

An Introduction to Processing

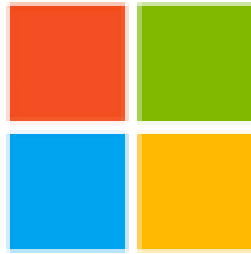
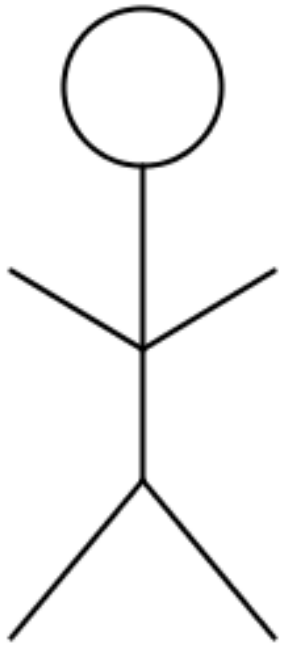
Basics of Animation

Produced Ms. Mairead Meagher
by: Dr. Siobhán Drohan



Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

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



So far, all of our
animations have been
static.

Topics list

1. The **setup()** function.

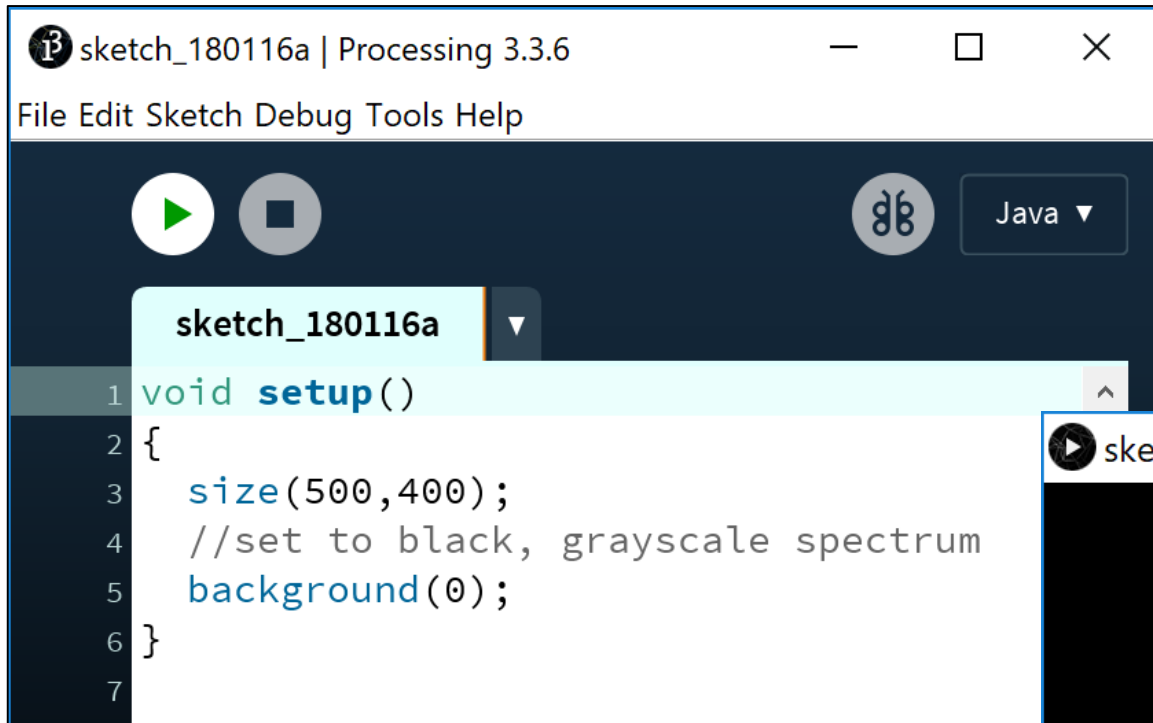
2. The **draw()** function.

3. **System Variables** in Processing.

void setup()

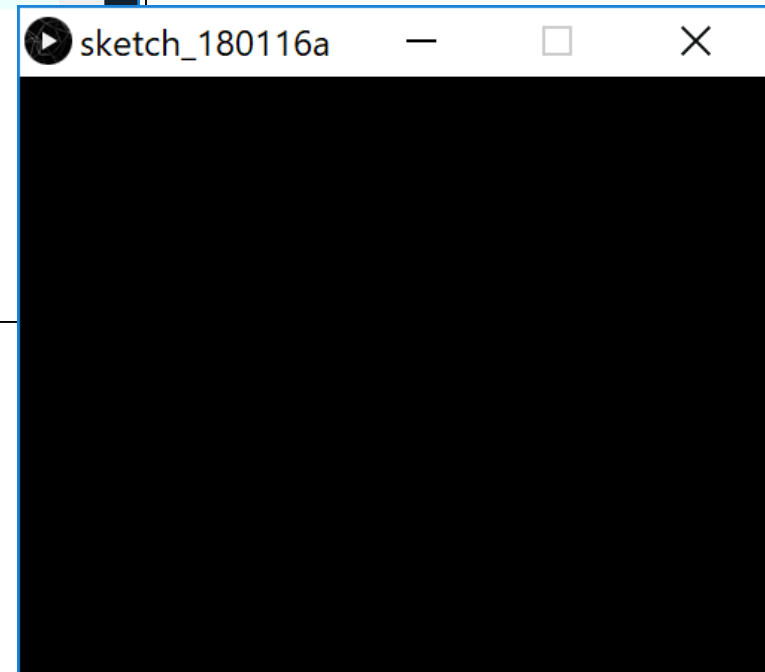
- **setup()** is called by Processing once (when the program starts). It should not be called again.
- **setup()** can set the screen size and background colour.
- There can only be one **setup()** function for each sketch.

