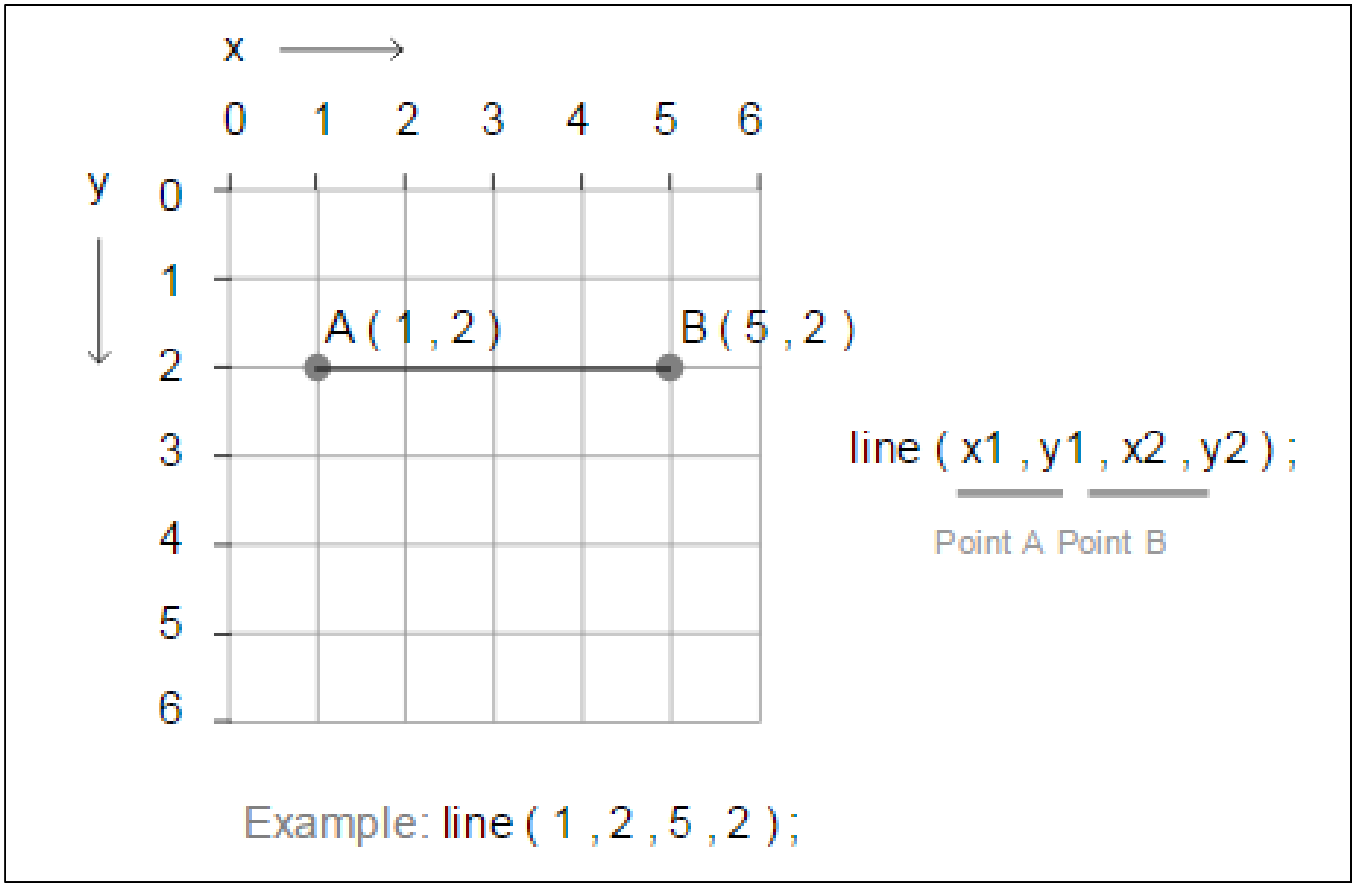
rect() – drawing a square


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a

```
rect(20,30,50,30);
rect(40,5,20,20);
```

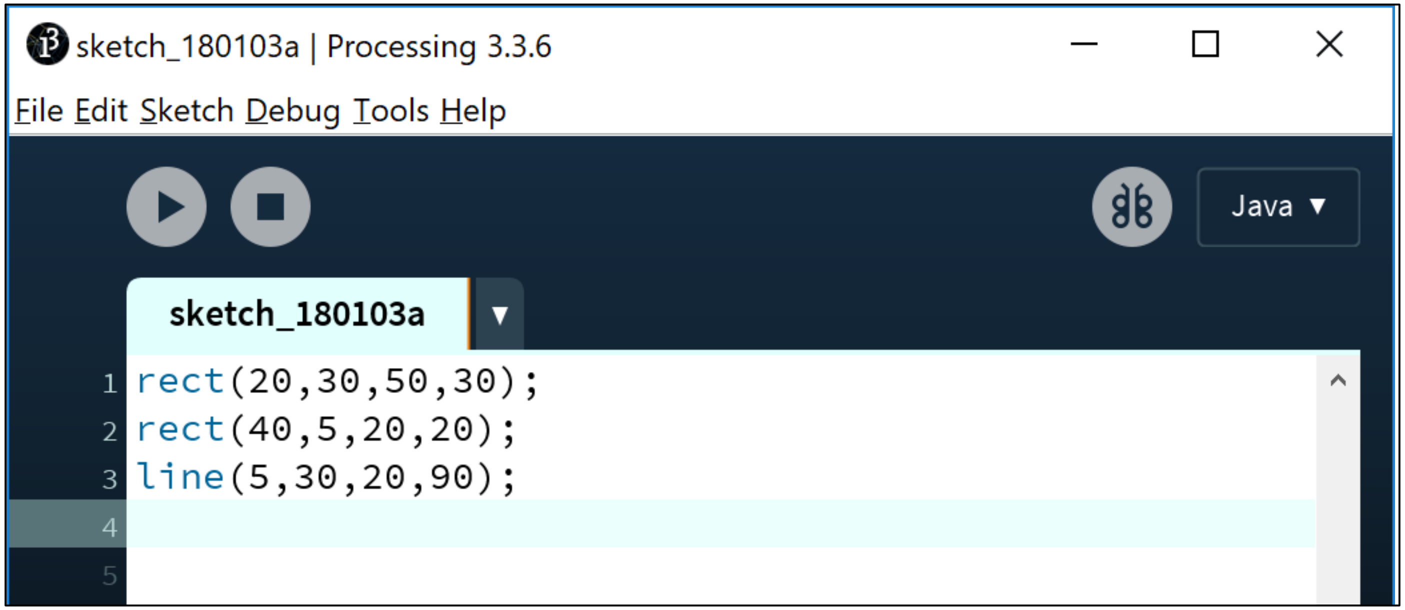
Note how each line of code has a semi-colon (;) at the end of it. This is called a statement terminator and must be included.

line()

