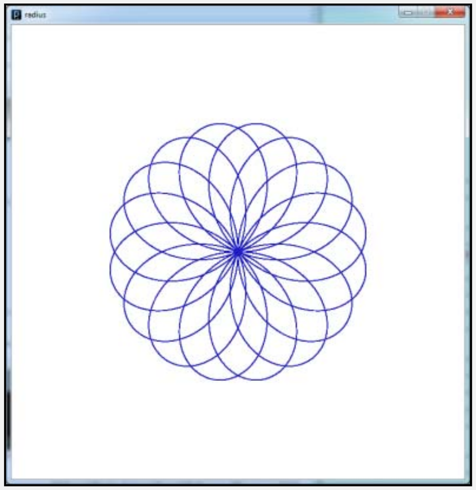
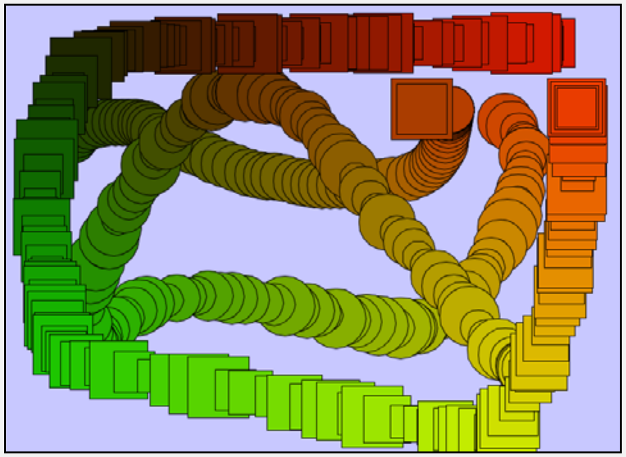
Writing Processing Programs

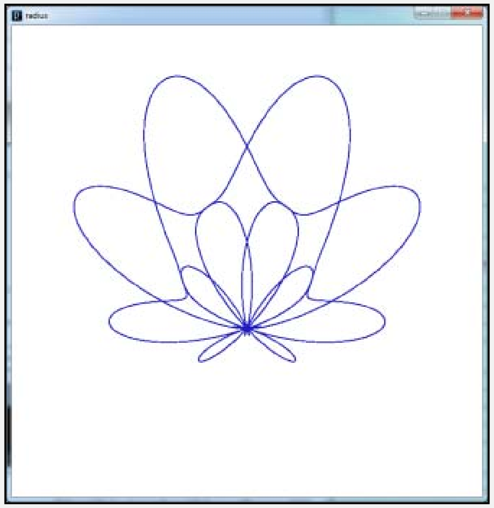
Oregon State University

Mike Bailey

mjb@cs.oregonstate.edu







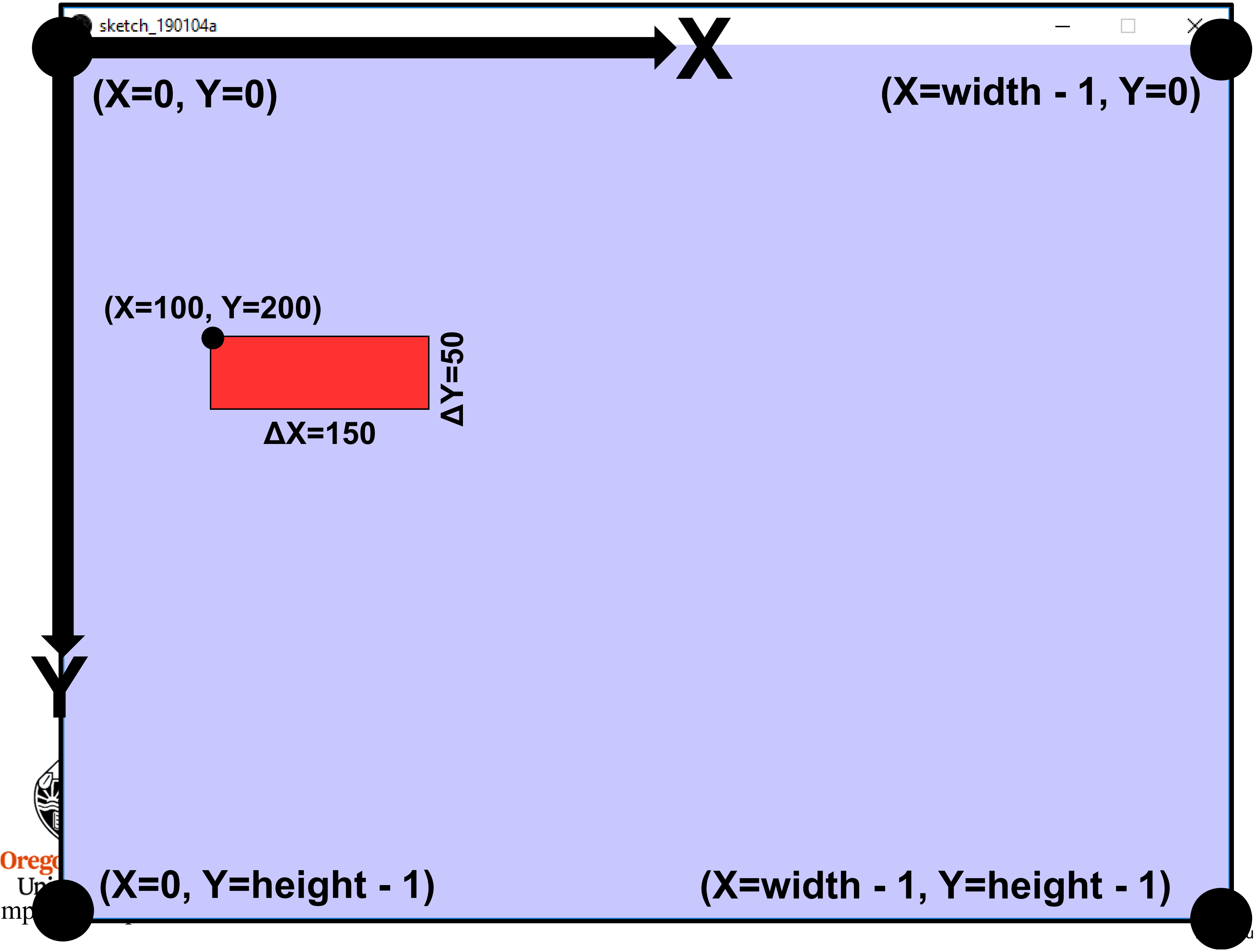
Introduction to Writing Processing Programs

With *Processing*, I have bad news, and I have good news.

