DATA VISUALIZATION FOR ARCHITECTURE, URBANISM, AND THE HUMANITIES
ARCH A4892
Friday 9-11am, Spring 2019

The first class will take place at Avery 600

Instructor:
Jia Zhang
jz3077@columbia.edu
Office Hours by Appointment

ABOUT THE CLASS

This is a class that combines critical engagement with information visualization and its hands on production .

In this class, students will be introduced to key concepts in design of information visualization(and interactivity). We will put theory to practice in discussions that critically engage with the visualizations we find in the world. We will also produce visualizations by hand and by code - students should be prepared to learn and experiment with both.

The practice of visualizing data is platform/technology agnostic. However, particular practices in programming currently dominate the field of its professional practice. In this class, we will be creating visualizations that are interactive, public facing, and live on the web. Specifically, we will be learning and using the javascript library D3.

D3(data driven documents) can have creative and artful uses, but it is not primarily for creative coding. Compared to P5(Processing javascript), D3 will not have the immediacy of visual experimentation and feedback that has introduced many artists and creatives to concepts of programming. D3 has a learning curve, however it is a powerful tool and both expandable and flexible. We will be covering technical material in each class session and students are asked to complete tutorials in D3 on their own. I ask you to bear with me through the technically challenging bits so that the skills we learn can be developed further and be useful beyond the scope of the course.

It is important to keep in mind that effective visualizations are built by judicious and appropriate use for techniques and technologies, not their evident abundance within a project. Your visualization projects are never evaluated on how much D3 code you are able to incorporate.

The practice of visualizing data has a long history, but it is also constantly evolving. Students will be introduced to this history and the state of the current field through lectures and readings.

Finally the practice of visualizing data sits at the intersection multiple disciplines, it is also useful in many fields. Students are encouraged to bring both datasets and expertise from their backgrounds to class discussions and projects.

CLASS OUTCOMES

- Given a dataset, students will be able to understand the dimensions, qualities, and limitations of the dataset, and to decide on best approaches and visual representations.

- Students will understand basic programming concepts, able to build basic visualizations, but most importantly know how and where to look to learn more.
- Students will be aware of best design practices, and able to think critically about when and where to use or not use them.
- Students will come away with a set of visualizations published on the web.

CLASS STRUCTURE

Each 2 hour class centers on a theme or specific learning objective, and is roughly divided into 3 parts: 1. lecture to cover key concepts, 2. code demo to illustrate technical knowledge that students will explore further in take home assignments, and 3. A in-class activity or discussion that complements the topic that week.

COURSE SCHEDULE

Readings are subject to change

WEEK 1 JANUARY 25

Lecture:

Introduction: Why visualize? Why Code? Why interactive? Why on the Web?

A Very Brief History

Overview of Course: logistics, structure, enrollment

Demo:

the basics of a webpage

Assignment:

1. Tutorial: create and publish your github page
2. Complete Student Surveys

Reading:

1. *Interactive Data Visualization for the Web* by Scott Murray: Chapter 1 and 2: Introductions, p. 1 - 16 (Skim Chapter 3: Technology Fundamentals, it is long and used as reference)
2. From Data to the Creation of Meaning
3. What is code?

Optional

4. Doing Data Science
5. Raw Data is an Oxymoron

WEEK 2 FEBRUARY 1

Lecture:

All About Data:

origins of information, the myth of raw data, models, structures, and data in your field

Demo:

1. Technology Fundamentals
2. Introduction to programming (taught with Javascript)

Assignment:

1. Project 1 Self data collection part 1: Start Collecting!

2. Tutorial: download and setup d3, follow *Interactive Data Visualization for the Web* by Scott Murray, Chapter 4

Reading:

1. Visual Semiotics
2. Graphical Perception

WEEK 3 FEBRUARY 8

Lecture:

Visual Encoding and Graphical Perception:
How do we make design choices when it comes to dataviz?
Intentional and unintentional bad practices

Demo:

1. Introduction to SVG, drawing with data
2. Debugging in the web

Assignment:

1. Project 1 Self data collection part 2: Exploratory visualizations
2. Tutorial: Drawing with Data, based on *Interactive Data Visualization for the Web* by Scott Murray, Chapters 5 & 6

Reading:

1. Narrative Visualization
2. Understanding Comics

WEEK 4 FEBRUARY 15

Lecture:

Narratives:
narrative structure and storytelling in the context of visualization and interactivity

Demo:

1. Working with selections
2. Basic types of mouse interactivity
3. Debugging in the web review

Assignment:

1. Project 1 Self data collection Part 3: Story and Presentation
2. Tutorial: D3 scales and axes, based on *Interactive Data Visualization for the Web* by Scott Murray, Chapters 7 & 8

Reading:

1. Video: So you want to scroll
2. Video: Powers of Ten
3. TBD

WEEK 5: FEBRUARY 22

Lecture:

The Idea of Interactivity
Forms + Stories + Interactivity: Bringing things together

Short Discussion Topic: Speculative and Adversarial Practices in Data Visualization

Demo:

1. More interactivity
2. Hybrids and related multiviews

Assignment:

1. Tutorial: Basic interactivity, based on *Interactive Data Visualization for the Web* by Scott Murray, Chapter 10
2. Project 2 Public Data Part 1: Project Planning - What are available datasets from your field(interests)? How would you use them? And why are they important?

