[ANONYMIZED]
COMM 362
16 October 2019

Assignment 2 Reflection

Semantics, or "related to meaning", are very important for making web pages on the World Wide Web; Sir Tim Berners-Lee wanted everyone to be able to understand HTML (HyperText Markup Language) (Sandvig 2019). Thus, he designed the commands to be fairly intuitive on their own. HTML, true to its full name, is considered a "markup" language. Quite literally, HTML was there to "mark up" the web page, akin to a teacher editing a student's paper. The tags and attributes in HTML are supposed to have meaning, or semantics. These meanings would make it easy for everyone who used the web to understand what they meant. In order to ensure HTML contained mostly, if not only semantic usage of tags, CSS (Cascading Style Sheets) was created to be a "layout" language and contain nearly all the tags and attributes related to the formatting of the page. That meant, several of the tags in HTML became obsolete (such as <font>) because their effects can, and arguably *should* be created using CSS. All the semantic usages of tags were supposed to be dealt with in HTML, and CSS would take care of the layout.

But that being said, are layout and meaning that separate? What constitutes if something has "meaning"? And, why does this matter? The third question is by far the easiest to answer. Semantics matter, because HTML as a language is designed for simplicity and human readability. Beyond that, machine readability is important too! Programs like screen readers and search engines need to read the contents of a web page for the visually impaired, or to list search results. If the HTML consists of a garbled mess of non-semantic tags like <div> and <span> (two examples of tags with no particular meaning), it will be much more difficult for the machine to understand what is important in the web page.

This discussion around semantics also has practical implications for web page designers. When multiple tags in HTML visually have the same effect, the one that is considered "better" should be used. Many years ago, before taking this class, I used my very primitive knowledge of HTML to design and format my tumblr blog. Now looking back at it, I cringe at the very poor coding and bad practice. For example, there are so many non-semantic tags like <b> and <i> strewn in my code, it is embarrassing. The <i> tag and <em> tag both have the effect of making something *italicized*. Despite them being visually identical, the coding is not the same; a screen reader will vocally emphasize something tagged with <em> but not with <i>. This logic does make some sense though; the 'i' within the tag stands for italics, but 'italic' speech does not have a lot of meaning. Emphasis (em), on the other hand, has actual meaning. The example of semantic differences with the bold and emphasis tags were just within the HTML language itself. Choosing between using tags in HTML and CSS is where many people disagree.

For example, in this assignment, I had to create a table in one of my webpages. The instructions explicitly told me to use the HTML <table> tag, so naturally I did so. However, when I checked my file using the HTML5 Validator, it informed me that using the <table> tag was bad practice and to use CSS for a table instead. The table looked perfectly fine when viewed in Google Chrome, but the problem was that the code was not good at all for a screen reader or some other application trying to read my table (Mills 2010). Thus, while it was decent in human understandability, it was lacking in machine readability due to me using poor practice. Although I did not change my web page to use CSS for the table (because the assignment instructions clearly stated to use the <table> tag), I am aware that the best practice was to use the CSS.

The dialogue between web developers surrounding semantics and their necessity relating to HTML and CSS is not entirely clear cut. Deciding if a tag or attribute has "meaning" (and

what constitutes meaning) feels rather arbitrary and subjective; the <p> tag means paragraph, which makes enough sense. Except, how do paragraphs have inherent meaning? Screen readers do not necessarily say when they move onto the next paragraph nor does it affect how they read the text. Thus, the answer to the discussion surrounding semantics is not so simple. That being said, the key point is understandability and readability both from the machine and the humans behind it; naturally the code that is easiest and most simple for both humans and computers to read should be the 'best practice'; that is what Sir Tim Berners-Lee wanted, after all.

References

Mills, C. (2010). Basic table elements. In M. J. Nolan (Ed.), Title of book (319). Berkeley, CA:

New Riders.

Sandvig, C. (2019). Semantics [PowerPoint slides]. Retrieved from a Google Drive Folder.