

Lesson 06 Demo 02

Demonstrating Custom Themes in Tailwind CSS

Objective: To demonstrate text font sizing, colors, and alignment options using Tailwind CSS and create and apply custom CSS classes using CSS variables (:root) while maintaining compatibility with Tailwind's utility classes

Tools required: Visual Studio Code

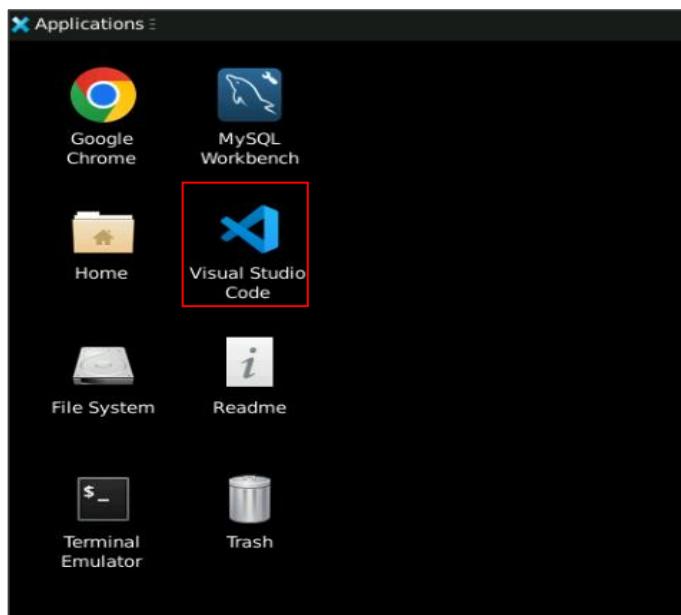
Prerequisites: HTML Basics

Steps to be followed:

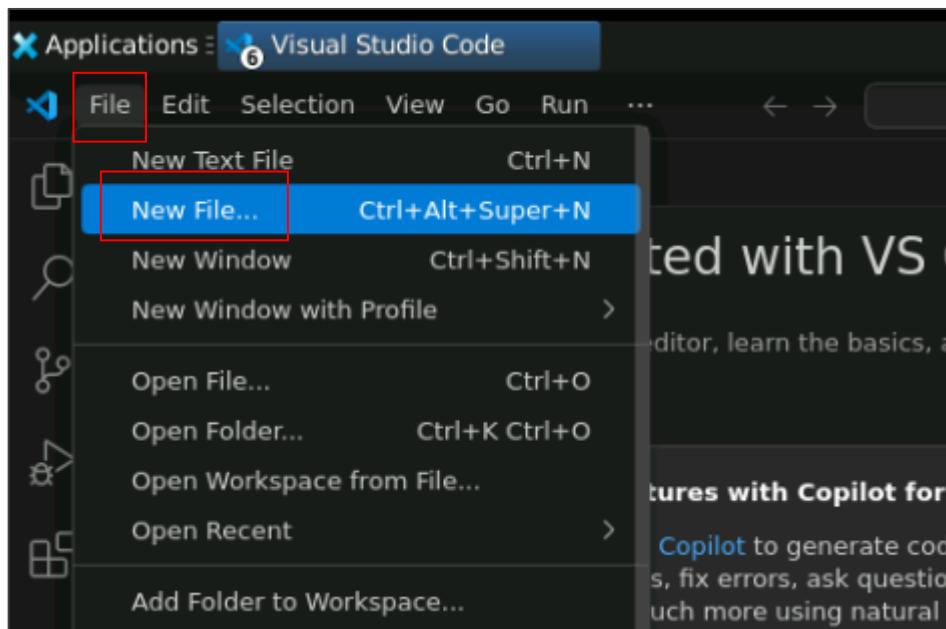
1. Open Visual Studio Code and create index.html
2. Add custom theme related code