The bad news is that you have to write a program. This will involve some learning.

The good news is that you ***get*** to write a program. You will end up being ever-so-more knowledgeable than you started out, and, once you get the hang of this, there is nothing you won’t be able to do with it!

Coordinate Systems for Processing Programs



Colors for Processing Programs

Colors are formed with combinations of **red**, **green**, and **blue**.

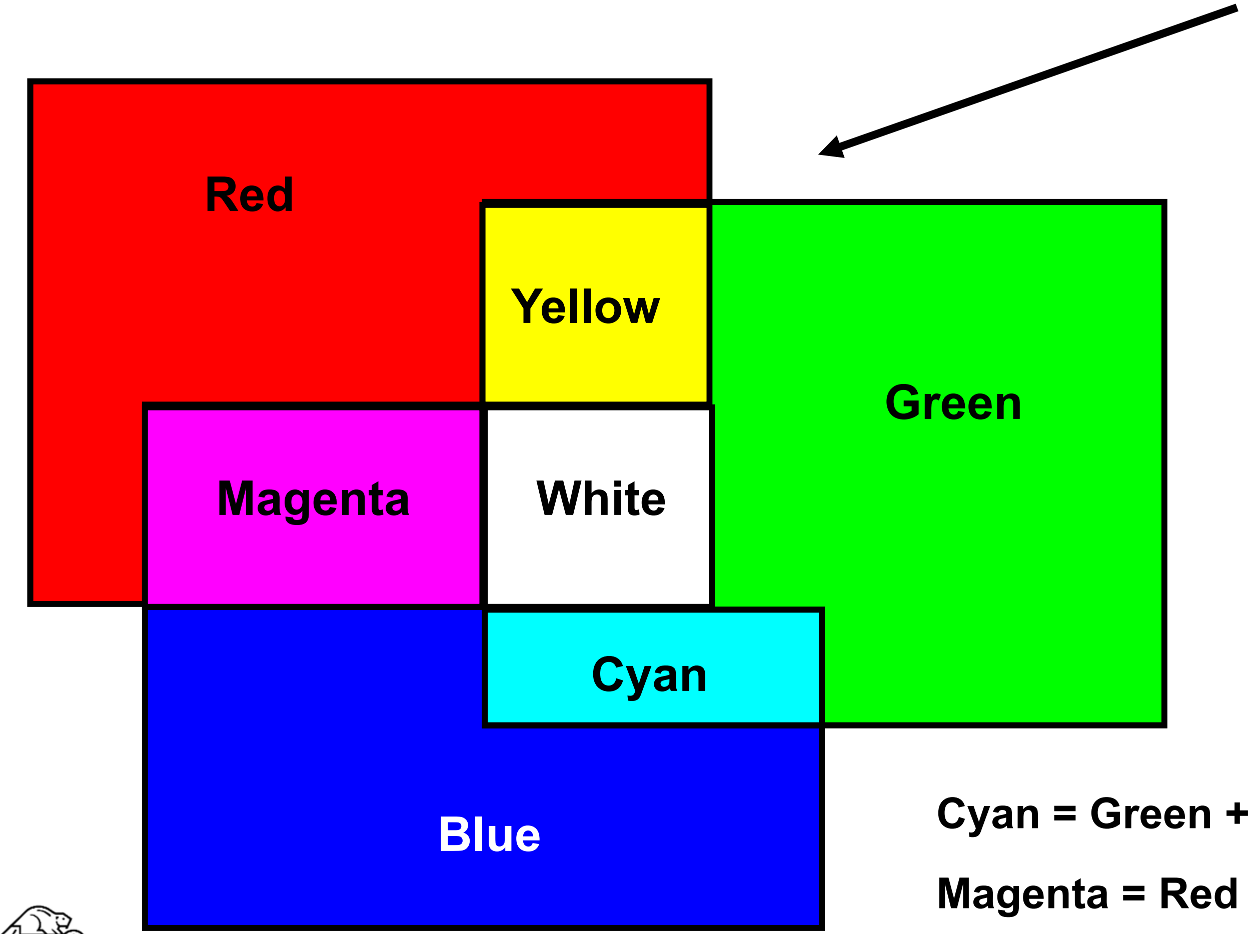
The smallest number you can use is **0**
The largest number you can use is **255**

Black	**0**	**0**	**0**
White	**255**	**255**	**255**
Red	**255**	**0**	**0**
Orange	**255**	**128**	**0**
Yellow	**255**	**255**	**0**
Green	**0**	**255**	**0**
Cyan	**0**	**255**	**255**
Blue	**0**	**0**	**255**
Magenta	**255**	**0**	**255**

Use the **Color Selector** from the Tools menu to pick your own color numbers.

