PER SCHOLAS

CP 325.3:
Adding Dynamic Web Content with JavaScript

Version 1.0, 08/14/23
Click here to open in a separate window.

Introduction

This document will challenge you with taking the next steps in your Capstone Project journey.

Assignment Objective

- Create interactive web content using JavaScript.

Submission

This portion of the Capstone Project will not be submitted on its own; only the final project will be submitted. You are encouraged to seek feedback from instructors and peers along the way.

Instructions

You now have the foundational programming knowledge that you need to create logic, manipulate web pages through the DOM, and create event-driven interactions.

Use these new skills to begin thinking about and implementing your application's flow of data and logic. You still have much to learn, but now you can start to see things come together! Buttons can work, forms can change data, and you can decide how everything should fit together.

Again, do not try to *finish everything* at this stage. You are laying the foundation for what is to come, and giving yourself the time to experiment. Do not be afraid of change or failure at this stage. Embrace the uncertainty, try new things, and make note of what works and what does not work. Failing fast and failing forward are key skills in the development world.

Once you have some interaction enabled, go back through your existing layout and ask yourself if there are things you want to change or build upon now that you have more context?

As you continue forward with this project, you can *always* make changes to what you have already done. Do not feel confined to the boundaries your past self set; you are constantly breaking those barriers.

Partial Requirements

Let's look at which Capstone Project requirements are related to this stage of development. Requirements not related to this stage of development have been omitted. For a full list of requirements, see CP 323.1 - Planning a Project, or CP 323.10 - Capstone Completion.

(20%) Project Structure, Standardization, and Convention	**Weight**
Project is organized into appropriate files and directories, following best practices.	**2%**
Project contains an appropriate level of comments.	**2%**
Project is pushed to GitHub, and contains a README file that documents the project, including an overall description of the project.	**5%**
Standard naming conventions are used throughout the project.	**2%**
Ensure that the program runs without errors (comment out things that do not work, and explain your blockers - you can still receive partial credit).	**4%**
Level of effort displayed in creativity, presentation, and user experience.	**5%**

(12%) Core JavaScript	**Weight**
Demonstrate proper usage of ES6 syntax and tools.	**2%**
Use functions and classes to adhere to the DRY principle.	**2%**
Use sound programming logic throughout the application.	**2%**
Use appropriate exception handling.	**2%**

The following section is NOT included in the requirements for this project. Completing this section is NOT required. This section will NOT negatively impact your grade if left unfinished.

This section is intended for learners looking to go the extra mile by showcasing additional skills such as project management, and optional technologies like TypeScript.

You must complete ALL other requirements to receive credit for this section.; however, this extra credit will not be included if you have already received the maximum 100% grade. The extra credit can only offset points lost elsewhere.

(5%) Extra Credit	**Weight**
Adhere to Agile principles and the Scrum framework. Perform stand-up sessions (with an instructor) when possible.	**1%**
Successfully track your project using a software like Jira.	**1%**