CSC103 How Computers Work

Week 6 — Fall 2017
Ben Fry on Processing...

http://www.youtube.com/watch?v=z-g-cWDnUdU
An Example

- Mouse 2D Example
  http://processing.org/examples/mouse2d.html
Another Example

http://wearechopchop.com/%E2%80%9Cunnamed-soundsculpture%E2%80%9D/

Sophisticated Coding
Reference. Processing was designed to be a flexible software sketchbook.

**Structure**
- () (parentheses)
- , (comma)
- . (dot)
- /* */ (multiline comment)
- /** */ (doc comment)
- // (comment)
- ; (semicolon)
- = (assign)
- [] (array access)
- {} (curly braces)
- catch
- class
- draw()
- exit()
- extends
- false
- final
- implements
- import
- loop()
- new

**Shape**
- createShape()
- loadShape()
- PShape
- arc()
- ellipse()
- line()
- point()
- quad()
- rect()
- triangle()
- Curves
- bezier()
- bezierDetail()
- bezierPoint()
- bezierTangent()
- curve()
- curveDetail()
- curvePoint()

**Color**
- Setting
  - background()
  - clear()
  - colorMode()
  - fill()
  - noFill()
  - noStroke()
  - stroke()
- Creating & Reading
  - alpha()
  - blue()
  - brightness()
  - color()
  - green()
  - hue()
  - lerpColor()
  - red()
  - saturation()
Examples. Short, prototypical programs exploring the basics of programming with Processing.

These examples are running online through p5.js using HTML Canvas for rendering. There are many more examples included with the Processing application; please look there if you don't find what you're looking for here.

Basic Examples. Programs about form, data, images, color, typography, and more...
**Examples.** Short, prototypical programs exploring the basics of programming with Processing.

These examples are running online through p5.js using HTML Canvas for rendering. There are many more examples included with the Processing application; please look there if you don't find what you're looking for here.

**Basic Examples.** Programs about form, data, images, color, typography, and more...

<table>
<thead>
<tr>
<th>Structure</th>
<th>Image</th>
<th>Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statements and Comments</td>
<td>Load and Display Image</td>
<td>Mouse 1D</td>
</tr>
<tr>
<td>Coordinates</td>
<td>Background Image</td>
<td>Mouse 2D</td>
</tr>
<tr>
<td>Width and Height</td>
<td>Transparency</td>
<td>MousePress</td>
</tr>
<tr>
<td>Setup and Draw</td>
<td>Alphamask</td>
<td>Mouse Signals</td>
</tr>
<tr>
<td>No Loop</td>
<td>CreateImage</td>
<td>Easing</td>
</tr>
<tr>
<td>Loop</td>
<td>Pointillism</td>
<td>Constrain</td>
</tr>
<tr>
<td>Redraw</td>
<td>Request Image</td>
<td>Storing Input</td>
</tr>
<tr>
<td>Functions</td>
<td></td>
<td>Mouse Functions</td>
</tr>
<tr>
<td>Recursion</td>
<td></td>
<td>Keyboard</td>
</tr>
<tr>
<td>CreateGraphics</td>
<td></td>
<td>Keyboard Functions</td>
</tr>
<tr>
<td>Form</td>
<td></td>
<td>Milliseconds</td>
</tr>
<tr>
<td>Points and Lines</td>
<td><strong>Color</strong></td>
<td>Clock</td>
</tr>
<tr>
<td></td>
<td>Hue</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Brightness</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Color Variables</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Transform</strong></td>
<td></td>
</tr>
</tbody>
</table>
Dan Shiffman

http://hello.processing.org/
The Sketch

- Integrated Development Environment (IDE)
- Run/Stop
- Name of sketch
- Programming area (editor)
- Program
The Sketch

```java
// circles
// D. Thiebaut

void setup() {
  size(500, 500);
  smooth();
}

void draw() {
  ellipse(mouseX, mouseY, 80, 80);
}
```
The Sketch
The Sketch
// circles
// D. Thiebaut
// displays circles following the
// mouse pointer.

void setup() {
    size( 500, 500 );
    smooth();
}

void draw() {
    ellipse( mouseX, mouseY, 80, 80 );
}
void setup() {
    size( 500, 500 );
    smooth();
}

void draw() {
    ellipse( mouseX, mouseY, 80, 80 );
}
Program Organization

Run

```cpp
// circles
// D. Thiebaut

void setup() {
  size( 500, 500 );
  smooth();
}

void draw() {
  ellipse( mouseX, mouseY, 80, 80 );
}
```

- Comments
- Functions
- `setup()`: once
- `draw()`: many times/sec
Program Organization

Run

// circles
// D. Thiebaut

void setup() {
    size( 500, 500 );
    smooth();
}

void draw() {
    ellipse( mouseX, mouseY, 80, 80 );
}

- Comments
- Functions
- `setup()`: once
- `draw()`: many times/sec
Program Organization

Run

// circles
// D. Thiebaut

void setup() {
  size( 500, 500 );
  smooth();
}

void draw() {
  ellipse( mouseX, mouseY, 80, 80 );
}

• Comments
• Functions
• setup(): once
• draw(): many times/sec
void setup() {
    size( 500, 500 );
    smooth();
}

void draw() {
    ellipse( mouseX, mouseY, 80, 80 );
}
Understanding `mouseX` & `mouseY`
Understanding $\text{size}(w,h)$
Understanding `background()`
Exercise

replace the circles by squares
Exercise

Mix circles and squares. Circles only move horizontally with the mouse.
Exercise

Circles move horizontally with the mouse, and squares move vertically.
Using Random Numbers
The random() function

- [https://processing.org/reference/random_.html](https://processing.org/reference/random_.html)

**Description**
Generates random numbers. Each time the `random()` function is called, it returns an unexpected value within the specified range. If only one parameter is passed to the function, it will return a float between zero and the value of the high parameter. For example, `random(5)` returns values between 0 and 5 (starting at zero, and up to, but not including, 5).

