2D Shapes

Creative Coding & Generative Art in Processing 2
Ira Greenberg, Dianna Xu, Deepak Kumar

Review: Drawing Basics

- Canvas
  size(width, height)
- Drawing Tools
  point(x, y)
  line(x1, y1, x2, y2)
  triangle(x1, y1, x2, y2, x3, y3)
  rect(x, y, width, height)
  ellipse(x, y, width, height)
  arc(x, y, width, height, startAngle, endAngle)
  curve(x1, y1, x2, y2, p1, c1, p2, c2)
  curveTo(x1, y1, x2, y2, p1, c1, p2, c2)
  beginShape() endShape(mode)
  vertex(x, y)
  curveVertex(x, y)
- Colors
  grayscale[0..255], RGB[0..255], [0..255], [0..255], alpha[0..255]
  background(color)
- Drawing & Shape Attributes
  stroke(color), noStroke(), strokeWeight(pixelWidth)
  fill(color), noFill()

Simple Program Structure

// Create and set canvas
size(width, height);
smooth();
background(color);

// Draw something
--
// Draw something else
--
// etc.

Variables: Naming Values

- Values
  42, 3.14159, 2013, "Hi, my name is Joe!", true, false, etc.
  - Numbers
    - Integers
      int meaningOfLife = 42;
    - Int year = 2013;
    - Floating point numbers
      float pi = 3.14159;
  - Strings
    String greeting = "Hi, my name is Joe!";
  - Boolean
    boolean keyPressed = true;

Simple Program Structure

// Draw a simple house
// Create and set canvas
size(300, 300);
smooth();
background(187, 193, 127);

// Wall
fill(206, 224, 14);
rect(50, 150, 200, 100);

// Door
fill(72, 26, 2);
rect(125, 200, 50, 50);

// Roof
fill(224, 14, 14);
triangle(50, 150, 150, 50, 250, 150);

Variables: Naming Values

- Values
  42, 3.14159, 2013, "Hi, my name is Joe!", true, false, etc.
  - Numbers
    - Integers
      int meaningOfLife = 42;
    - Int year = 2013;
    - Floating point numbers
      float pi = 3.14159;
  - Strings
    String greeting = "Hi, my name is Joe!";
  - Boolean
    boolean keyPressed = true;
Variables: Naming Values

- **Values**
  - Integers
    - `int meaningOfLife = 42;`
  - Floating point numbers
    - `float pi = 3.14159;`
  - Strings
    - `String greeting = "Hi, my name is Joe!";`
  - Boolean
    - `boolean keyPressed = true;`

Variables have a Name

- **Names begin with a letter, an underscore ( ), or a dollar sign ($)**
  - Examples: `weight`, `_meaningOfLife`, `$value`
- **Names may include numbers, but only after the initial character**
  - Examples: `value1`, `score5`, `5bestFriends`
- **No spaces are permitted in names**
  - Examples: `value 1`, `dollar sign`
- **Processing Conventions**
  - Names begin with a lowercase letter
    - Example: `meaningOfLife`, `highestScore`
  - Constants are written in all caps
    - Example: `DAYS_IN_WEEK`, `PI`

Variables: Declaring & Initialization

- **Declaring variables**
  - `int meaningOfLife;`
  - `int year;`
  - `float pi;`
  - `String greeting;`
  - `boolean keyPressed;`
- **Initializing values in declarations**
  - `int meaningOfLife = 42;`
  - `int year = 2013;`
  - `float pi = 3.14159;`
  - `String greeting = "Hi, my name is Joe!";`
  - `boolean keyPressed = true;`

Expressions: Doing Arithmetic

- **Assignment statement**
  - `<variable> = <expression>;
  - Example: `meaningOfLife = 42;`
  - `area = length * height;`
  - `perc = statePop/totalPop*100.0;`
- **Operators**
  - `+` (addition)
  - `-` (subtraction)
  - `*` (multiplication)
  - `/` (division)
  - `%` (modulus)
  - Example: `mouth_x = (leftIris_x + irisDiam/2 + eyeWidth) / 4;`

The color type

- **Processing has a type called color**
  - `color firebrick = color(178, 34, 34);`
  - `color chartreuse = color(127, 255, 0);`
  - `color fuchsia = color(255, 0, 255);`
  - `fill(firebrick);`
  - `rect(50, 100, 75, 125);`

Using Variables

```java
// Draw a simple house
// Create and set canvas
size(300, 300);
smooth();
background(187, 193, 127);

// wall
fill(206, 224, 14);
rect(50, 150, 200, 100);

// Draw Door
fill(72, 26, 2);
rect(125, 200, 50, 50);

// Draw roof
fill(224, 14, 14);
triangle(50, 150, 150, 50, 250, 150);
```
Arithmetic with `int` and `float` values

```
int x = 42;   vs   int x = 42.0;
float x = 42.0 vs float x = 42;
float x = 7/2; vs float x = 7.0/1.0;
```

- Type of variable is important and determines the value that can be assigned to it.
- Result of division depends upon operands

<table>
<thead>
<tr>
<th>Type of Operation</th>
<th>Examples</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>int/int</code></td>
<td></td>
<td>yields an integer result</td>
</tr>
<tr>
<td><code>float/int</code></td>
<td></td>
<td>yields a float result</td>
</tr>
<tr>
<td><code>int/float</code></td>
<td></td>
<td>yields a float result</td>
</tr>
</tbody>
</table>

Processing: Predefined Variables

- `width, height`  
The width & height of the canvas used in the sketch
- `PI, HALF_PI, TWO_PI`  
  For different values of π. Note that
  
  ```
  HALF_PI = PI/2
  TWO_PI = 2*PI
  ```
- `displayWidth, displayHeight`  
The width and height of the monitor being used. This is useful in running fullscreen sketches using:
  ```
  size(displayWidth, displayHeight);
  ```
- `mouseX, mouseY`  
The current mouse location in sketch (...coming soon!)