Lab 2

Due Sep 14, 2017 by 11:55pm   **Points** 100   **Submitting** a file upload
File Types zip

CS-554 Lab 2

Advanced CSS Concepts

For this lab, you will submit a semantically valid HTML document that styles the following content.

The Home Page

You will write a semantically valid HTML document that describes 10 of your favorite television shows.

You will create a navigation with links to each television show article element, and a container element filled with these article elements that list a picture, title, and short description of your favorite television shows. All content is on one page; the links in the navigation point to the ids (ie: if *The Office* is one of your favorite shows, your navigation element would look something like `<a href="#the-office">The Office</a>` and your article would start with `<article id="the-office">`

When the browser is below 500px wide, the navigation element will sit on top of the container, not to the side of the container.

When the browser is above 500px, the navigation will take up approximately 25% of the page and will be oriented on the left side of the page. The container element will take up approximately 50% of the page, and will be next to the navigation. Offsets and exact widths are up to you. The container will have a max-width of 800px.

The container will be styled with flex-box and the number of items visible on each row will change with your current screen size.

When the browser is below 500px wide, it will show 1 television show article per row in your container area.

When the browser is between 501px and 800px wide, it will show 2 television show articles per row in your container area.

When the browser is greater than 800px wide, it will show 3 television show article per row in your container area.

The look

Besides the specified settings, as long as your HTML is valid and your page runs passes a **tota11y (http://khan.github.io/tota11y/)** test, the general look and feel is up to you.

The Server

You will load an express server that will allow static assets to be loaded from a directory called `/public`

You will have a single route, `/`, that will load your description page.

All other routes will load a 404 page.

Notes

1. Remember to submit your `package.json` file but **not** your `node_modules` folder.