



Data Modeling for Systems Development

CSCE 411/811

Programming Assignment 3

Spring 2020

Build Web-based Interactive Visualization Data Models using D3

Basic Info

The programming code will be graded on **both implementation and correctness.**

There is no written report for this assignment. However, for question no. 9 you will have to document your observation and submit it as a PDF file.

Assignment Goals

This assignment is intended to build the following skills:

- Building web-based interactive visualization data models using D3
 - Understand the efficiency of using D3 by comparing it with vanilla JavaScript
-

Score Distribution

Part I: 411 & 811: 10 pts

Part II: 411 & 811: 15 pts

Part III: 411 (15 pts) & 811 (10 pts)

Total: 411 (40 pts) & 811 (50 pts)

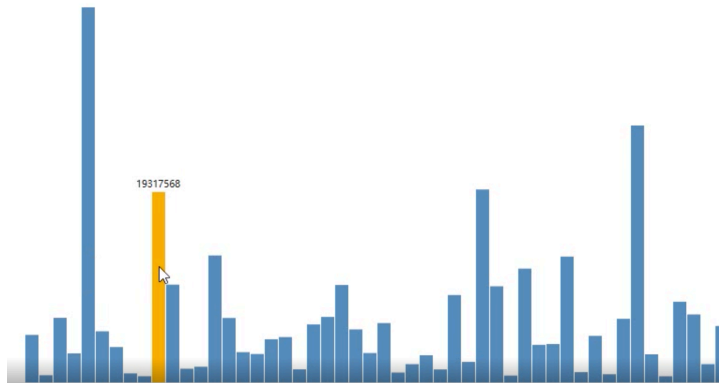
Dataset: Use the following file containing GDP and population numbers of the U.S. states: *state_population_gdp.tsv*

Part I (411 & 811: 10 pts)

Part I (a):

1. Display the bar chart of the population numbers of the states. You need to display the bars vertically. [2 pts]
2. Since the population number is rather long, the image would be busy if we display all of them. Thus, you need to implement tooltips in this case: only display the population number on the top of a bar if a user moves the mouse over the bar. The number disappears if the mouse is moved out of the bar. [3 pts]
3. In the implementation of the tooltips, you may have already tried to “get” the attributes or styles of the bar. Can you also “set” some attributes or styles of the bar? Please change the color of the bar to orange when the mouse is over the bar. Reset to the original color when mouse is moved out. [3 pts]

The visualization would look like this:



Part I (b):

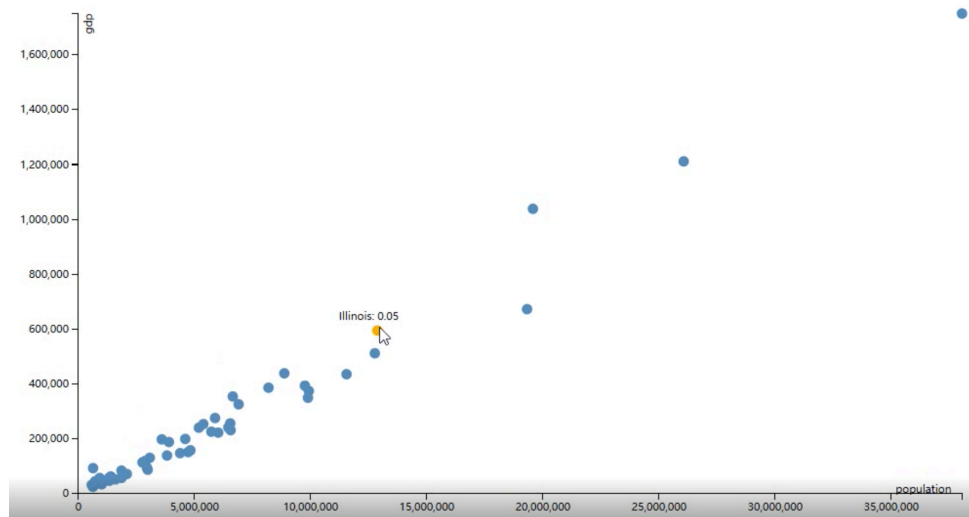
4. Then, use the same visualization to display the GDP numbers. [2 pts]

Part II: (411 & 811: 15 pts)

More insights of the data may be obtained if we can display the population and GDP together. In this step, implement a **scatter plot** of the data: the horizontal axis represents the population, the vertical axis represents the GDP, and each state is represented as a dot in this space.

5. Use a circle to represent each state. A circle can be created in a way similar to a rectangle, where you can specify the center of the circle through the attributes “cx” and “cy”, and the radius through the attribute “r”. A detailed example can be found here (<https://www.dashingd3js.com/svg-basic-shapes-and-d3js>). [6 pts]
6. When you move the mouse over a circle, instead of the original numbers, you can show some **derived number**. In this step, please show the state name and the per-capita GDP. Also change the circle color during the mouse over event. [6 pts]
7. Displaying the axes can make the visualization more meaningful. Please add the two axes. You may refer to this tutorial (<http://bost.ocks.org/mike/bar/3/>). [3 pts]

The visualization would look like this:



Part III: (411 & 811: 15 pts)

As we discussed in the lectures, one of the fundamental challenges for on-line visualization is to maintain the correspondence between data elements and visual elements, where both elements can be large-scale, complex, and dynamic; and can be presented using DOM. The D3.js is one possible solution to address this challenge.

In our lectures we also showed that on-line visualization can be implemented by employing the native JavaScript DOM methods (https://www.w3schools.com/js/js_htmlDOM_document.asp) to manage (e.g., create, append, remove, etc.) visual elements according to data elements.

8. In this step, implement the visualization in **Part I(a)** using the native JavaScript DOM methods (https://www.w3schools.com/js/js_htmlDOM_document.asp) **without D3.js**. [10 pts]
9. Compare your code in this step and the previous one in **Part I(a)** from the perspectives of the programmer (e.g., the implementation, maintenance, and extensibility of the code) and the user (e.g., the learning and deployment of the code, using it on the browser). Document your observation and explanation on the comparison. [5 pts]

[Following question is Mandatory for 811 and Extra Credit for 411]

- 10 Modify your program from **Part I(a)** to enable user to sort the bars in the ascending order. For example, after the bar chart is displayed a user could sort those by clicking the mouse pointer on a bar. [10 pts]

Some useful articles and examples:

- <https://github.com/d3/d3/blob/master/API.md#requests-d3-request>
- <https://observablehq.com/@d3/lets-make-a-bar-chart>
- <http://mbostock.github.io/d3/talk/20111116/iris-splom.html>
- <https://bost.ocks.org/mike/join/>
- <https://bost.ocks.org/mike/selection/>

Submission Guideline:

- Create following directories corresponding to Part I(a), Part I(b), Part II & Part III: *part1_a*, *part1_b*, *part2*, and *part3*.
- Put the data, html, JavaScript, and CSS files of each step in its directory so that we can open the webpage to see the result.
- A report will be also included in the part3 directory. In addition, screen capture your web browser to generate the image of the result of each step. Your web page of each step should be able to display on our web browser. Compress everything into a file (such as zip, tar.gz, etc.) and turn in the compressed file via Webhandin.