HW 04 - Course Catalog Analytics and Visualization

In this homework you will work with MIT's course catalog data. The URL address for the site is:

<http://student.mit.edu/catalog/index.cgi>

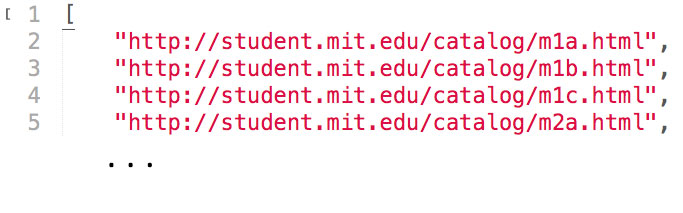
* Node.js is the virtual machine interpreter of javascript within the browser
* What is ayscn computation
* What is a request ;
* What is in a sync fashion

# Question 1. - Return an array with the links to every page

Return the address of all the html pages in the MIT course catalog - return a string array. For example, the first page for Course 1 is:

http://student.mit.edu/catalog/m1a.html

Sample Data:

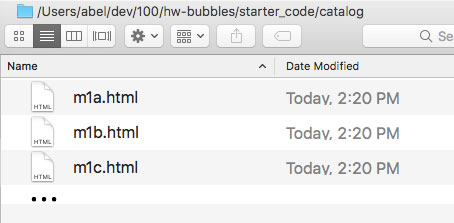


# Question 2. - Download the Data

Download every course catalog page.

You can use the NPM package request. Or curl with the NPM package shelljs.

Save every page to your\_folder/catalog:



# Question 3. - Combine all files into one

Combine all files into one, save to your\_folder/catalog/catalog.txt

You can use the file system API, https://nodejs.org/api/fs.html

# Question 4. - Remove Whitespace

Remove line breaks and whitespaces from the file. Return a string of scrubbed HTML. In other words, HTML without line breaks or whitespaces.

You can use the NPM package html-minifier.

# Question 5. - Load data into DOM, get courses

Load your scrubbed HTML into the DOM. Use the DOM structure to get all the courses.

Return an array of courses.

You can use the NPM package cheerio.

# Question 6. - Get titles

Return an array of course titles.

You can use the NPM package cheerio.

# Question 7. - Clean titles

Filter out punctuation, numbers, and common words like "and", "the", "a", etc.

Return clean array.

# Question 8. - Make words array

Make an array of words from all the titles.

Return array of words.

# Question 9. - Count the word frequency.

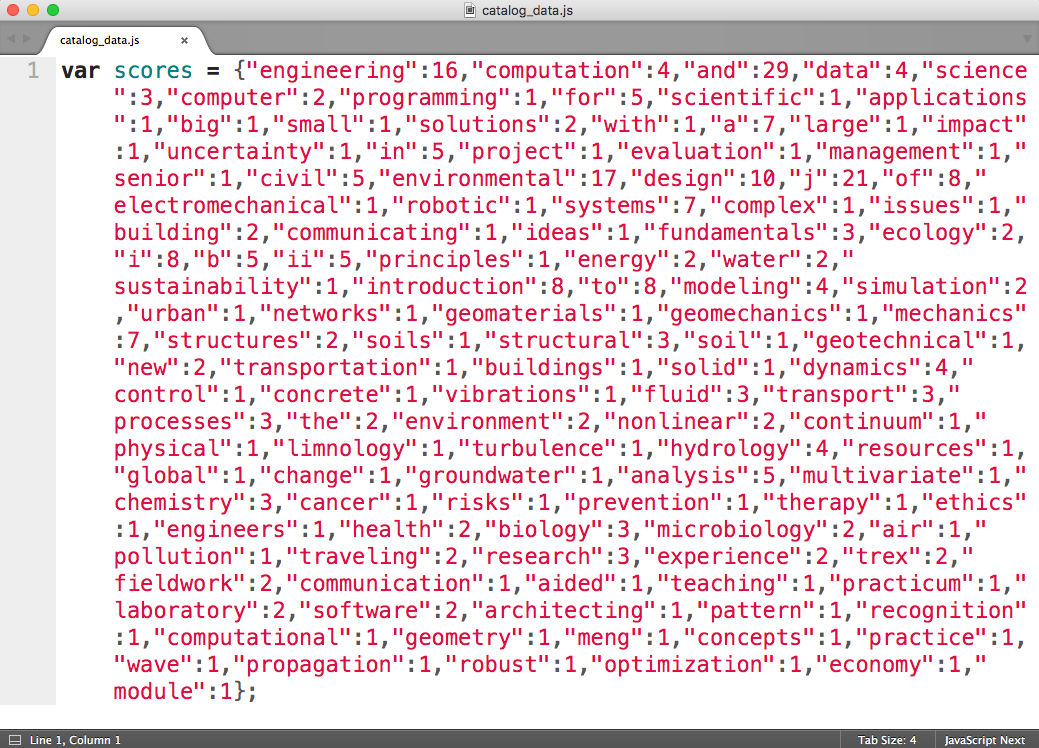
Count the word frequency. Return a word count array.