void setup() – defining a method

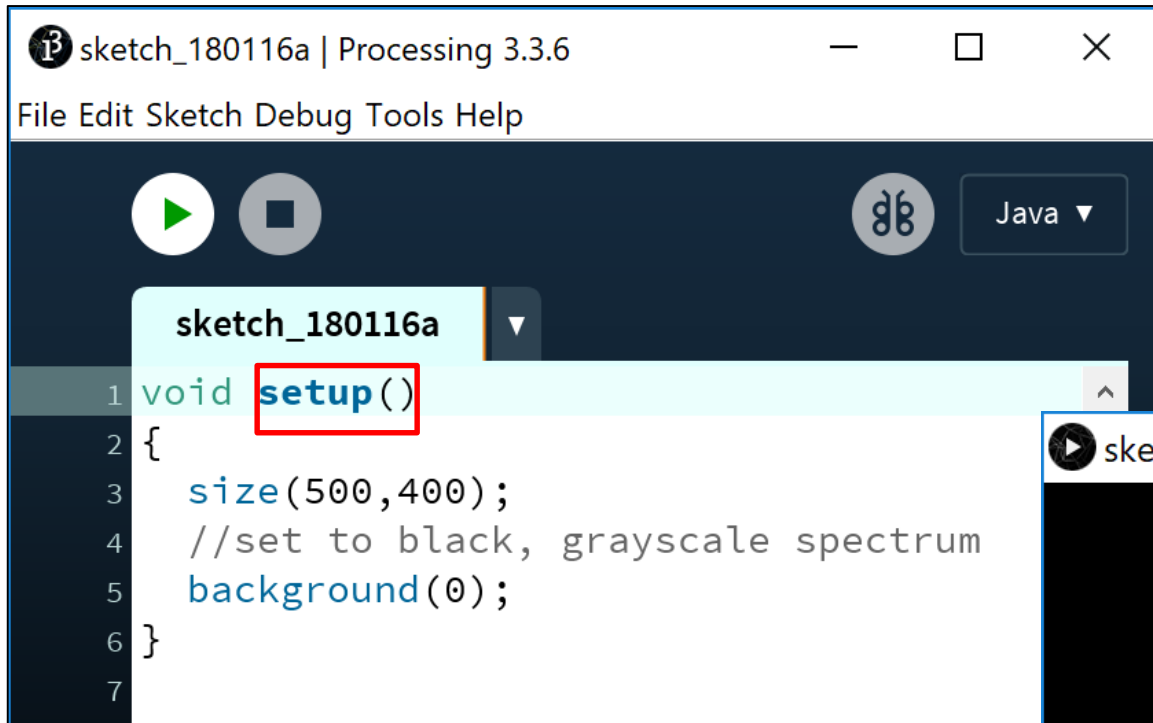


The screenshot shows the Processing IDE window titled "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". The toolbar contains a play button, a stop button, a palette icon, and a "Java" dropdown menu. A tab labeled "sketch_180116a" is active. The code editor displays the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4   //set to black, grayscale spectrum  
5   background(0);  
6 }  
7
```



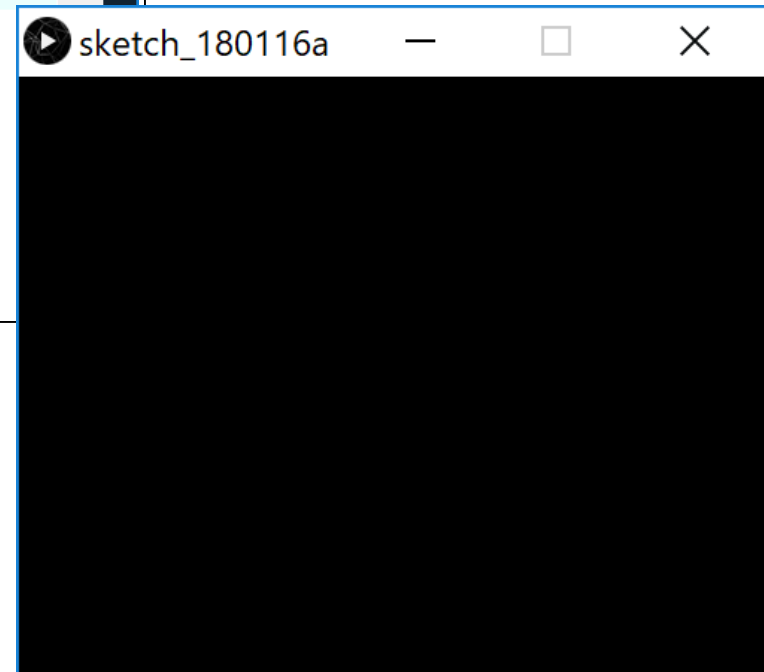
void setup() – defining a method



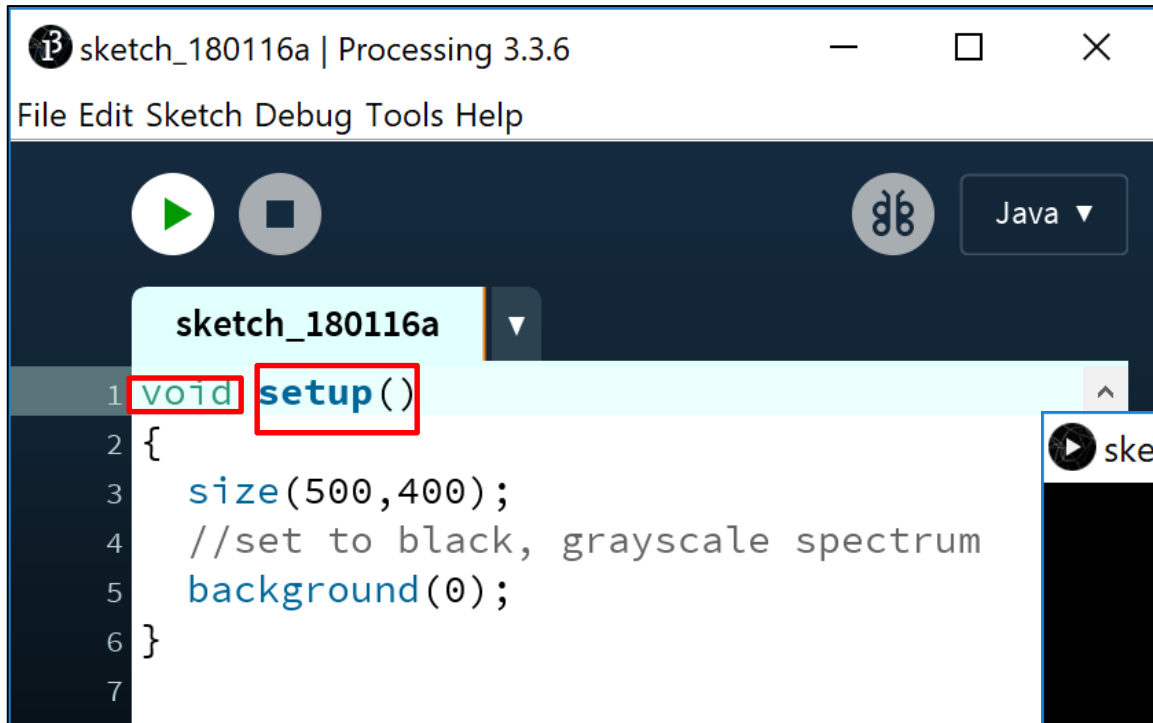
The screenshot shows the Processing IDE window titled "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". The toolbar contains a play button, a stop button, a palette icon, and a language dropdown set to "Java". Below the toolbar, a tab labeled "sketch_180116a" is active. The code editor displays the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4   //set to black, grayscale spectrum  
5   background(0);  
6 }  
7
```

The `void setup()` line is highlighted in light blue, and the text `setup()` is enclosed in a red rectangular box.



