APIs, AJAX, and JSON

September 27

Today's Goal: Learn how your small piece of the internet interacts with the rest of the internet.

Application Protocol Interface (API)

In general, APIs are parts of one application that are exposed to be used by another application, developer to developer.

The API could expose part of the application, i.e. Twitter API vs Twitter. Or it sometimes the entire application could be dedicated to being an API, i.e. meowfacts

APIs can facilitate communication between applications in many different scenarios, whether it is between different computers entirely (such as client-server interactions), or just between different applications on one computer.

Browser APIs

Web browsers, being very complex have many different APIs that they expose to JavaScript coders. These we refer to as browser APIs.

Some common browser APIs you will come across in your web development journey:

- DOM
- Fetch API
- Canvas API
- File API
- Clipboard API
- WebAudio API
- WebGL API

Third Party APIs

Whenever you are using an API in a client-server type interaction, especially if you (or your team) didn't code the server side yourself, in the web dev world it's often referred to as using a third party API.

A term you will often hear associated with third party APIs: REST which stands for Representational State Transfer. It is just a very common design framework for creating APIs.

Incorporating third party APIs into your code means creating bigger and greater experiences than you would ever be able to without. But it also means leaving your program longevity in the hands of of the API maintainers.

Asynchronous JavaScript and XML (AJAX)

AJAX describes the program design model most commonly used throughout the web for the last two decades now.

It generally entails a client-server architecture that allows processing to truly happen in the background (remember, asynchronous JavaScript on its own is not actually multi-threaded).

Even though XML is in the name, the even modern-er approach to AJAX is to use JSON instead.

JavaScript Object Notation (JSON)

JSON is a data storage file format that is the web industry standard for passing data from client to server and back.

It was designed around the web and JavaScript in particular hence the name, though is also heavily used by many other languages now, because it is easy to read, both by computers and by humans.

Syntax is the exact same as creating objects in JavaScript, except double quotes are required for strings, and they are required around the keys of the key value pairs as well.

Example:

```
{
    "name": "Front-End Web Development",
    "id": 4630,
    "gradId": 5630
}
```

Fetch API

The fetch API is a browser API that enables us JavaScript developers to access third party API services.

There are other (older) browser APIs that accomplish the same result with more work, such as the XMLHttpRequest API, but fetch is the modern standard for AJAX programming, so it's the one we will be covering in class.

Fetch allows you to make requests to other servers and process them in JavaScript like promises.

Example:

```
fetch(url)
   .then(response ⇒ response.json())
   .then(json ⇒ doYourThing(json))
   .catch(e ⇒ console.error(e))
```