**Student: Sabyrov Sanzhar  
Group: SE-2425  
Link to the GitHub:**

**Objective: Create a webpage styled with CSS (Cascading Style Sheets) to improve its design, layout, and visual appeal. You may continue with the HTML page from Assignment 1 or create a new HTML page. This assignment focuses on using basic CSS selectors and properties, requiring students to create their own class names and ID names.  
Description of Steps Taken:  
Step 1: External CSS and Structure**

* **I linked an external CSS file to the HTML file using the <link> tag in the <head> section, which helps in keeping the styling separate from the content structure.**

**Step 2: Typography (Text Styling)**

* **I applied a consistent font style for all headers (h1, h2, h3) using the Roboto font, with text alignment set to center.**
* **Paragraphs were styled with a class .paragraph-text to set the font size, line height, and justification.**

**Step 3: Colors and Backgrounds**

* **Backgrounds and text colors were applied to sections and elements on the page to create visual contrast and improve readability.**

**Step 4: Layout and Box Model**

* **I used the CSS box model (padding, margin, border) to structure the webpage and create space between elements. Tables, buttons, and other elements were styled with margins and padding to enhance visual appeal.**

**Step 5: Navigation Menu**

* **I styled the navigation links with the class .nav-link, adding hover effects that change the text color and underline the links when hovered.**

**Step 6: Buttons**

* **Buttons were styled with background color, text color, padding, and rounded corners. I added a hover effect using :hover to change the background color when the user hovers over the button.**

**Step 7: Two-Column Layout**

* **I implemented a two-column layout using a table, dividing the content into a left column (menu) and a right column (main content). The left column has a light background, while the right column holds the main content.**

**Step 8: Form Styling**

* **I styled the form elements, including input fields for name, email, and favorite color. The color picker input field was styled to appear as a small circle with a shadow effect.**

**1. External CSS:**

* **Stylesheet Link:**

**Link an external CSS file to your existing HTML pages. Ensure that your CSS file is properly linked using the <link> tag in the <head> section of the HTML document.**

* **Class and ID Creation:** 
  + - * **Create your own class names and ID names for styling. Each class and ID**

**should be descriptive and relevant to its content or layout function.**

* + - * **Avoid using generic class names like class1 or header-style; instead, use**

**meaningful names like .header-section, .highlighted-text, or #top banner.**

* **Selectors to Use:**

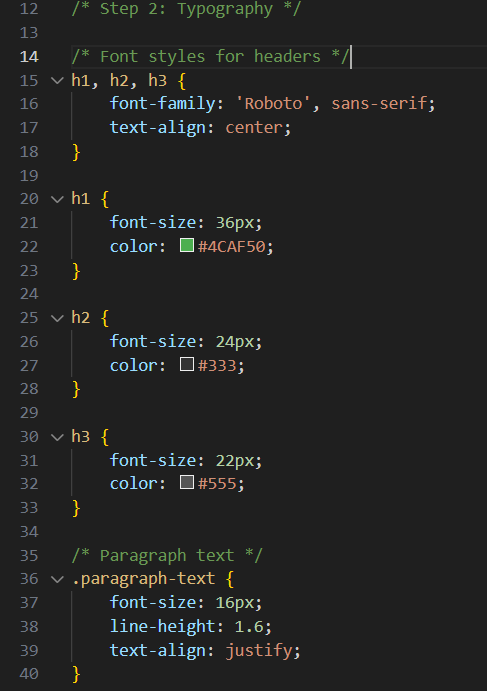
**Apply various CSS selectors:**

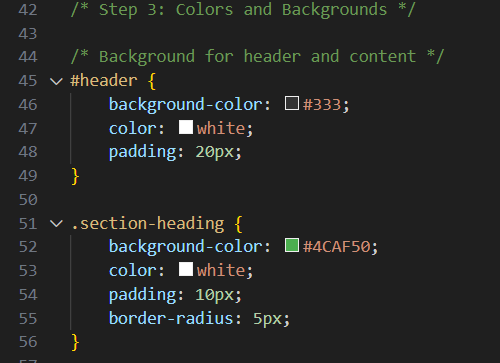
* + - * **Element selectors (e.g., h1, p) for general styling of HTML elements.**
      * **Class selectors (e.g. .button-style) to style elements that share a class.**
      * **ID selectors (e.g., #header, #main) to target specific, unique elements.**
      * **Descendant selectors (e.g., .nav a) to style elements nested within other elements.**

**  
2. Typography:**

* **Font Styling:**

**Change the font family, size, and weight across your webpage. You may use standard web fonts or import fonts using external services like Google Fonts.**

* **Text Alignment and Decoration: Align text (e.g., left, center, right) and style links (e.g., change color, remove underline).**
* **Class and ID Naming: Use appropriate class names for different text styles. For example, create .section heading for section headers and .paragraph-text for body text. 👇** **3. Colors and Backgrounds:**
* **Color Scheme: Develop a consistent color scheme for your webpage. Use CSS to define background colors, text colors, and link colors.**
* **Background Styling: Apply background colors or images to different sections of your webpage, such as headers, footers, or content sections.**
* **Class and ID Naming: Use meaningful names like .header-background, .footer-bg, or #content wrapper to organize your styles.**