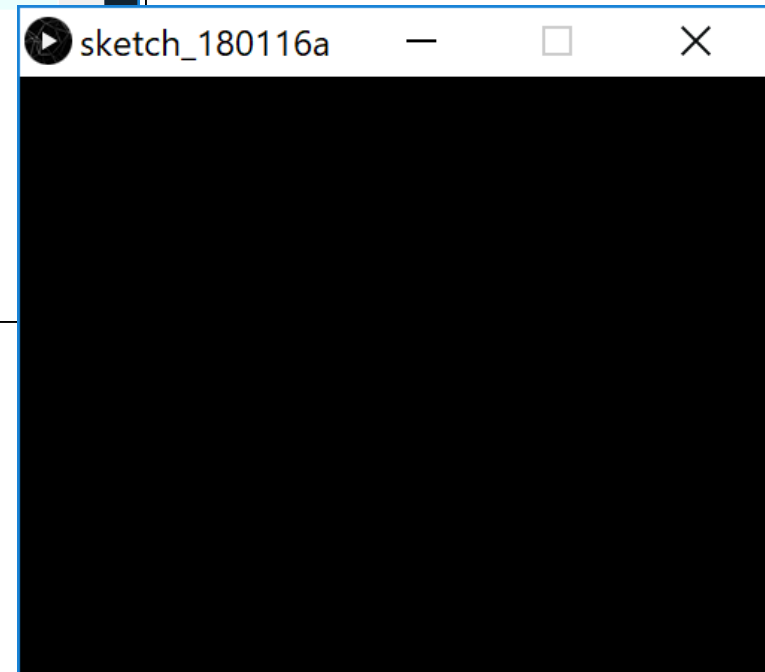
void setup() – defining a method



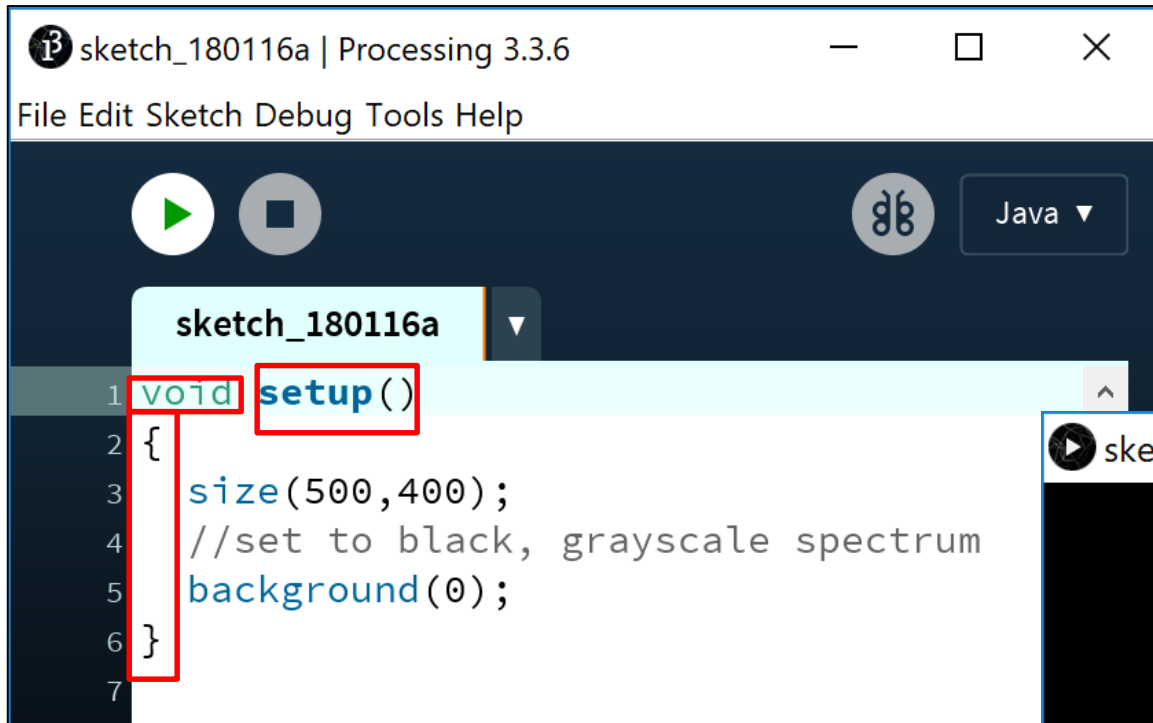
The screenshot shows the Processing IDE window titled "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". The toolbar contains a play button, a stop button, a palette icon, and a language dropdown set to "Java". Below the toolbar, a tab labeled "sketch_180116a" is active. The code editor displays the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4   //set to black, grayscale spectrum  
5   background(0);  
6 }  
7
```

In the first line of code, the word "void" is highlighted with a red box, and the word "setup()" is also highlighted with a red box.



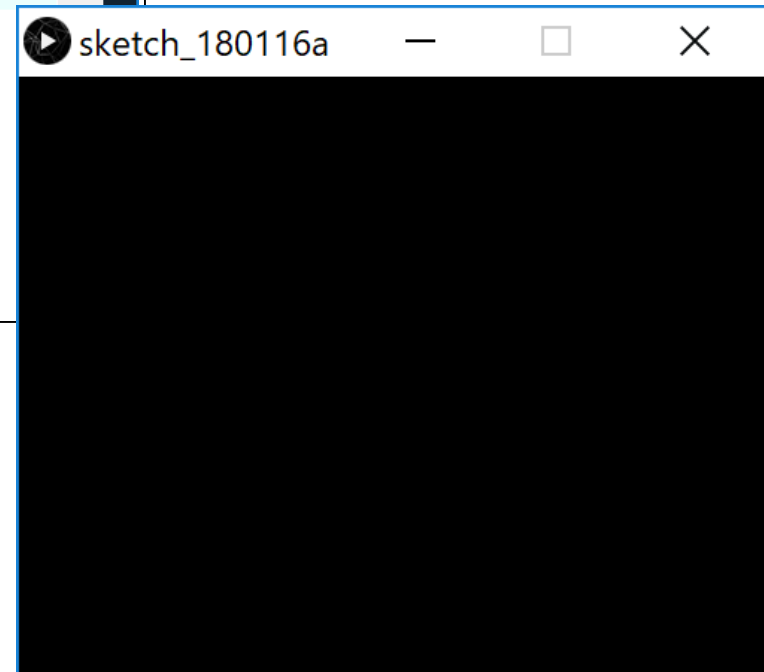
void setup() – defining a method



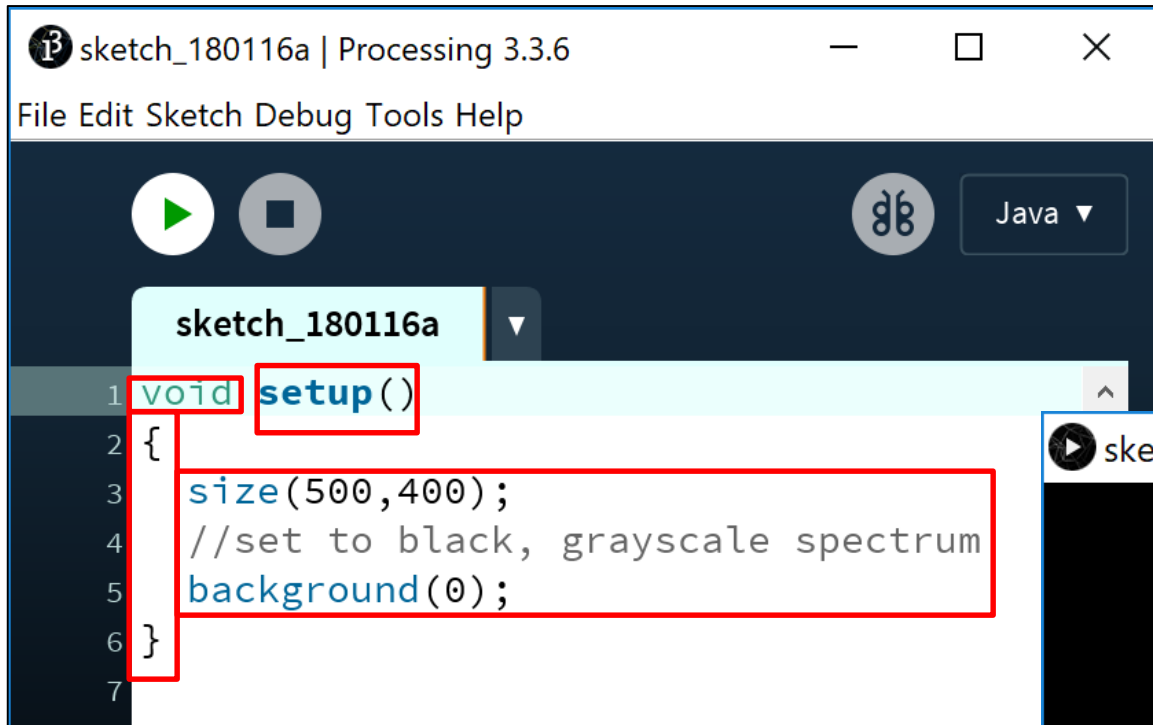
The screenshot shows the Processing IDE window titled "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". Below the menu bar are icons for running (a green play button), stopping (a gray square button), and a palette icon, along with a "Java" dropdown menu. A tab labeled "sketch_180116a" is active. The code editor displays the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4   //set to black, grayscale spectrum  
5   background(0);  
6 }  
7
```

