

Cascading Style Sheets (CSS)



For week 4, fellows will take their webpage to the next level by learning how to style it with CSS! They would learn to transform the look and feel of your creation, adding personality and visual appeal. By the end of this week, their webpage will be transformed with a unique and personalized style, showcasing your creativity and understanding of CSS. They will be well on their way to becoming a software developer who can not only build but also bring their vision to life visually.

Learning Objectives for the week

At the end of this week you should be able to;

- ★ **CSS Basics:** Understand CSS syntax (selectors, properties, values) and its role in styling web pages.
- ★ **Selectors:** Use element, class, ID, and advanced selectors (e.g., pseudo-classes, descendant) to target elements.
- ★ **Text and Layout:** Style text (fonts, colors) and apply the box model, **display**, **position**, and **float** for layouts.
- ★ **Responsive Design:** Implement responsive layouts with media queries, flexible grids, and mobile-first design.
- ★ **Stylesheets:** Differentiate between inline, internal, and external stylesheets; link external CSS files.
- ★ **Flexbox and Grid:** Use **Flexbox** for flexible layouts and **CSS Grid** for complex, two-dimensional layouts.



Software Development Week 4



Online Learning Modules

This is the online module that you have to complete this week.

- CSS Full Course: [Click Here](#)
- **Module 3:** Understanding CSS: [Click Here](#)

Additional Resource links

1. [Click here](#) - Text
2. [Click here](#) - Text



Software Development Week 4



Weekly Applied Learning Assignment

For this week:

- Create a new pen on CodePen.io
- Write and test your code within the editor.
- Once you're satisfied, save your work and copy the pen's URL.
- Submit the URL in the assignment submission tab of your dashboard.

Below is the task:

1. **Selectors and Styling:**

Style a webpage using different CSS selectors (element, class, ID, pseudo-classes) and advanced selectors like descendant or sibling.

Responsive Design:

2. Build a responsive webpage using media queries that adjusts for mobile, tablet, and desktop views.

3. **Flexbox**

and

Grid:

Create one page layout using **Flexbox** and another using **CSS Grid**, each with a header, content section, and footer.

Transitions and Animations:

4. Add transitions to buttons or links and create a simple animation using **@keyframes** (e.g., bouncing or sliding effect).