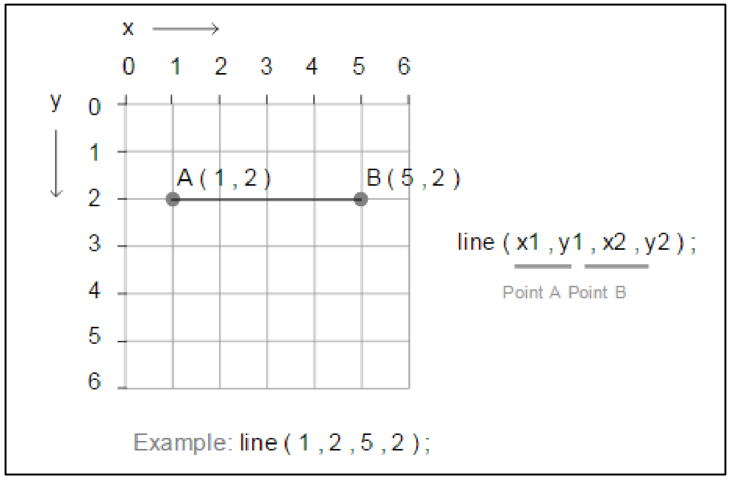


https://processing.org/

line () – drawing a line

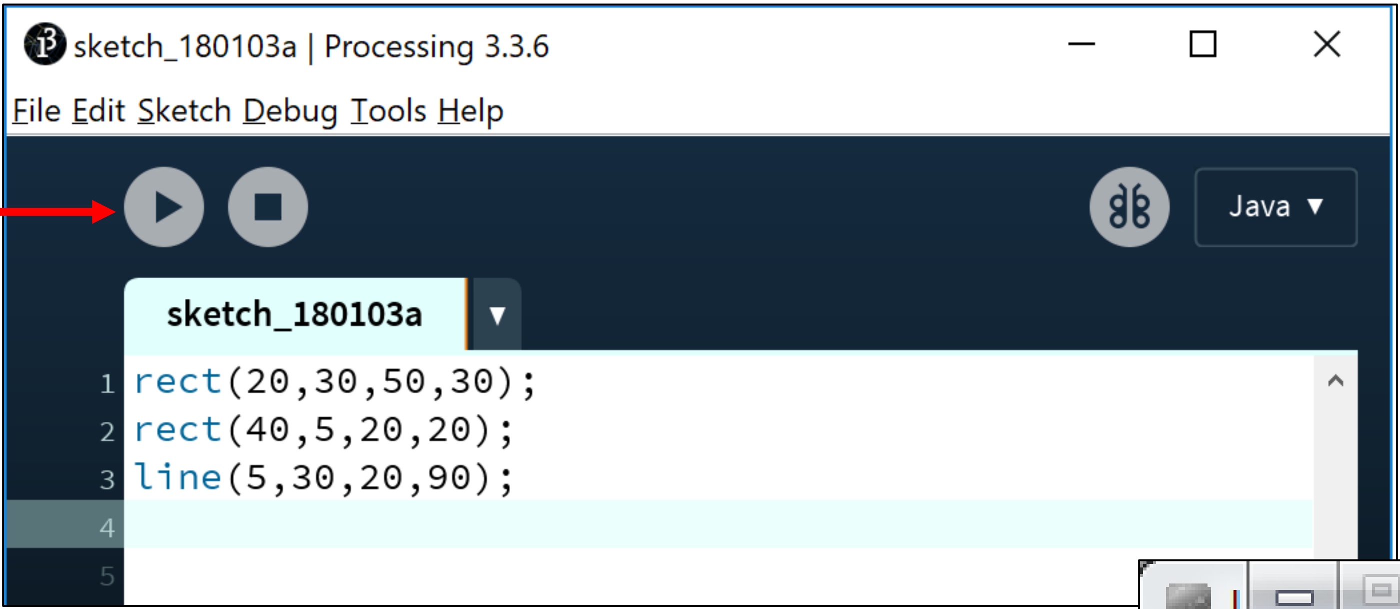

sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a

```
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
```

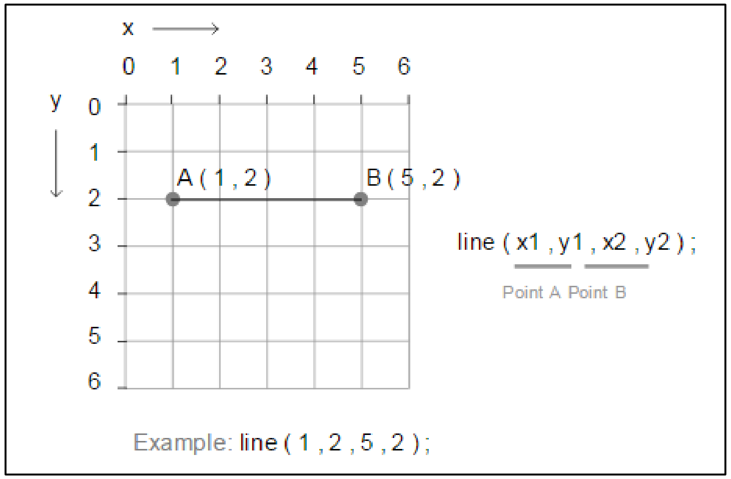

x
y
0 1 2 3 4 5 6
A (1 , 2)
B (5 , 2)
line (x1 , y1 , x2 , y2) ;
Point A Point B
Example: line (1 , 2 , 5 , 2) ;