In the code, the word "void" is highlighted with a red box, and the word "setup()" is also highlighted with a red box. The opening curly brace "{" is highlighted with a red box.



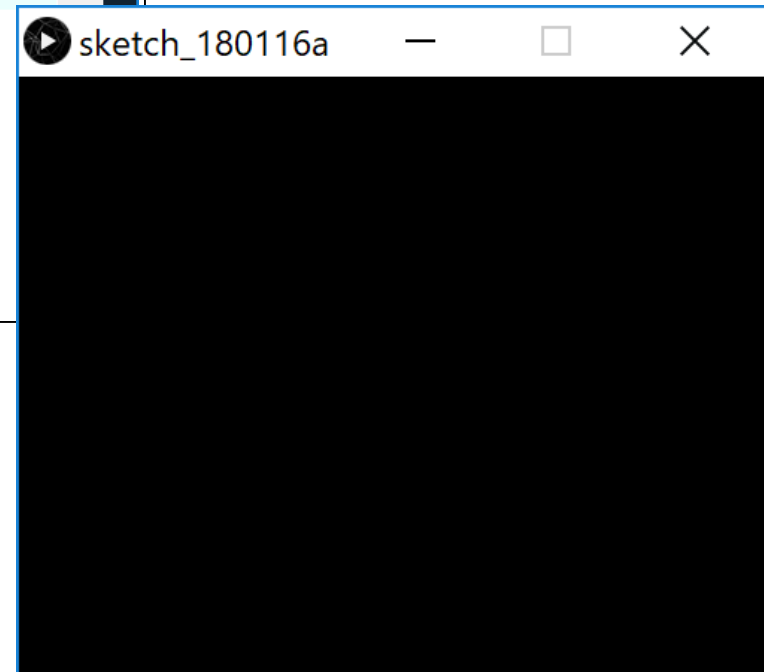
void setup() – defining a method



The screenshot shows the Processing IDE window titled "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". The toolbar contains a play button, a stop button, a palette icon, and a "Java" dropdown menu. The sketch name "sketch_180116a" is displayed in a dropdown. The code editor shows the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4   //set to black, grayscale spectrum  
5   background(0);  
6 }  
7
```

Red boxes highlight the `void` keyword, the `setup()` method name, and the entire function body.



Topics list

1. The **setup()** function.
2. The **draw()** function.
3. **System Variables** in Processing.

void draw()

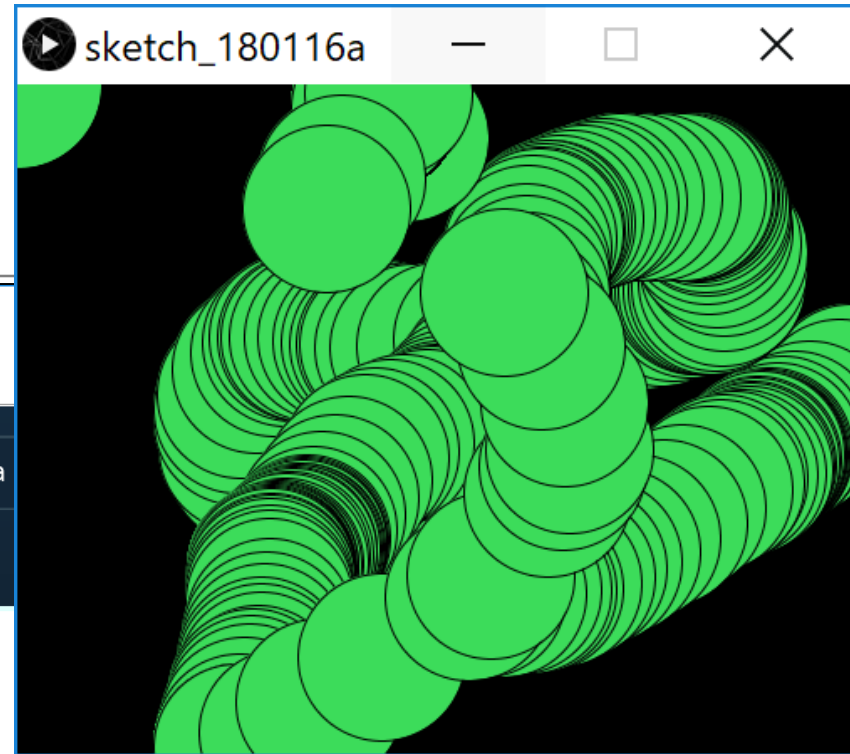
- You should never call the **draw()** function.
 - Processing automatically calls it straight after the **setup()** call.
- **Draw()** continuously executes the code contained inside it.
 - (60 times a second by default)
- There can only be one **draw()** function for each sketch.

void draw()