Colors for Processing Programs

This is referred to as **"Additive Color"**



Cyan = Green + Blue

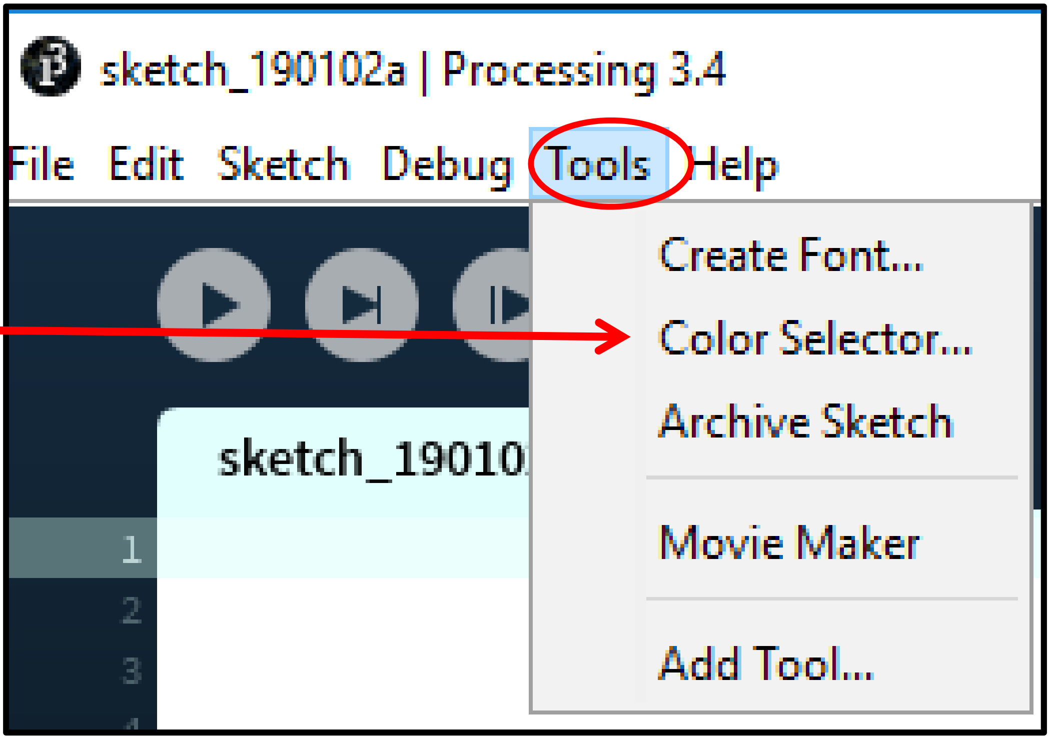
Magenta = Red + Blue

Yellow = Red + Green

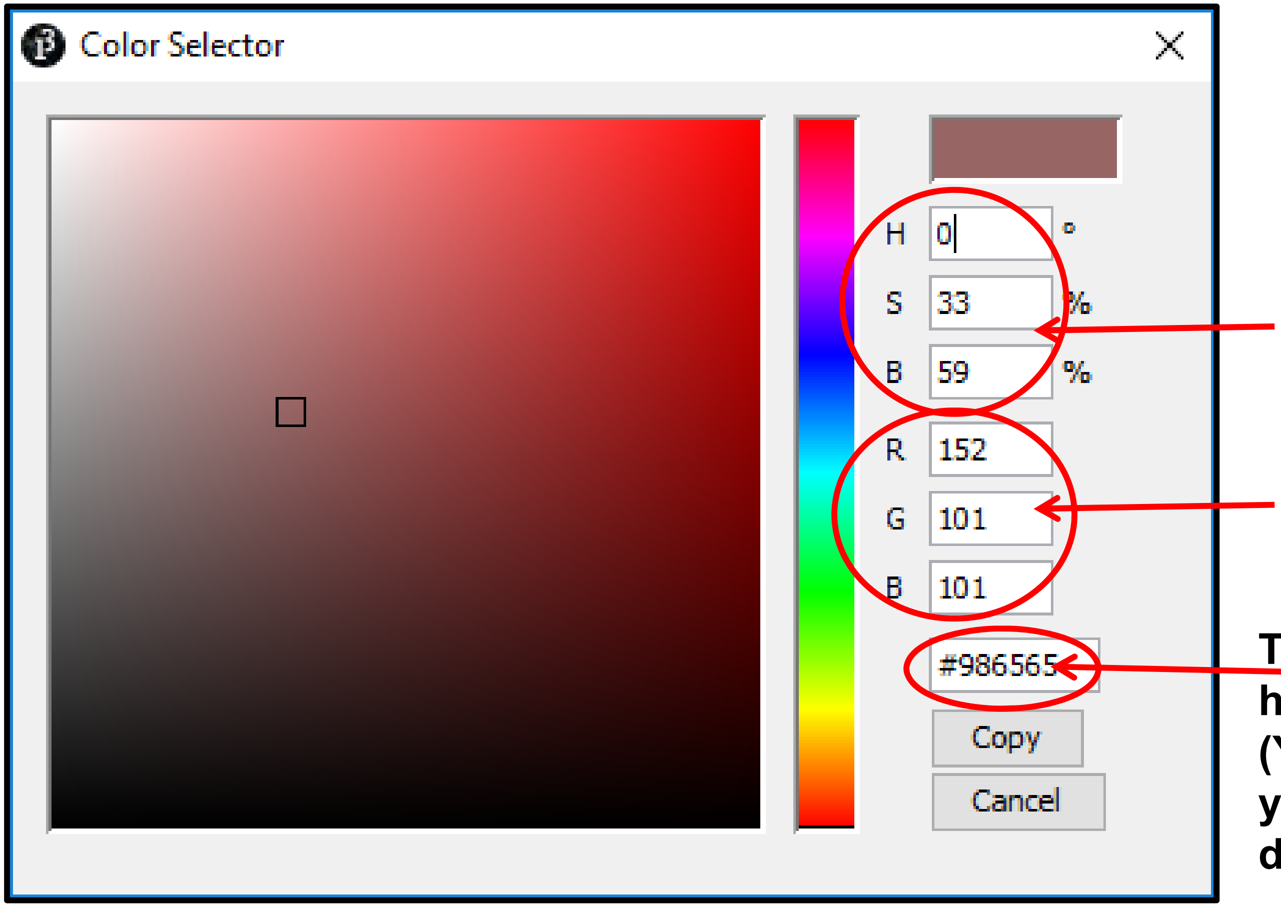
White = Red + Green + Blue

The Tools Menu

Interactively select and edit a color



The Color Selector from the Tools Menu



H 0 °

S 33 %

B 59 %

R 152

G 101

B 101

#986565

The color in Hue-Saturation-Brightness measurements.

