Project report

Questions related to your web page

How have you incorporated accessibility, usability, reliability and readability in your website?

I have mainly left the font sizes alone, so the user's preferences should adjust the font size as needed, rather than specifying a specific pixel size.

Ive attempted to maximise readability by choosing light backgrounds and dark text or dark background with light text. Ive checked it on other websites to see if my contrast ratio was high enough. Most colour combinations are quite high 8.45:1. Only very rarely using a light yellow on dark green for highlighting.

To make the website more usable, i made sure to follow typical header and navigation conventions that are typical in most other websites, like having a coloured header that stands out, making the logo redirect the user to the homepage, having elements underline and change colour when highlighted and so on. This navigation bar should be familiar to the user and will make it easier for them to navigate the website and find other pages.

Always using alt text in case images dont load correctly so the user still knows what should appear.

I have also made the webpage's general layout and design consistent throughout the website.

How have you maintained the Quality of your HTML and CSS code?

Ive tried to maintain code quality by keeping style elements inside external .CSS files and style sheets rather than inline elements or styles specified in the html. I have also tried to make it so that elements remain consistent throughout.

I have also made sure to use correct indentation to make the html code as readable as possible. I've included comments to explain sections needed and seperate elements as needed.

All ids and classes have been named appropriately to what they do, what they should or what they contain look like. I've attempted to minimize duplicated code by re-reading my CSS files and attempting to group elements with similar attributes together.

Keeping multiline elements like padding, margins and borders as a single line rather than having them in multiple lines

Research Theme Questions

CSS Module 2

Due to CSS not having some typical features of a programming language, such as variables, function and so on, it is easy for CSS to have lot of redundant, repeated lines of code. Thus, making the code harder to read and parse, and because a large portion of a browser's processing power is taken by analysing and applying CSS to HTML files, its worth attempting to clean up the CSS code as much as possible to reduce processing time. By using scripts to detect "clones" of information, the researchers have analysed a collection of websites and found that most have redundant CSS code, like information and attributes that were stated more than once. Most redundant code being "Declarations having lexically identical values for given properties." Or what the article puts as Type 1. To avoid this bloat read through your own style sheets and check for any redundancies.

What are various types of duplication in CSS code mentioned in this paper?

There were 3 main types of CSS duplications

1. Type 1 was declaring identical properties for more than 1 selector. The exact same attributes being used twice in 2 separate selector tags rather than putting a comma between the 2 tags and declaring the attributes in the one chunk

2. Type 2 was having attributes with the same values but different names, while this doesn't reduce the size of the CSS documents, it makes them easier to read.

3. Type 3 was having declarations be seperated when they there is shorthand available to make the attributes easier to read

How can you minimize this duplication while designing websites?

To minimise type 1 duplication, if you find chunks of CSS declarations that have the same attributes, you can declare them both in the same selector separated by a comma. Then if there are any differences, they can be described separately in their own selectors. Because of the comma, all properties described in the declaration will be given to both selectors.

To minimise type 2 read through the CSS code to find any duplicates of colour values, font size, length, angles or frequency and replace them so all declarations are consistent with each other.

To minimize type 3, look through the CSS documents for certain CSS properties that have shorthand available. E.g. padding, margin, background, if there are any declarations that have multiple lines describing the properties, then refactor the code so that it is all in one like. For example, if padding was described as.

In the article scripts are described to automatically refactor large CSS files without changing anything about the appearance of the website

CSS Module 3

Summarize this research.

“All bugs are not created equal. Some bugs occur over and over again in novices’ programs, while others occur rarely.”

This article looks into the what bugs and errors people might encounter, based on their experience level. Different bug types that occur in HTML and CSS. When most people encounter or think about bugs, they typically think of syntactical errors, i.e. forgetting semi colons or closing brackets. When people who are inexperienced with coding attempt to write things in HTML they find issues with the interactions between different pieces of code or boundary conditions the programmers weren’t taking into account.

70.9 percent of errors occurred at the skill-based, 16.9 percent at the rule-based, and 12.1 percent at the knowledge-based levels. A scant 4.3 percent of skill-based errors were unresolved, while 39.6 percent of rulebased and 52.1 percent of knowledge-based remained so. Tables 7 to 9 provide a description and example for each type, and a count of total and unresolved occurrences.

While knowledge errors are the least common of the types of errors, they are the ones that will remain unsolved

What are the errors the researchers detected in terms of HTML and CSS code writing?

There are 3 main errors that the article accounts for, skill based, rule based, knowledge based

- Skill based
    - Errors here are caused by unintentional actions, slip of the finger, typo, minor syntax issues, closing of parenthesis and so on. Letting the person know the presence of the error is enough to fix it

- Rule based
    - Due to not knowing correct syntax or conventions of the language e.g. not knowing classes cannot begin with a number.
    - These errors occur intentionally and are not caused by a slip of the hand. They are consistent as it occurs due to an unfamiliarity with a certain language
    - Elaboration of a certain rule i.e. “classes cannot begin with a number” would easily solve this error
- Knowledge based
    - Errors that need to be learnt over time. Where there are gaps in knowledge and the person making the error need to be taught more concepts before the error can ve fixed. Web searches may help in understanding the error but more substantial learning may be needed to avoid them and fix in the future
    - This occurs from a lack of relevant knowledge

How do you think you can minimise such errors?

Skill based errors will typically be solved quickly for me since I code on VSCode, the program will automatically colour my code as I am writing it. So missing a quotation mark will colour the code differently and items inside a parenthesis will be coloured differently. Missing delimiters will be underlined. These issues will be solved quickly. If something is syntaically wrong, VSCode will also underline to alert me of the issue

Rule based errors are trickier but will generally be solved with google searches more easily or asking other more experience people for help.

Knowledge based errors are mainly avoided with experience, these are much less easily solved with google searches so also asking for help from tutors or other people.