# Question 10 - Graph the word frequency

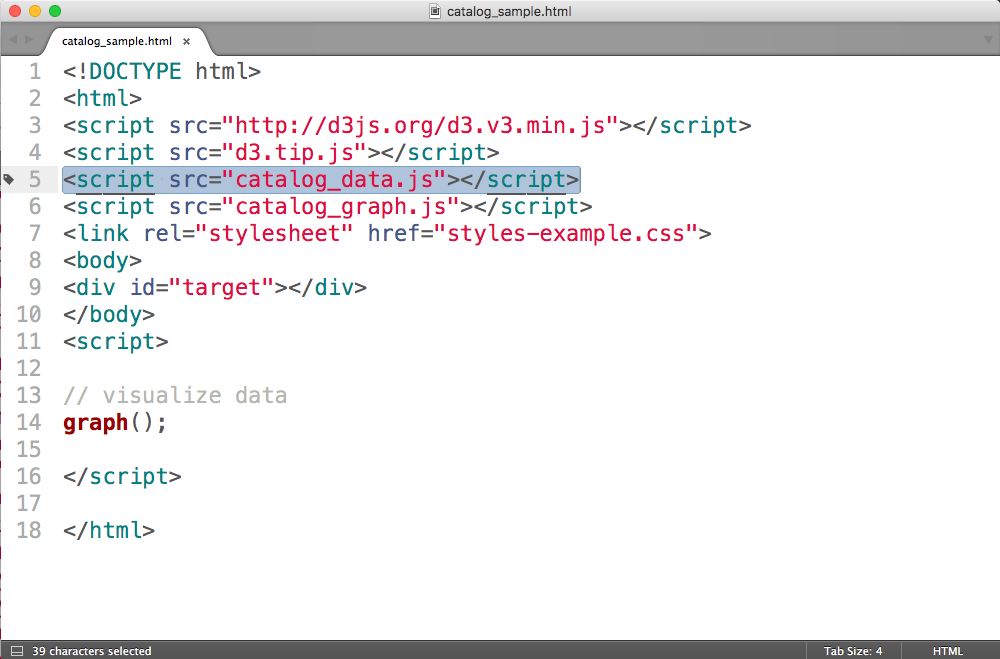
The included zip file, catalogSample.zip, contains everything you need to graph your word frequency data.

## Example

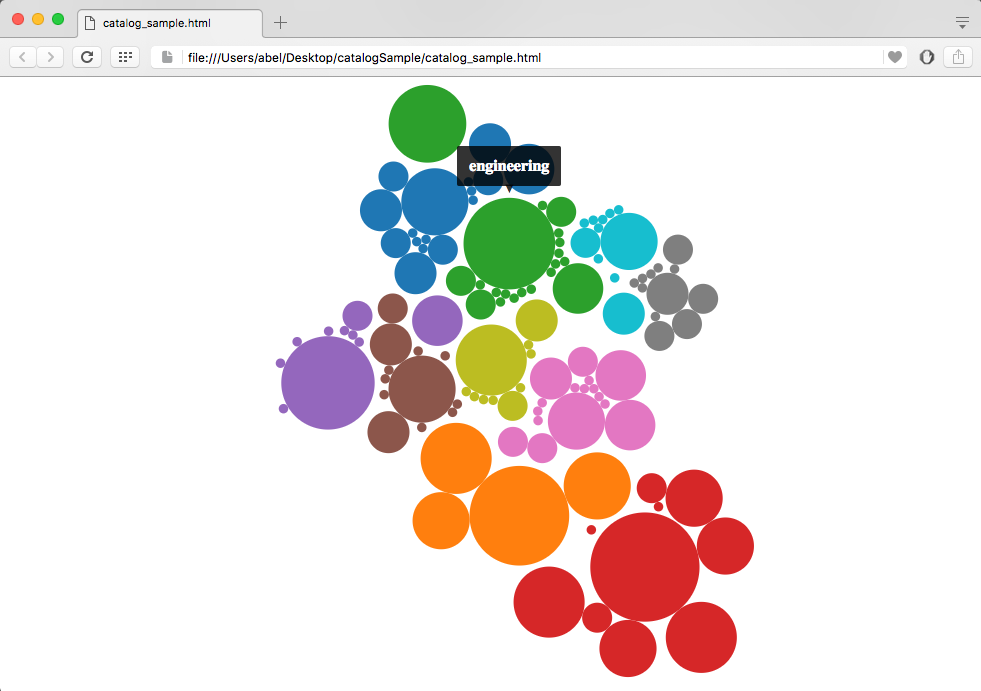
Sample word frequency data (catalog\_data.js):



Sample html file (catalog\_sample.html):

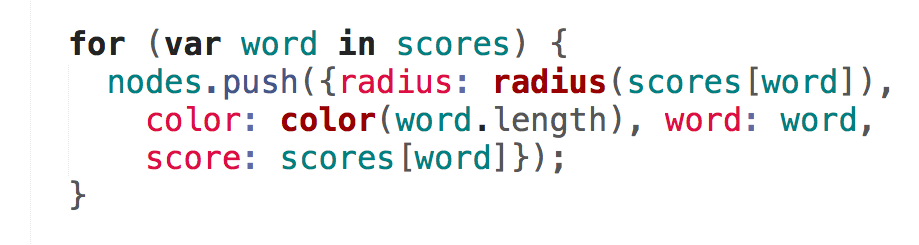


Sample html file in browser (catalog\_sample.html):



# Question 11. - (OPTIONAL) Improve Graphing Logic

The graph logic is based on the scores you calculated for words. You can find the code in the graphing JavaScript file.



Can you improve the graph?