The color in Red-Green-Blue measurements

The color in RGB hexadecimal (base 16). (You care about this if you are doing web development.)

```
colorMode( RGB );
fill( 152, 101, 101 );
```

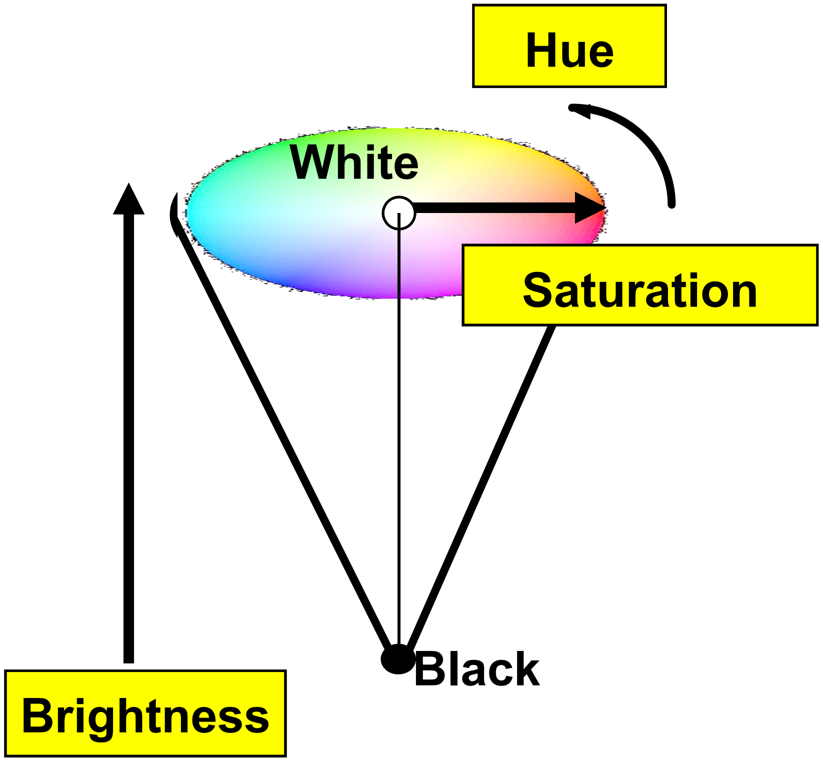
$0 \le$ r, g, b ≤ 255

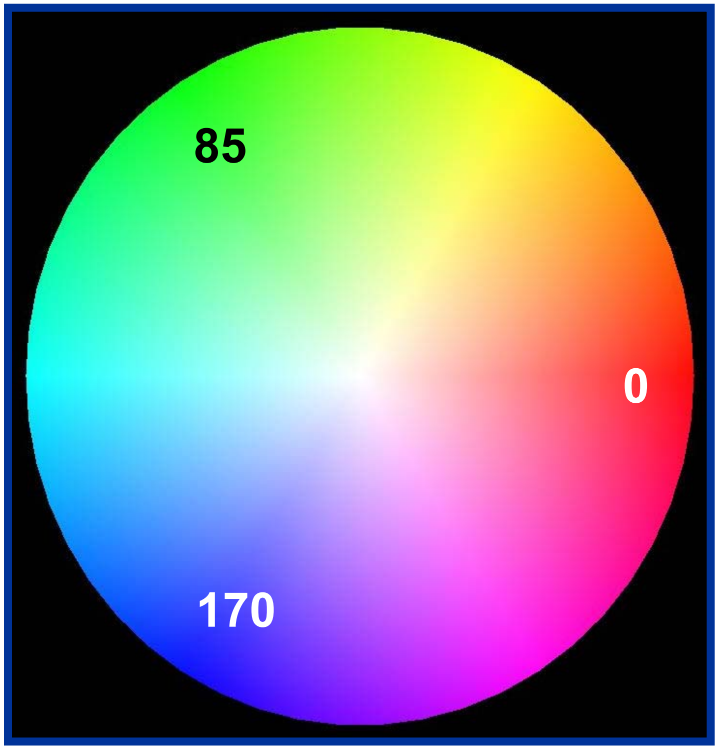
or

```
colorMode( HSB );
fill( 0, 33, 59 );
```

$0 \le$ h, s, b ≤ 255

Hue-Saturation-Brightness (HSB) -- Another way to specify additive color





0 ≤ h, s, b ≤ 255

Writing Processing Programs

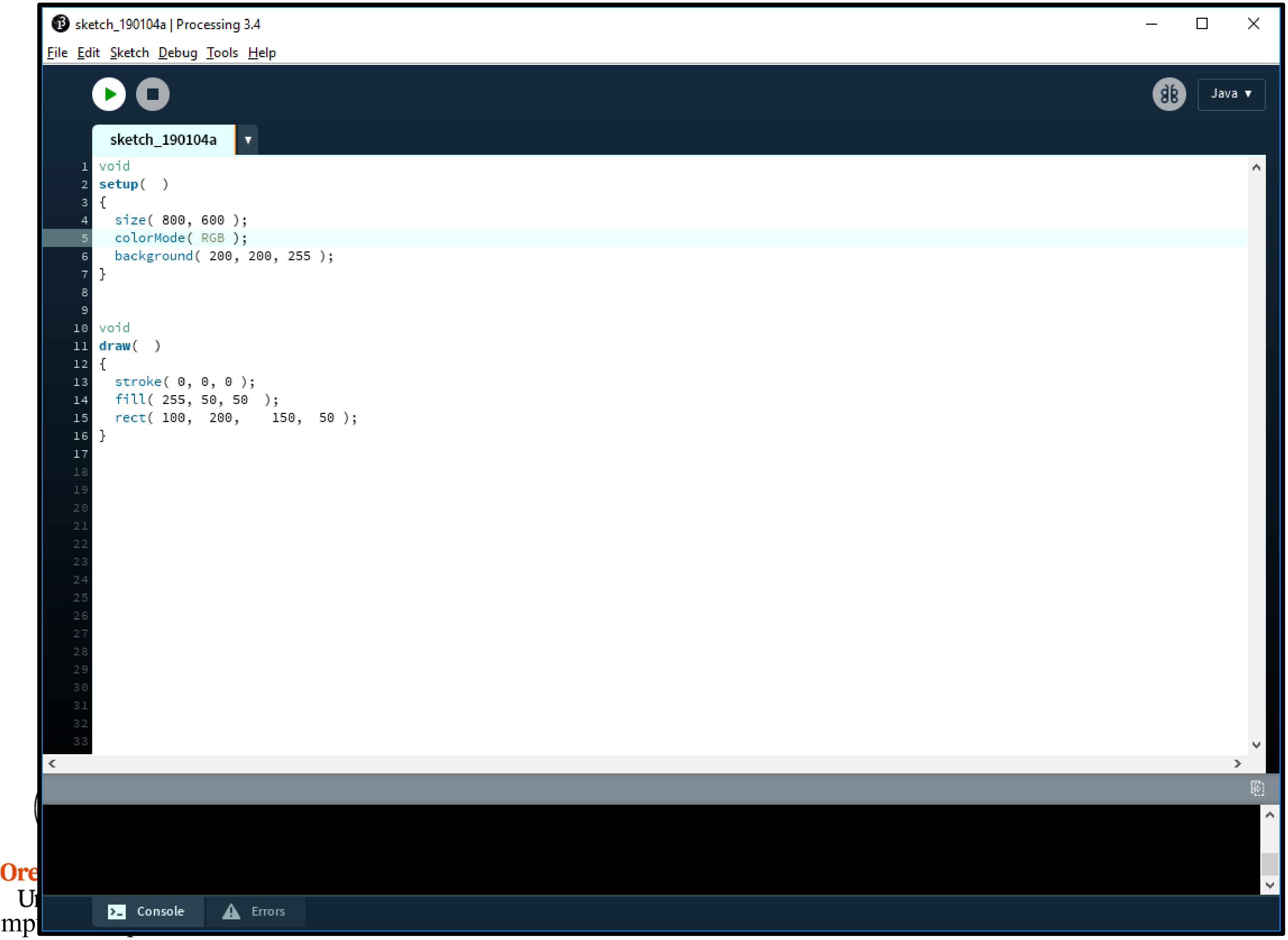
```
void
setup( )
{
  size( 800, 600 );
  colorMode( RGB );
  background( 200, 200, 255 );
}


void
draw( )
{
  stroke( 0, 0, 0 );
  fill( 255, 50, 50  );
  rect( 100,  200,    150,  50 );
}
```

