Keegan Jones

CSD – 360 JavaScript

Module 2

5/25/2021

There are 4 main JavaScript display possibilities, innerHTML, document.write(), window.alert(), and console.log(). The first one, innerHTML is a property of an element that can be used to either get or set the HTML markup. I found a great explanation and example of how to use innerHTML to get the contents of an HTML element. (JavaScript Tutorial, n.d.)

Graphical user interface, text, application

Description automatically generated

You can also use this to set the contents to something else or even clear the contents, this is very powerful and could cause some pretty serious problems if not handled correctly. It will literally delete everything on the page and replace it with whatever you write, whereas with innerHTML you can make a change to just one part of the HTML.

The second option, document.write() also looks like it could be risky to use. If you use this after the HTML document has loaded it will delete all existing HTML in the document. This is used for testing purposes. I found that the best example for this was the one at w3schools, <https://www.w3schools.com/js/tryit.asp?filename=tryjs_output_write_over>, as it shows how everything else goes away. (W3 Schools, n.d.) Generally on most of the forums that I looked at, developers tend to consider document.write() to be bad practice and they recommend not to use it, although if used correctly has its place.

The third option, window.alert () is fun. It will create a pop-up or alert box that the user is required to dismiss to continue. These are a great option for when there is no other requirement on the part of the user except acknowledgement. It is important to note, though, that sometimes the user may not be required to dismiss it in certain circumstances such as if the user switches to a different tab and then returns. You can also choose to omit the first part, “window.”, and it will still work as seen in the example below. (MDN Web Docs, n.d.)

Graphical user interface, application

Description automatically generated

There is also a window.confirm(), which is very similar but it allows for a cancel button. If the user selects ok the box returns true, but if the user selects cancel the box returns false. With an alert there is no return. (W3 Schools, n.d.)

The last one that I will discuss is, console.log (). This is predominantly used to debug as it makes the output be printed to the console. It is native, easy to read and is easy to use but most of what I see when researching its usefulness leads me to believe that many developers have a love/hate relationship with console. Another great source of information that I found comes from Code Academy, <https://www.codecademy.com/forum_questions/5371069f52f863ceb5000a64>. In this article they discuss that console.log() is a function that you can call to print something to the console. The console is a debugging environment where you can see temporary results of testing of code. Console.log() in not part of JavaScript and must be used with a console environment. (Codecademy, n.d.)

The most common method of display possibilities in JavaScript, is probably innerHTML. It is easy and safe. For debugging the console.log() seems to be very safe, easy to use and useful as well.

# Works Cited

*Codecademy*. (n.d.). Retrieved from codecademy.com: https://www.codecademy.com/forum\_questions/5371069f52f863ceb5000a64

*JavaScript Tutorial*. (n.d.). Retrieved from javascripttutorial.net: https://www.javascripttutorial.net/javascript-dom/javascript-innerhtml/

*MDN Web Docs*. (n.d.). Retrieved from developter.mozilla.org: https://developer.mozilla.org/en-US/docs/Web/API/Window/alert

*W3 Schools*. (n.d.). Retrieved from w3schools.com: https://www.w3schools.com/js/tryit.asp?filename=tryjs\_output\_write\_over

*W3 Schools*. (n.d.). Retrieved from w3schools.com: https://www.w3schools.com/js/js\_popup.asp