# The Internet in Laymen's Terms

A basic webpage is just an HTML file with a CSS file linked to it that sits on someone else's (really big) computer in a file directory (fig. 1) like the one in your "Finder" app.

When you type in a web address, you're telling your browser where that file is: the computer where it's stored and which file on that computer that you want.

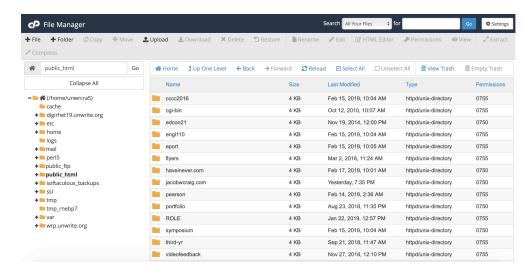


Figure 1. What a server looks like

# HTML & CSS in Laymen's Terms

HTML is a markup language used to name and group elements in a document CSS is a language used to tell browsers how elements should look in the document

In an ideal world HTML and CSS live in separate files, joined only by the link included in the heading of the HTML file...

<head><link rel="stylesheet" text="style/css" href="style.css"></head>

```
1
    <!DOCTYPE html>
 2
    <html>
 3
        <head>
 4
            <title>Example<title>
 5
            <link rel="stylesheet" href="styl</pre>
 6
        </head>
 7
        <body>
 8
            <h1>
 9
                 <a href="/">Header</a>
10
            </h1>
11
            <nav>
12
                 <a href="one/">0ne</a>
13
                 <a href="two/">Two</a>
                 <a href="three/">Three</a>
14
15
            </nav>
```

```
p
{
    color:Red;
    font-size:x-large;
    font-femily:Verdana;
    background-color:Gray;
}

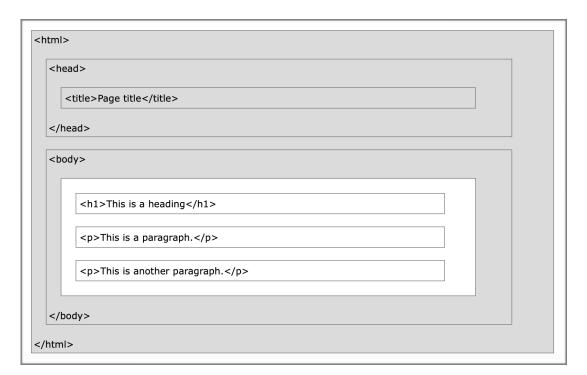
hi
{
    color:Blue;
    background-color:Yellow;
    font-family:Comic Sans MS;
}

.pForHyPage3
{
    color:Gray;
    font-size:x-large;
    font-femily:Verdana;
    background-color:Red;
}

.ButtonControl
{
    background-color:Red;
    color:Yellow;
}
```

Let's live in an ideal world for a bit.

- 1. Download the folder shared through the workshop page on the WRP blog
- 2. Open the HTML file in your preferred browser. This is all the browser knows without markup. So, let's tell the browser some things about this file.
  - 1. Tell the browser it's an HTML file <!DOCTYPE html>
  - 2. Tell the browser that everything in the file is HTML—it's not always. <html></html>
  - 3. Tell the browser that there are two sections in the HTML file <head></head> and <body></body>
  - 4. Use the <head> section to tell the browser about the file
    - 1. There's a title (browsers use this to label tabs): <title><title>
    - 2. There's a stylesheet rel="stylesheet"
    - 3. and where the style sheet is href="style.css"



Tell the browser what's in the <body> section with tags. In HTML, all tags come in pairs. The first tag is the start tag. The second tag is the end tag. Tags always bookend whatever they're defining—whether it's the whole file like in the <html> tag or part of the file like in the <body> tag or particular elements in the document like in the tags below.