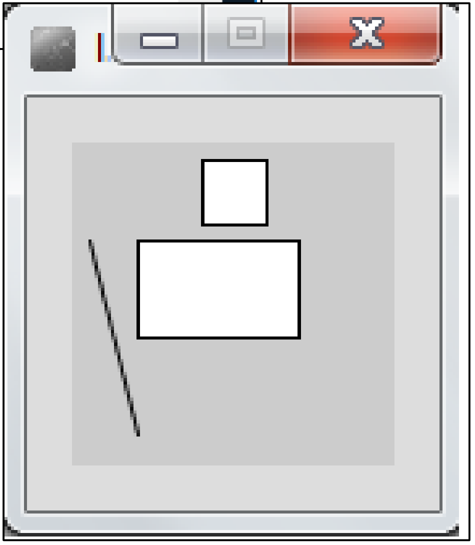
line () – drawing a line

Click to Run

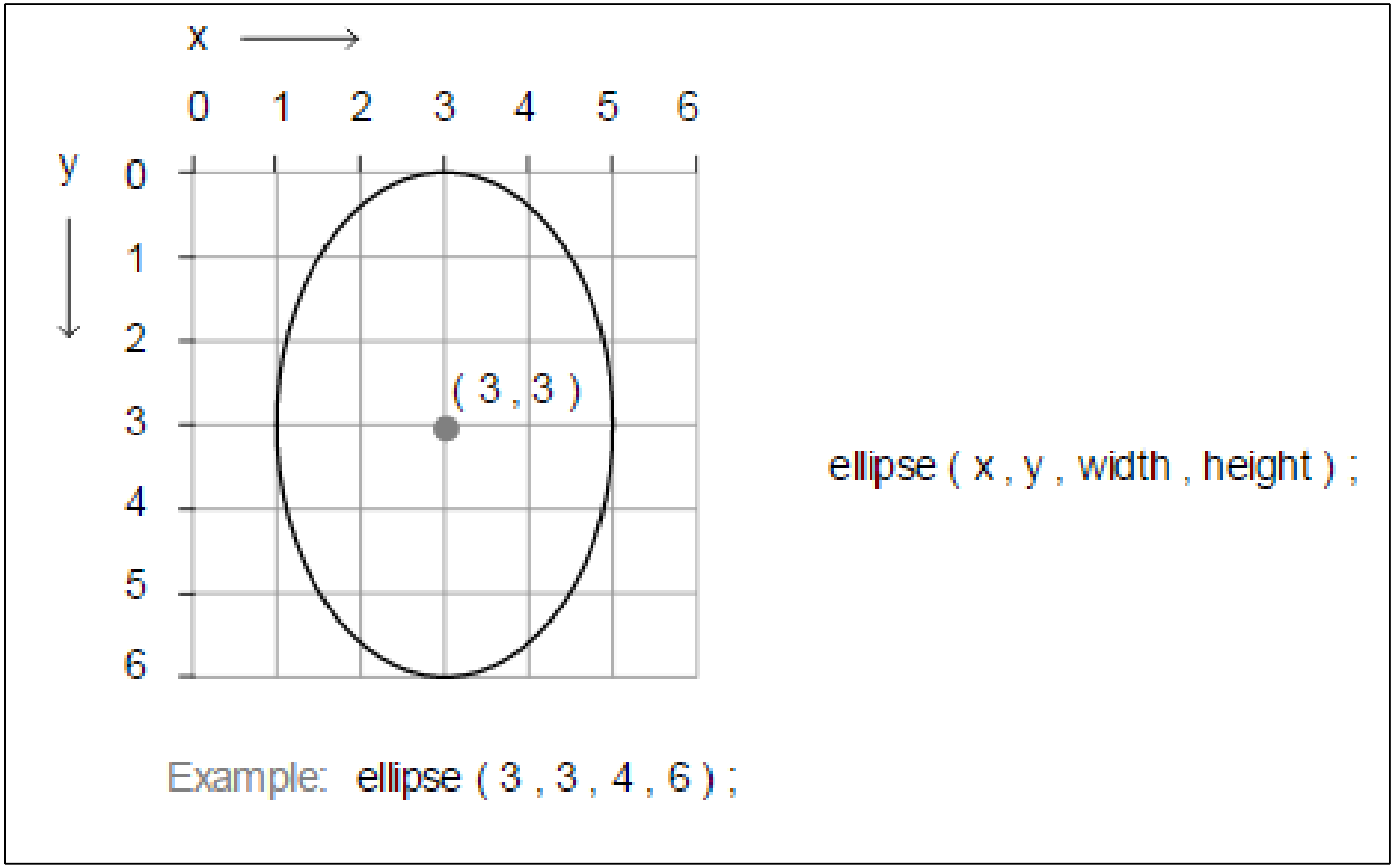

sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a

```
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
```





ellipse()

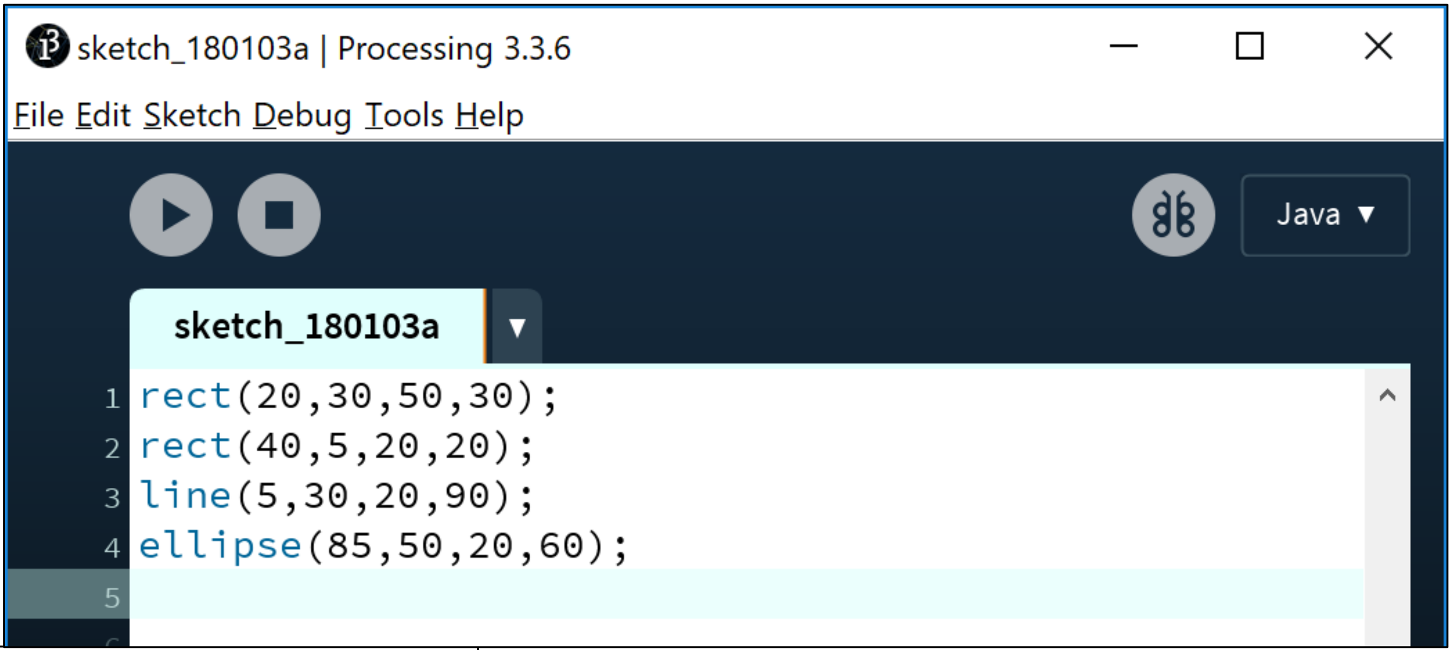


ellipse (x , y , width , height) ;

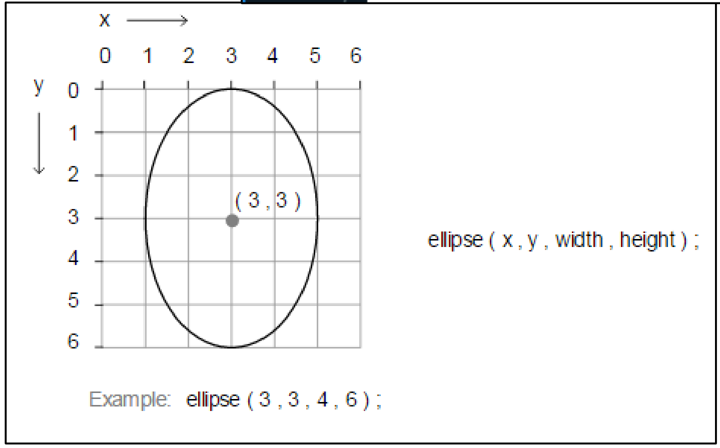
Example: ellipse (3 , 3 , 4 , 6) ;

https://processing.org/

ellipse() – drawing an oval


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a

```
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
ellipse(85,50,20,60);
```