You must add code to the **setup( )** function.
Processing calls this once when your program starts,

You mist add code to the **draw( )** function.
Processing calls this every time it wants to re-draw the scene.

Typing in Processing Programs

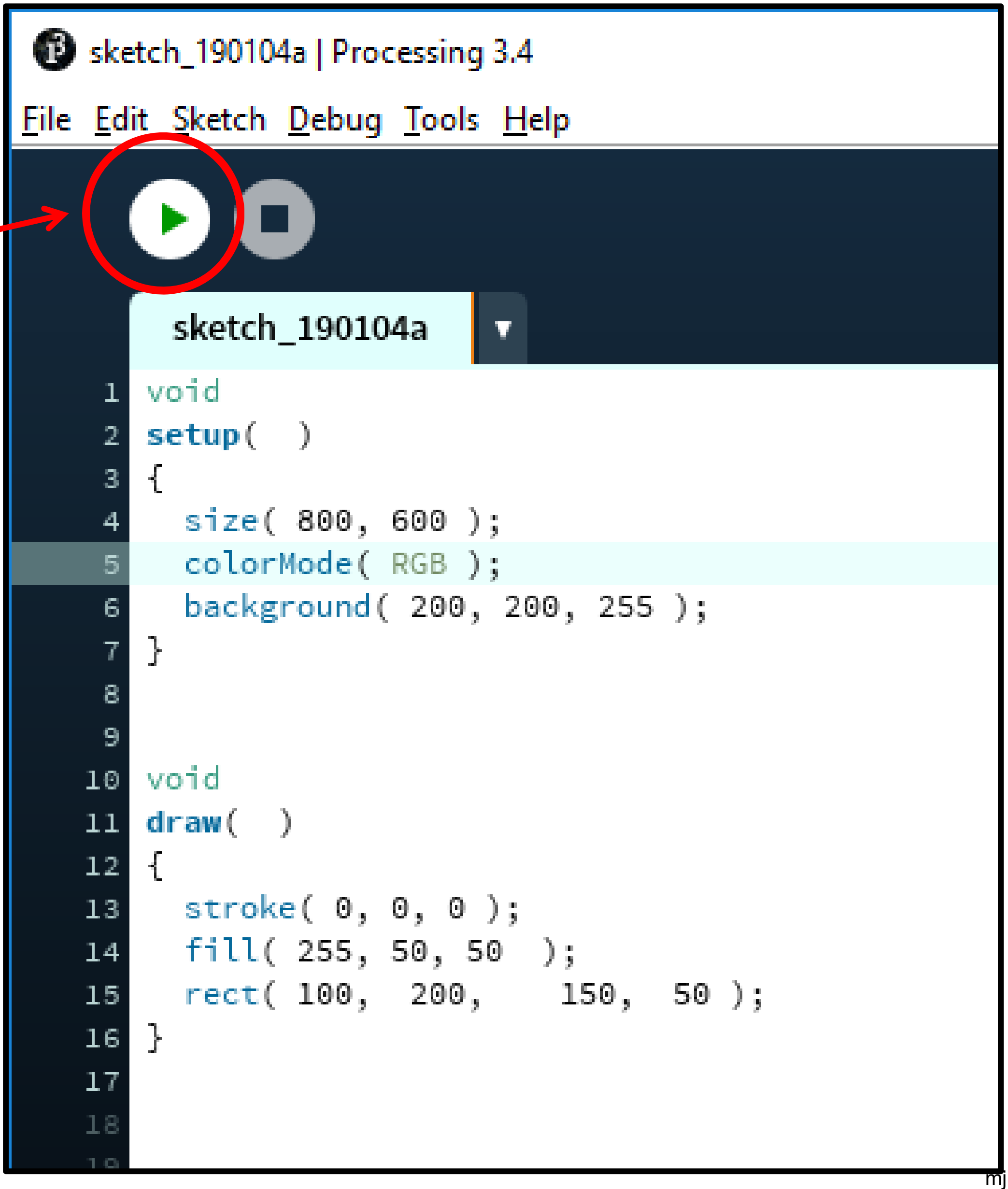

sketch_190104a | Processing 3.4
File Edit Sketch Debug Tools Help
Java ▾
sketch_190104a
Console
Errors

```
void
setup(  )
{
  size( 800, 600 );
  colorMode( RGB );
  background( 200, 200, 255 );
}


void
draw(  )
{
  stroke( 0, 0, 0 );
  fill( 255, 50, 50  );
  rect( 100,  200,    150,  50 );
}
```

