Lab 1 - Web Systems review(-ish): HTML5, CSS3, JavaScript, JSON

**Due: Friday, January 21, before class. *Whatever is on GitHub at that time will be graded. Committing after the due date will result in the lab being marked late!***

This should help get your brain to think in code again, and perhaps spur some ideas for your projects.

A long time ago (in a galaxy far far away…), we made a Tweet ticker for Lab 1. I provided pre-scraped Tweets in JSON format. But they’re old now, and mostly broken. We can do better.

We are going to create a news ticker. You will still be using pre-scraped data, but it will be up to you from where you will scrape that data. Yes, you may use multiple news sites. Many news sites have an RSS feed, which will give you XML descriptions of news articles. I bet there are some free tools out there that can convert XML to JSON if you’d like to make your life easier and parse JSON for your labs…

You should aim to scrape **at least 200 articles** for the lab. This may take several news sites to accomplish. Again, that’s fine. Concatenate all the news items into a single JSON (or XML) file. If using JSON, you should run your finalized JSON file through a JSON linter: <https://jsonlint.com/>

Need a refresher on JSON? Find the official documentation here: <https://json.org/>

Like I mentioned before, you get to use all the fancy things right from the beginning now. Bootstrap would be quite handy for this lab. Read up on its documentation here: <https://getbootstrap.com/>

Use XAMPP for now as your web server.

**The lab:**

Create a news ticker, which will show the news items from the JSON file, five articles at a time. The ticker must slowly cycle through the articles no faster than once every three seconds. How you choose to cycle is up to you: you could replace all five articles every three seconds, replace one every three seconds, or anything in between.

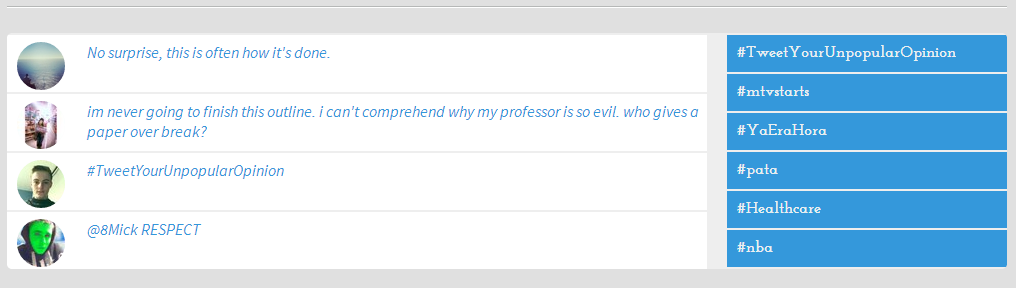
Use CSS3 transitions and animations (or jQuery animations, if you prefer) to make the articles cycle through the ticker smoothly.

Make the site responsive and clean using Bootstrap. Ideally, it should look good on both desktop and mobile. For this lab only, if you can only do one, prioritize the desktop experience. But do your best to make it look good on both. Remember the dev tools built into Chrome and Firefox! There are even buttons that squeeze everything down to phone-screen size. Things like @media-queries might be useful to getting everything right (if you’re not going to go the Bootstrap route).

Please write a README.md file that documents your thinking throughout the development process. I want to know where you got stuck, how you got unstuck, what was easy, what was difficult, etc. You should also document your individual creativity there. Remember too that you’re allowed to use tutorials and open source code if you want, but those things have to be cited in the README.md file. Additionally, open source code needs to actually follow the terms of the license (this usually means including an **unmodified copy of the license** somewhere in your repo–it is against the law to modify someone else’s license in any way, for any reason).

Did you remember to invite me and the TA to your private lab repo? We can’t grade what we don’t have access to.

**Potential example (from back when this was a Tweet ticker):**



Remember: if you copy this exactly, you get a C. Creativity is key! Study other news ticker apps out there: see what you like and what you don’t (and then cite the news ticker apps you studied in your README.md file!).

**Grading rubric:**

Formatting and code style: 10 points

Objective 1, processing JSON/XML: 10 points

Objective 2, displaying output: 10 points

Creativity: 10 points

README.md: 10 points

**Total: 50 points**