# Week 4: CSS Part I

**Agenda**

* Introduction to CSS
  + What is CSS?
  + How can we add styles to a page?
  + CSS Examples
  + First CSS Exercise
  + CSS Research Initial Post & Response
* Next Week

## Introduction to Cascading Style Sheets (CSS)

**What is CSS?**

CSS stands for Cascading Style Sheets. Unlike HTML, which is used to represent the structure of a web page, CSS is the language for the determining the presentation of a page in terms of colors, fonts, positioning, etc. We use CSS to select specific HTML elements in order to give them styles. After selecting specific elements (either directly or based on certain criteria), we can then choose whatever properties we’d like and give them different values. An example would be targeting paragraph tags and giving them a red font color. We can do that with the code below:

p {

color: red;

}

“p” is the selector (all paragraph tags), “color” is the property, and “red” is the value we are assigning to the property. The selector comes first followed by curly brackets “{}”. Inside of the curly brackets is where we choose our properties and values. The properties and values are always separated by a colon and the whole statement (color: red) is ended with a semicolon. The entire set of code from the “p” up until the closing curly bracket is a CSS declaration. You can have as many statements as you want inside of curly brackets.

It is **best practice** to separate your HTML and CSS. The reason for this is because it is easier to update your code if it is in a different location. If you have a consistent banner style on many pages of your website and use inline/internal styling, you would have to go into the code for each page and edit it. If you use external styling, you can edit the code in one place and the changes would persist across all pages.

Additional Reading

* <https://www.htmldog.com/guides/css/> - CSS Tutorials for all levels
* <https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps> - Excellent resource with guides/tutorials for CSS
* <https://www.tutorialspoint.com/css/css_syntax.htm> - Specific syntax examples
* <https://css-tricks.com/> - Cool website for code snippets and articles

Besides the links provided above, there is a PowerPoint document available on Blackboard under today’s Lecture Notes or in the Resources folder which explains everything in this document in addition to things we will learn in future weeks.

**How can we add styles to a page?**

There are three main ways for implementing CSS on a web page and each one is explained below with a link to an example. To see the code in the example, right-click the page and select “View Source” or “Inspect Element” and look around the window the appears.

Inline CSS

This is typed directly into the HTML tag as part of the style attribute.

Internal CSS

This is CSS typed between <style></style> tags which are located inside of the <head></head> tags of your HTML document.

External CSS

This is CSS placed in a separate document with a “.css” extension. It is connected to our HTML through the usage of a <link> tag which is placed inside of the <head></head> tags.

**Example Link:** <https://iinf362.000webhostapp.com/examples/using-css/>

**If you haven’t done so already, please read the “Creating and Viewing our Web Pages.docx” file on Blackboard. You will not be able to submit anything for the assignment without completing that portion first.**

**First CSS Exercise**

Download the “First-CSS-Exercise.zip” folder from Blackboard under today’s Lecture Notes or the Assignment folder. Inside of this zip folder, there will be an HTML file, a css folder with a style.css file in it, and an image. Using CSS, make t­­­­he web page look as close as possible to the image provided. It doesn’t have to be an exact replica (I’m not providing padding, margin, border sizes) but try to get the general styles placed correctly. Make sure to pay attention to the image because there are small nuances that can easily be missed with the styles (list type, text decoration, borders, etc.). You also don’t have to use the CSS file I’ve provided but your CSS should still be externally placed.

To get started, open the HTML/CSS files inside of a text editor and add code as necessary. I would recommend using classes ­on each of the main three divs to target the content inside of them. Feel free to code your CSS as you see fit though. Please note that no HTML should be removed…only classes and ids should be added as attributes to help you with selecting content using CSS. No inline or internal styling is permitted.

For this assignment and all future assignments after this one, your CSS must be **externally placed.** This is because it will be easier to manipulate styles on your page if the styles are separated. Below I have listed the requirements based on selectors, properties, and things that aren’t visible in the image.

Requirements

* Add a hover effect the <a> tag in the unordered list
  + When I hover over the <a> tag, the background of the <a> tag should become blue and the text should become white
  + You will need to use the “:hover” pseudo selector on the <a> tag for this part
* Background
* Color
* Text-align
* Border
* List-style-type
* Border-collapse
* Text-decoration
* Nth-child selector
* The grey color you see is #ccc, otherwise the color values used are white, red, blue, and green
* There are three different types of borders being used
* The light blue line you see around the list in the first section can be ignored…that’s something from Brackets when it selects HTML during screen presentation and isn’t really part of the CSS.

To receive credit for the assignment, submit **ONE** link in total. The link should direct me to the page you have created for this assignment. Make sure the CSS is placed correctly as mentioned above. The link is due Wednesday, February 19th at midnight.

**CSS Research Initial Post & Response**

In your **initial post**, do some research on any of the CSS properties listed below and report your findings. In addition to writing about what you discovered, you must create an original example demonstrating some capability of that CSS property and submit a link to that example with your initial post. The example doesn’t have to be very elaborate, but it should clearly depict what the CSS property you chose does and include an explanation of how it works. You can either submit a write up with a link or you can submit the write up inside of the HTML file used to create your example. I’ve listed some starting points, but you can include other pieces of information you find that are interesting as well.

* What is the property?
* What does it do?
* Does it require other properties for its usage?
* What version of CSS was it introduced in?
* What browsers support this property? Which ones don’t?
* Do you think this property is used very often?

Properties to Research

* Visibility
* Text-shadow
* Transform
* Clip
* Float
* Grid
* Text-transform
* Content
* Table-layout
* Animation
* Position
* Border-radius
* Box-shadow
* Flex
* Overflow
* @font-face
* Text-decoration
* Z-index
* Word-wrap
* Media queries
* Filter

In your **response post**, choose a property done by another student that isn’t the same as the one you selected initially, do some research on it as well, and create an example for it. In your post, you should include a link to this new example and an explanation as to what you did differently. If you found out some piece of information the student didn’t list, please share it.

To receive credit, you must submit **TWO** posts in total. The first post should include your write up along with a link to the example of the CSS property you chose. This initial post is due on Sunday, February 16th at midnight. The second post should include a link to an example you created based on another student’s selected property in addition to an explanation as to what you did differently. This response post is due Wednesday, February 19th at midnight.

## Next Week

Next week we will look at more complicated CSS properties that will allow us to manipulate the positioning of the content on our page. This is one of the tougher areas of web development because manipulating positioning on a web page can be difficult based on your content and layout needs. I would recommend reading up on the CSS Box Model and Floats at the very least. Below are some links to get you started:

1. <https://www.htmldog.com/guides/css/intermediate/layout/>
2. <https://css-tricks.com/all-about-floats/>
3. <https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/The_box_model>
4. <https://css-tricks.com/the-css-box-model/>