Running Your Processing Programs

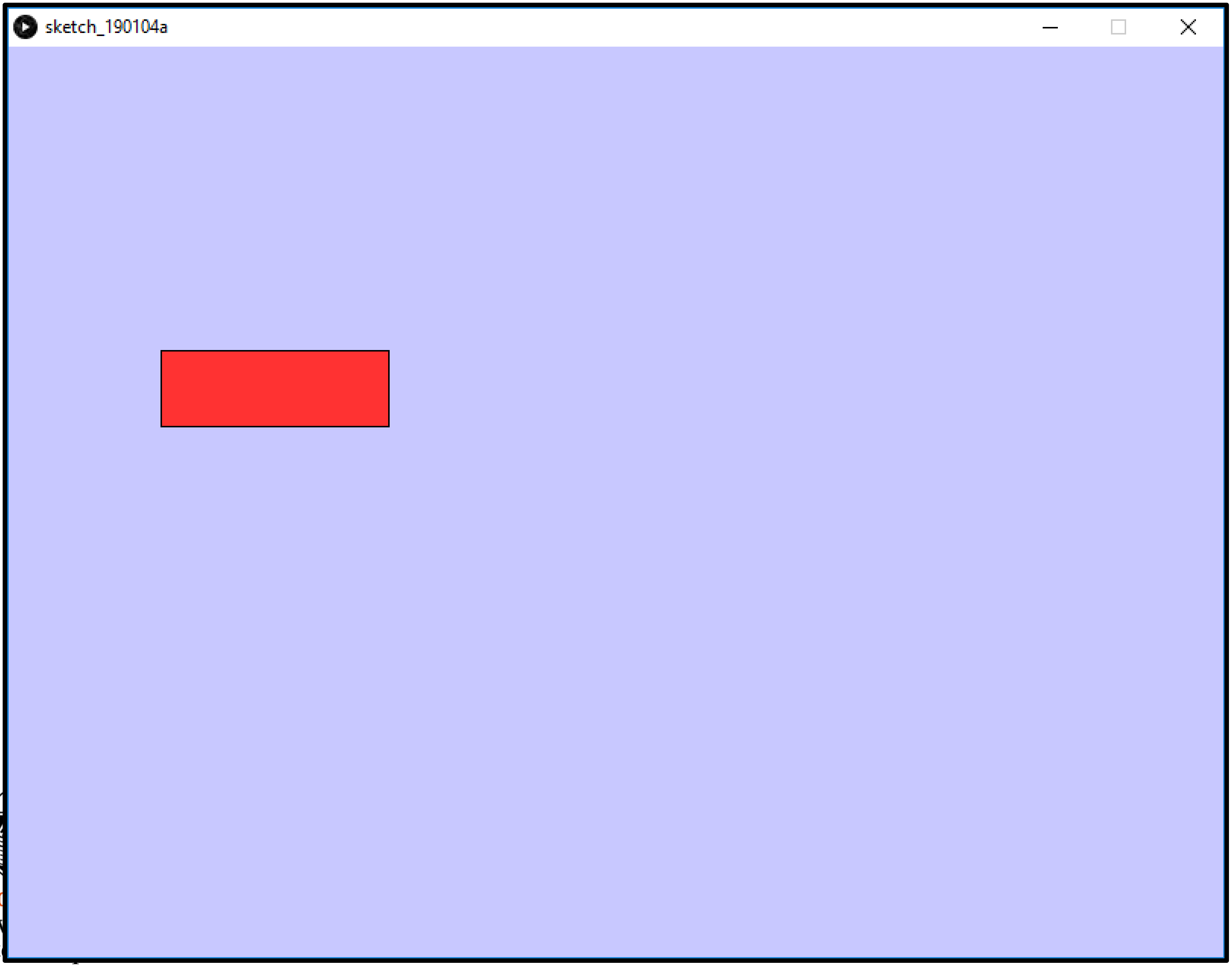
Click here to run your program


sketch_190104a | Processing 3.4
File Edit Sketch Debug Tools Help
sketch_190104a

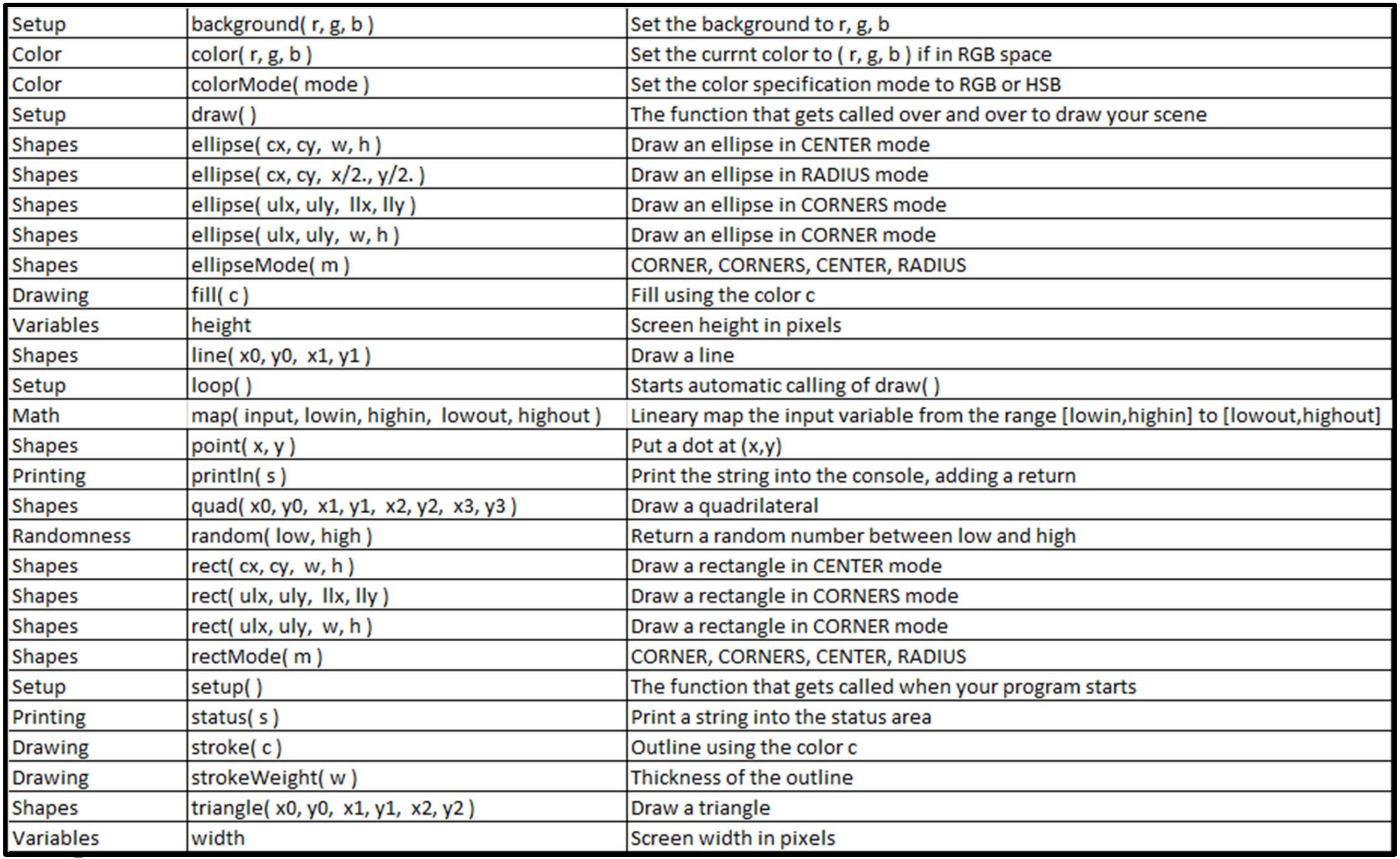
```
void
setup(  )
{
  size( 800, 600 );
  colorMode( RGB );
  background( 200, 200, 255 );
}


void
draw(  )
{
  stroke( 0, 0, 0 );
  fill( 255, 50, 50  );
  rect( 100,  200,    150,   50 );
}
```

