Lab: HTML and CSS

Lab Overview

This lab consists of one HTML/CSS project. You will find a folder for with design mockups for mobile, desktop and active states, optimized images, a style guide and a starter index.html file with text content (not yet formatted and not styled). Open the folder using VS Code to get started.

While performing this lab, incorporate the principles and practices that you've learned from prior lessons in this course; for this lab, be sure to consider:

- Proper file formatting for HTML and CSS
- Make good code comments
- Try to match the designs exact in all details

Where to start:

1. Look through the designs in the design directory to start planning out how you'll tackle the project. This step is crucial to help you think ahead for CSS classes to create reusable styles.
2. Before adding any styles, structure your content with HTML. Writing your HTML first can help focus your attention on creating well-structured content.
3. Write out the base styles for your project, including general content styles, such as `font-family` and `font-size`.
4. Start adding styles to the top of the page and work down. Only move on to the next section once you're happy you've completed the area you're working on.
5. Learning to use Google and Stack Overflow is a crucial tool to learn as a developer – use these to help you find answers to any problems you run into during development.

Submission Instructions

Create a new feature branch in your TEKbootcamp repository. Commit and push frequently. When your work is done and ready for review, please merge the feature branch into the develop branch.

Following instructions for CodeGrade submission:

1. Click on the "Launch exercise" button on the exercise page at TEKsystems Academy.
2. In the new CodeGrade window, click on the "Connect Git" button.
3. Select GitHub as your Git host. Connect your TEKbootcamp repository to the exercise*.

After selecting a repository, this repository will be cloned to CodeGrade as a first submission. After this, you can start to use Git like you usually would. Every time you push to the develop branch, it will automatically result in a new submission in CodeGrade. Pushes made after the submission deadline will not be taken into consideration.

* You will only have to log in and authorize CodeGrade once; after that, it will be available for all your other assignments inside CodeGrade too.

WARNING: Be sure to connect the proper repository to the exercise. You will not be able to connect to another one immediately using the CodeGrade connection. You can undo your Git connection with CodeGrade by revoking access to the External Tool "CodeGrade" in your GitHub account only.

Grading Expectations

A solution that meets all the requirements in this lab constitutes a grade of Proficient (3/5). To earn a higher grade, your solution must also apply the code quality concepts you've learned up to this point. Additional functionality that does not improve the quality of the code and only serves to make your assignment look more impressive will not help to raise your grade.

Plagiarism Warning

This assignment is a demonstration of your understanding of the topics covered to-date in the boot camp. You may need to reference tutorials or Q&A sites such as StackOverflow to complete parts of the assignment. It is acceptable to use tutorials/Q&As to learn how to add specific functionality to your project, but you cannot simply copy a tutorial repository, follow a single tutorial from start to finish, or copy the code from a Q&A site to develop your application. See guidelines below for using tutorials:

- Do not fork a tutorial repository. This would be considered plagiarism and result in an automatic grade of zero and potential removal from the boot camp.
- Do not simply follow the tutorial from start to finish.
- Do not copy code from a Q&A site and then change it for your application (a few lines of code are permissible)
- You must be able to explain how all functionality included in your project works.
- List ALL tutorials or aids used, either in your README file, and/or as code comments. Include:
    - The name of the tutorial,
    - Which section(s) you used, and
    - The functionality you learned from the tutorial.
- Failure to identify tutorials/Q&A responses will result in significant reduction of your grade.
- If you work with other boot camp members – or instructional team members – to find solutions, that is acceptable, but must also be clearly identified

Estimated Duration

3-8 hours

Resources

1. **HTML_CSS-Lab Resources** zip folder with the following folder inside:
    a. Three Column Component folder

Project: *Three Column Component*

Project Overview

- Your task is to build out the project to the designs inside the design folder.
- Your users should be able to:
  - View the optimal layout for the page depending on their device's screen size.
  - See hover states for all interactive elements on the page.
- The designs are in JPG static format. Using JPGs will mean that you'll need to use your best judgment for styles such as `font-size`, `padding` and `margin`.
- You will find all the required assets in the images folder. The assets are already optimized.
- Some images are svg images that are becoming popular.
  - Do a search to see how these are implemented with html and styled compared to .jpg and .png file formats.
- Use the style-guide.md found in the Exercise folder for your page styles.
  - ** ALL styles must be in an external stylesheet you need to create and attach to the index.html file in the correct location.
- Convert the hsl colors to hex or rgb colors.
  - Use VS Code editor to cycle through color formats when hovering over color in editor.

Project Setup Instructions

Open the Three Column Component Exercise folder In VSCode:

- As you complete a section of the design mockup push your code to your student repo frequently.
- Use the same setup guidelines for working html/css as Traversy provides in the Udemy course in Section 1.4 – VS Code Setup as well as anything VS Code he presents in the course.

Exercise Notes:

- Do a css box-sizing reset for all exercises
- Use flexbox for your grid
- Use proper semantics for html markup
- NO frameworks – example – can't use Bootstrap nor Foundation etc.
- Must be fully responsive using MOBILE FIRST approach
- Be detail oriented as you look at the design, noting places where there are rounded corners, box shadows, font-size variances, spacing, active states needing pseudo classes etc.
- Create a README-challenges.md file and add it to this exercise directory after you have completed this exercise and note any challenges you faced while working this exercise.