<h1></h1> — First level headings <h2></h2>—Second level headings



—paragraphs —lists —unordered lists —ordered lists <b></b>—bolded text <i></i>—italicized text

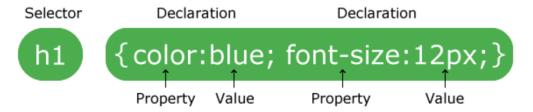
Use the tags listed above to tell your browser a little bit about what's in the <body>

- 5. Indicate the title of the website through the **<h1>** tag
- 6. Indicate the title of the webpage through the <h2> tag
- 7. Indicate other sections through the <h3> tag
- 8. Tell the browser where each paragraph starts and ends through the tag
- 9. Define your two lists, placing the tag at the beginning of each list and the end tag at the end of each list
- 10. Tell the browser what's in the list by placing the <il> tag at the beginning of each item in the list that you want bulleted and the </il> tag at the end of each item in the list you want bulleted
- 11. Repeat the process for the list that you want numbered, using the tag
- 12. Bold something with the <b> tag
- 13. Italicize something with the <i> tag

Save and open your document in your browser; it should look a little different now—although not great yet.

Open your CSS file. Here's where we'll tell the browser how things should look. For now, we'll just make a few rules.

#### A CSS rule-set consists of a selector and a declaration block:



The most powerful rule you can write is for the **body** because it will cover everything you tagged in the HTML file. Whatever rules you make for the **body** will apply to specific elements in the document. In other words, if you make a rule for the **body**, that rule will automatically apply to all the p's, h1's, h2's, h3's, etc.

There are lots of properties and values possible in HTML. Here's a sample:

#### Defines color of text in hex or color names

background-color color

### **Defines font-type**

font-family

Defines where the element is aligned left, right, center

text-align

### Defines size measured in pixels (px)

width

height

font-size

# So make a rule for the body defining

- a background color
- a font-family
- an alignment
- a font color
- a font size
- the width of the whole page

Save both files and open the HTML file again; things should now look different.

Your task, now, is to make specific rules for three of the elements you specified in your HTML file:

h1 {	}
h2 {	}
h3 {	}
ul {	}
ol {	}

#### **Customizing pages**

Create a blog in Google's Blogger using the Contempto template and start a new post that includes text from the *blogger-text.rtf* file and images from the *blogger-img* folder. Make it look as close as you can to the screenshot on the right.

<sup>\*</sup>for your ul and ol rules, in addition to naming a font-size and color, also make rules for list-style-type by putting the ordered list in roman numerals (**upper-roman**) and the unordered list in square bullets (**square**)

We'll be using 3 pieces of in-line CSS to style the post

border to create a border around your imageswidth to make images a uniform widthpadding-left to create uniform space between image and textCustomizing templates

Open your published blog as readers would see it in Firefox and open a new file in Brackets/Notepad +++

We're going to customize 6 things about this template

- 1. Change the size and color of the header
- 2. Change the size and color of the date of the post
- 3. Change the color of the background of the post
- 4. Change the size and color of the post title
- 5. Change the body text font to Verdana
- 6. Change the header fonts to Bookman

After each change, copy your line of CSS into your Brackets file. We'll need this.



"The FSU Symposium: Origins, Revisions, and Reflections." (with Rory Lee and David Bedsole). Computers and Composition Online. (2018).

In this article, we offer a history of Florida State's (FSU) Digital Symposium, a celebration of student work that supports FSU's Signification, and a student working programs as well as FSU's Digital Studio. In recounting the history of FSU's Digital Studio. In recounting the history of FSU's Digital Symposium, we show how everst like FSU's Symposium can be a powerful means of institutional change to cultivate a departmental and programmatic culture that values digital writing and multimodal composition.



"Navigating a Varied Landscape: Literacy and the Credibility of Networked Information," Literacy, Democracy, and Fake News, special issue of Literacy in Composition Studies, vol. 5, no. 2, pp. 24-42. (2017).

"Craig takes note of the rhetorical skills of the Macedonian teenagers who compose fake news stories to profit from click-bait addj...] As Craig discusses, such efforts to teach civic reasoning can be enhanced by attending to how the rhetorical dynamics of network literactes complicate print-centric assumptions that close reading teaches students to be reasonable. As Craig's article notes, such conceptions of information literacy do not attend to the rhetorical complexities involved in surfing across diverse platforms, media, and