Assignment 1: Modern Interactive Website with JavaScript

For this project, you will make a meaningful, fully functional, ES6 based web application that, among others, uses conditional dropdowns (cascading selects), a form, and a web storage to store data locally within the user's browser.

Note 1: You need to decide what kind of a web application you would like to make, and you need to present your idea to your instructor. You will start building your web application once the topic gets approved by your instructor.

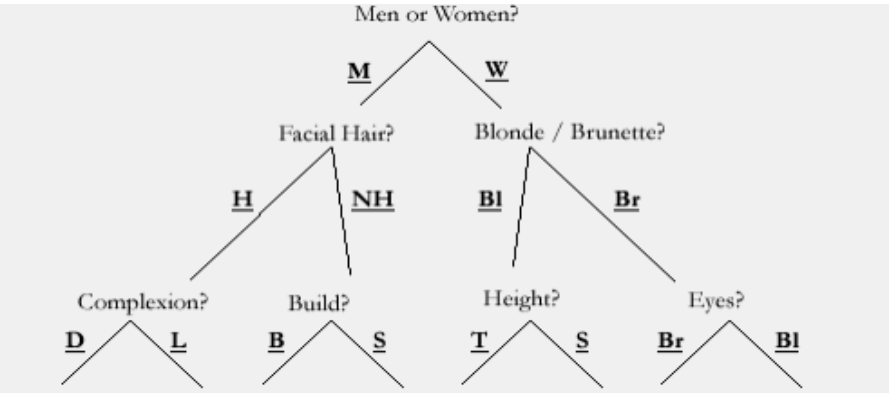
Note 2: Your web application needs to follow all the good/modern design and programming principles and techniques, especially those we practiced during our class sessions: using ES6 features, properly structuring your application (OO approach, MVC architecture, using modules)

Requirements (Web Design and Programming Principles)

- ☐ Your web application needs to follow all the good/modern design and programming principles and techniques, especially those we practiced in class: OO approach using modules, MVC architecture
- ☐ Your application needs to provide meaningful content (with paragraphs).
- ☐ The code needs to be properly aligned and indented.
- ☐ The code needs to be documented using properly formatted comments (JSDoc format) that clearly explain the code functionality. More at https://jsdoc.app

Requirements (Conditional/Cascading Selects)

- ☐ Dynamically create selects with options depending upon the user's previous select option.
- ☐ Each select must have at least 2 options (<option> elements)
- ☐ There must be a depth of 3 different selects (<select> elements)
- ☐ Example for *Find a Date* app:



- ☐ The selection process includes a dynamically generated GUI element that changes in response to the user selection. For example, in a *Find a Date* app, a person image will change depending on the user's choices - a man with facial hair and light complexion.
- ☐ Give the user the option to start over at any time.

Requirements (Form Element)

- ☐ Once the user is done with the selection, create a form that shows the user's selections, AND is used to gather additional information. Additional information is application specific, depending on the website topic you chose. For example, in a *Find a Date* app, additional information like the user's email and the age may be required.
- ☐ The form inputs need to be validated – make sure that the data the user provided is in the expected format. In case the data is invalid, provide an informative feedback (e.g. "Please enter your phone number in the format xxx-xxxx"), and, restyle the form to visually contrast the element with the invalid input.
- ☐ Provide at least 2 inputs with an expected format to validate. For example, a credit card number in an expected format, and a phone number in an expected format.
- ☐ The validation needs to be performed by JavaScript not by HTML5.
- ☐ In case of invalid user input, do not make the user fill the whole form, just the invalid inputs.

Requirements (Web Storage)

- ☐ Your web application should use the *Web Storage* to hold the application data locally.
- ☐ User selection process will result with data that need to be stored locally.
- ☐ Submitting the form will result with all the form data stored locally.

Requirements (Browser Support)

- ☐ Your application needs to check how well does the user's browser support the web application's features.
- ☐ You do not need to check every feature, just the modern ones you are using. For example, you can check support for the ECMAScript 2015 (ES6) features (only those you use): Promises, Modules, Classes, Template Literals, Arrow Functions, Let and Const, Default Parameters, Generators, Destructuring Assignment, Rest & Spread, Map/Set & WeakMap/WeakSet and many more.
- ☐ If the feature is not supported redirect the user to a different web page indicating what the user needs to do to have the application work as expected (e.g. downloading a browser that supports the application features). Do this at the beginning of the program, before allowing the user to do any interaction.
- ☐ You may use 3rd party libraries to help you out with this task, libs such as Modernizr (https://modernizr.com)

Grading

An excellent project (grade A) will have all of the requirements above PLUS something extra such as those listed below. The extra features (whether they are listed below or not) need to be discussed and approved by your instructor in order for you to proceed with the implementation.

- ☐ Explores an area of code we didn't cover in class (e.g. using a CSS preprocessor, front-end framework, ...)
- ☐ Scalability: Your code is generic enough to adapt to have more than 3 selects of unknown depth. You have provided 2 data sets in this regard that the user can load and test by interacting with the web page.
- ☐ Extended Functionality: Request and use additional data from 3rd party APIs (e.g. Google Street View, New York Times, ...). You may use the following web app to find an API: https://www.programmableweb.com/apis/directory

You will also have to document your solution using the provided project1-readme.docx file. For every rubric in the document, if applicable, explain what you did and where to find your features in the code. Failing to submit the document will result in no points for the assignment. Also, missing to properly explain a rubric will result with no points for that rubric.

Zip up the entire project (file structure intact, project1-readme.docx file included) and dropbox it. The project root should be named as LastNameFirstNameInitial (e.g. MarasovicK)

DUE DATE: Beginning of W06C1 class (check the Assignments folder MyCourses)