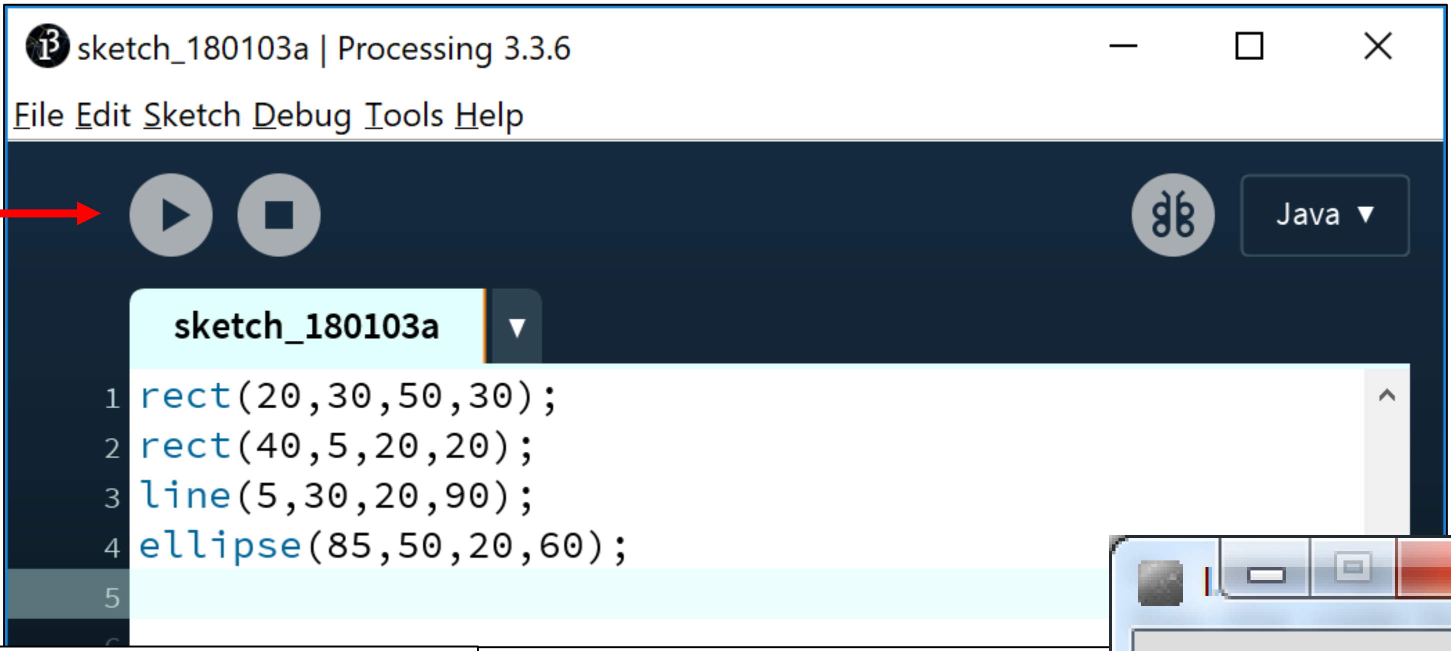


ellipse (x , y , width , height) ;

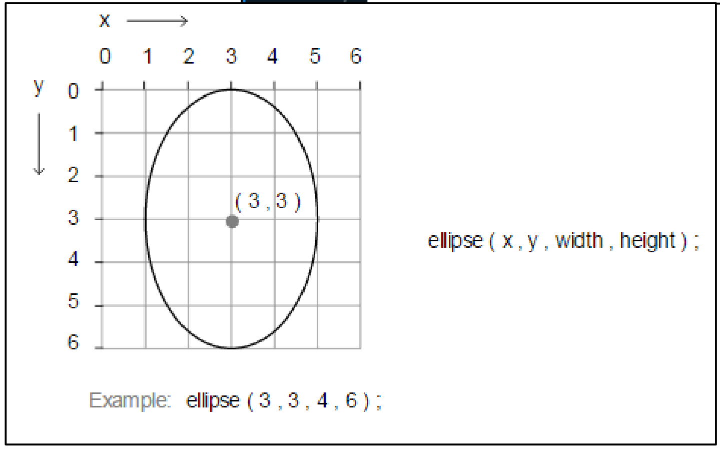
Example: ellipse (3 , 3 , 4 , 6) ;

ellipse() – drawing an oval

Click to Run

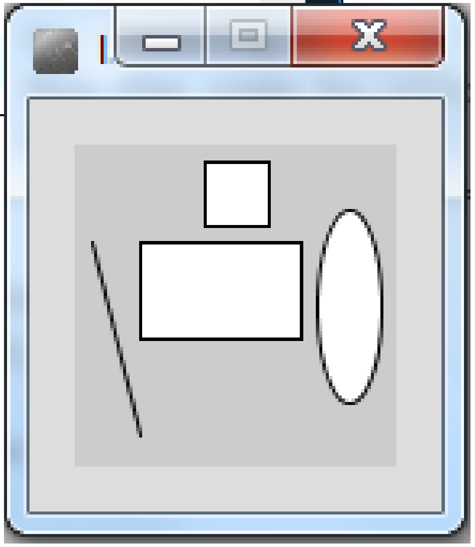

sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a

```
rect(20,30,50,30);
rect(40,5,20,20);
line(5,30,20,90);
ellipse(85,50,20,60);
```

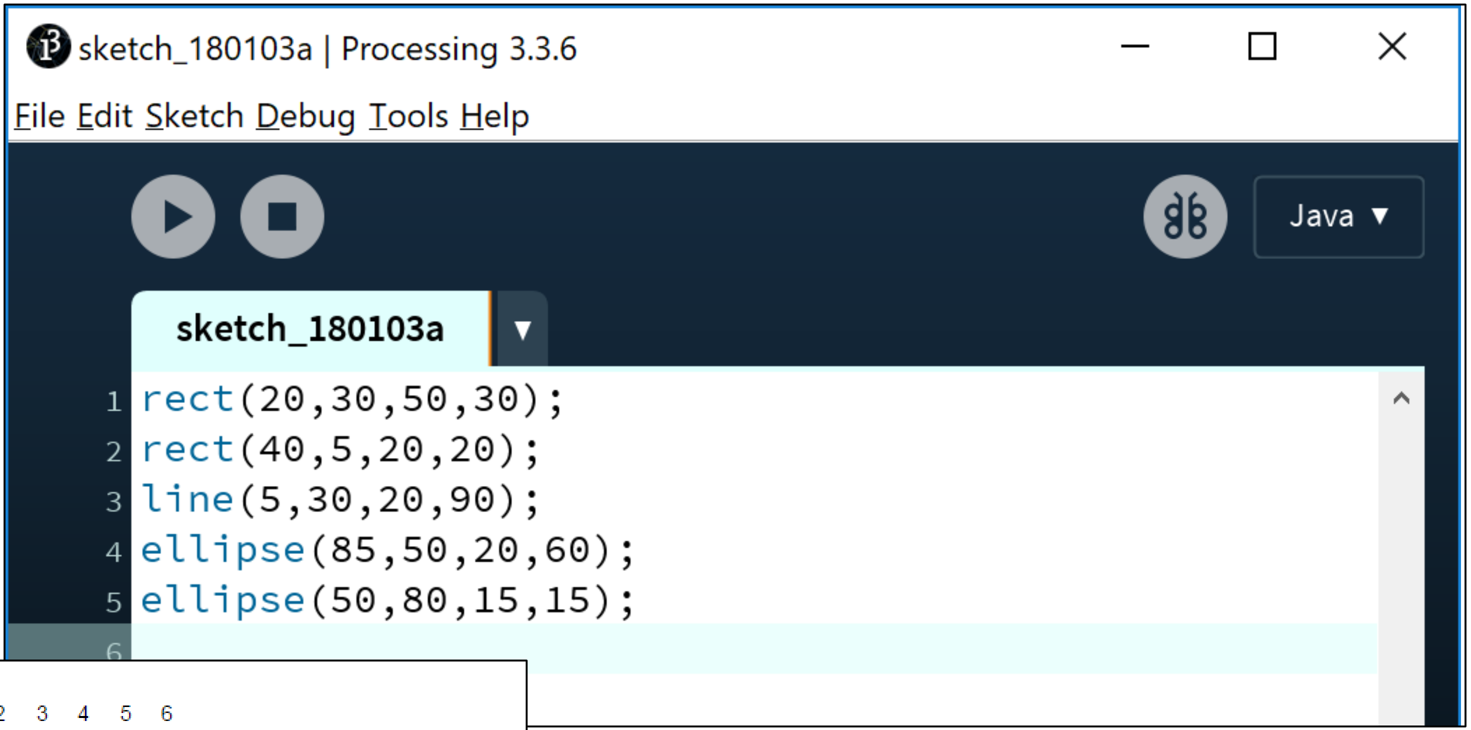


ellipse (x , y , width , height) ;