Enjoying the Output of Your Processing Programs

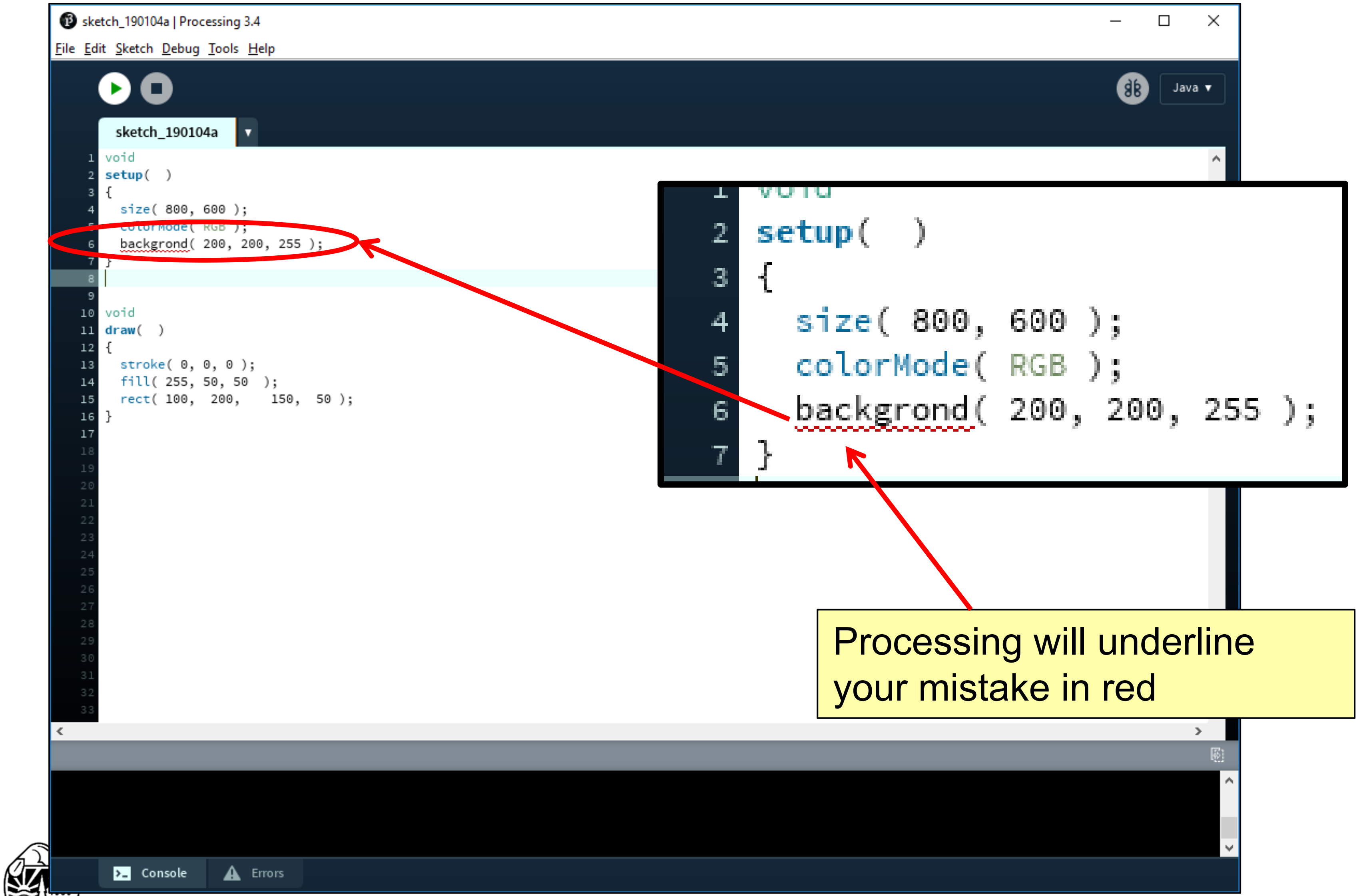


Other Functions to use when Writing Processing Programs



Setup	background(r, g, b)	Set the background to r, g, b
Color	color(r, g, b)	Set the currnt color to (r, g, b) if in RGB space
Color	colorMode(mode)	Set the color specification mode to RGB or HSB
Setup	draw( )	The function that gets called over and over to draw your scene
Shapes	ellipse(cx, cy, w, h)	Draw an ellipse in CENTER mode
Shapes	ellipse(cx, cy, x/2., y/2.)	Draw an ellipse in RADIUS mode
Shapes	ellipse(ulx, uly, llx, lly)	Draw an ellipse in CORNERS mode
Shapes	ellipse(ulx, uly, w, h)	Draw an ellipse in CORNER mode
Shapes	ellipseMode(m)	CORNER, CORNERS, CENTER, RADIUS
Drawing	fill(c)	Fill using the color c
Variables	height	Screen height in pixels
Shapes	line(x0, y0, x1, y1)	Draw a line
Setup	loop( )	Starts automatic calling of draw( )
Math	map(input, lowin, highin, lowout, highout)	Lineary map the input variable from the range [lowin,highin] to [lowout,highout]
Shapes	point(x, y)	Put a dot at (x,y)
Printing	println(s)	Print the string into the console, adding a return
Shapes	quad(x0, y0, x1, y1, x2, y2, x3, y3)	Draw a quadrilateral
Randomness	random(low, high)	Return a random number between low and high
Shapes	rect(cx, cy, w, h)	Draw a rectangle in CENTER mode
