**Lab 3 - Interactive Mapping with Leaflet**

**Lab purpose:**

This lab marks a turning point in the quarter. To start the quarter, we moved quickly through the fundamentals of what modern web mapping is and what technologies web maps tend to use. We covered a lot of ground, especially with respect to the core of how the interactive web works, and it’s understandable if all of the material is still coming together for you; but, this lab marks a shift in focus in **two** ways for the class. On one level, we are turning our focus continually to building interactive maps from their base layers to their highest levels of interactivity. On another, now that we understand the *fun*damentals, we’ll be emphasizing and practicing how we can leverage existing knowledge (tutorials, example code, etc.) to build our own maps. Your workbook calls this “Google Fu” and feel free to refer to that if you want, but the basic idea is gaining the ability to parse through documentation and examples to adapt them to your own circumstances - this is a skill that requires significant practice, but it’s applicable to all sorts of technical environments.

In our first lab, we used QGIS to both remind ourselves how to get, process, and share spatial information; but, also, to walk through how a web map can be built using different tools. In our second, we focused on *computational thinking* in javascript - breaking apart big problems into little ones, using the ***basics*** of programming to solve problems.

This week, we’re going to hone in on getting the basics of Leaflet down. We’ll do so by *working through* examples and tutorials provided by Leaflet. You’ll be receiving ***less*** guidance ***in*** the lab and being asked to start using your developing skills (the basics of thinking computationally, the ability to read documentation, etc.) to solve problems.

This lab is in two parts and you have two weeks to complete it. I recommend that you work on the first part in week one and the second part in week two, but that’s less important for this lab than in previous weeks. Each part of the lab is worth five points and has a rubric for how those points are allocated.

**A note of caution:** This lab is meant to serve as a sort of ‘mid-term’ for you. It is a bit harder than our previous labs in that it provides less guidance. That’s intended as it’s meant to both test what skills you’ve already acquired ***and*** push you towards more independent map creation. **But**, you have plenty of help available should you need it! Come to office, send an email to schedule a meeting, etc. You have two weeks, don’t leave this to the last minute!

**Lab Deliverables and Grading Scale:**

This lab is due Monday, February 19th, by 5:00pm.

You will turn in one ‘website’ as a zipped file for your lab. This ‘website’ will

have an index.html. When unzipped and loaded on a server, index.html will

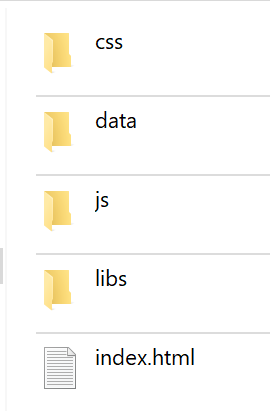
open and display two maps. One will be a point map with proportional

symbols, the other will be a choropleth map.

The file you turn in should be titled LastNameLab3.zip.

**Alternatively**, you may host the entire website on your own github account and submit a **link** on canvas. Either approach is acceptable for this lab.

You are allowed to use any file structure you wish for your website; however, your workbook and class examples all use something similar to the following and it is strongly encouraged (however, if your page loads and the maps display, you can use whatever structure you want):



**Proportional Symbols Map Rubric:**

Your first map will be a proportional symbol map of ***some*** data (In the

instructions, a data set is suggested, but you may use any you wish).

The map is worth five points total and will be graded as such:

1 pt - The map has a title (the title need not be ‘in’ the map and can be

html text above or below it.

2 pt - The map contains point data that is visualized as proportional

symbols.

1 pt - When clicked on, the proportional symbols have a pop up that

contains additional information (the exact amount, etc.)

1 pt - The symbols are loaded from a geojson file

**2** Bonus Pts - Your map has a legend internal to it (see chapter 6 of

your workbook on how to do this)

**Choropleth Map Rubric:**

Your second map is a choropleth map. It can be of the same data (you’d

need to convert it to an areal unit, how would you do that? Hint: spatial joins would be one approach) or it can be of different data. In the instructions, a data set is suggested, but you are free to use whatever you’d like.