Step 1: Open Visual Studio Code and create index.html

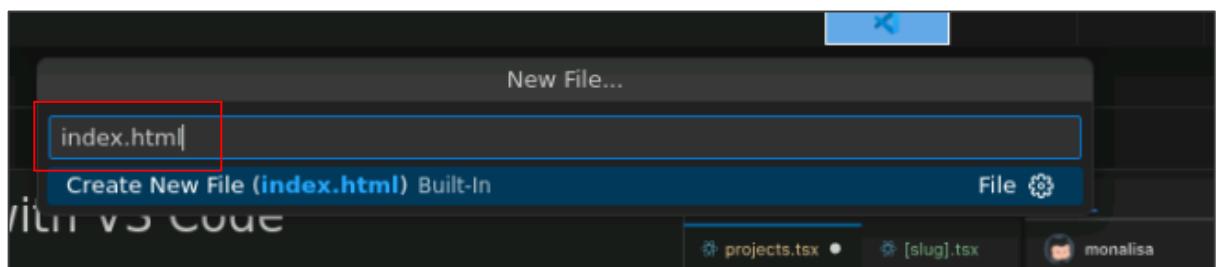
1.1 Open Visual Studio Code



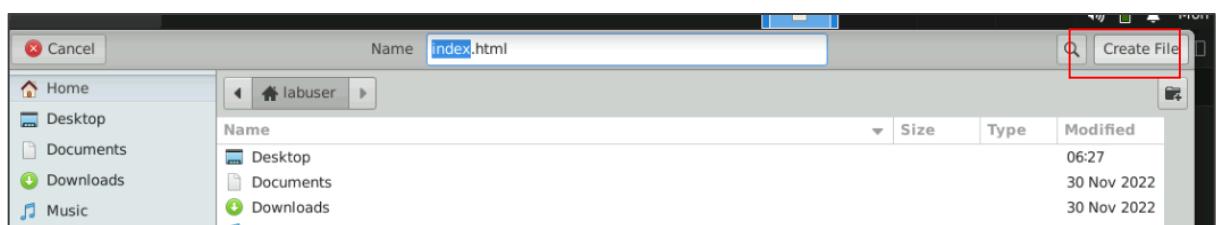
1.2 Click on **File** and then select the **New File...** option



1.3 Create an **index.html** file and press **Enter**



1.4 Click on **Create File**

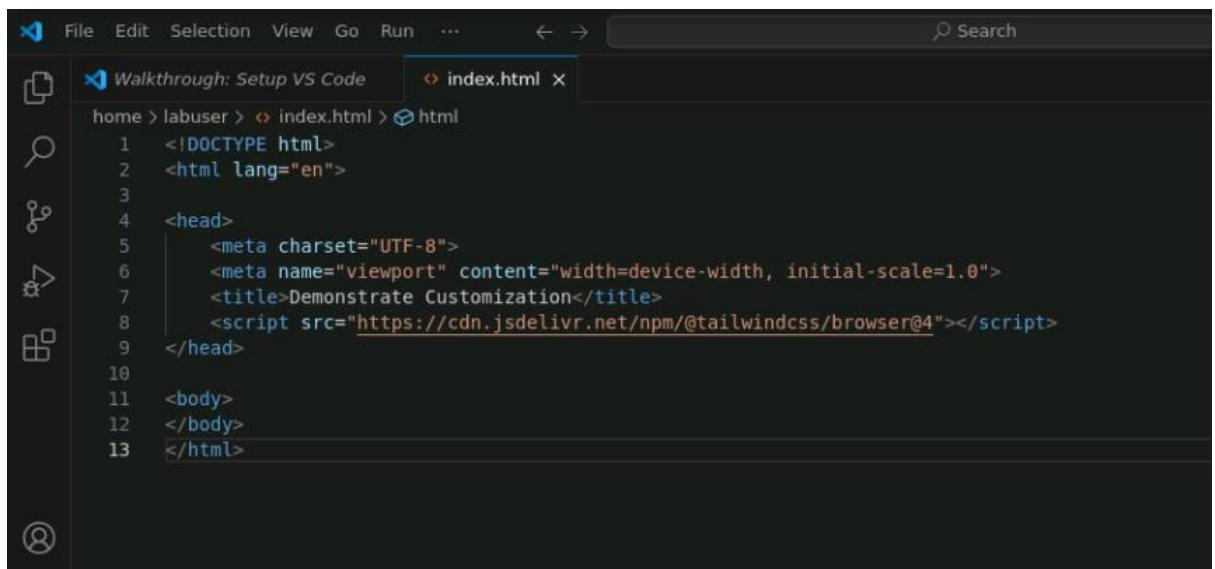


1.5 Add the following code to the index.html file to include custom font sizes and colors:

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Demonstrate Customization</title>
  <script src="https://cdn.jsdelivr.net/npm/@tailwindcss/browser@4"></script>
</head>

