Discussion Board 1.1 – Code Organization

Eric McCool

Bellevue University School of Science and Technology

Web231-339A Enterprise JavaScript I

Professor Krasso

October 17, 2022

This week's discussion is on Code Organization and some general rules. The topics include naming and declaring variables and functions, indentation, and comments. These topics are then broken down further to include the 'why' these rules are important.

The first topic is the naming convention for declaring variables and functions. It is recommended that these names, or identifiers (w3schools, 2022), be written in 'camelCase'. This style of naming combines multiple words run together with no spaces. The name should begin with a lowercase letter and any subsequent words begin with a capital letter e.g. camelCase or hourlyPay. If the name requires just one word, it should be in all lowercase. JavaScript identifiers are also case-sensitive, so firstBase and FirstBase represent two different variables (w3schools, 2022).

This rule is important to maintain a standard that helps to eliminate confusion and errors. Variables and functions should not start with a number, as the JavaScript compiler or web browser may try to interpret the name as a value and throw an error. Underscores and the dollar sign symbols can be used (w3schools, 2022), but should also be avoided because these symbols can represent other methods or have a different syntax use such as using strings in the console (Mozilla, 2022). To further avoid errors at runtime, do not use JavaScript keywords such as new or this (w3schools, 2022).

The second topic is indentation. The convention for code indentation in JavaScript is to use two spaces. It is recommended that the tab key not be used. There are many editors and browsers and each, if not configured properly, will translate the tab function differently (w3schools, 2022).

This rule is important to maintain a clear standard. By using this standard, other developers who will work with or read the code know what to expect. This improves readability and decreases confusion (Mozilla, 2022).

Lastly is rules regarding code for comments and using white space. There are both block comments and single-line comments. Here, there are many views and arguments both for and against one or the other. Mozilla suggests avoiding block comments and using single-line and in-line comments (Mozilla, 2022), whereas w3schools suggests using block comments for formal documentation (w3schools, 2022). Another point of contention regarding comments is the use of punctuation. While Mozilla suggests no punctuation at the end of a comment (Mozilla, 2022), one of the instructor's handouts recommends punctuation (Krasso, 2022). Seeing as how Professor Krasso is the overseeing authority, punctuation in the comments should be applied.

The rules for whitespace are loose due to the fact that JavaScript and some other languages ignore whitespace. Mozilla makes some recommendations on where to use spaces, particularly around operators (+, -, *, /, =). A leading space before the opening brace enclosing a block of code is also recommended (Mozilla, 2022). This is a valuable leniency and should be employed liberally in the interest of clarity and readability of code since it has no bearing on application performance.

All of the above mentioned rules and conventions are important. They can increase the readability of code while reducing the chances for confusion and errors to occur.

References

Krasso, R. (n.d.). *Coding Guidelines.* Enterprise JavaScript I. Bellevue University, Bellevue, NE

68005, 2022. https://cyberactive.bellevue.edu/ultra/courses/_513850_1/cl/outline

Mozzila, 2022. *Guidelines for Styling JavaScript Code Examples.* The MDN Web Docs Project.

Mozilla Corporation, 1998-2022.

https://developer.mozilla.org/en-US/docs/MDN/Writing_guidelines/

Writing_style_guide/Code_style_guide/JavaScript

#general_guidelines_for_javascript_code_examples

W3Schools, 2022. *JavaScript Style Guide.* JavaScript Tutorial. W3Schools.com. Refsnes Data,

1999-2022. https://www.w3schools.com/js/js_conventions.asp