

# **COMP 266: Unit 3**

Cascading Styling Sheets

Richard Sullivan Andison

2026, January

# Unit Review

Cascading Style Sheets (CSS) are a means of dictating an HTML element's shape, position, and appearance via attribute parameters, tag inheritance structures (basic, element, class, and ID), and styling frameworks (basic, grid, and flexboxes) to form a document's presentational guidelines [1]-[2]. CSS was invented as a means to address issues with structuring sites and documents solely with XML or HTML for presentation [1]. CSS presents HTML elements drawing upon object-oriented principles. Firstly, CSS files are designed to be polymorphic in such a manner that a single file can be used across multiple HTML documents or settings can be applied across different scopes of elements. Secondly, CSS sheets also use properties and terms to abstract away certain complex elements; for example, flexboxes and grids are a means of mentally picturing how elements fit on the page and interact with one another without needing to understand what logic is involved. Third, CSS encapsulates styling elements which allow for a cleaner and more logical organization of site elements. And finally, CSS uses inheritance via class hierarchies and identification to organize and simplify element presentation. Currently, CSS is in its third version.

The New 3DS browser provides full support of CSS 1 and 2, but only has limited functionalities for CSS 3 (Nintendo's official browser specifications state "some" CSS 3 support is available, but any elaboration to what these limitations are have not been found [3]. Some exploration to determine what features were available was made,

such as viewing the open-source files for the web browser; the files did reveal some information about the 3DS's Javascript functionality, but not much could be gathered on the CSS [5]. When developing the 3DS library, most of the testing for CSS implementations occurred through Firefox's browser inspector and style editing tools, but each change had to be tested on an actual 3DS system to assure cross platform functionality; some features that are not supported are flexboxes, grids, and opacity. Additionally, considerations for memory and computing limitations were made; this included resizing background images and gifs to lower resolutions, and staying to simplistic UI formats. Finally, assuring screen elements displayed correctly and centered on both New 3DS and desktop browsers required several iterations to achieve a satisfying result. For the scope of the website, it seemed reasonable to contain all site styling rules within a single CSS sheet; the first sheet would act as a base template for future themes or modifications.

In creating the first styling sheet, some elements required going back and modifying the HTML. One example of this was the creation of an additional “backing div” that served as a container for both screens on the 3DS. The backing div allowed for several improvements by allowing both screen elements to be re-positioned together (this element was later replaced by the body tag to simplify the site structure). A second benefit to setting a backing element was to deal with a quirk of the 3DS browser. Other 3DS sites, such as Wolfyon's site center the screen to the 3DS's default page position to assure it is centered, but on a desktop browser it appears as left-

justified. In order to fix this for my site, a conditional function was used (from Wolfyon’s “3DSWebStuff github repository) that updated the bodys positional CSS (via my new code) if a 3DS system was not detected [4]. The first successfully finished CSS theme was the “starry” theme that featured a moving gif as the site background. In creating new themes from the starry theme some issues were discovered to have been hidden by the dark backgrounds of the original theme, such as subtitle and heading background widths, photo alignment on the index screen, and screen centering/bordering issues were all difficult to detect due to the dark background. To amend this in future themes, borders for each div were established so each elements position could be clearly seen. Furthermore, a more contrasting, bright theme, “blueberry avo papaya,” was created to serve as a better template for future site themes. At the time of writing this, there are four site themes for the user to choose from: “starry”, “blueberry avo papaya,” global village,” and “old school.”

Other major considerations involving CSS involved displaying of HTML literature from various sources with different formatting rules. Formatting the literature had several difficulties and developments. Early iterations of the library utilized UTF-8 encoded files to display literature, but limitations were found in parsing text in meaningful chunks and displaying titles, headers, and images were deemed too complicated to implement; UTF-8 literature was first chosen due to the limited functionality of Javascript on the 3DS preventing HTML modification, (functions for inserting and removing elements did not work) but was later replaced

after it was discovered that changing the inner HTML of an element was still possible (therefore, element can be “created” and “destroyed” by inserting HTML tags as raw text into preexisting elements). Formatting literature for the 3DS typically consisted of removing unnecessary elements (such as page numbers and preexisting stylings, classes and id’s) before being added to the catalogue page/site storage. Future development could implement a system for changing font size and type, setting a display screen (top, bottom, both), and UI modes (touchscreen, buttons, keyboard, mouse), and implementing a more desktop-friendly UI.

In terms of software, Firefox and KATE were the primary means for development. At the start, elements were modified via the inspector tool with Firefox’s default web utilities. Later, the “style editor” tab was used for the majority of modification after it was discovered. In this way, changes could be seen before the original CSS is overwritten and pushed to the site repository. A final note is the use of Git and GitHub as a means of tracking major and minor changes to the site; I found the ability to roll back to previous versions or see what was changed to have been helpful in several circumstances.

To summarize, fundamental elements of CSS, such as position, margins, paddings, and color schemes were applied to decorate and organize the 3DS library’s HTML elements and provide theming capabilities. While the overall structure of the site is fairly simplistic, there were several hurdles in discovering and overcoming

limitations to the 3DS system which required learning a deeper understanding about CSS attributes and how their modification affects and interacts with other elements. In general, limitations were solved by rolling back to a more simplistic and less abstract form of CSS, such as using margins, absolute sizing, and positioning. A primary focus for this unit was developing a workflow that was enjoyable yet effective for creating CSS files that adhered to commonly accepted rules and practices (this would be reinforced via implementing multiple website themes). After some time, a cycle of making changes via Firefox's browser tools before saving and pushing changes to GitHub via KATE was followed and yielded four site themes: "starry," "papaya avo blueberry," "global village," and "old school." While generally simplistic, the overall CSS framework for the New 3DS Library is promising in that new themes can be created quickly with the use of sensible classes, element identification, and attribute settings. A final note would be that while I do feel as if I've developed a firm grasp on basic CSS syntax and methodologies, I do feel that learning to implementing newer and more advanced methodologies could prove useful.

# Resources Used

[1] GeeksforGeeks, “CSS History and Versions,” *GeeksforGeeks*, Feb. 07, 2024.

<https://www.geeksforgeeks.org/css/css-history-versions/>

[2] GeeksforGeeks, “CSS Selectors,” *GeeksforGeeks*, Sep. 24, 2021.

<https://www.geeksforgeeks.org/css/css-selectors/> (accessed Jan. 31, 2026).

[3] “Nintendo 3DS Internet Browser Specs | Nintendo Support,” *Nintendo.com*, 2026.

[https://en-americas-support.nintendo.com/app/answers/detail/a\\_id/13802/~nintendo-3ds-internet-browser-specs](https://en-americas-support.nintendo.com/app/answers/detail/a_id/13802/~nintendo-3ds-internet-browser-specs) (accessed Jan. 31, 2026).

[4] “3DS Web Stuff,” *Github.io*, 2024. <https://wolfyxon.github.io/3ds-web-stuff/>

[5] “Nintendo Open Source Software source code distribution page / ニンテンドー オープンソースソフトウェアのソースコード配布ページ,” *Nintendo*, 2026. <https://support.nintendo.com/jp/oss/index.html> (accessed Feb. 02, 2026).

