Passion Project Site

JavaScript functions, API calls and DOM manipulation

Objectives

- Become familiar with DOM manipulation
  - Make API calls and extract useful data from them
  - … or add other JavaScript-based functionality to your site.
- Challenge yourself!
- Explore web development with an interest you have

Description

The goal of this deliverable is to produce a website (one or more HTMLs file, a CSS file, and one or more JavaScript files) related to an interest or passion of yours.

“Interest or passion” is intentionally ambiguous - you can decide what the topic of your website is. The minimum *technical* requirement is that there be either **a.** 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.
or **b.** some kind of other JavaScript functionality - using JavaScript to add some kind of interactivity.

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*.

API Selection

Finding an API can be a challenge; thankfully, a few people have compiled large lists of publicly available APIs that you can use. Two lists can be found here and here - although you can feel free to search for an API relating to a topic of interest to you. These lists are not guaranteed to be up to date or accurate - some services mentioned on them are no longer available (shut down, moved, etc.).

Both these GitHub repositories sort these APIs by topic, and have three columns with important information about the API - “Auth”, “HTTPS” and “CORS”. Two of these are very important.

Authentication Requirements

If an API in these lists has an “Auth” value of “No”, you need no authentication to use this API and make calls to it. Otherwise, you’ll need to either sign up for an API key or set up an OAuth or OAuth2 token.

Getting an API key is typically pretty straightforward - the one thing you want to know is that some of these registrations can take a few days to process. If you see an API that interests you that needs a key, register for it sooner rather than later.

We haven’t really discussed OAuth or OAuth2 tokens, but some API documentation has instructions on setting that up. If you really want a specific API that needs OAuth, try setting it up yourself and ask if you need a hand.

CORS

CORS stands for cross-origin resource sharing - to put it simply, CORS ensures that requests to an API are only made from specific domains.

If an API in these lists has a value of “Yes” for CORS, *you will not be able to use it for this deliverable*. CORS will prevent us from calling an API from any address other than ones permitted by the API - this means that you cannot call it from “localhost” (i.e. your computer).

Some of these APIs have a value of “Unknown” for CORS - you may have to investigate the API’s documentation, or just try making a request and seeing what happens. Some APIs have routes that permit CORS but otherwise do not support it, in the same way that some APIs have public routes available without an API key and routes that require one. Every API is different.

Other JavaScript Usage

Some of your chosen topics may not really have a suitable or accessible API to use. In this case, you are free to use JavaScript to add some other element of interactivity to your website via DOM manipulation. The sky really is the limit here - come up with an idea and see if you can figure out what you need to do to accomplish it.

Can I Change My Topic?

Yeah, sure. If you decide you want to do something different, go for it. Again, the purpose is for you to learn via engaging in something that is interesting to you - if you lose interest, pick another topic/idea.

Requirements

The HTML file must link a CSS file and at least one JavaScript file. That JavaScript file must either make some kind of API call to an API successfully, then use data from that API call to perform DOM manipulation to show some data in the web browser *or* perform some other kind of DOM manipulation using JavaScript. Or, combine an API call or API calls with some other kind of JavaScript functionality.

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 **Tuesday, July 9th 2024**. There will be a presentation to alum or some other group of people tentatively scheduled for **Wednesday, July 10th, 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”.

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:

- Hobbies you have - sports are a good example. https://developer.sportradar.com/ has APIs for many American sports, and trial API keys are available
- Previous professional or work experience - for example, there are APIs for data relating to food and drink (recipes, nutritional information, etc) for people who have worked food service or hospitality before
- Other websites may have an API to access and use your account - there are APIs for Reddit and Twitter (or X. Thanks, Elon.), for example.
- If all else fails, look through the list - something may jump out at you that you hadn't considered before

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
- A good guide to using `fetch()` can be found here
- 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.