****

**4. Layout and Box Model:**

* **Spacing:**

**Apply the CSS box model (padding, margin, border) to structure your webpage and**

**create space between elements. Ensure the layout is well-structured and visually**

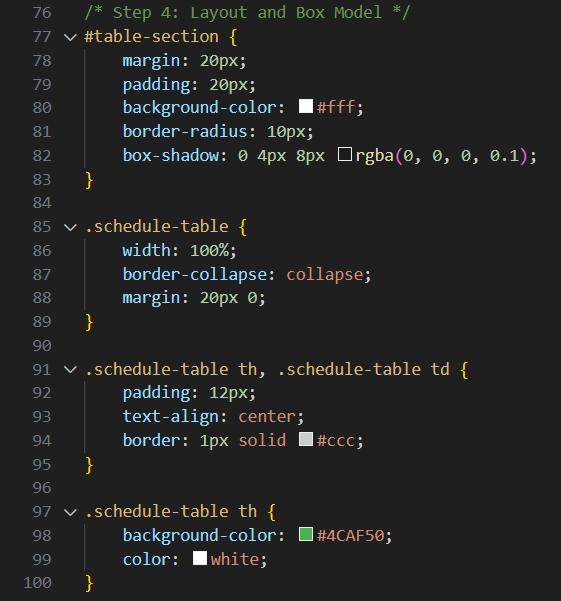
**appealing.**

* **Width and Height:**

**Set fixed or relative widths and heights for elements to control their size.**

* **Class and ID Naming:**

**For layout purposes, create meaningful names like .layout-section, .contentbox, or #footer-container. 👇**



**5. Navigation Menu:**

* **Styling Navigation:**

**Style the <nav> element using CSS. Ensure that navigation links are spaced properly, and consider adding a background to the navigation bar.**

* **Hover Effects:**

**Add hover effects to navigation links using pseudo-classes like :hover. This can**

**include changing text color or adding an underline effect when a user hovers over a**

**link.**

* **Class and ID Naming:**

**Assign names like .nav-item, .nav-link, or #navbar for the styling of your**

**navigation elements.**

**Изображение выглядит как текст, снимок экрана, Шрифт, число

Содержимое, созданное искусственным интеллектом, может быть неверным.**

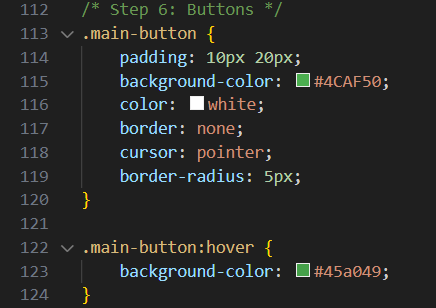
**6. Images and Media:**

* **Image Styling:**

**Set widths and heights for images to ensure they scale appropriately. Apply borders or box shadows to images for additional styling.**

* **Class and ID Naming:**

**Name your image styles logically, such as .image-thumbnail, .featured-image, or #main-image. 👇**

****

**7. Forms (Optional):**

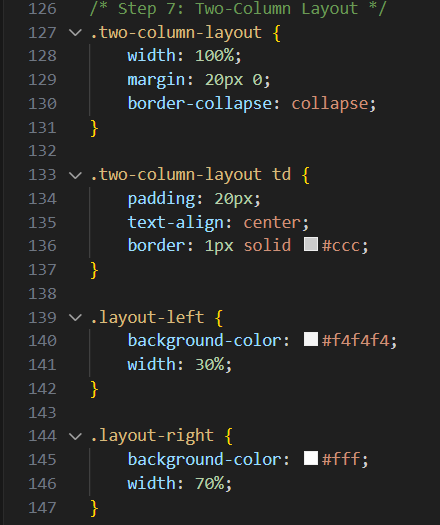
* **Styling Forms:**

**If your webpage includes forms, style them with custom inputs, buttons, and text**

**areas. Ensure forms align with your overall design.**

* **Class and ID Naming:**

**For form elements, use names like .form-input, .submit-btn, or #user-form.**

****

**8. Advanced Styling (Optional):**

* **Pseudo-elements:**

**Use pseudo-elements like ::before and ::after to add decorative content to**

**specific elements.**

* **Transitions:**

**Apply transitions to make changes smoother when a user interacts with elements,**

**such as hovering over buttons or images.**

* **Class and ID Naming:**

**Use creative class names like .btn-hover-effect, .card-shadow, or #animation**

**box for advanced styles.   
👇**

**👇**

**👇**

**👇**

**👇**

**Изображение выглядит как текст, снимок экрана, программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.** **Изображение выглядит как текст, снимок экрана, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.**  
**Conclusion:**

**In this project, I enhanced the appearance of the webpage using CSS. As part of the assignment, I:**

* **Linked an external CSS file to the HTML document to separate content and styling.**
* **Styled the text (headers, paragraphs) to ensure readability and consistency.**
* **Applied background colors, text colors, and hover effects for a better visual experience.**
* **Used the box model to create spacing between elements, making the layout visually appealing.**
* **Styled buttons and navigation links with hover effects for a more interactive user experience.**
* **Created a form with input fields, including a color picker, and styled them to match the design.**

**These changes improved the design of the webpage, making it more interactive and visually appealing for users.**