**HOMEWORK:** JS libraries Hackathon!

**Explore - Play - Fiddle**

Pick one of the libraries below and build something cool with it! This time, don’t worry about a strict plan, just make sure that whatever you do build, you have something to present by the end!

Chances are quite high that you’ve never worked with or seen these libraries before - perfect! Once you pick a library you want to use, we strongly recommend you find a ‘getting started’ guide, or a tutorial to walk you through an example first. Then look back, modify, make something your own, customize.

What you should get out of this exercise:

* Ability to get up and running quickly with a new technology
* Broadened view of what you are currently capable of building, what’s possible
* Exposure to different coding philosophies, structures, and patterns
* Become more comfortable with reading documentation, diving into something unfamiliar
* Fun

Pick one (only one) library to use:

* Chartjs (charts)
* Flotjs (charts)
* P5js (visuals)
* Threejs (3d)
* Slickjs (carousel)
* EaselJS + TweenJS (animation)
* Trackingjs (camera Tracking)
* Howlerjs (audio)
* Bootstrap (CSS)
* Leafletjs (maps)
* Firebasejs (hosted)
* D3 (https://d3js.org/)
* Bandjs (audio)
* Aframevr (virtual reality)
* Anything else.

**Getting started:**

Do a little bit a research, and pick one library from the list above. Then learn how to use the library, use "the getting started", or follow a tutorial. Then customize what you made to be something unique or interesting. Make sure your idea / theme is still professional and does not have anything inappropriate.

**Caution:**

Do NOT follow parts of any instructions that say use “npm install”, “npm i ..”,“bower install”, “yarn install” - or have any kind of “import” / "export" statements in the code (unless you’re comfortable with them). We’ll learn about those later, but for now, you’ll bring the libraries in using script tags e.g. <script src=””> in your HTML. You’ll likely need to download those javascript library files to your computer and link them in your HTML like you would any other JS file. Look for directions that show you what to put in your script tag. In some cases you’ll just have to find the github repo.

**# Libraries**

​

The libraries below each have great documentation and allow for a wide variety of interesting

pages and apps.

​

*- [Pixi.js](https://pixijs.com/): A fast renderer like Flash (rip) that uses your GPU to quickly and efficiently update.*

​

*- [D3.js](https://d3js.org/): A powerful visualization tool with a wide array of uses, from fancy graphs to interactive displays.*

​

*- [Anime.js](https://animejs.com/): No not that anime. This is a lightweight library that lets you control css animations easily and to great effect.*

​

*- [AOS](https://michalsnik.github.io/aos/): Animate on Scroll. A simple libray which lets you add neat animations to elements based on the scroll level of your page.*

​

*- [Chart.js](https://www.chartjs.org/): For all of your graphing and data visualization needs.*

​

*- [Cleave.js](https://nosir.github.io/cleave.js/): Gives you great control over any input fields, leading to consistent data acquisition and intersting interactivity.*

​

*- [P5.js](https://p5js.org/get-started/): You can use P5 to make your own visuals or to allow the user to draw or manipulate shapes.*

​

*- [Three.js](https://threejs.org/): Create and animate 3D models.*

​

*- [Bootstrap](https://getbootstrap.com/): Pre-built webpage elements for anything you can imagine.*

​

*- [Leaflet.js](https://leafletjs.com/): Let's you embed and interact with maps.*

​

*- [AFrame](https://aframe.io/): Create, model, and view anything for virtual reality.*

​

*- [Choreographer.js](https://christinecha.github.io/choreographer-js/): Another animation library with a focus on reacting to user input.*

​

*- [fullPage](https://alvarotrigo.com/fullPage/): A tool for creating sleek scrolling movements in a single page app.*

​

*- [Parsley](http://parsleyjs.org/): A tool for form validation, giving you greater control over user inputs and giving users a clear understanding of what they need to do.*

​

*- [Popper](https://popper.js.org/): A small library that makes it easy to make pop-ups, dropdowns, tooltips, and any other dynamic label.*

​

*- [Hammer](http://hammerjs.github.io/): Make your site mobile-ready by giving you greater control over touch-based events and interactivity.*

​

*- [Socket.io](https://socket.io/): A tool to help create chatrooms, multiplayer games, or anything else that requires constant 2-way communication.*

​

*- [Currency.js](https://github.com/scurker/currency.js): You may have heard of rounding erros in computing. This library allows you to do any math with money while maintaining perfect accuracy, and even allow operations like adding `'$1,500.50'` and `1000` together.*

​

*- [MouseTrap](https://craig.is/killing/mice): Don't think about the URL. With this you can set up keyboard shortcuts, including combinations like ctrl+s, easily. It's also very lightweight!*

​

*- [Words to Numbers](https://github.com/finnfiddle/words-to-numbers): Convert any written numbers into into actual numbers. `'One Thousand and Fifty point Two'` becomes `1050.2`. Even handles spelling errors.*

​

*- [Devices.css](https://devicescss.xyz/): Templates for showing nearly any device on your webpage, like mobile phones and MacBooks. You can even dynamically set what is shown on the device screen.*

​

*- [Sentinel.js](https://github.com/muicss/sentineljs): Adds just one thing: an event listener for when anything is added to the DOM. Honestly not much you can do with this on it's own.*