Published: 05 Sep 2018
Updated: 07 Sep 2018
No longer requiring the nonexistent "line-length" property.

6.170 Assignment 1: Personal Website

Due: 11 Sep 2018 @ 11:59 a.m. — Submission Form

Since this is the first assignment, please take a look at this course's ***collaboration and grading policy*** *in the General Information before beginning.*

Overview

Your challenge in this assignment is to build a website using HTML and CSS. Your website should be purely informational with no dynamic or interactive elements.

The content of the website is up to you. Our recommendation is to create a personal website that presents your biographical information, interests, projects you have completed, and anything else you want the world to know about you. A personal website can serve as your digital identity and a portfolio of your work. Alternatively, you can create a website on a topic of your choosing such as a city you have enjoyed visiting or a historical figure your admire.

Whatever you choose, you should present your content in a clean and organized manner that takes advantage of the features of HTML and CSS. Although we do not cover them deeply in this class, we expect you to apply the good software engineering practices taught in this class's prerequisites (i.e. 6.031) to all assignments in this class, including this one.

Objectives

In this assignment, you will explore and gain experience with HTML and CSS, two of the most fundamental web technologies. HTML and CSS define the content and look of the web, so it is important for a web developer to have a good grasp of their fundamentals and be familiar with their features.

You will make design decisions regarding the content, layout, and styling of web pages. While visual design is not the focus of this class, web developers should aim to design applications that are easy for users to understand and operate. Thinking about design now will help you as we move to more interactive applications.

To complete this assignment, you will almost certainly need to read and learn from online documentation. Doing this effectively is a critical skill for web developers and software engineers in general.

Finally, while not the primary focus of this class, you will learn how to apply good software engineering practices (like the ones taught in 6.031) to the declarative languages of HTML and CSS. Writing maintainable code is an important skill for all professional software engineers to have, and web developers are no exception.

Specification

Your deliverable is a static website presenting information about yourself or a topic of your choosing. Your website should be written in plain HTML and CSS of your own authorship.

You must push your files to the GitHub repository we provide for you. You also must deploy your website to make it visible to others at a URL (see the deployment guide).

To ensure that you are exposed to some important aspects of HTML and CSS, we place some additional requirements on your website.

Separation of Concerns: Modularizes website so pieces can be developed independently.

- Use multiple HTML files. At least two must be used to get full credit, but you will likely want more. Different types of information should generally be on different pages (e.g. for a personal website: interests.html, projects.html, etc.).
- Use external style sheets to separate HTML from CSS. Styles should not be in HTML.
- Take advantage of CSS selectors and cascading rules to reuse styles and reduce repetition.
- Use at least one CSS variable, and take advantage of them whenever you must repeat values. Variables are particularly useful for defining color schemes.

DOM Structure: Helps make your website accessible and maintainable.

- Include (and properly use) p, h1, h2, meta, and title elements.
- Include (and properly use) at least one semantic element, and use them wherever it is appropriate to. You should not be using div if a more specific tag exists.

Color and Typography: Makes website readable and aesthetically pleasing.

- Use a font from Google Fonts.
- Set color and background-color to provide a pleasant color scheme.
- For at least one chunk of text on your website, set the font-size, line height, and other text properties for readability.

Links: How people navigate your site and the web.

- Provide a navigation bar to connect the pages of your website.
- Use an anchor element to link to some external source (e.g. your LinkedIn, a video, etc.).

Images: Useful medium for presenting information.

- Use an image on your website that has the proper alt-text, aspect ratio (should not be stretched), resolution (should not be blurred), and size.

Layout: Organize website elements into coherent structure.

- Take advantage of the power of flexible box ("flexbox"). You must use at least two distinct flex-flow settings, set the flex properties on the elements of at least one flexbox, and use one of the justification or alignment properties on at least one flexbox.

Flexibility: Display nicely regardless of screen size.

- Pages should have all of their content fit without horizontal scrolling on screens 320px and wider. Flexible size units such as % and vw can make elements adjust to the screen's width, and flexbox can be used make content wrap that does not automatically.

You are not allowed to use JavaScript in your website. We have only covered HTML and CSS so far and want you to focus on learning them well. JavaScript would only be a distraction.

You are not allowed to use any third-party libraries such as Bootstrap because they hide underlying HTML and CSS concepts that we want you to learn.

You are also not allowed to use a CMS (e.g. WordPress), website builder (e.g. Dreamweaver), or any other tool that will allow you to build a website without having to complete the assignment requirements yourself.

Fill out the submission form at the top of this document when you are done with the assignment. If you fail to fill out the form before the deadline, your assignment will be marked late.

Grading

The grading is out of 100 points. We will be grading your assignment using Google Chrome version 68+ on a laptop computer. We will not be testing your website on a smartphone or tablet, but we will slowly resize our browser window to make sure that your website satisfies the flexibility specification.

This is the rubric we will be using to grade your assignment.

Hints

Sample Student Designs: These are examples of A-level (but not necessarily perfect) work from students in the Fall 2016 offering of the class. Note that the assignment specifications were not exactly the same as they are now.

- Danny Tang
- Eric Wang
- Malika Shahrawat
- Melissa Slaughter

Reference: We recommend MDN web docs and CSS Tricks for reference. The W3C provides the official HTML5 and CSS3 specifications which are thorough but can be a bit harder to read. W3Schools (not to be confused with the W3C) is popular but occasionally provides outdated or incorrect information.

Tutorials: The Mozilla Developer Network (MDN) has some helpful tutorials on HTML and CSS. If you have never worked with HTML and CSS before, we recommend you try the tutorials before jumping into the assignment: it may save you time in the long run.