Shapes	rect(ulx, uly, llx, lly)	Draw a rectangle in CORNERS mode
Shapes	rect(ulx, uly, w, h)	Draw a rectangle in CORNER mode
Shapes	rectMode(m)	CORNER, CORNERS, CENTER, RADIUS
Setup	setup( )	The function that gets called when your program starts
Printing	status(s)	Print a string into the status area
Drawing	stroke(c)	Outline using the color c
Drawing	strokeWeight(w)	Thickness of the outline
Shapes	triangle(x0, y0, x1, y1, x2, y2)	Draw a triangle
Variables	width	Screen width in pixels

What if You Mis-type Something?

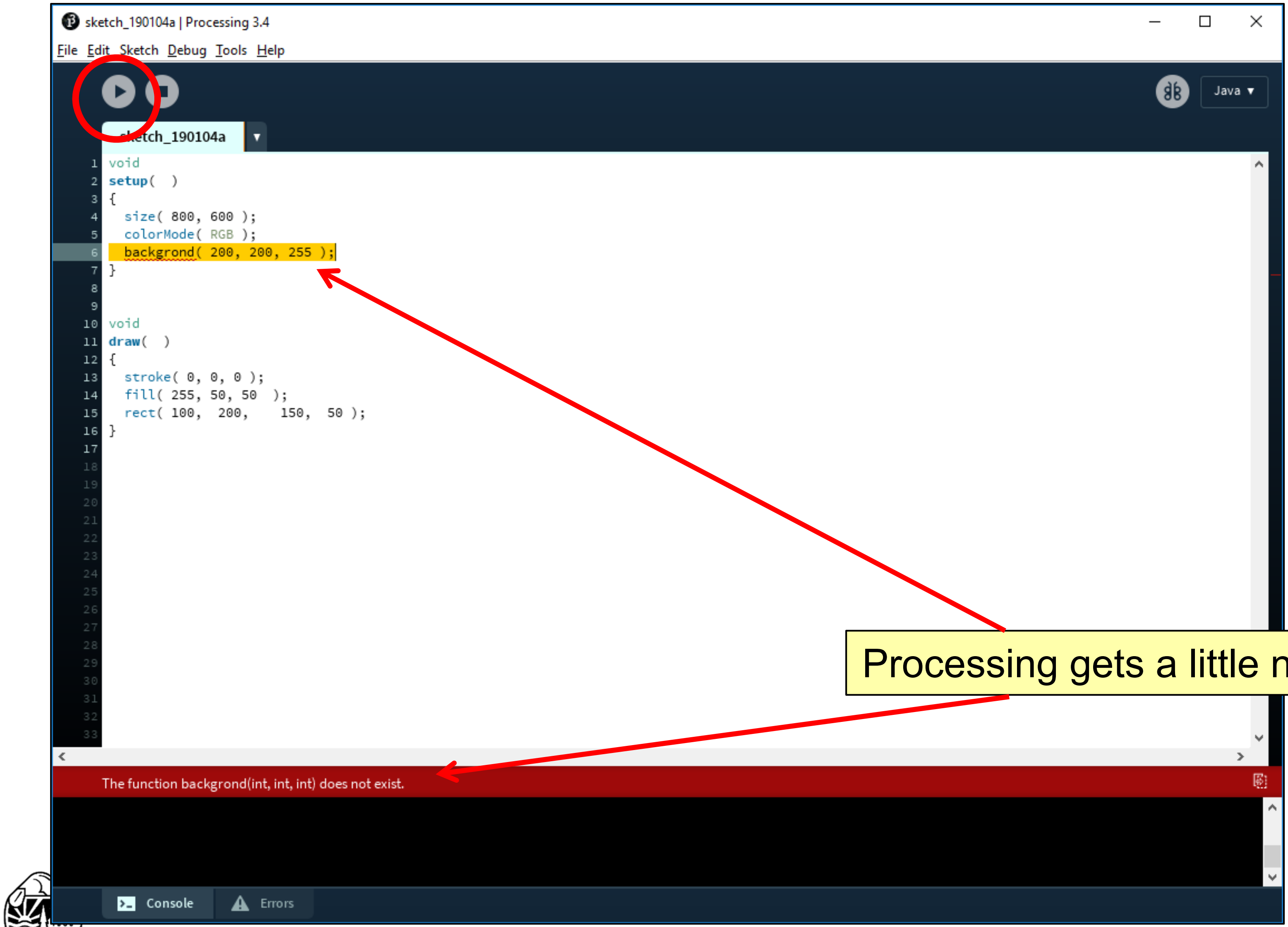

sketch_190104a | Processing 3.4
File Edit Sketch Debug Tools Help
Java ▾
sketch_190104a
Console
Errors

```
void
setup(  )
{
  size( 800, 600 );
  colorMode( RGB );
  backgrond( 200, 200, 255 );
}


void
draw(  )
{
  stroke( 0, 0, 0 );
  fill( 255, 50, 50  );
  rect( 100,  200,    150,  50 );
}
```

Processing will underline your mistake in red

What if You Try to Run it Anyway?



sketch_190104a | Processing 3.4
File Edit Sketch Debug Tools Help
Java ▾
sketch_190104a
Console
Errors

```
void
setup(  )
{
  size( 800, 600 );
  colorMode( RGB );
  backgrond( 200, 200, 255 );
}


void
draw(  )
{
  stroke( 0, 0, 0 );
  fill( 255, 50, 50  );
  rect( 100,  200,    150,  50 );
}
```

```
The function backgrond(int, int, int) does not exist.
```

Processing gets a little nastier