CTA API Use

API calls and DOM manipulation

Objectives

- Become familiar with DOM manipulation
- Make API calls and extract useful data from them
- Challenge yourself!

(*Read this entire document! It contains useful information. I didn't write it for fun - I could be writing code for my* own *projects right now. - Ryan*)

Description

The goal of this deliverable is to produce a single page (an HTML file, a CSS file, and one or more JavaScript files) with the purpose of showing a user some kind of information about the city of Chicago's train and/or bus system.

"Some kind of information" is intentionally ambiguous - you can decide what this means, based on your personal estimation of your skills and your desire to push yourself. The minimum *technical* requirement is that there be at least one API call to get data, and that some data from that API call's result is placed into the DOM so that a user can read the information.

You should aspire beyond the minimum, however! The true purpose of this deliverable is to get you interested in setting your own goals, then pursuing the means to achieve those goals. The best way to learn to code is to pursue projects that are interesting to *you*.

There is also a minor prize of no real value on the line for the most interesting project - how exciting! Read on for more details.

The CTA Bus Tracker API

In-person, Ryan introduced the CTA Train Tracker API to you and walked you all through signing up for it. If you're more interested in the bus system, there is a separate API for CTA's bus system available here.

What about Metra?

Metra also has an API service, available here. It appears to use Google Transit, and may be a bit more complicated than it seems at first glance. You might want to try working with the CTA train and bus data first to make sure your basic skills are firm before dealing with this.

Requirements

The HTML file must link a CSS file and at least one JavaScript file. That JavaScript file must make some kind of API call to a CTA or Metra API successfully, then use data from that API call to perform DOM manipulation to show some data in the web browser.

Delivery Method and Due Date

Your work must be completed and pushed to a repo on GitHub, and a link to that repo must be emailed to Ryan (rmagley@icstars.org) and Jess (jmoton@icstars.org) by **8pm** on **Friday, February 23rd, 2024**.

Remember that the purpose of this deliverable is your personal growth and experimentation - do not be late in turning it in because you were attempting to make it “perfect”. Prizes will be given out sometime afterwards.

A Prize?!? Tell Me More Please :^)

Don’t get too excited - it has a monetary value of something like a tenth of a penny and will, as a physical object, probably not impress your friends and family much. But some people get fired up by competition, and we want you to put some effort into this - and for all you know, the emotional value will be *priceless*.

Multiple prizes will be given based on Tech staff’s personal whims. Maybe two, maybe ten. Deliverables will be judged based on code quality, conceptual technical proficiency, aesthetic quality, attention to detail, the omens of the stars, evidence of personal intellectual growth, believing in yourself, and general vibes.

If you really want detail about how to “win”: think about what you’ve been told during Night School and conversations you’ve had with Tech and Program staff. Remember that there’s more to code than just accomplishing a goal.

Don’t Forget Your Presentation

You still have a presentation with your teams to be working on. Both of these are important - please remember that this deliverable has no real requirements other than “turn *something* in” and “grow your skills”. Do not skip out on your team in an effort to win a prize that, this *cannot be stressed enough*, has little actual value other than the fun of competition. It is in no way as important as making a good impression with your team, it is not more important than attending regularly scheduled meetings, etc.

Suggestions, Resources and Encouragement

If you're looking for suggestions for *what to display* to the user, or *what data to look for*, think about:

- Data relating to the closest bus or el stop near your home, or on the route you use to take to get to the i.c. stars office
- Information about the state of a specific bus or train line that you or your family use frequently
- Data that can be used to plan a trip from one landmark or personally important place in the city to another
- The status of a bus or el stop that a child or another family member regularly uses
- Keeping a tally of how many trains are on time (starting when the page loads) at the Chicago stop near the i.c. stars office

If you're looking for suggestions about *what to implement technically*, think about:

- Using JS functionality in HTML with attributes like `onclick`, `onsubmit`, etc.
- Getting data from the user and using that to customize the API call in some way
- Making multiple API calls and storing that data in an array or object for organization or later access
- Using JS DOM functionality like `document.createElement()` or `element.innerHTML` to perform DOM manipulation

If you have an idea but aren't sure of how to implement it:

- Read more about JavaScript and DOM manipulation online! Some specific pages that might be useful:
  - The MDN documentation on DOM Elements
  - The MDN documentation on Document methods
  - W3 Schools' introductory reading on the DOM
- You can ask your cycle mates for assistance! This is not technically a group activity, but there's nothing stopping you from advising each other or helping each other find and fix bugs with your code
- If you *know what you want to do* and *have been trying for some time without making progress*, you can find Ryan or Jess and ask for assistance. Please only come to us if you feel like you've exhausted other options - we'll want to see what you've already tried and what ideas you've had

If you are frustrated, remember:

- The purpose of this is just to try something new! We just want you to make an attempt at doing something with minimal handholding so that you have a better idea of your strengths and weaknesses
- This should be, on some level, a little fun - feel free to personalize it in some way, laugh at your mistakes, and don't be afraid to get a little messy
- We believe in you! Any progress is good progress. Your project may not be as technically proficient as someone else's, and that's fine! The important thing is that *you* learned something.