Name: Li Jiaxin

Student Number:2310245

Learning Diary

Over the past few weeks, I embarked on an exciting learning journey by attending the online course "Software Development Skills: Front-End Lobby." The course focused on three key technologies: HTML, CSS, and JavaScript. By following the video tutorials, I successfully built a website from scratch. This experience not only deepened my understanding of front-end development but also allowed me to apply the theoretical knowledge I had gained into practical web development. Here’s a detailed account of my learning process and how I utilized these technologies in building the website.

Understanding HTML (HyperText Markup Language)

I was introduced to HTML, the foundation of any website. The course started with the basics, explaining the structure of HTML documents. I learned that HTML is responsible for creating the skeleton of a webpage, using tags to define the structure and conten. I built the initial structure of my website by writing the HTML code. I used the <html>, <head>, and <body> tags to set up the basic document structure. Inside the <head>, I included meta tags, linked external stylesheets (CSS), and used <title> to define the webpage's title.

Next, I populated the webpage with various HTML elements such as headings (<h1>, <h2>), paragraphs (<p>), images (<img>), and links (<a>). Each of these elements serves a specific purpose. For example, I used <h1> for the main heading to introduce the website and <p> tags for descriptive text.

One of the more interactive parts of the website was creating forms. I learned how to use the <form> tag along with input elements like <input>, <textarea>, and <select> to allow users to input data. I created a contact form where users could enter their name, email, and message, which I later connected to JavaScript for validation.

I had a basic but structured HTML document, which was the backbone of my website. I realized how important clean and well-organized HTML code is for a webpage’s accessibility and readability.

Styling with CSS (Cascading Style Sheets)

For me this could be most difficult part. The course transitioned to CSS, which is responsible for the visual aspect of the website. CSS allowed me to transform the raw HTML elements into an aesthetically pleasing layout. I started by creating an external CSS file and linking it to my HTML document using the <link> tag in the <head>. This separation of structure (HTML) and style (CSS) is a fundamental concept in web development. I learned how to target specific HTML elements using selectors like tag selectors (h1), class selectors (.classname), and ID selectors (#idname) to apply styles.

Understanding the CSS box model was crucial in positioning and sizing elements. Each element on the webpage is a box, and I learned how to manipulate properties like margin, padding, border, and width to control the spacing around elements. For example, I set margin: 20px; to add space around each element, ensuring that the text didn’t stick to the edges of the container.

I also learned about Flexbox, a powerful layout module in CSS that made it much easier to align and distribute space among items in a container. Using Flexbox, I created a navigation bar (navbar) and centered its items both vertically and horizontally. I applied properties like display: flex; to the navigation container, followed by justify-content: center; and align-items: center; to ensure that all items were perfectly aligned. This helped me create a responsive design that adjusted smoothly across different screen sizes.

In addition to Flexbox, I explored CSS Grid, which is another powerful tool for layout design. I used Grid to create a responsive gallery section on my website, allowing me to define rows and columns with specific sizes. This enabled me to create a structured layout that automatically adjusted based on the screen size.

To improve readability and aesthetics, I spent time learning about fonts and colors. I applied Google Fonts to my website by embedding links in the HTML <head> and specified custom font families in the CSS. For instance, I used font-family: 'Roboto', sans-serif; to set the font for the body text. I also learned to use hexadecimal and RGB color values to style the text and background. For buttons and hover effects, I added transitions, which gave the website an interactive feel.

I implemented hover effects on buttons and links to enhance interactivity. This made the buttons change color smoothly when hovered over, providing users with visual feedback.

Throughout the week, I learned the importance of responsive design. I used media queries to adjust styles based on the device’s screen size.

Enhancing Interactivity with JavaScript

In the third and final week, the course introduced JavaScript (JS), which brings interactivity and dynamic behavior to websites. This was one of the most exciting parts of the course as I could now make the website respond to user actions. Basic JavaScript Syntax: I started by learning the basics of JavaScript syntax, including variables, data types, functions, and loops. I wrote my first JavaScript function to display an alert message when a button was clicked. It was a simple task but gave me insight into how JavaScript can interact with HTML elements.

This course provided me with a comprehensive understanding of front-end development, combining the power of HTML, CSS, and JavaScript to create a functional and interactive website.