Report Day: 3.5

Submitted By: Ayesha Abid

Submitted To: Mr. Ali Hyder Hidayat

Group: Frontend Development @ProSensia

Dated: 11/07/2025

Learning Outcomes:

CSS Variables:

Introduction:

CSS Variables, also known as Custom Properties, are a powerful feature introduced in **CSS3** that allow developers to store values in reusable variables, similar to variables in programming languages. This makes stylesheets more maintainable, dynamic, and easier to scale in large projects. CSS Variables enhance the flexibility and reusability of code, and they are especially useful in modern web development involving **responsive design, themes, and dynamic UIs**.

CSS Variables:

CSS Variables are user-defined property values that can be reused throughout the CSS document. They follow a specific syntax and are defined within a CSS selector using the -- prefix.

1. Scope

CSS Variables are scoped similarly to regular CSS properties. When defined in the :root pseudo-class, they have global scope

2. Inheritance

CSS Variables follow the natural **inheritance model** of CSS. This means if a child element does not have a defined variable, it will inherit from its parent or from the root.

3. Cascading

As with all CSS properties, **cascading rules** apply to CSS Variables. A variable defined later in the code or with higher specificity will override earlier definitions.
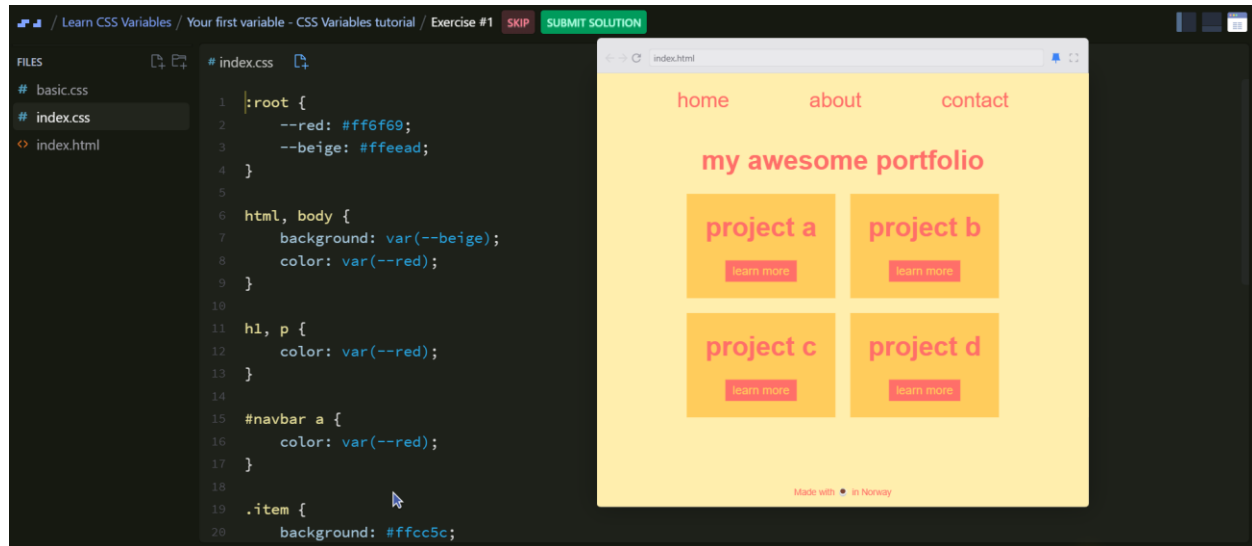
4. Fallback Values

When using var(), a fallback value can be provided if the variable is not defined:

color: var(--main-color, black);

CSS Variables with Media Queries:

Here, we first declare a new local variable named --fontsize for the .container class. We set its value to 25 pixels. Then we use it in the .container class further down. Then, we create a @media rule that says "When the browser's width is 450px or wider, change the --fontsize variable value of the .container class to 50px."

Conclusion:

CSS Variables represent a significant evolution in the way styles are written and managed in web development. Their power lies in simplicity, reusability, and dynamic behavior, especially when paired with JavaScript.