```
sketch_180116a | Processing 3.3.6
File Edit Sketch Debug Tools Help

[Run] [Stop] [Java]

sketch_180116a ▼
1 void setup()
2 {
3   size(500,400);
4   //set to black, grayscale spectrum
5   background(0);
6 }
7
8 void draw()
9 {
10  stroke(0, 0, 0);    //black outline
11  fill(60, 220, 90);  //green
12  ellipse(mouseX, mouseY, 100, 100);
13 }
14
```



System Variables ➡

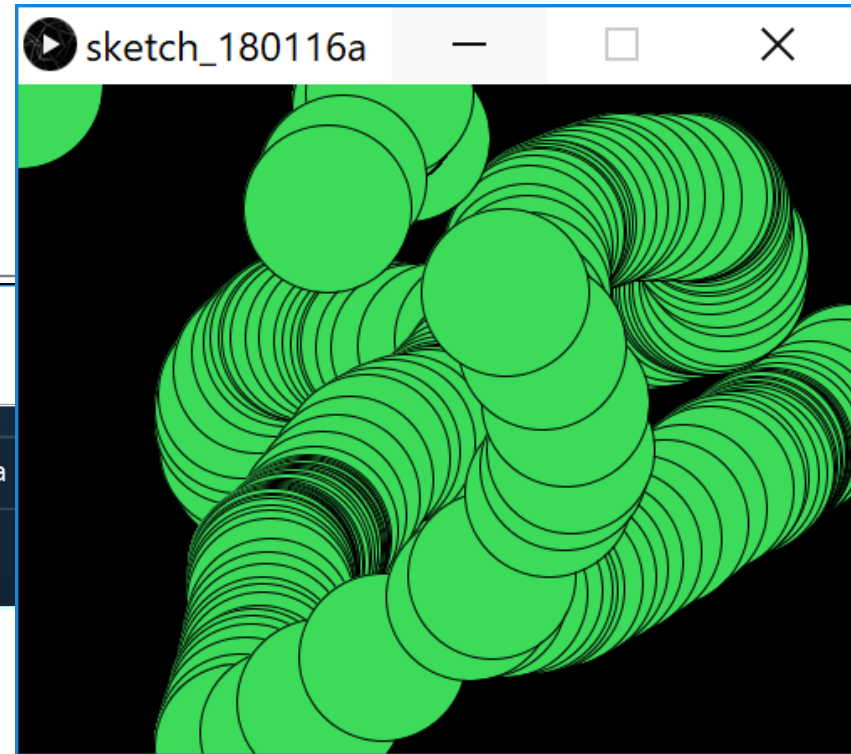
mouseX = x co-ordinate of mouse pointer
mouseY = y co-ordinate of mouse pointer

void draw()

```
sketch_180116a | Processing 3.3.6
File Edit Sketch Debug Tools Help

[Play] [Stop] [Pencil] [Java]

sketch_180116a ▼
1 void setup()
2 {
3   size(500,400);
4   //set to black, grayscale spectrum
5   background(0);
6 }
7
8 void draw()
9 {
10  stroke(0, 0, 0);    //black outline
11  fill(60, 220, 90);  //green
12  ellipse(mouseX, mouseY, 100, 100);
13 }
14
```



Q: Why many circles?

System Variables ➡

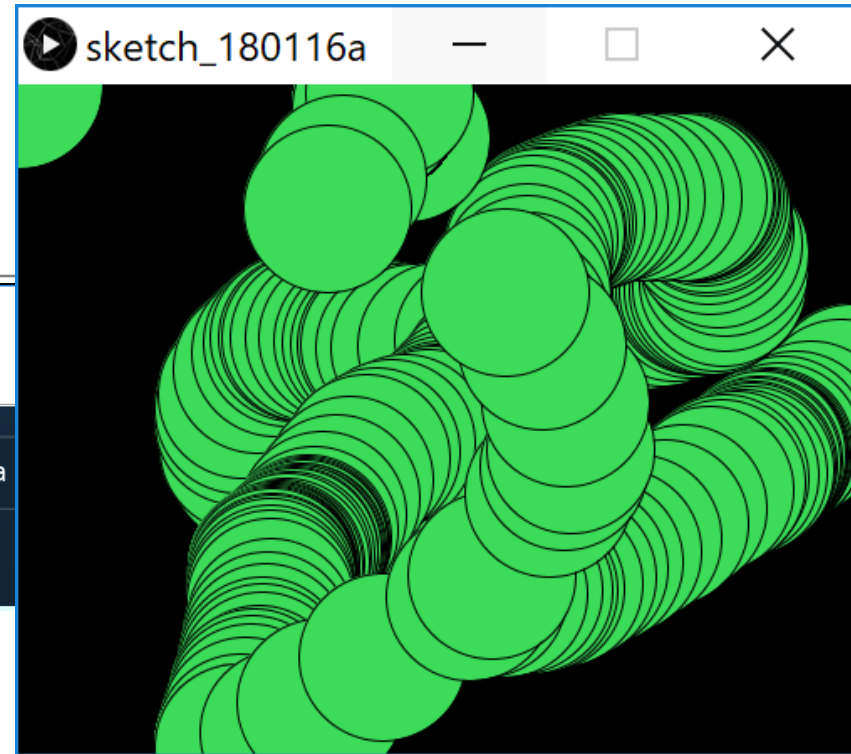
mouseX = x co-ordinate of mouse pointer
mouseY = y co-ordinate of mouse pointer

void draw()

```
sketch_180116a | Processing 3.3.6
File Edit Sketch Debug Tools Help

[Play] [Stop] [Pencil] [Java]

