CSE 154: Web Programming — Spring 2018

Homework Assignment 1: Writing Readable Code

Due Date: Wed. April 4th: 11pm

Overview

This assignment tests your understanding of basic HTML and CSS. You will create a webpage using a PDF as the basis for the expected appearance.

Learning Objectives

- Practice following CSE 154 assignment specifications, and more broadly, webpage specifications given a PDF and/or other visual and text based artifacts as a design basis.
- Be able to write valid HTML and CSS pages.
- Be able to test for valid HTML and CSS pages.
- Be able to choose appropriate HTML tags with an understanding of the semantic meaning of each tag and the context they are used in a webpage.
- Practice separating *content* from *presentation* with your HTML and CSS.
- Practice reducing redundancy in your CSS styling when producing expected output.

Note for Assignment 1:

We expect you to read the content of your resulting webpage as well, as it provides an overview of the expectations of CSE 154 coding practices and documentation for the quarter (which are also relevant to many industry settings).

Files to Submit:

For full credit, you must turn in your files via Grade-It.

1. `readable.html`: The file containing the HTML for your "Writing Readable Code" webpage
2. `readable.css`: The style sheet for `readable.html`

Files Provided:

1. `readable.txt`: Text content for your HTML page, used as the starting point to your HTML (do not change any of the text content, such as specific words or ordering of sentences and paragraphs)
2. `writingreadablecode.pdf`: Expected output PDF that should be used as a visual reference when writing your HTML and CSS

Links For Images:

Your page should use the following absolute links for the corresponding images in the expected output:

- Validation images:
  a. HTML Validator icon: https://webster.cs.washington.edu/images/w3c-html.png
  b. CSS Validator icon: https://webster.cs.washington.edu/images/w3c-css.png

- Other images (in order of appearance in the expected output):
    a. https://webster.cs.washington.edu/images/hw01/goodvsbad.png
    b. https://webster.cs.washington.edu/images/hw01/htmlandcss.png
    c. https://webster.cs.washington.edu/images/hw01/jsdoc.png
    d. https://webster.cs.washington.edu/images/hw01/php.png

External Requirements

- Your webpage should match the overall appearance of the PDF provided, although it is not required to match "pixel-perfect" (remember that browsers render the same webpage differently, so its often unrealistic to achieve a pixel-perfect output!).
- The title of your webpage should be "Writing Readable Code".
- All the headings on the webpage should use a foreground color of `#4ac3fd` and all except the main heading should be underlined.
- The main heading of the page as well as the other sub-main headings should be center-aligned, whereas all other headings should retain their default alignment property. All the headings should be capitalized as well, which you should achieve using an appropriate CSS rule.
- All the headings should use the Acme font family (you'll need to import from Google fonts), Helvetica, Arial, or, in the case that none of these are available on a user's system, the default sans-serif font.
- The main heading should have a font size of `22pt`, sub-main headings a font size of `18pt` and all other headings the font size of `16pt`.
- Just below the main heading, there is a `2px` wide horizontal divider with a color of `#4ac3fd`.
- The overall page's body should have a white background and the text should have the default foreground color.
- Text in the body should have a font size of `12pt` and use the Rokkitt font family (imported from Google fonts), Garamond or the default serif font available on the user's system.
- Any quotes in the body should be centered and use the 'Amatic Sc' font family, or the default cursive font available on the user's system. The quotes should have the same font size as the page's body text, and should also be italicized with a background color of `#1bd8d8`.
- Any code snippet in your body should inherit the body's font size, but should be defined as the default monospace font available on the user's system. It should also have a background color of `#c7f3dc` to distinguish it from normal text.
- Any occurrence of the term "JSDoc" on the page should link to the JSDoc documentation website (http://usejsdoc.org/).
- You should use appropriate HTML tags to strongly stress words that are bolded in the expected output.
- All the images except the validator icons in your code should be center-aligned.
- There should be two validation buttons at the bottom right of the page (see expected output). You should use the absolute URL provided for each image (See Overview -> Links for Images).

- The two validators should link to their corresponding validation websites:
  1. HTML: https://validator.w3.org/#validate_by_input
  2. CSS: https://jigsaw.w3.org/css-validator/#validate_by_input

Internal Requirements

- For full credit, your HTML and CSS should follow the rules listed in the Style Guide on the class website.
- Choose appropriate HTML tags to match the structure of the content on the page. Use `<header>`, `<footer>`, `<section>`, and `<article>` tags where appropriate to categorize the various blocks of content in your webpage.
- Do not express style information in HTML with inline styles or presentational tags such as `b` or `font`.
- You should only use line break when required, for more information on this, check the Style Guide on the class website.
- Your `readable.html` page must also pass the W3C HTML5 validator with no errors (a green bar). Your page is fine as long as you see the green bar and text "Document checking completed. No errors or warnings to show." We will click your W3C validation buttons to validate your page, so test them for validity by clicking those same buttons. *Note: the validator buttons will not work by default when viewing your page on Cloud9. We will show you several ways to validate your HTML in lecture. As the final check to make sure that your page passes HTML and CSS validation, you should click your validation buttons on the Webster copy of the assignment after you have turned in.*
- You only need to worry about your page's appearance in standards-compliant browsers such as Firefox or Chrome. You will not be tested in browsers that do not comply to web standards. Express all stylistic information on the page using CSS defined in `readable.css`.
- For full credit, your style sheet must successfully pass the W3C CSS validator.
- Part of your grade comes from expressing your CSS concisely and without unnecessary or redundant styles. For example, if the page uses the same color or font family for multiple elements on the page, you must group those elements into a single CSS rule, so that it would be possible to change the page's color/font by modifying a single place in the CSS file.
- You should only use HTML and CSS constructs that have been discussed in lecture, section, and Chapters 2-3 of the textbook.
- As we introduce IDs and classes as attributes to target HTML elements for styling, its important that you do not overuse these in your HTML. If there is already a suitable tag for representing a given piece of content, favor the use of that tag rather than a less appropriate tag with a class or id attached for styling purposes. Refer to the Style Guide for more examples of good use of ID's/classes.
- Format your HTML and CSS nicely so that it is as readable as possible, similar to the examples shown in class. You should include a comment header in each file containing your name and section and a brief description of the assignment and the file's contents.
- Your use of spacing and indentation in your source code should be consistent and follow the examples shown in class and in the Style Guide. Do not place more than one block element on the same line or begin any block element past the 100th character on a line.

Academic Integrity

As with any CS homework assignment, you may not place your solution to a publicly-accessible web site, neither during nor after the school quarter is over. Doing so is considered a violation of our course academic integrity policy. As a reminder: The University of Washington has an entire page on Academic Misconduct on their Community Standards and Student Conduct Page. Please acquaint yourself with the University of Washington's resources on academic honesty, and in particular how academic misconduct will be reported (which has been changed for 2017).