Example: ellipse (3 , 3 , 4 , 6) ;

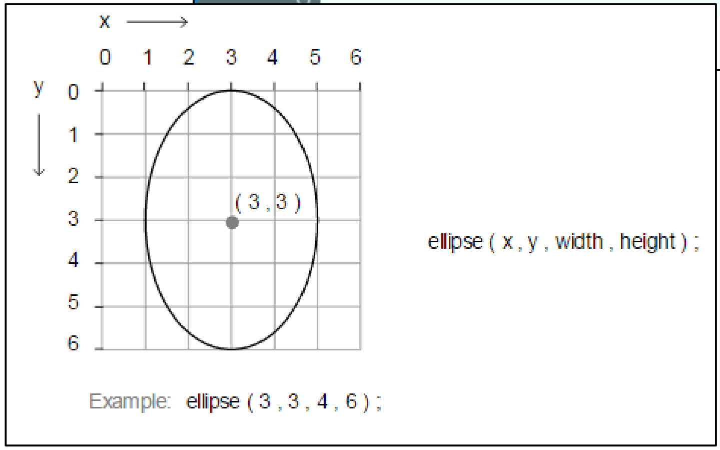


ellipse() – drawing a circle



sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a

```
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);
```

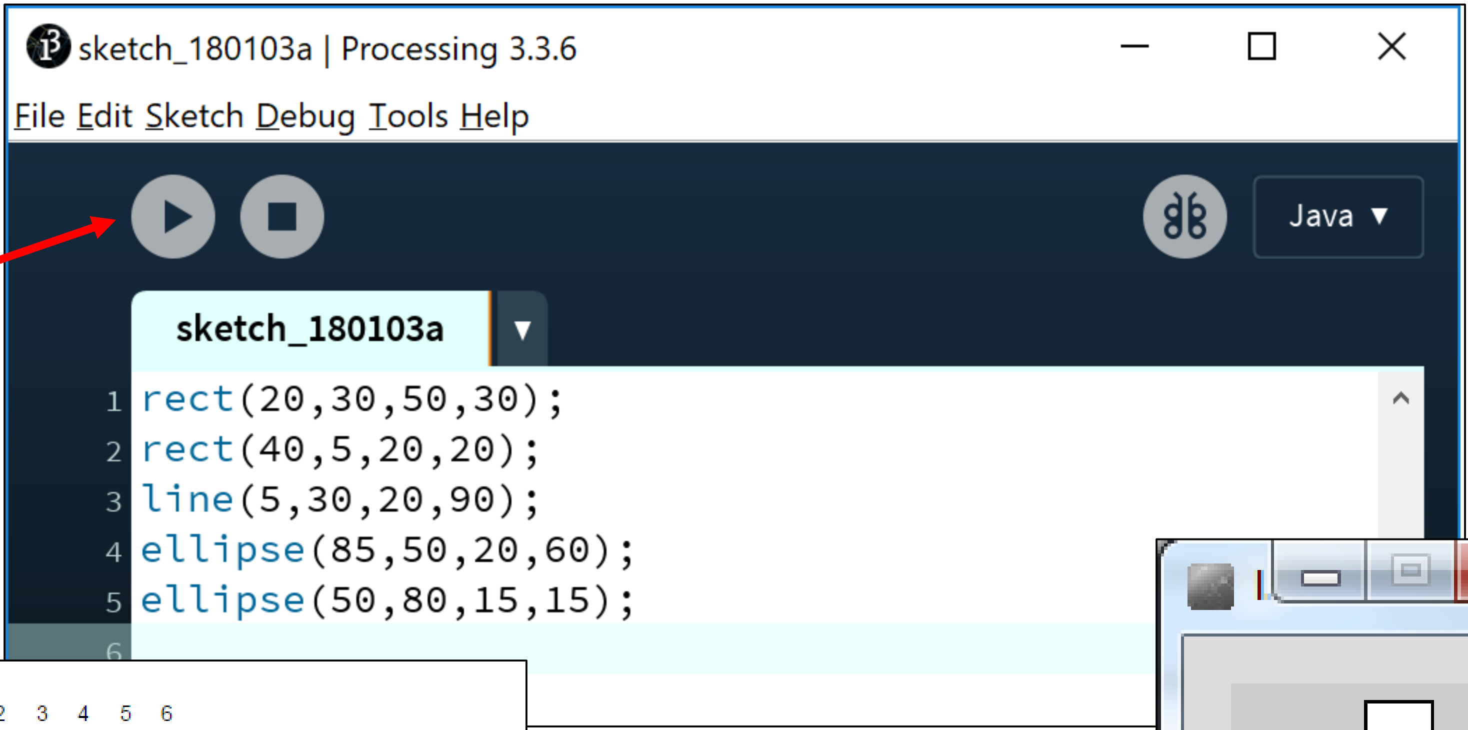


ellipse (x , y , width , height) ;

Example: ellipse (3 , 3 , 4 , 6) ;