The map is worth five points total and will be graded as such:

1 pt - The map has a title (the title need not be ‘in’ the map and can be

html text above or below it, but it needs to be clearly associated with this map and not the other)

1 pt - The map displays data in a choropleth format

1 pt - When hovered over, additional information (the precise values

and location name, for example) are displayed on the map (this can be in a set place as per the Leaflet choropleth tutorial, or a pop up which appears on hover)

1 pt - The map has a legend (see the Leaflet choropleth tutorial)

1 pt - The data for the map is loaded from a geojson file

**1-3** Bonus pt - The data used is the same as that used in your

proportional symbol map (you may prepare the geojson = i.e. handle any necessary joins - using a desktop gis software like arc or qgis; but you’ll get **two more** points if you handle it with javascript)

**Lab Instructions**

As noted, this lab serves both as an informal mid-term and an opportunity to practice your ability to find and adapt code for your own purposes. As such, the ‘instructions’ are more overarching suggestions than step-by-step directions.

**Proportional Symbols Suggested Workflow:**

**If you’ve forgotten what proportional symbols are or when to use them, Axis Maps has a very nice** [**tutorial/discussion here**](https://www.axismaps.com/guide/proportional-symbols)**.**

1. Acquire some point based data that has a scale or rank associated with it that can be visualized with proportional symbols.
   1. You could use our old friend, airports.geojson, and try to build this off of one of its rank variables (but would that be appropriate according to the above discussion?)
   2. You could also find other data. Population is a common one.
      1. I’ve put a USA\_Major\_Cities.zip file in your lab repository. It has a shapefile with US cities and their populations. You’ll need to convert it to geojson, but we’ve gone over that in class, in your workbook, etc. But, remember that error I was running into in class? If you do, how might you solve it?
      2. This is a good time to remember that part of making web maps is ‘cobbling’ different tools together to create what you want; make use of mapshape, of qgis, of arcgis, etc.
2. Create your index.html and set up a basic webmap to load in it
3. Load up your data
4. Create proportional symbols
5. Create popups, style the map, etc.

**Suggested resources:**

We’ve converted shapefiles into geojson in class (using mapshaper) and

chapter three, lesson 1 of your workbook goes over this as well.

Creating a basic map object on a web page was covered in class, in chapter4 lesson 1 of your workbook, and in the Leaflet tutorial ‘getting started’.

Loading geojson data was covered in class, in chapter 4 lesson 2 of your workbook, and in the Leaflet tutorial ‘Using GeoJSON with Leaflet’

Lesson 1 of Chapter 5 in your workbook has a ***great*** walkthrough of loading data and creating proportional symbols.

**Choropleth Suggested Workflow:**

**Just like before, if you’d like a quick refresher on choropleth maps, when to use them, and how to design them, I recommend Axis’** [**discussion here**](https://www.axismaps.com/guide/choropleth)**.**

1. Find some areal data that has values you can visualize as a choropleth.
   1. A common source here might be demographic data from the US Census.
   2. Or, you can use the data provided in the Leaflet choropleth tutorial, which can be found [here](https://leafletjs.com/examples/choropleth/us-states.js).
2. Add a new map object to your index.html and the appropriate titles, positioning, etc. (Use the DOM and CSS to your advantage here).
3. Load up your data.
4. Add Interaction
   1. Remember you need information to display when you hover over an area
5. Build a legend
6. Take a break!

**Suggested resources:**

Many of the resources used in the above map will be applicable here.

Perhaps the *most useful* will be the “Interactive Choropleth Map” tutorial that Leaflet provides [here](https://leafletjs.com/examples/choropleth/).

You’ll need to ***adapt*** the code to your situation. Lesson 2 of Chapter 4 of your workbook provides some tips on how to do this (it’s also a core learning outcome of this lab).