sketch_180116a ▼
1 void setup()
2 {
3   size(500,400);
4   //set to black, grayscale spectrum
5   background(0);
6 }
7
8 void draw()
9 {
10  stroke(0, 0, 0);    //black outline
11  fill(60, 220, 90);  //green
12  ellipse(mouseX, mouseY, 100, 100);
13 }
14
```



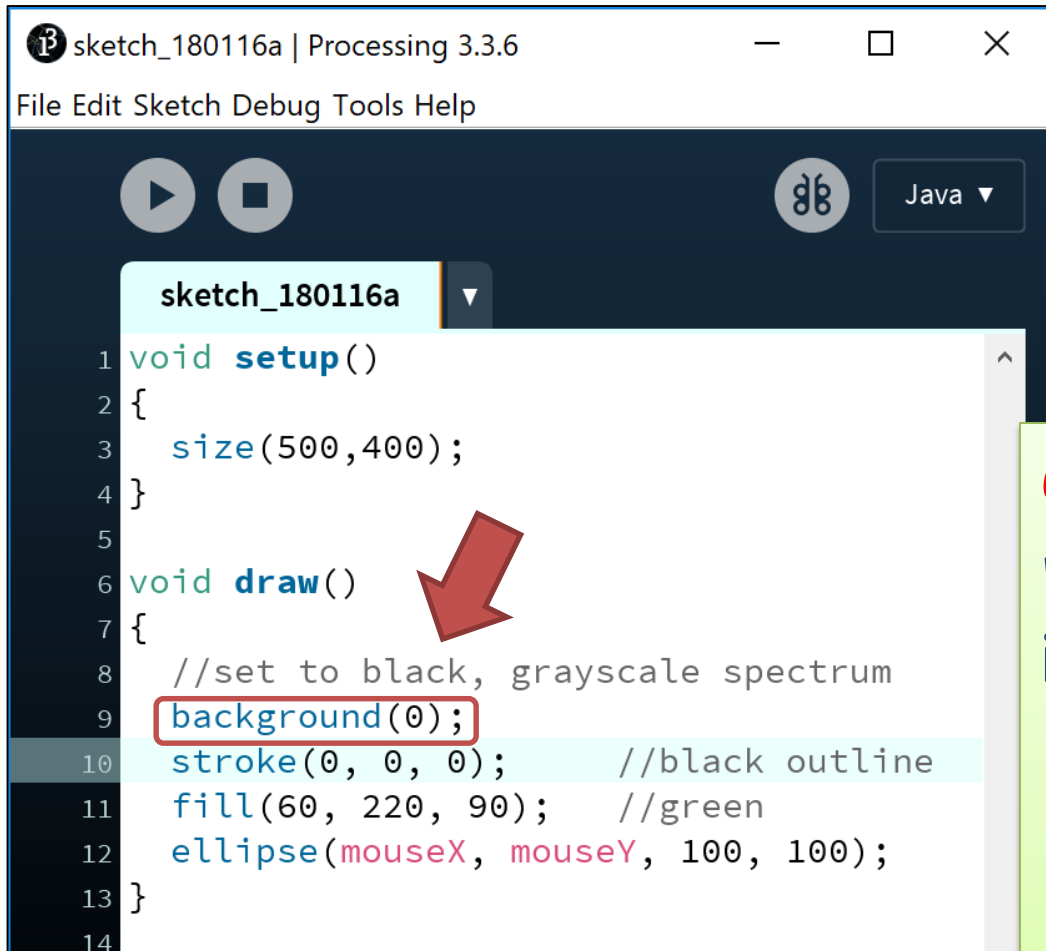
Q: *Why many circles?*

A: **background(0)** is
in the setup
function.

System Variables ➡

mouseX = x co-ordinate of mouse pointer
mouseY = y co-ordinate of mouse pointer

void draw()



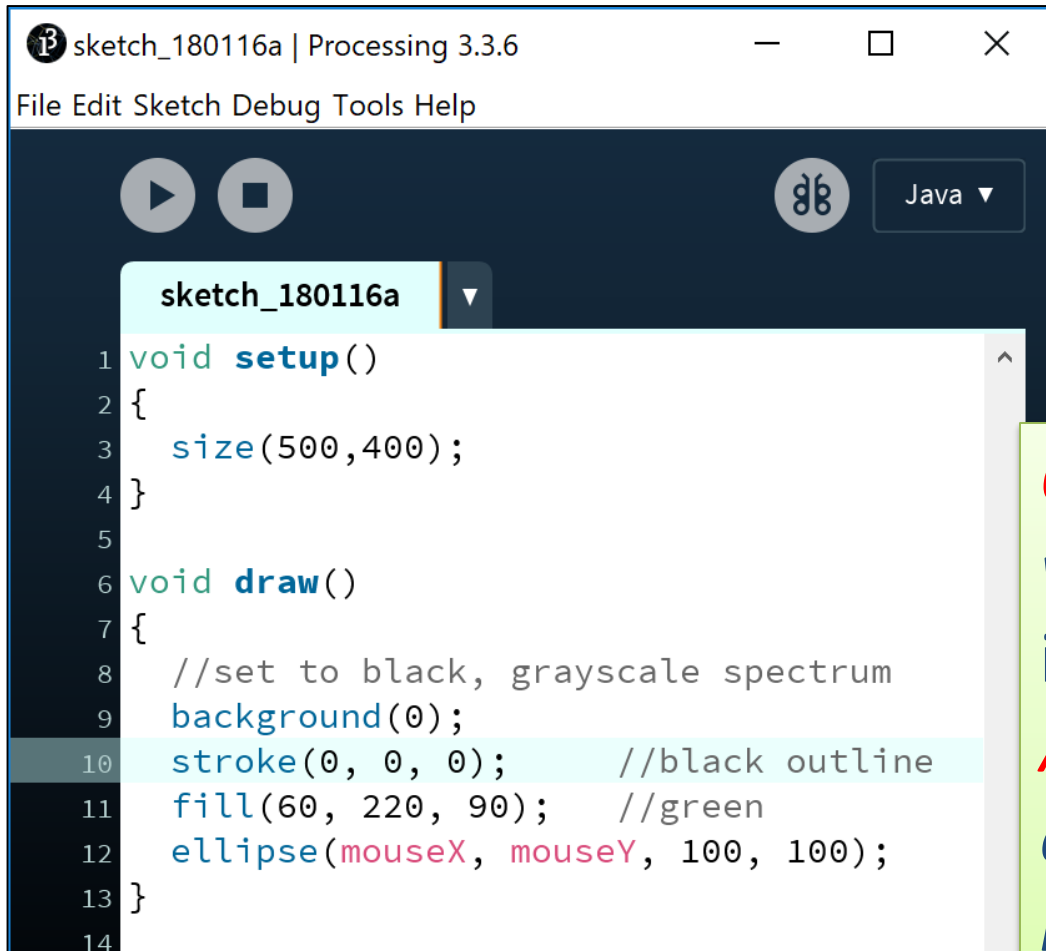
The screenshot shows the Processing IDE interface. The title bar reads "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". The toolbar contains a play button, a stop button, a palette icon, and a language dropdown set to "Java". The sketch name "sketch_180116a" is displayed in a tab. The code editor shows the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4 }  
5  
6 void draw()  
7 {  
8   //set to black, grayscale spectrum  
9   background(0);  
10  stroke(0, 0, 0);    //black outline  
11  fill(60, 220, 90);  //green  
12  ellipse(mouseX, mouseY, 100, 100);  
13 }  
14
```

A red arrow points to the `background(0);` line on line 9, which is also highlighted with a red box.

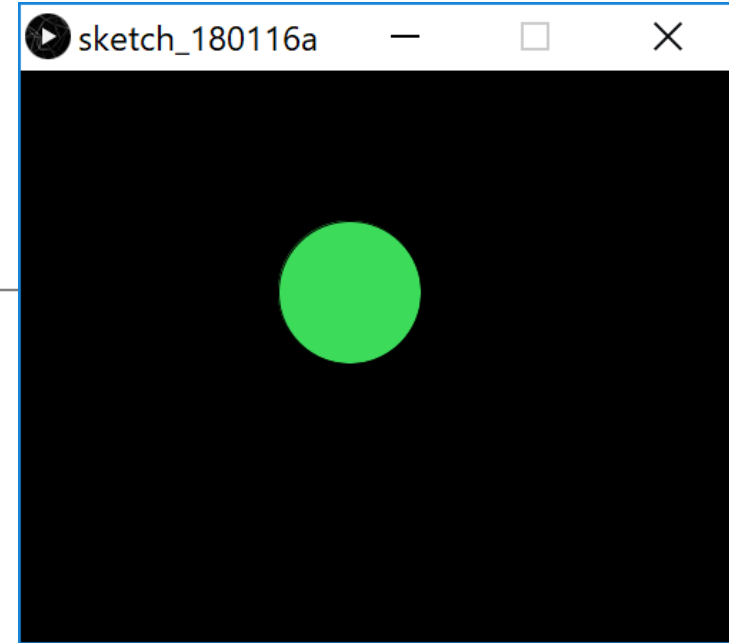
*Q: What happens when we move **background(0)** into the draw function?*

void draw()



The image shows the Processing IDE window for sketch_180116a. The title bar reads "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". Below the menu bar are icons for running (a play button), stopping (a square button), and a palette icon. A dropdown menu shows "Java". The code editor displays the following code:

```
1 void setup()  
2 {  
3   size(500,400);  
4 }  
5  
6 void draw()  
7 {  
8   //set to black, grayscale spectrum  
9   background(0);  
10  stroke(0, 0, 0);    //black outline  
11  fill(60, 220, 90);  //green  
12  ellipse(mouseX, mouseY, 100, 100);  
13 }  
14
```



*Q: What happens when we move **background(0)** into the draw function?*
A: Before each circle is drawn, the background is painted black, so it clears the previous circle.