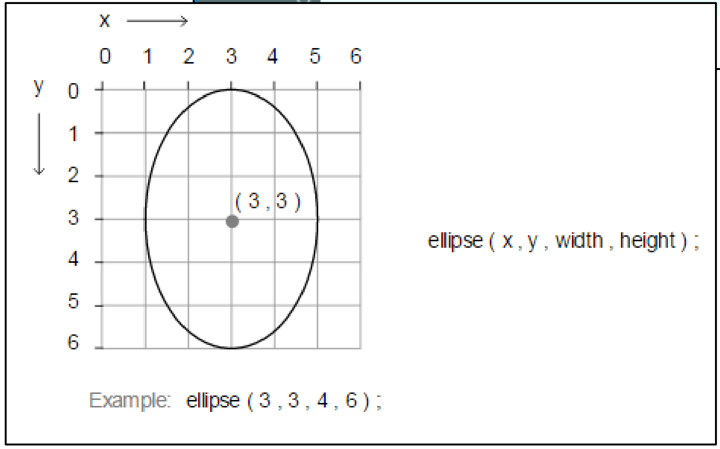
ellipse() – drawing a circle

Click to Run



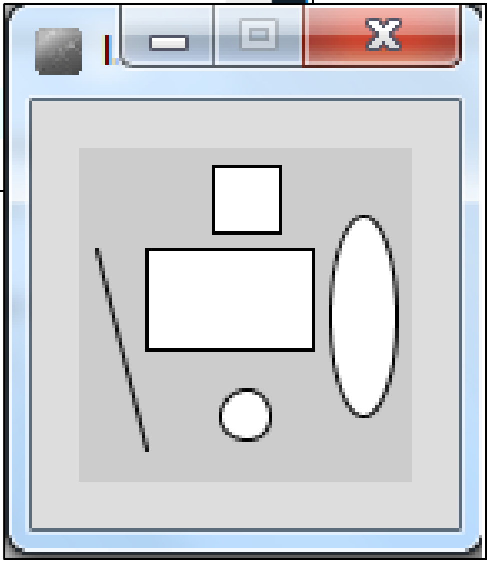
sketch_180103a | Processing 3.3.6

```
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 (x , y , width , height) ;

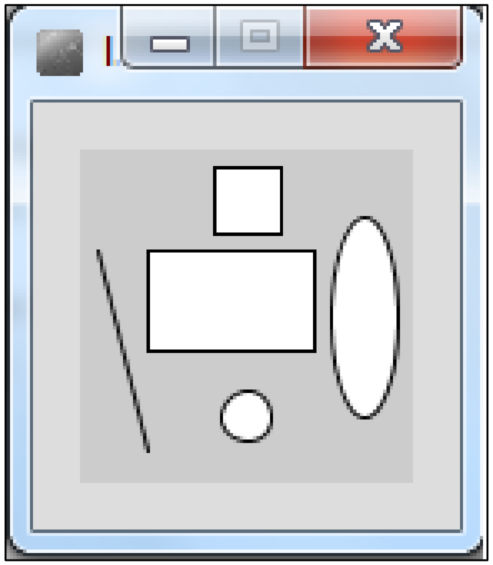
Example: ellipse (3 , 3 , 4 , 6) ;



Formatting the Display Window

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.

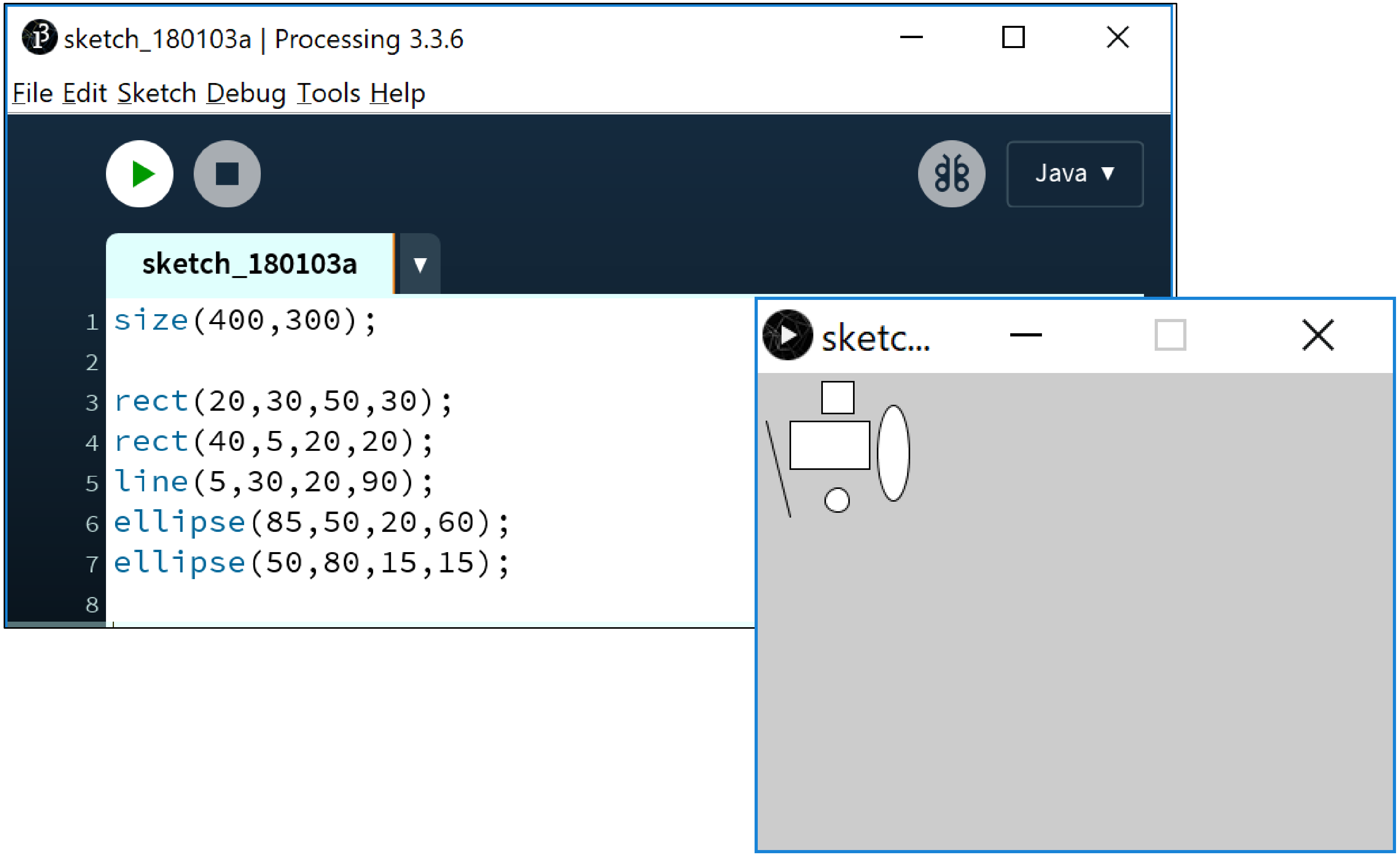


Formatting the display window

- 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
```

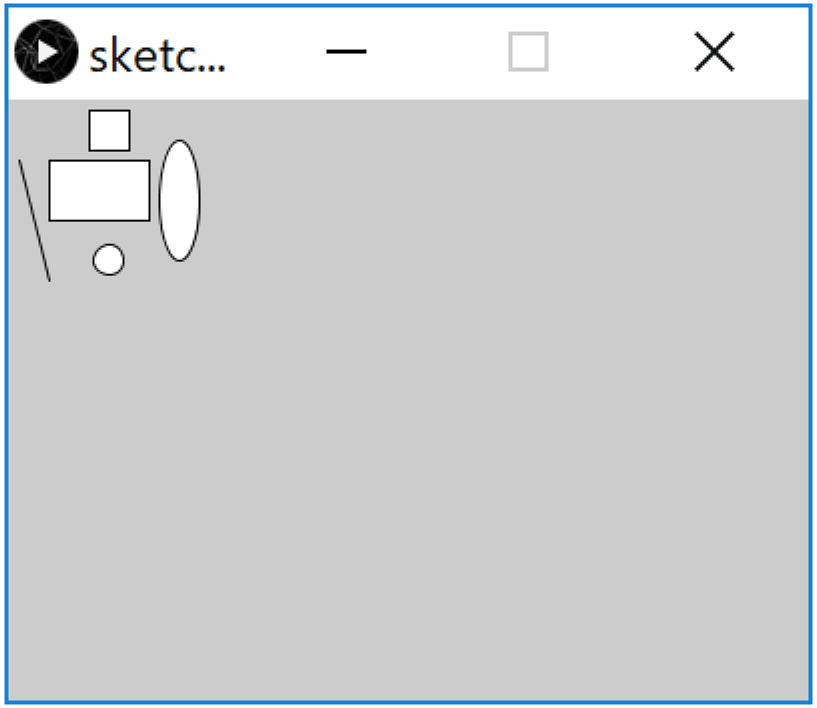
size()


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a
sketc...