Reading:

1. Data Humanism
2. you say data, i say system OR thinking in systems introduction
3. Algorithmic bias
4. Video: Humanizing Data

WEEK 6: MARCH 1

Lecture:

Power of Representation (and underrepresentation)

Data Humanism

Introduction to Public Data Resources

Short Discussion Topic: The cost of data

Demo:

1. Data cleaning and regular expressions

Assignment:

1. Tutorial: preparing public datasets
2. Project 2 Public Data Part 2: Processing and Understanding your data

Reading:

1. New Media, New Civics?
2. TBD
3. TBD

WEEK 7: MARCH 8

Lecture:

Review of Everything So Far

Introduction to Public Data APIs

Short Discussion Topic: Issues of Accessibility, Sustainability, Reciprocality in Public Datasets and Interactive Visualizations

Demo:

1. Data cleaning and programming clinic - submit your questions

Assignment:

1. Project 2 Public Data: Presentation
2. Tutorial: Review of D3 functionality
3. Tutorial: Review of programming concepts

Reading:

1. The Death of Interactive Infographics? & In Defense of Interactive Graphics
2. Read case studies of processes

WEEK 8: MARCH 15

(MIDTERM)

Lecture:

Introduction to special topics: networks, maps, text

Planning your final project, discussions, ideas, and pitches

Demo:

1. Introduction to external libraries for special layouts
2. Introduction to special topics tools
3. Survey of student interests and priority for special topics

Assignment:

1. Final Project Proposal

Reading:

1. Visit the sites and projects of relevant people
2. Visit publications
3. TBD

SPRING BREAK MARCH 22

WEEK 9: MARCH 29

Lecture:	Special Topic 1 Networks (order subject to change)
Demo:	Networks
Assignment:	Final Project Data and Paper Prototypes Optional Tutorial: Networks
Reading:	Iterative design

WEEK 10: APRIL 5

Lecture:	Special Topic 2 Maps
Demo:	Web Maps
Assignment:	Final Project Prototype 1st Iteration Optional Tutorial: Web Maps

WEEK 11: APRIL 12

Lecture:	Special Topic 3 Text
Demo:	Working with Text
Assignment:	Final Project Prototype 2nd Iteration Optional Tutorial: Text

WEEK 12: APRIL 19

Lecture:	Desk Crits
Demo:	Troubleshooting Clinic
Assignment:	Final Project Prototype final version before guests

WEEK 13: APRIL 26

Final review with guests!

WEEK 14: MAY 3

Final Showcase!
Visit each other's projects, chat about next steps, have a good time.

Final Assignment: Publish your final project and all the documentation
Complete exit survey

EVALUATION

Participation 10%	Tutorials: 30%	Short Projects: 30%	Final Project: 30%

COURSE POLICIES

Students in this course will work in accordance to the student honor code and the statement of academic integrity for Columbia University. Please refer to the Faculty Statement of Academic Integrity[1] for all submissions of your work, and contact the instructor if you have any questions about a specific case related to the contents of this course.

HOMEWORK

Tutorials

Tutorials are assigned every 1-2 weeks and are designed to take no more than 1.5 hours to complete. They allows hands on experience with code that is addressed in class. They are cumulative and should be completed in sequence. Please contact the TA or Instructor when you encounter issues.

Submission Deadlines

Take home assignments are to be submitted before noon on Thursday via student's individual websites. The TA and instructor will start reviewing submissions at that time.

Completeness

Sometimes your code will have bugs that you cannot fix on your own. In those cases, please get in touch with your TA or instructor well ahead of the due date to address the issues together.

Code

Commenting your code and attributing code you use are required as part of every programming assignment. Writing comments inline with the code helps you to think through how to best complete particular programming tasks and also helps the instructor troubleshoot with you when there are issues.

Plagiarism in Coding

Plagiarism can be a serious issue in programming courses.[2]
In the context of this course, it is permissible to use pieces of code you find in the wild that suits your project under these circumstances:

[1] http://www.college.columbia.edu/faculty/resourcesforinstructors/academicintegrity/statement
[2] https://www.nytimes.com/2017/05/29/us/computer-science-cheating.html

1. You have checked the author’s permissions and their work is under either a creative commons, MIT, or similar license in which noncommercial use is permitted.

AND

2. That you include clear attributions of the code you incorporate from others as comments directly within your work. This includes to the extent available, the author’s name, the URL of the work, and a sentence explaining the function of the code.

Extra Credit

Throughout the semester, there will be opportunities to make presentations based on optional readings or completing code projects(demos) on special topics for extra credit throughout the term.