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”

Red
Yellow
Magenta
White
Cyan
Blue

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

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 \leq r, g, b \leq 255
```

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

```
0 \leq h, s, b \leq 255
```

---

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

```
Hue
```

```
White
```

```
Saturation
```

```
Black
```

```
Brightness
```

```
8
```

```
0 \leq h, s, b \leq 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 must add code to the draw( ) function. Processing calls this every time it wants to re-draw the scene.

Typing in Processing Programs
Running Your Processing Programs

Click here to run your program

Enjoying the Output of Your Processing Programs
Other Functions to use when Writing Processing Programs

- `background( r, g, b )`: Set the background to r, g, b
- `color( r, g, b )`: Set the current color to (r, g, b) if in RGB space
- `colorMode( mode )`: Set the color specification mode to RGB or HSV
- `draw()`: The function that gets called over and over to draw your scene
- `fill( c )`: Fill using the color c
- `height()`: Screen height in pixels
- `line( x0, y0, x1, y1 )`: Draw a line
- `point( x, y )`: Put a dot at (x, y)
- `println( s )`: Print the string into the console, adding a return
- `quad( x0, y0, x1, y1, x2, y2, x3, y3 )`: Draw a quadrilateral
- `random( low, high )`: Return a random number between four and High
- `rect( ox, oy, w, h )`: Draw a rectangle in CENTER mode
- `rect( ub, ul, bx, by )`: Draw a rectangle in CORNERS mode
- `rectMode( mode )`: CORNER, CORNERS, CENTER, RADIUS
- `setup()`: The function that gets called when your program starts
- `status( s )`: Print a string into the status area
- `stroke( c )`: Outline using the color c
- `strokeWeight( w )`: Thickness of the outline
- `width()`: Screen width in pixels

What if You Mis-type Something?

Processing will underline your mistake in red.
What if You Try to Run it Anyway?

Processing gets a little nastier