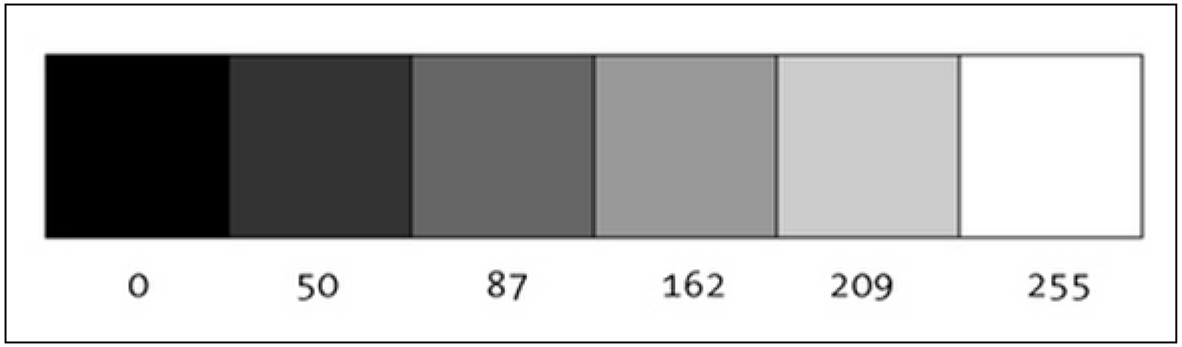
```
size(400,300);

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);

```

Formatting the display window

- Our display window looks less cramped now.
- But maybe we want to change the default gray colour?
- We could use the background function to set the colour to something else.



A note on colour first…Grayscale



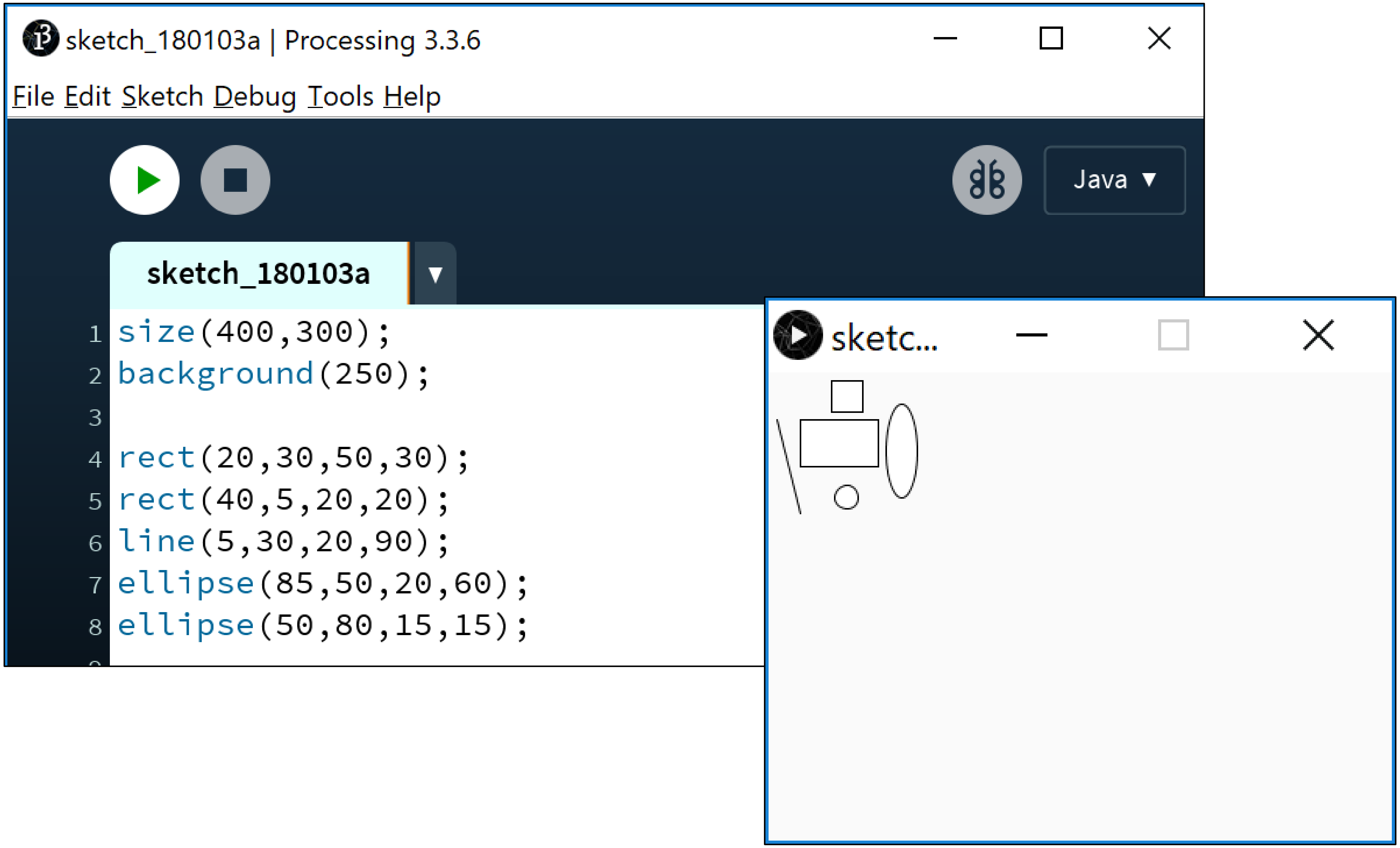
"0 means black, 255 means white. In between, every other number - 50, 87, 162, 209, and so on - is a shade of gray ranging from black to white."

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

background() - syntax

```
background(grayscale)

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

background()


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a
sketc...