Topics list

1. The **setup()** function.
2. The **draw()** function.
3. **System Variables** in Processing.

System Variables in Processing

Some **examples** of system variables in Processing:

mouseX (x co-ordinate of the mouse pointer on the display window)

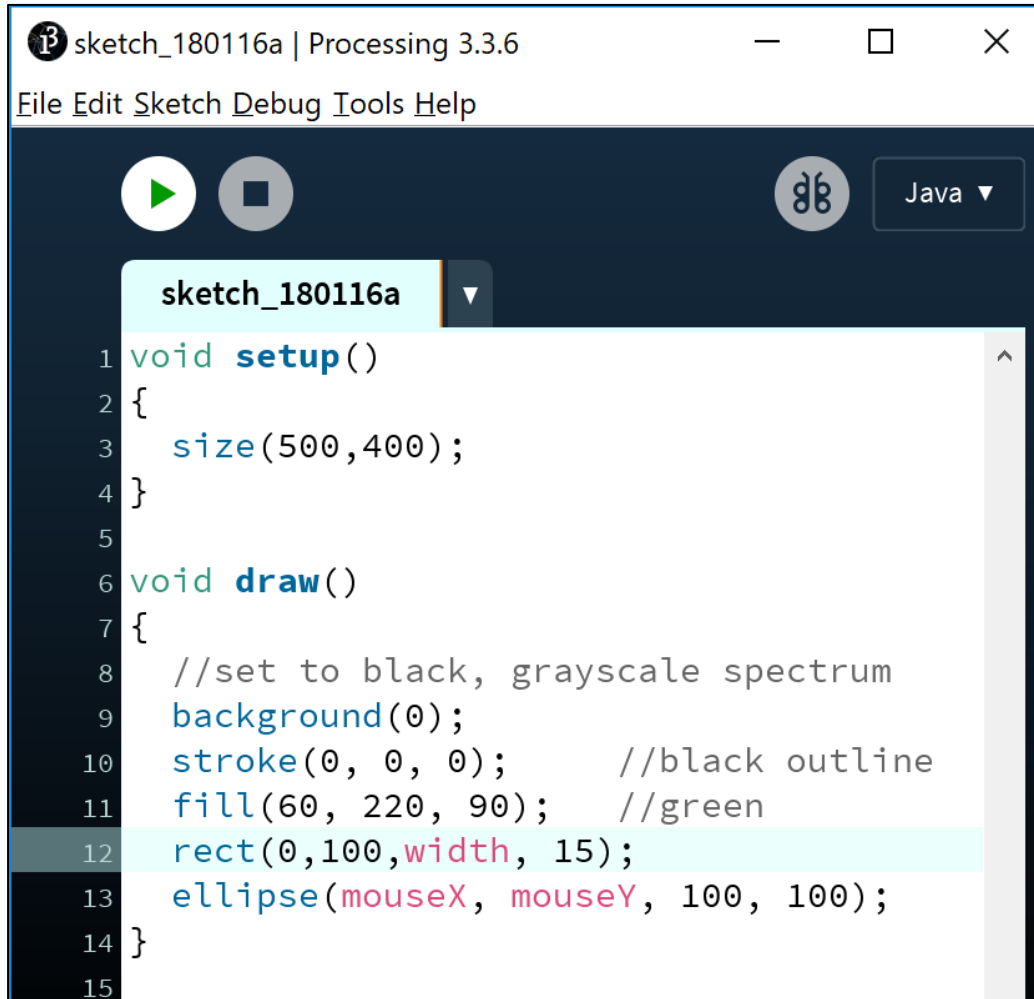
mouseY (y co-ordinate of the mouse pointer on the display window)

width (width of the display window)

height (height of the display window)

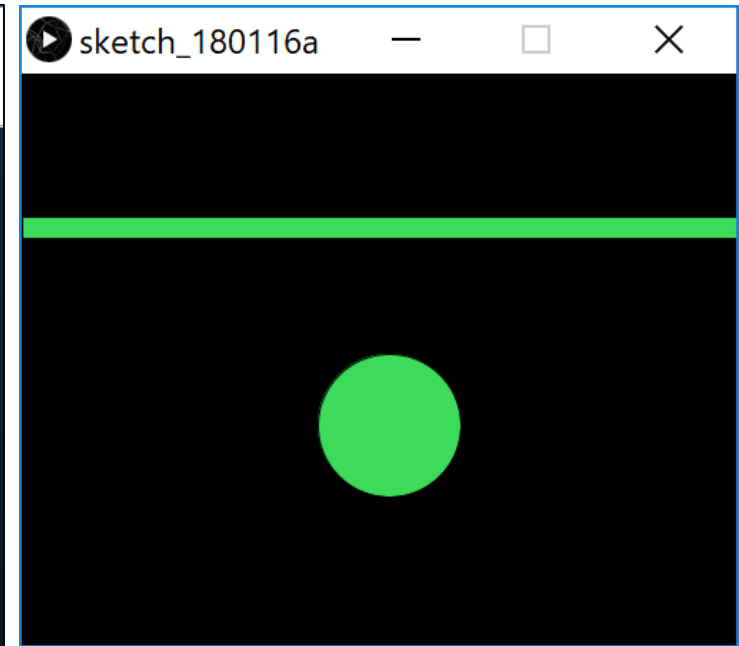
We **don't have to define/create** these; just use them.

