**D3 Homework - Data Journalism and D3**

Welcome to the newsroom! You've just accepted a data visualization position for a major metro paper. You're tasked with analyzing the current trends shaping people's lives, as well as creating charts, graphs, and interactive elements to help readers understand your findings. The editor wants to run a series of feature stories about the health risks facing particular demographics. She's counting on you to sniff out the first story idea by sifting through information from the U.S. Census Bureau and the Behavioral Risk Factor Surveillance System.

The data set included with the assignment is based on 2014 ACS 1-year estimates: https://data.census.gov/cedsci/, but you are free to investigate a different data set. The current data set includes data on rates of income, obesity, poverty, etc. by state. MOE stands for "margin of error."

**Before You Begin**

1. Create a new repository for this project called D3-challenge. Do not add this homework to an existing repository.
2. Clone the new repository to your computer.
3. Inside your local git repository, create a directory for the D3 challenge. Use the folder name to correspond to the challenge: D3\_data\_journalism.
4. This homeworks utilizes both html and Javascript so be sure to add all the necessary files. These will be the main files to run for analysis.
5. Push the above changes to GitHub or GitLab.

**Your Task**

**Core Assignment: D3 Dabbler (Required Assignment)**

You need to create a scatter plot between two of the data variables such as Healthcare vs. Poverty or Smokers vs. Age.

Using the D3 techniques we taught you in class, create a scatter plot that represents each state with circle elements. You'll code this graphic in the app.js file of your homework directory—make sure you pull in the data from data.csv by using the d3.csv function. Your scatter plot should ultimately appear like the image at the top of this section.

1. Include state abbreviations in the circles.
2. Create and situate your axes and labels to the left and bottom of the chart.
3. Note: You'll need to use python -m http.server to run the visualization. This will host the page at localhost:8000 in your web browser.

**1. More Data, More Dynamics**

You're going to include more demographics and more risk factors. Place additional labels in your scatter plot and give them click events so that your users can decide which data to display. Animate the transitions for your circles' locations as well as the range of your axes. Do this for two risk factors for each axis. Or, for an extreme challenge, create three for each axis. Hint: Try binding all of the CSV data to your circles. This will let you easily determine their x or y values when you click the labels.

**2. Incorporate d3-tip**

While the ticks on the axes allow us to infer approximate values for each circle, it's impossible to determine the true value without adding another layer of data. Enter tooltips: developers can implement these in their D3 graphics to reveal a specific element's data when the user hovers their cursor over the element. Add tooltips to your circles and display each tooltip with the data that the user has selected. Use the d3-tip.js plugin developed by Justin Palmer—we've already included this plugin in your assignment directory.

Check out David Gotz's example to see how you should implement tooltips with d3-tip.

**Assessment**

Your final product will be assessed on the following metrics:

1. Creation of a new repository on GitHub called D3-Challenge (note the kebab-case).
2. Completion of all steps in the core assignment
3. Coherency of scatter plot (labels, ticks)
4. Visual attraction
5. Professionalism