# Collaborative Processing

Part 1 October 2016

### What is Processing?

"Processing is a programming language, a development environment, and an online community"

#### Reasons to use Processing:

Well documented

Many examples and libraries

A community around it

Very easy to get started

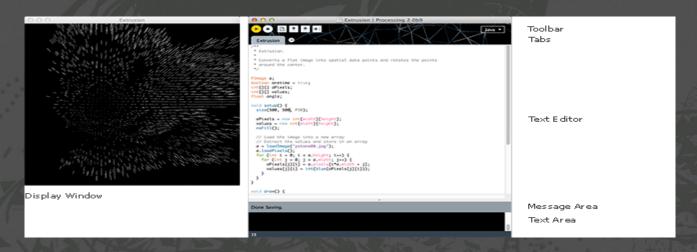
Free, Open Source and multi-platform

Can be used for professional work

#### The Processing Environment

"The Processing Environment includes a text editor, a compiler, and a display window"

"It enables the creation of software within a carefully designed set of constraints"



https://processing.org/reference/environment/

### Processing help and links

IDE Reference Examples menu

#### Online

http://processing.org

http://forum.processing.org

http://openprocessing.org

http://funprogramming.org

http://fyprocessing.tumblr.com

# "Hello World" program - drawing a circle

ellipse(50, 50, 20, 20);

- 1. Look at the reference for ellipse().
  - 2. Change the size of the circle.
- 3. Place the ellipse at the top left corner, then at the center.
  - 4. What happens if you remove the semicolon?
- 5. What happens if you remove the comma? The parenthesis?
  - 6. Study https://processing.org/tutorials/drawing/

# ellipse() is a function

Functions perform tasks. To call a function (to run it, to execute it), we write its <u>name</u> followed by a pair of <u>parenthesis</u>.

Inside the parenthesis we include zero or more <u>arguments</u>, separated <u>commas</u>.

The line ends with a <u>semicolon</u>.

The Processing reference details how many arguments to use and their meaning.

The ellipse function expects 4 arguments:

ellipse(30, 30, 40, 40);

# Choosing colors

```
size(400, 400);
fill(#FF0000);
stroke(#000000);
ellipse(200, 200, 50, 50);
```

Some functions have an immediate effect on the screen (ellipse). Others like fill() and stroke() affect following drawing operations.

1. Use the Processing color selector to choose colors.

2. Try rect(), line(), strokeWeight() and background() and create a simple composition.

#### Comments

```
// beautiful
size(400, 400);
background(0, 0, 0);
fill(200, 100, 0);
rect(0, 0, 200, 200);
```

Lines that begin with // are comments. The computer ignores them.

Use comments to leave explanations to yourself.

1. What do the values "0, 0, 0" stand for?2. Put 3 small rectangles inside the existing rectangle.

# Printing (on the console)

```
println(100);
println("hello");
println(70 + 30);
println(width);
```

println() is useful for finding out what is happening in your program. It displays text in the console (not in the main screen).

### Randomness

```
// print a random number between 0 and 100
println(random(100));
// print a random number between 1000 and 2000
println(random(1000, 2000));
size(600, 400);
// print display dimensions
println(width, height);
// ellipse at random position
ellipse(random(width), random(height), 20, 20);
```

1. Create a composition using different drawing functions and randomness.

### Saving an image

```
ellipse(width/2, height/2, 40, 40);
save("circle.png");
```

The save() function saves an image to your hard drive.

Tip: to save an image with a unique name, you can use

```
save(System.currentTimeMillis() + ".png");
```