

PROCESSING FOR LANDSCAPE ARCHITECTURE

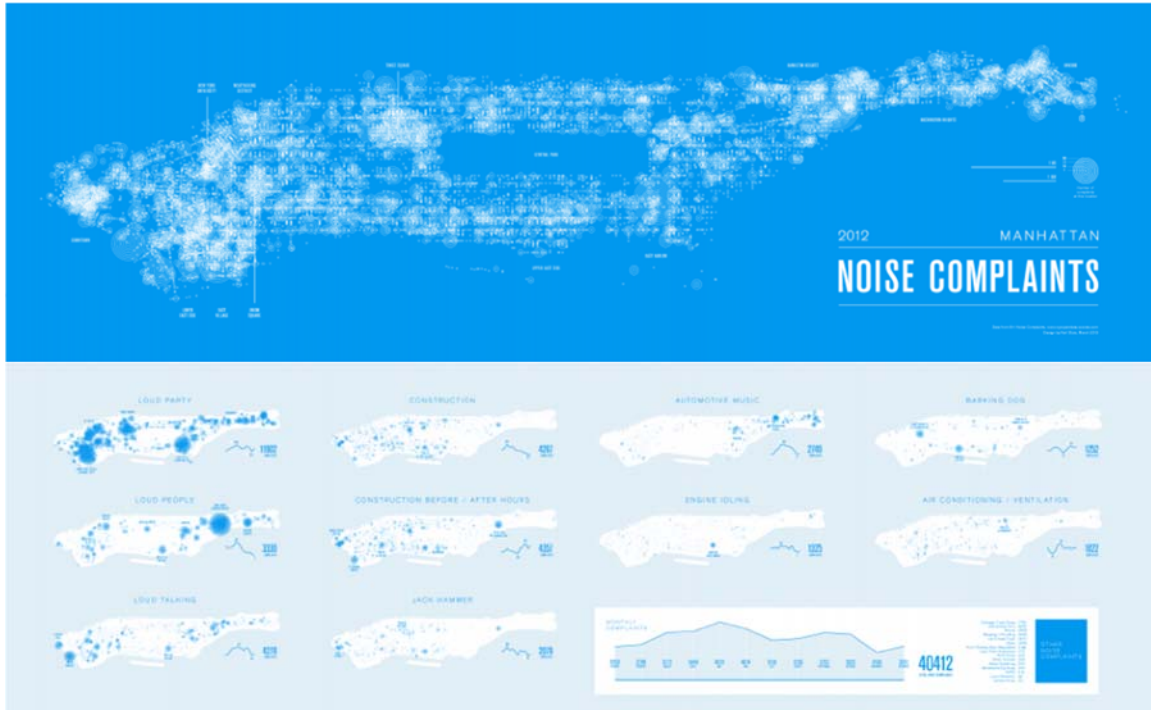
CUNY/Spitzer/MLA program

Workshops- 5hrs/each

Saturdays & Sundays, April 25, 26, 2015

noon-5pm

Instructor: Meg Studer (no office-hours, studer@siteations.com for questions)



[*Noise Complaints, 2012* by Karl Sluis using 311 call data from Open Data NYC, (visualizing.org)]

This introduction to Processing focuses on 2D data plotting and visualization. By the end of this course students will know how to a) access, manipulate, and display different data types using Processing, b) export graphics, animations, and interactions for presentation and integration with the Adobe Suite and, with a basic conceptual grasp of code, c) be able to move and choose between GIS, Processing, and Excel for communicating spatio-temporal processes.

NO CODING EXPERIENCE NECESSARY! THIS IS AN INTRODUCTION!

Landscape is awash with parameters and data—climate trends, population distributions (flora, fauna, mineral, human), species attributes, logistical networks, and so on. As designers, we must be able to communicate these relationships when explaining sited strategies and particular interventions. This introduction to Processing is designed to help you access, explore and think about how to plot relevant, publicly accessible data.

While GIS provides a great interface for spatial analysis, its statistical displays and temporal features are very limited. Processing offers an alternate plotting environment that, while offering x/y or lat/long referencing, enables you to prioritize and feature other types of relationships, time-series, and develop different relational-displays. The end goal of this workshop is to both know what's possible, given code structures and datatypes, and to get started experimenting with data displays for your semester projects.

DO NOT BE INTIMIDATED. YOU WORK AND THINK WITH THIS INFORMATION ALL THE TIME.
Processing just gives you another, integrated way of imaging it.

WORKSHOP BASICS:

Processing.org:

Download latest version at <https://www.processing.org/download/>
Processing 'plug-ins' are call Libraries- <http://www.processing.org/reference/libraries/>
Grab pieces of code from Examples- <http://www.processing.org/examples/>
Look up code and functions at References- <http://www.processing.org/reference/>

Background Readings:

Fry, Ben. *Visualizing Data: Exploring and Explaining Data with the Processing Environment*. 1st ed. O'Reilly Media, 2008.