If two parameters are specified, the function will return a float with a value between the two values. For example, `random(-5, 10.2)` returns values starting at -5 and up to (but not including) 10.2. To convert a floating-point random number to an integer, use the `int()` function.

**Syntax**
- `random(high)`
- `random(low, high)`
Printing `random()`
Exercise

Replace mouseX and mouseY by calls to random(500)…
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors

Pixel

{ 23 red
   13 green
   19 blue }
RGB System

• Each pixel corresponds to 3 numbers
• 1st number = shade of red in the color of the pixel
• 2nd number = shade of green in the color of the pixel
• 3rd number = shade of blue in the color of the pixel
• Red-Green-Blue = RGB
• Each shade is represented by 8 bits
# RGB System

<table>
<thead>
<tr>
<th>1 bit</th>
<th>4 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000</td>
</tr>
<tr>
<td>1</td>
<td>0001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3 bits</th>
<th>8 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>00000000</td>
</tr>
<tr>
<td>001</td>
<td>00000001</td>
</tr>
<tr>
<td>010</td>
<td>00000010</td>
</tr>
<tr>
<td>011</td>
<td>00000011</td>
</tr>
<tr>
<td>100</td>
<td></td>
</tr>
<tr>
<td>101</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2 bits</th>
<th>4 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0100</td>
</tr>
<tr>
<td>01</td>
<td>0101</td>
</tr>
<tr>
<td>10</td>
<td>0110</td>
</tr>
<tr>
<td>11</td>
<td>0111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4 bits</th>
<th>8 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>00000000</td>
</tr>
<tr>
<td>0001</td>
<td>00000001</td>
</tr>
<tr>
<td>0010</td>
<td>00000010</td>
</tr>
<tr>
<td>0011</td>
<td>00000011</td>
</tr>
<tr>
<td>0100</td>
<td></td>
</tr>
<tr>
<td>0101</td>
<td></td>
</tr>
<tr>
<td>0110</td>
<td></td>
</tr>
<tr>
<td>0111</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td></td>
</tr>
<tr>
<td>1001</td>
<td></td>
</tr>
<tr>
<td>1010</td>
<td></td>
</tr>
<tr>
<td>1011</td>
<td></td>
</tr>
<tr>
<td>1100</td>
<td></td>
</tr>
<tr>
<td>1101</td>
<td></td>
</tr>
<tr>
<td>1110</td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td></td>
</tr>
</tbody>
</table>
Color Selector

D. Thiebaut, Computer Science, Smith College
Good Reference on Color

- Dan Shiffman's page: https://processing.org/tutorials/color/
Example

Make the circles blue, and the squares red.
Exercise

Emulate this graphic output
Exercise

Emulate this graphic output

Note that some ellipses are flat, others are tall and skinny...
Exercise

Emulate this graphic output

Note that all ellipses are perfect circles…
Variables
Variables

• They contain data

• **Real** numbers, **integer** numbers, **characters**

• Variables have a name

• Names always start with a letter or an underscore, and can contain letters or digits: x, y, i, alpha, r2d2, c3pO, x1, pi…
```
void setup() {
  size( 500, 500 );
  smooth();
}

void draw() {
  float radius = random( 180 );
  float red = random( 255 );
  float green = random( 255 );
  float blue = random( 255 );
  fill( red, green, blue );
  ellipse( random(500), random(500), radius, radius );
}
```
float x = 3.5;
float y = 5.2;
float sum;
sum = x + y;
println( "sum = " + sum );
Types

- float
- int
- String  
  (see https://processing.org/reference/String.html)
- boolean
// define geometry of rectangle
float w = random(100, 300);
float h = random(100, 300);
float x = random(500);
float y = random(500);

// draw a transparent grey version of it
// at 5,5, offset from mouse cursor
noStroke();
fill(200, 200, 200, 100);
rect(x+5, y+5, w, h);

// draw a randomly colored rectangle at mouse
// cursor
float red = random(255);
float green = random(255);
float blue = random(255);
fill(red, green, blue);
rect(x, y, w, h);
Loops

(See https://processing.org/reference/for.html)
Example 1

```java
for ( int i=0;  i <= 10;  i++ ) {
    println( i );
}
```
for ( int i=0; i <= 10; i++ ) {
    println( i );
}

Example 2

```java
int sum = 0;
for ( int counter=0; counter <= 10; counter++ ) {
    sum = sum + counter;
}
println( "sum = ", sum )
```
Example 3

```java
void draw() {
    fill( 200, 100, 0 );

    for ( int x=50; x<500; x = x + 100 ) {
        ellipse( x, 100, 40, 40 );
    }
}
```
```java
void draw() {
    fill( 200, 100, 0 );

    for ( int x=50; x<500; x = x + 100 ) {
        ellipse( x, 100, 40, 40 );
    }
}
```
Example 5

```c
void draw() {
    fill( 200, 100, 0 );
    for ( int x=50; x<500; x = x + 100 ) {
        for ( int y=50; y<500; y = y + 100 ) {
            ellipse( x, y, 40, 40 );
        }
    }
}
```
Testing Quantities