<body>
</body>
</html>
```



A screenshot of the Visual Studio Code interface. The title bar shows "Walkthrough: Setup VS Code" and the file path "index.html". The main editor area displays the provided HTML code. The sidebar on the left contains icons for file operations like copy, paste, search, and refresh.

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5    <meta charset="UTF-8">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>Demonstrate Customization</title>
8    <script src="https://cdn.jsdelivr.net/npm/@tailwindcss/browser@4"></script>
9  </head>
10
11 <body>
12 </body>
13 </html>
```

Step 2: Add custom theme related code

- 2.1 Modify the index.html file and add the following code to include custom font sizes and colors:

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Demonstrate Customization</title>
  <script src="https://unpkg.com/@tailwindcss/browser@4"></script>
  <style>
    :root {
      --font-xxs: 0.6rem;
      --font-xxxl: 4rem;
      --color-primary: #4CAF50;
      --color-secondary: #FF5722;
      --color-accent: #607D8B;
    }

    .text-xxs {
      font-size: var(--font-xxs);
    }

    .text-xxxl {
      font-size: var(--font-xxxl);
    }

    .text-primary {
      color: var(--color-primary);
    }

    .text-secondary {
      color: var(--color-secondary);
    }

    .text-accent {
```

```

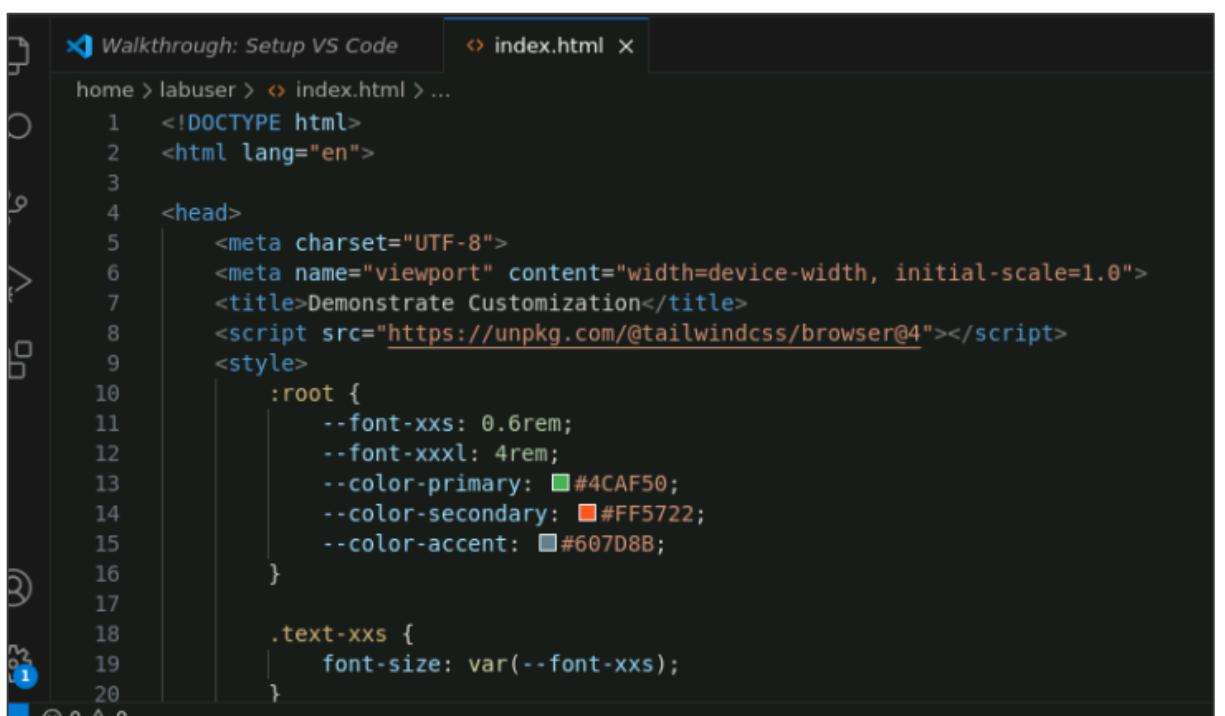
        color: var(--color-accent);
    }
</style>
</head>

<body>
<div class="container mx-auto p-4">
    <!-- Custom Font Sizes -->
    <h1 class="text-xxxl font-bold">This is custom XXXL text</h1>
    <p class="text-xxs mt-2">This is custom XXS text</p>

    <!-- Custom Colors -->
    <h2 class="text-2xl font-bold mt-5">Custom Colors</h2>
    <p class="text-primary mt-2">This text uses the custom primary color</p>
    <p class="text-secondary mt-2">This text uses the custom secondary
    color</p>
    <p class="text-accent mt-2">This text uses the custom accent color</p>
</div>
</body>