Although we will start with different data, the core visualization processes covered in this workshop are adapted from *Visualizing Data*. The workshop tutorials will provide links to Fry's original web materials, as well as, page numbers here. *This is a highly recommended read.*

Reas, Casey, and Ben Fry. *Getting Started with Processing*. 1st ed. Make, 2010.

This will give you more background on the overall environment and structure of Processing. If you have questions about the structure of code we are using, this will walk you through the background structures in an easy to understand format.

Background Watching and Inspiration:

Shiffman, Daniel. *ICM/Learning Processing Videos*.

<https://vimeo.com/channels/introcompmedia>

Dan is the author of *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction* (1st ed. Morgan Kaufmann, 2008) and provides another good background overview.

Miscellaneous Data Representation Project samples:

The following list of links comes from the Processing Data Representation Course at NYU's Interactive Telecommunications Program. Most are working with more dynamic data than we'll cover (api twitter feeds, facebook streams, the gps/gpx iphone tracks), but they are working from the same building blocks. Many will have process videos and code for copying/editing. Explore! (Don't be overwhelmed: the post are most recent to least, skim back to the beginning of term for simpler examples).

<http://itpcourtney.com/?p=563>

<http://www.craigprotzel.com/?cat=24>

<http://supboon.com/blog/?cat=10>

<http://dougkanter.wordpress.com/category/itp/data-rep/>

Visualizing.org:

Flip through their competitions and search 'processing' to find useful examples.

Most projects will link back to author's websites and the original data source. A fair number of projects won't have their code posted, but it's great to help spur ideas for simple visualizations

<http://www.visualizing.org/explore - tools=177&sort=recent>

GitHub:

This is where developers trade info, so sign up for a free account and see what you can find (both free dataset and code)...

<https://github.com/>

<https://github.com/shiffman>

etc.

Code Source, Data Sources, Inspirational Examples at end of Syllabus (so much inspiration!)

WORKSHOP BASICS:

Structure:

Timing for Workshops: 5 hr = (3) 1.5hr sessions with two 15min breaks

General Breakdown

Session 1: Conceptual foundations and Code Syntax

(Thinking code when you know CAD, GIS, and Ai),

Very Simple Interactive Samples and Data Explorations

Session 2: Co-Coding/Assembling program pieces

Session 3: Independent review, Q & A (schedule tbd)

SESSION 1: Processing Basics- Finding, Reading, and Plotting Data

- o sample files, pdf tutorials (in the works) to be distributed before class
 - o please load the latest Processing version onto your laptops
1. VISUALIZATION- A Re-introduction:
 - a. You already work with big data (but you might not know it)
 - b. You already think parametrically (but you might not know it)
 - c. Conceptualizing Code vs. Coding from the Ground-Up (Steal like an artist) (30 minutes)
 2. THINKING LIKE PROCESSES (or any of your familiar digital drawing environments):
 - a. Processing as a plotting environment (the interface, typical program structures, libraries to extend the interface)
 - b. Code Syntax (CAD commands to processing lines, recognizing the familiar hierarchy of your existing keystrokes made EXPLICIT, internal hierarchies and code blocks)
 - c. Digital Spatial Orientations (All options/spaces to processing structures, counting and plotting as compared with CSV files)
 - d. Understanding Interaction (stable vs. dynamic code sections, key interactive shortcuts, references)
 3. USING PROCESSING TO READ DATA:
 - a. There is a ton of data but there is no 'raw' data (finding and cleaning data for beginners, preferred formats, using multiple, related files)
 - b. Delving into Data (the structure of shortcuts for reading tables, effective use of variables, think in relative difference)

(1 hr)

Breath/Stretch/Caffeinate

4. PLOTTING DATA-EXAMPLE 1-A: NYC Open Data (a simple waste management set)
 - a. Adding Materials to a Processing program (imports, file management, adapting basic display parameters)
 - b. Modifying existing code samples for typical display structures– we'll walk/type through how each an example structure is working and how it would be altered to read different data (Bars, Network graphs, Geographic plots- working through two+ of multiple sample types)
 - c. Enabling Interactions and Live Reading (refining representations to read-out data values, interactive options, export types- still and live uses)
- (2.5 hr with breaks)

Breath/Stretch/Caffeinate

Assignment- To be elaborated during Test Scenario 1 (handouts, references)

Students should forward assignments to instructor so they can be compiled for live discussion/review (min. 1 days before class)

- Students should zip their entire working pde folder. (If you don't include the internal data folder, the visualization program you edited together will not run!)
- As a matter of documenting process, write in either key or mouse-triggered saveFrame() commands and hand in a selection of shots from your running program
- Recommended working habits:
 - Comments (//tags) detailing what your code is defining are encouraged. If/When you find yourself coding for a group project, such comments will be key for clarity and file-management in shared files.
 - Test a lot and save multiples as you work (you never know when you'll put in a line of code that completely throws things off and you'll want to start with a working base if you can't debug!)
 - Processing is not particularly forgiving, but it will tell you where you've eliminated required structures. Check your spelling and typos!
- If you get stuck on a 'bug' and your code is not working, refer to both the tutorials but also the on-line resources. Try alternate code and comment-out old or dysfunctional sections (so basically show your working process and attempts, even if your final graphics aren't quite what you were hoping for...)