```
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);
```

Flow of Control

Problem Solving

Programming IS problem solving.

Flow of Control in a Program

- Each program you write will typically have:

Sequence	Things that will be done in a particular order
Selection	Things that will be done conditionally
Iteration	Things that will be done repetitively

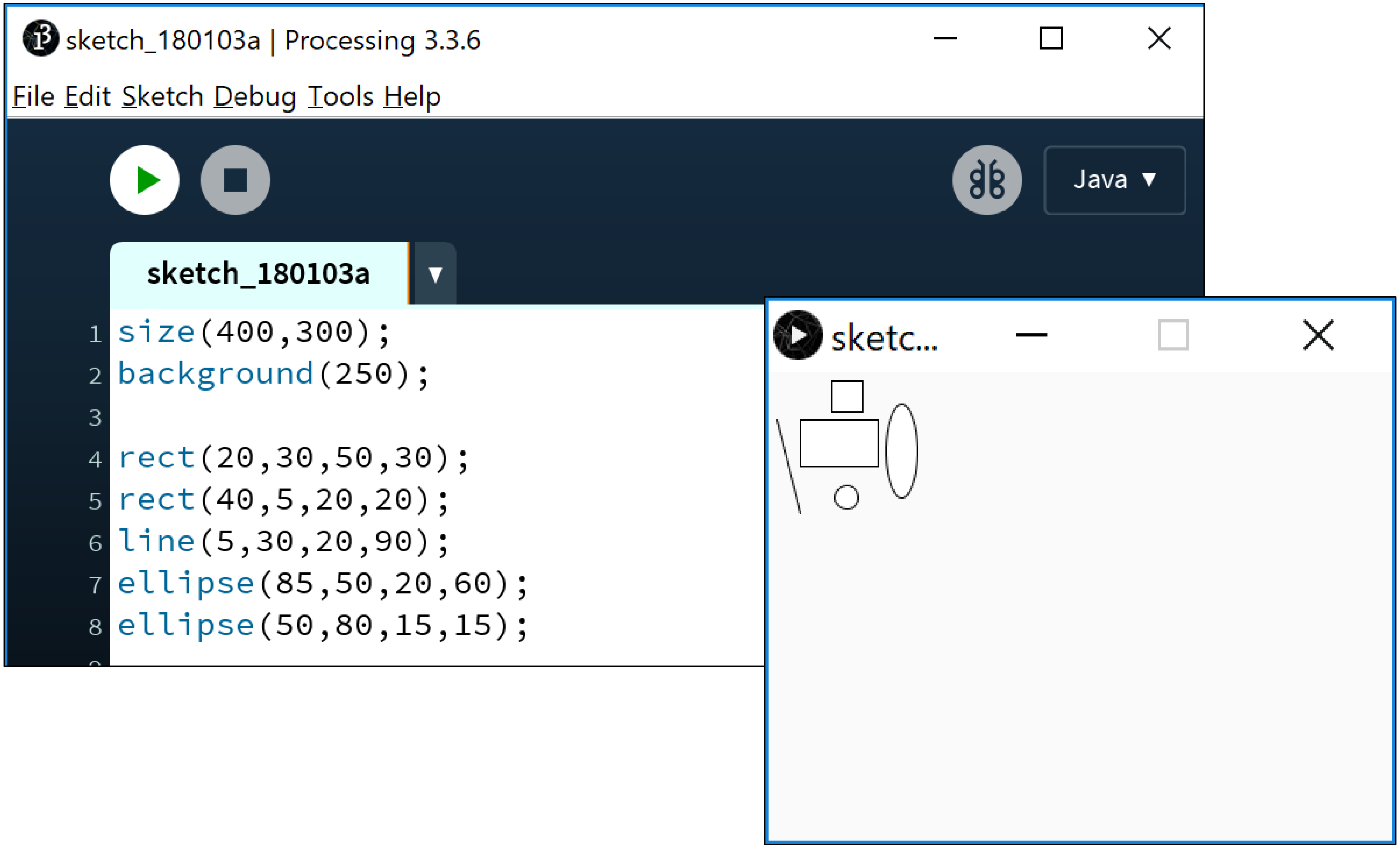
Flow of Control in a Program

- Each program you write will typically have:

Sequence	Things that will be done in a particular order
Selection	Things that will be done conditionally
Iteration	Things that will be done repetitively

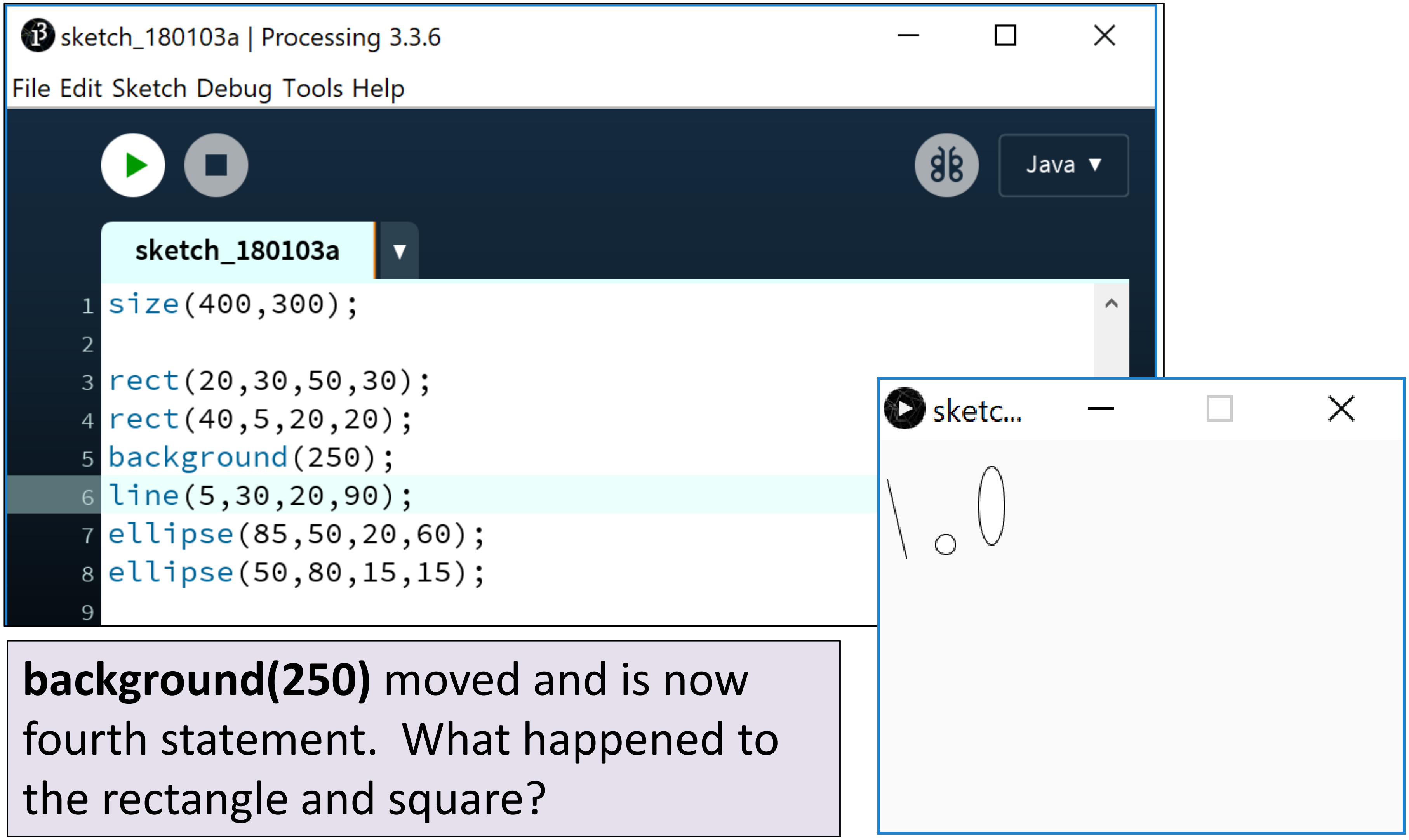
- The following example demonstrates ***Sequence***.
- We will cover ***Selection*** and ***Iteration*** in future weeks.

Sequence of Instructions – Example


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a
sketc...

```
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);
```

Sequence of Instructions – Matters!!!


sketch_180103a | Processing 3.3.6
File Edit Sketch Debug Tools Help
Java
sketch_180103a
sketc...

```
size(400,300);

rect(20,30,50,30);
rect(40,5,20,20);
background(250);
line(5,30,20,90);
ellipse(85,50,20,60);
ellipse(50,80,15,15);
```

background(250) moved and is now fourth statement. What happened to the rectangle and square?

Questions?