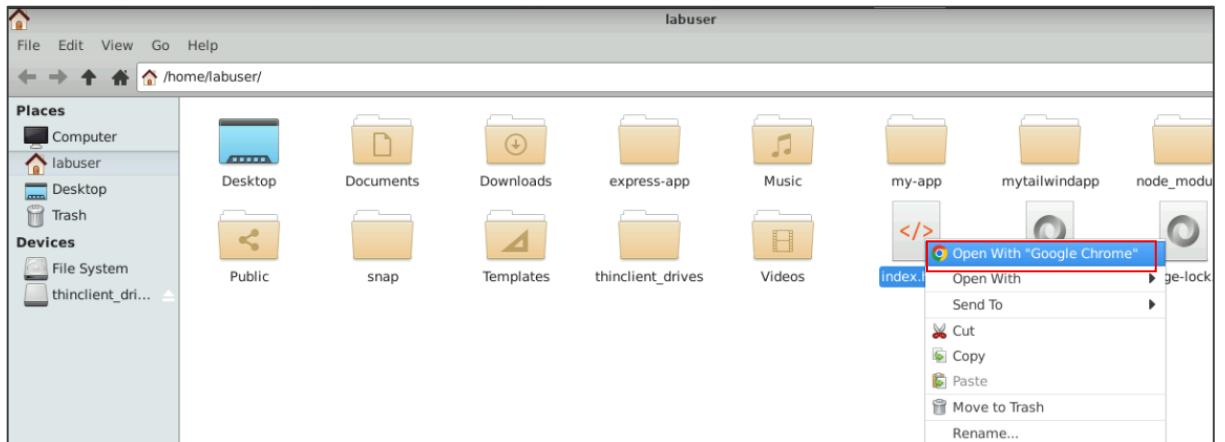
</html>

```

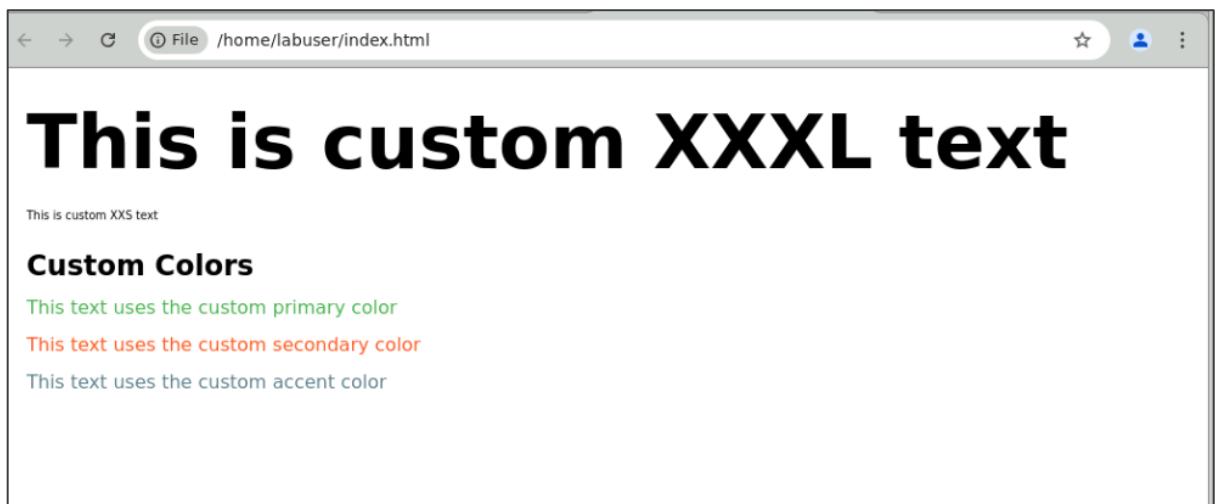


The screenshot shows the VS Code interface with the 'index.html' file open. The file content is identical to the one provided in the text block above, demonstrating Tailwind CSS customization. The code includes a head section with meta tags and a script tag pointing to Tailwind's CDN. The body section contains a large heading and several paragraphs with different text classes applied to show the customization.

2.2 Navigate to the index.html file, right-click on the file, and select **Open With “Google Chrome”**



2.3 Verify that the displayed output matches the expected appearance



By following these steps, you have successfully demonstrated how to create and apply custom themes in Tailwind CSS using CSS variables. This ensures flexibility while maintaining compatibility with Tailwind's utility classes.