

COMM 2555: Interactive Digital Communication / Spring 2017

Assignment 3: Adding CSS to your web page

Due: Monday, February 13, 2017, by 11:59 p.m.

Assignment Description

The goal of this assignment is to get you started with writing CSS code. You will create a basic web page and add some simple content. Then you will create a stylesheet (which is what we usually call a CSS file), link it to your web page and add some basic styling to your HTML.

Step 1. Setup

Download the file source.txt - that's your web page content.

On your computer, create a new folder and call it "assignment3". Inside this folder, create two text files:

- index.html (this is your web page); or better, use the [blank HTML template page](#).
- style.css (this is your stylesheet)

Step 2. Add your content

Open your index.html file in your favorite text editor and add the content from the source.txt file (as usual, make sure to place it between the opening and closing body tags).

Step 3. Add markup

- 3.1. Surround the first line with `<h1>`, `</h1>` tags: now this is a level-1 heading
- 3.2. Surround the word "Vision" with `<h2>`, `</h2>` tags: this is a level-2 heading
- 3.3. Surround the three remaining headings with `<h3>`, `</h3>` tags: these are level-3 headings
- 3.4. Add `<p>` tags to each of the 5 paragraphs (that's the grey text)
- 3.5. Make the last line a link to this page: <https://www.w3.org/Consortium/mission>
- 3.6. Validate your HTML

That's what your browser will display:

Assignment 3

Vision

W3C's vision for the Web involves participation, sharing knowledge, and thereby building trust on a global scale.

Web for Rich Interaction

The Web was invented as a communications tool intended to allow anyone, anywhere to share information. For many years, the Web was a "read-only" tool for many. Blogs and wikis brought more authors to the Web, and social networking emerged from the flourishing market for content and personalized Web experiences. W3C standards have supported this evolution thanks to strong architecture and design principles.

Web of Data and Services

Some people view the Web as a giant repository of linked data while others as a giant set of services that exchange messages. The two views are complementary, and which to use often depends on the application.

Web of Trust

The Web has transformed the way we communicate with each other. In doing so, it has also modified the nature of our social relationships. People now "meet on the Web" and carry out commercial and personal relationships, in some cases without ever meeting in person. W3C recognizes that trust is a social phenomenon, but technology design can foster trust and confidence. As more activity moves on-line, it will become even more important to support complex interactions among parties around the globe.

Source: [The World Wide Web Consortium](#)

Step 4. Add a few styles

Open your style.css file. Add the following code:

```
▼ body {  
    margin: 50px;  
}  
▼ h1 {  
    color: maroon;  
}  
▼ h2 {  
    color: navy;  
}  
▼ h3 {  
    color: teal;  
}  
▼ p {  
    color: grey;  
}
```

Like with your HTML editing, I strongly suggest you save the file and view it in the browser (refresh the page each time to see the updated version) after making each change: that way you will see the effect of each edit.

Save and close your style.css file. Your web page should look like this:

Assignment 3

Vision

W3C's vision for the Web involves participation, sharing knowledge, and thereby building trust on a global scale.

Web for Rich Interaction

The Web was invented as a communications tool intended to allow anyone, anywhere to share information. For many years, the Web was a "read-only" tool for many. Blogs and wikis brought more authors to the Web, and social networking emerged from the flourishing market for content and personalized Web experiences. W3C standards have supported this evolution thanks to strong architecture and design principles.

Web of Data and Services

Some people view the Web as a giant repository of linked data while others as a giant set of services that exchange messages. The two views are complementary, and which to use often depends on the application.

Web of Trust

The Web has transformed the way we communicate with each other. In doing so, it has also modified the nature of our social relationships. People now "meet on the Web" and carry out commercial and personal relationships, in some cases without ever meeting in person. W3C recognizes that trust is a social phenomenon, but technology design can foster trust and confidence. As more activity moves on-line, it will become even more important to support complex interactions among parties around the globe.

Source: [The World Wide Web Consortium](#)

Step 5. Validate your CSS

Use the W3C validation service to validate your CSS code by uploading your style.css file:

https://jigsaw.w3.org/css-validator/#validate_by_upload

Like with the HTML validator, if you see a long list of errors, look at the line number of the first one and find what's causing it (you must have made a typo or missed a character). Fix it and revalidate (you must re-upload the updated file). Repeat until you have no errors.

Code requirements

1. Your tags are nested correctly: `<tag1><tag2></tag2></tag1>`
2. You are not missing any required closing tags
3. Attribute values are surrounded with quotation marks: `<tag attribute="value">`
4. Use proper indentation to make your code easier to read
5. Use lowercase for your tags (except `<!DOCTYPE html>`)

Submissions

Submit 2 files to eLearning: **index.html** and **style.css**

Grading

The assignment is worth **45 points**, which is **approximately 5.7%** of your grade.

- Your index.html file contains all the required HTML code (**10 points**)
- Your style.css file contains all the required CSS code (**10 points**)
- Your HTML code validates without errors (**10 points**)
- Your CSS code validates without errors (**10 points**)
- Your files are named correctly (all lowercase, no spaces) (**5 points**)
- Points will be deducted for code requirements that were not fulfilled (see code requirements above)