a) your XXXXXX.pde working file

b) the directly linked data folder of tvs, cvs, png reference files

c) Simple workflow screenshots from your programming process

d) samples of the original data source, with notes on origin, your cleaning process

e) a short explanation of how the data relates to your studio work: why you chose to display certain variables, why you chose the visualization type and how its argument aids your studio project

f) grab some adjacent and related data

SESSION 2: Multi-Data Displays and Time-Series

- See session 1 prep instructions

1. REVIEW

- a. Clarification on code references, variables, assignment questions (30 minutes)

2. EXAMPLE 1-B- MULTI-DATA PANELS:

- a. Combining display types (within spatial setup- reviewing map() functions and transform possibilities- and how to extend multi-file use)
- b. Linking information between displays- text highlights, figure highlights/color-coordination, other remote interactions from one graph to the other (1.5 hr)

Breath/Stretch/Caffeinate

3. EXAMPLE 1-C- TIME-SERIES, ANIMATION and INTERACTIONS:

- a. In Multi-Display Updates- (1-B) update tables from Interaction (using graphs as buttons, organizing tables so that user interaction loads additional data into secondary panels)
- b. Extend as controlled animation, interpolating values between table moments (multiple tables for serial, ordered display)
- c. Alternate ways to store temporal data for display- discussion and review of examples (internal code, table formats, web and database access) (3 hrs, split structure)

Breath/Stretch/Caffeinate

4. TEST SCENARIO 2: Individual problems: (builds on Test 1)-

- a. Introduction of Exercise: Students take their original plot and work on developing either:
 - i. A time-series, controlled by mouse or keyboard interactions (if they had temporal elements-internal to their file or easily collected)
 - ii. A linked, dual display with interaction (using overlapping data sets to re-contextualize their initial argument)
- b. Q & A, individual help (1.5 hrs)

Assignments- See session 1 assignment instructions

SESSION 3: Other Data Sets/Other Display Structures (DATE TBD)

- o See session 1 prep instructions

1. Questions/reviews

- a. (30 minutes)

2. EXAMPLE 2: New Data tables and simple vectors! A fresh build, combing forms

- a. Vector and shape imports (svg, scaling, stroke and fill over-rides)
- b. Constructing/combining with data Roses or a Network Graph (data tables + vectors) (1hr)

Breath/Stretch/Caffeinate

5. EXAMPLE 2-B- DISTRIBUTE AND REORDER:
 - a. Shifting Display structures- reorder grid/network based on Variable/Button Interactions (builds on week2, uses code to write out new tables for order read-in variations)
 - b. Additional table writing techniques and uses (for creating internal feedback in data manipulations covering read/write, plot/print, etc.)
 - c. Materials and Structures to be covered by request (1.5 hrs)

Breath/Stretch/Caffeinate

6. TEST SCENARIO 3: Individual problems: (builds on Test 1, 2)-
 - a. Students continue work on their data
 - b. Q & A, individual help (1.5 hrs)

Open discussion/questions

ADDITIONAL COURSE MATERIALS

INSPIRATION AND CODE

Shiffman, Daniel. *ICM/Learning Processing Videos*.

<https://vimeo.com/channels/introcompmedia>

Dan is the author of *Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction* (1st ed. Morgan Kaufmann, 2008) and provides another good background overview of the program/programming. This covers similar territory covered as in *Getting Started with Processing*.

Miscellaneous Data Representation Project samples:

The following list of links comes from the Processing Data Representation Course at NYU's Interactive Telecommunications Program. Most are working with more dynamic data than we'll cover (api twitter feeds, facebook streams, the gps/gpx iphone tracks), but they are working from the same building blocks. Many will have process videos and code for copying/editing. Explore! (Don't be overwhelmed: the post are most recent to least, skim back to the beginning of term for simpler examples).

<http://itpcourtney.com/?p=563>

<http://www.craigprotzel.com/?cat=24>

<http://supboon.com/blog/?cat=10>

<http://dougkanter.wordpress.com/category/itp/data-rep/>

Visualizing.org:

Flip through their competitions and search 'processing' to find useful examples. Most projects will link back to author's websites and the original data source. A fair number of projects won't have their code posted, but it's great to help spur ideas for simple visualizations

<http://www.visualizing.org/explore - tools=177&sort=recent>

GitHub:

This is where developers trade info, so sign up for a free account and see what you can find (both free dataset and code)...

<https://github.com/>

<https://github.com/shiffman>

etc.

Other To Be Added:

Nick Feltron

Tulp Interactive

Etc.

DATASETS AND SOURCES

To Be Added:

