

CS200 Assignment 5 – HTML and CSS

Due Monday June 15 2020, 11:59 pm

Readings and Resources

On the web: <http://validator.w3.org/> : a site that will check a web page for faulty HTML tags
<http://jigsaw.w3.org/css-validator/> : a site that will check a web page or external CSS file for faulty CSS
<https://www.student.cs.uwaterloo.ca/~cs200/#handouts>: commented HTML

Texts: The Non-Designer's Design Book, by Robin Williams, chapters 1–6, pages 117–120
Learning Web Design 4th Ed., by Jennifer Niederst
CSS Pocket Reference, by Eric Meyer, chapters 2, 4–9, 11, 12

Other Resources: <http://www.w3schools.com/html/default.asp>
<http://www.htmlCodeTutorial.com>
http://www.westciv.com/style_master/academy/css_tutorial/
<http://codeacademy.com/learn/web>
Check our Pinterest page for other resources: <https://www.pinterest.com/cs2000143/>

Assignment Objectives

- To create a webpage for yourself using HTML and gain an understanding of how a webpage works.
- To use tables, hyperlinks and images correctly.
- To familiarize yourself with and use Cascading Style Sheets (CSS) to format your webpages.
- To gain a stronger understanding of clients and servers and how webpages are shared.

Assignment Strategy

- This assignment must be done individually.

Your Tasks

For this assignment, you will be creating a small website for yourself using a text editor (**BEdit** is recommended, but you can use any text editor of your choosing). Each question contributes to creating your website. While you won't be able to do all the questions in the first lab, you should read through the entire assignment before beginning, to get a sense of what you'll be doing. In particular, read through all of question 1 before beginning.

1. [55%] Your site must include at least 4 HTML pages and at least 1 external CSS file, as described below (the instructions for the 4th html page is in question 2). In part C you will be applying CSS to style these webpages. It will be helpful to read through all of question 1 before beginning, and you might want to consider all parts simultaneously, though you won't be able to complete all of question 1 until you've listed to the CSS lecture.

Part A:

- ☐ A page named root.html. This will be your homepage, which must include:
 - ☐ Your name and username at the top of your web page;
 - ☐ Your name as the <title>
 - ☐ An image (such as the photo you edited for A3, or something else). Note that you may only use the gif, jpeg and png image file formats in web pages.
 - ☐ Two absolute hypertext links to other web pages (ie. sites you find particularly interesting);
 - ☐ A list of something;
 - ☐ A table of something;
 - ☐ A mail-to link to your UW email account;
 - ☐ This page should be visually appealing, easy to read, well designed, and consistent with the guidelines from *The Non-Designers Design Book*. Make sure you follow the four basic principles of visual design described in the book. Part B asks you to define at least 2 of these basic principles and explain how you've applied them in root.html. You will likely find this easier after learning and applying CSS in part C.
- ☐ A relative hypertext link to each of:
 - ☐ design.html (see part B),
 - ☐ username.html (see below),
 - ☐ form.html (see question 2)
- ☐ This file must be named root.html.

- A page named `username.html`, containing the following:
 - Text of your choosing (ex. this could be more detail about something from `root.html`, or anything else you'd like).
 - There must be a relative hypertext link from `root.html` to this page, as listed above.
While you are not required to have a relative hypertext link in `design.html` to go back to `root.html`, think about why this would be a good website design choice.

Part B: Appearance of `root.html`. You will be marked for the appearance of `root.html` as displayed in Firefox or Chrome. This might be easier to do after learning about CSS and doing part C. You are also required to have the following:

- A page named `design.html`, containing the following:
 - A definition of 2 of the 4 basic principles from the *Non Designer's Design Book*.
 - A brief explanation (few sentences each) of how you applied these 2 principles to your `root.html` page. Note that you should not have violated any of the 4 principles, but you only have to define and explain your use of 2 of them. There will be bonus marks for defining and explaining your use of the other 2 principles.
 - There must be a relative hypertext link from `root.html` to this page, as listed in question 1. While you are not required to have a relative hypertext link in `design.html` to go back to `root.html`, think about why this would be a good website design choice.

Part C: You will now apply CSS to the webpages you just created. You can be as creative as you want, as long as you follow the principles in the *Non-Designer's Design Book* and satisfy the following. In either `root.html` or `username.html`, you must have:

- At least 9 distinct user-defined CSS styles defined and applied, as follows:
 - At least 3 of these must be defined in the `<head>` section of the page using a `<style>` tag pair.
 - At least 3 of these must be defined in `styles.css`, which you must link to either `root.html` or `username.html`
 - At least 3 of these must use the `style` attribute to directly apply CSS to the contents of a tag pair.

Notes:

- You must apply these 9 styles to either `root.html` or `username.html`, or you can apply some in each of those pages, as long as you have 9 total and have satisfied the above requirements.
- `Root.html` is marked for appearance in Part B based on the *Non-Designer's Design Book*, so it will be helpful to use some styles in `root.html`, though you can choose to use less than 9 if you want, and apply the others in `username.html`, which will not be marked on appearance.
- You are welcome to define and apply more than 9 styles, and you're welcome to use CSS styles in `design.html` and `form.html` (question 2) as well.
- While this is not required, you should also think about why you would want to link `styles.css` to all 4 of your webpages, and consider doing so.

Marking

70% of question 1 will be marked based on the criteria listed above.

20% of this question will be marked based on the attractiveness of your `root.html` page as displayed in Firefox and Chrome. This will be based on your use of the principles from *The Non-Designer's Design Book*, and your explanation of this in `design.html`.

10% of this question will be marked on how well organized and readable your source file is, including appropriate indentation.

Bonus: Extra credit will be awarded for an especially attractive and creative webpage, for a definition and use of the remaining design principles, for a non-trivial and interesting use of JavaScript, or for entire page created in XML.

2. [35%] Create a file named form.html containing a web form with the following items. (Note that labels are what the user of a form sees—just text—while names are what the cgi uses to identify a piece of information—they are equivalent to a “field name” in a database—and values are the actual data that is entered on the form.) You will need to look up the tags for some of these form elements. Your form must have:

- ☐ your name and username at the top of your web page;
- ☐ a hidden field with the name “identity” whose value is your full name (ie. “John Smith”);
- ☐ a one-line text input field with the label “Registration Number” and the name “idNumber”;
- ☐ a multi-line, scrolling text input field with the label “Comments” and the name “comments”;
- ☐ three radio buttons labeled “830 TT”, “1030 TT”, and “1430 WF” with the name “lab” that submit the values “Lab101”, “Lab102” and “Lab103” respectively, when selected;
- ☐ one pop-up menu or scrolling menu with the label “Application” and the name “app” that allows the user to select from “Word”, “Excel”, “HTML”, “Pixel Graphics”, “Social Media”, and “FileMaker”;
- ☐ three checkboxes labeled “Macintosh”, “Windows”, and “Linux” with names “mac”, “win”, and “unix”, respectively, each of which returns the value “Yes” when checked;
- ☐ a Submit button.
- ☐ Your form should invoke the following URL using either of the GET or POST methods.

<https://www.student.cs.uwaterloo.ca/~cs200/cgi-bin/Responder.cgi>

Clicking on the submit button of your form will cause Responder to echo back a list of the names and values of the form parameters sent to it. At the right, for example, is what Responder returned when sent data from the table-formatted grade request form discussed in lecture (which had text fields for surname and id number, and checkboxes for assignments, the midterm, the final, and the course mark).



3. [5%] Once you have completed Questions 1 and 2 you will move your webpages to your public_html folder, by doing the following:
- Under your personal network drive, find the folder labeled “public_html”.
 - Drag root.html, username.html, design.html, form.html, styles.css, and any other necessary files into this folder.
 - Your webpage should now be available at <https://www.student.cs.uwaterloo.ca/~username/root.html>, however, it will say “Forbidden”, as you will now need to change the permissions on these files...
 - Under Applications, go to Utilities and open the application Terminal.
 - To login to Terminal type ssh followed by your_userID@linux.student.cs.uwaterloo.ca, where your_userID is your Quest/student.cs username. For example:
ssh bmzister@linux.student.cs.uwaterloo.ca
 - You might be asked if you want to continue. Type “yes”.
 - When asked, type in your student.cs password and hit enter. Note that the cursor won’t move while you type your password, but it still works.
 - When you see [xx]% where xx is a number, you can continue typing...
 - Type chmod -R go+rx public_html (do not copy and paste this; make sure you type it) and hit Return.
 - All of your files should now be accessible from a browser at the url:
<https://www.student.cs.uwaterloo.ca/~username/root.html>

Note: If you want to do this from your own machine, you will first have to connect to the server: <smb://smb-files.student.cs.uwaterloo.ca> and mount your network drive. If you are doing this from off campus, you will first have to connect to UW’s VPN. Instructions on how to do this can be found here: <https://uwaterloo.ca/information-systems-technology/services/virtual-private-network-vpn/about-virtual-private-network-vpn>

Submission Instructions

- Create a folder called username_Assign5
- Move root.html, design.html, username.html, styles.css, form.html, and any other necessary files into the folder.
- Compress this folder and name it username_Assign5.zip and submit it to the **Assignment 5 DropBox** on **Learn**.