System Variables in Processing



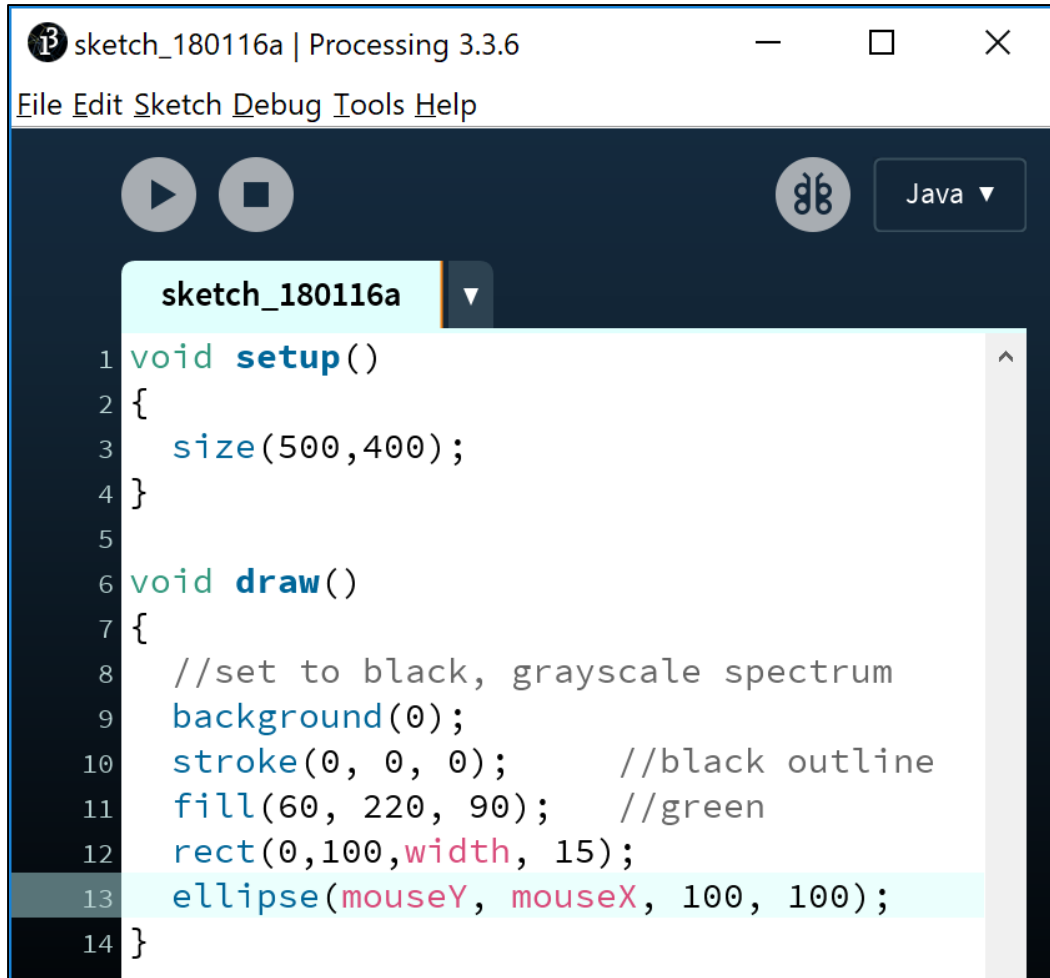
The screenshot shows the Processing IDE window titled "sketch_180116a | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". Below the menu bar are buttons for running (a green play button), stopping (a gray square button), and a palette icon. A dropdown menu shows "Java". The code editor displays the following code:

```
1 void setup()
2 {
3   size(500,400);
4 }
5
6 void draw()
7 {
8   //set to black, grayscale spectrum
9   background(0);
10  stroke(0, 0, 0);    //black outline
11  fill(60, 220, 90);  //green
12  rect(0,100,width, 15);
13  ellipse(mouseX, mouseY, 100, 100);
14 }
15
```



Using the **width** system variable in the **rect** function to draw a thick line.

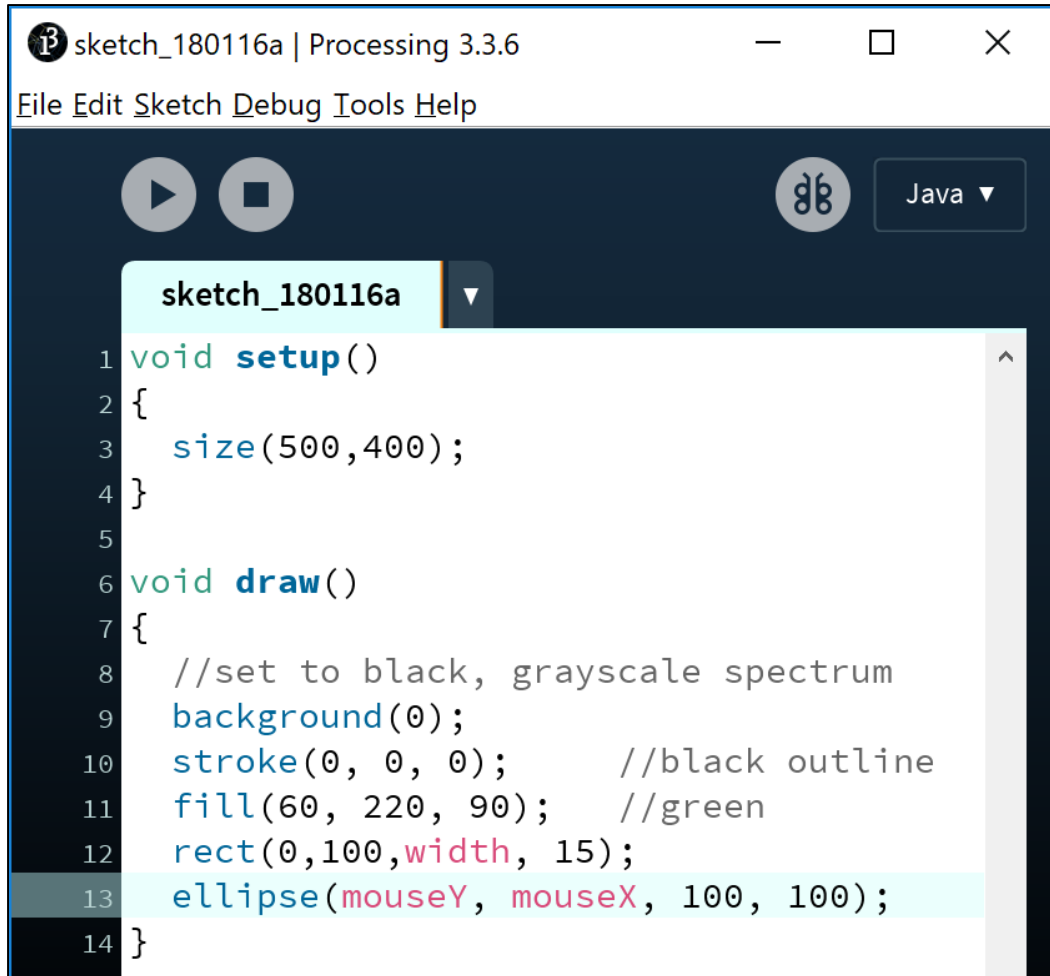
System Variables in Processing



```
1 void setup()
2 {
3   size(500,400);
4 }
5
6 void draw()
7 {
8   //set to black, grayscale spectrum
9   background(0);
10  stroke(0, 0, 0);    //black outline
11  fill(60, 220, 90);  //green
12  rect(0,100,width, 15);
13  ellipse(mouseY, mouseX, 100, 100);
14 }
```

***Q:** What would happen to our animation if we swapped the **mouseX** and **mouseY** variables in the **ellipse** function with each other?*

System Variables in Processing



```
sketch_180116a | Processing 3.3.6
File Edit Sketch Debug Tools Help

sketch_180116a
1 void setup()
2 {
3   size(500,400);
4 }
5
6 void draw()
7 {
8   //set to black, grayscale spectrum
9   background(0);
10  stroke(0, 0, 0);    //black outline
11  fill(60, 220, 90);  //green
12  rect(0,100,width, 15);
13  ellipse(mouseY, mouseX, 100, 100);
14 }
```

Q: *What would happen to our animation if we swapped the **mouseX** and **mouseY** variables in the **ellipse** function with each other?*

A: *As you move your mouse right on the x axis, the circle will move down on the y axis and vice versa.*

Questions?

