**Developer Tools - Client-Side Debugging**

Assignment 9.4

By: Diandra McKenzie

The Google Chrome Developer Tools, also known as Chrome DevTools, are web authoring and debugging tools built right into the browser. Chrome DevTools can greatly improve workflow by helping to develop, test, and debug websites from right within the Chrome browser. They provide developers deeper access into their web applications and the browser. Everything from testing the viewport on a mobile device to even measuring the performance of an entire website or individual assets can be done by utilizing Chrome DevTools. (Smith, 2019) Once a developer has opened Chrome DevTools, there are eight panels from in which they can select and then edit code. Let’s explore two of these panels and examine exactly what a developer can do within each of them.

The first panel we will discuss is the Elements Panel. The Elements panel is where you can see the elements that are currently available in the DOM (Document Object Modal). This panel allows a developer to inspect and modify a site from the front-end. It’s used to change the appearance and content of a web page by editing its CSS and HTML files. This panel includes the DOM pane and CSS pane. The DOM pane is used to change the layout. The CSS pane is used to modify the style rules by manipulating CSS properties. (L., 2019) By default, CSS modifications are not permanent, changes are lost when you reload the page. Any edits should be copied and then added to the root CSS file.

The next panel we will discuss is the Sources panel. From the Sources panel you can view your files, edit CSS and JavaScript but there are a few different features that will also allow a developer to create, save and run snippets, and debug JavaScript. Within the editor pane is where a developer can edit CSS and JavaScript. Chrome DevTools updates the page to run the edited code so changes take effect immediately. But again, keep in mind any changes are not saved when the page is reloaded. As far as snippets, they are scripts which can be run on any page. Code can be saved in a snippet and then saved to your file system instead of having to repeatedly type out the same code in the Console. Instead of using console.log() to gather where a JavaScript code has an issue, a more efficient way for developers to debug is to utilize Chrome DevTools. By setting a breakpoint, the code can be paused and then stepped through one line at a time, so it can be viewed and values can be changed for all currently-defined properties and variables. (Basques, 2019)

Besides being user friendly and easy to navigate Chrome DevTools offers rapid debugging, the console pane shows errors in code. Testing responsive design for many different interfaces is also a huge benefit to using Chrome DevTools. I’m looking forward to becoming more acclimated the all the benefits offered by this resource.

# References

Basques, K. (2019, February 12). *Sources Panel Overview*. Retrieved February 6, 2020, from Tools for Web Developers: https://developers.google.com/web/tools/chrome-devtools/sources

L., L. (2019, September 13). *How to Inspect and Change Page Style Using Google Chrome*. Retrieved February 6, 2020, from Hostinger Tutorials: https://www.hostinger.com/tutorials/website/how-to-inspect-and-change-style-using-google-chrome

Smith, M. (2019, November 5). *Chrome DevTools - 20+ Tips and Tricks*. Retrieved February 6, 2020, from keycdn: https://www.keycdn.com/blog/chrome-devtools