WEB222 Assignment 4 – DOM Manipulation

Due date

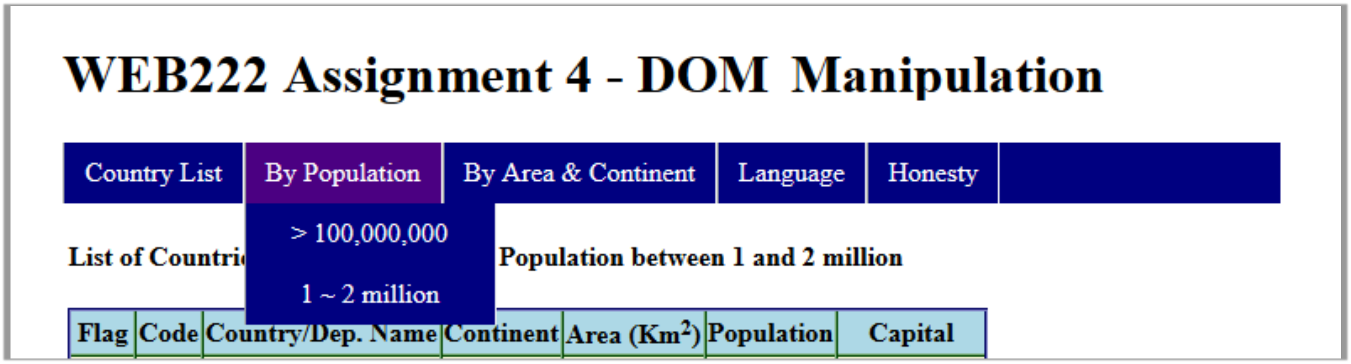
Monday Nov 16, 2020 by midnight

Grade value: 5% of your final course grade

Objective

This assignment will help you learn and practice interactive DOM programming. Please do this assignment on your own and submit only your own work. Gaining experience with JavaScript and the web takes a lot of personal practice and working on these problems yourself will help build your skill.

We will be creating a web application for exploring country data. This web app allows the user to dynamically alter the page at runtime by choosing between various menu options:



WEB222 Assignment 4 - DOM Manipulation

List of Countri Population between 1 and 2 million

Flag	Code	Country/Dep. Name	Continent	Area (Km^2)	Population	Capital

Doing so allows the user to see a list of country data based on population, area, language, etc.

Setup

Begin by downloading the assignment Zip file and setting up the necessary dependencies:

```
cd assignment4
npm install
```

This will download and save all the necessary files to a folder named node_modules/ in the current directory. It will also allow you to run a number of scripts, including:

1. `npm start` – this will start a local webserver on http://localhost:5000 which you can use to run and debug your web app.
2. `npm run eslint` – this will run eslint and tell you about any warnings or errors in your code. You should have no warnings/errors before you submit your assignment.
3. `npm run prettier` – this will run pretter.io on your source code and automatically reformat it correctly.
4. `npm run prepare-submission` – this will create your final submission.zip file to be handed in, see more details below on submitting.

Instructions

You are asked to complete the code for the web app in the src/ folder. The program is broken up into multiple files, and you need to work on the following files, and in the following order:

1. src/util.js has methods for working with the country data
2. src/ui.js has methods for working with the country data and DOM
3. src/index.js has our app's main entry point, and handles event listeners

Each file is focused on a particular aspect of the program, and you will complete the program by using all of the Objects and methods together.

Read the comments above each function and write your code in the empty function bodies to implement the behaviours described in the comments. Your goal is to make it possible to click on any of the menu items and have the table's contents update to show the requested data.

The table takes the following form:

- **Flag** - an <img> with the flag for the country. All flag images are dynamically loaded from https://www.countryflags.io/ and you will write code in ui.js to generate the appropriate URLs
- **Code** - a string with the country's 2-letter ISO country code
- **Country/Dep. Name** - a string with the name of the country. This will either be in English, or one of the provided languages.
- **Continent** - a string with the continent for this country
- **Area (Km2)** - a number with the area in square km, formatted for the chosen language
- **Population** - a number with the population of the country, formatted for the chosen language
- **Capital** - a string with the name of the country's capital

All country data comes from the file src/js/countries-data.js (see this file for format details).

Finally, make sure you update `src/index.html` and `src/honesty.html` with your name and information in the footer and other relevant sections.

Use the browser Developer Tools to debug your code. See https://developers.google.com/web/tools/chrome-devtools and https://developer.mozilla.org/en-US/docs/Tools for information on how to use the developer tools for your browser. For example, you can call all of your functions directly in the Console to test that they are working as you expect, set breakpoints in the Sources/Debugger tab to inspect and step through code, etc. If you have questions on how to do any of this, ask your professor.

NOTE: when developing web apps it is highly advisable to disable your browser cache while the dev tools are open. See https://nicholasbering.ca/tools/2016/10/09/devtools-disable-caching/ for instructions on doing this.

Submission

When you have completed your assignment, you need to prepare your submission. To do so, use the npm command:

```
npm run prepare-submission
```

This will do a number of things automatically for you:

1. Run prettier on your assignment code, formatting it
2. Create a directory called submission/
3. Run eslint on your code and save the result to submission/eslint.log
4. Create submission.zip from the contents of submission/*

You can upload and submit the submission.zip to Blackboard.