

Day 13 of training

Mini Project 2: Building a Portfolio Website (Part 1 - Structure & Layout)

1. Project Goal: Creating a Modern Personal Portfolio

- Today, we began building our second mini-project: a complete, single-page personal portfolio website. This project is a significant step up in complexity and will integrate all the HTML and CSS skills we've learned so far, while also introducing some more advanced techniques. The goal for this first part was to establish the complete HTML structure and create the fundamental layout for all the major sections using modern CSS, particularly Flexbox and advanced positioning.

2. Semantic HTML Structure for a Multi-Section Page

- The entire project is built on a foundation of semantic HTML, which makes the code more readable, organized, and better for accessibility.
- **Main Wrapper and Container:** The entire page is enclosed in a `.wrapper` div, which controls the overall scrolling viewport. Inside it, a `.container` div is used for the main content sections. This container has a `max-width` and is centered with `margin: 0 auto;`, a classic technique for creating a centered, fixed-width layout on larger screens.
- **Semantic Sections:** The page is clearly divided into logical sections using semantic tags, which is great practice:
 - **`<div class="nav">`:** Serves as the header and navigation bar at the top of the page.
 - **`<div class="hero-section">`:** The main introductory "hero" area.
 - **`<div class="project-section" id="project">`:** A dedicated section to showcase projects.
 - **`<div class="skills-container" id="skills">`:** A section to display technical skills.
 - **`<div class="contactus-section" id="contact">`:** A contact form section.
 - **`<footer>`:** The footer at the very bottom of the page.

- **IDs for Navigation:** Key sections like "project," "skills," and "contact" are given id attributes. These are then linked from the navigation bar's anchor tags (e.g., ``), allowing for smooth scrolling to different parts of the single-page layout.

3. Layout with CSS Flexbox

- Flexbox is the primary tool used to structure almost every part of this project, demonstrating its power and versatility in creating complex layouts.
- **Navigation Bar (.nav):** `display: flex;` is used on the main nav container. `justify-content: space-between;` pushes the logo to the far left and the navigation links to the far right, while `align-items: center;` vertically aligns them perfectly. The navigation links themselves (.nav-items) are also in a flex container to create horizontal spacing with the `gap` property.
- **Hero Section (.hero-section):** This is another key use of Flexbox. `display: flex;`, `justify-content: space-between;`, and `align-items: center;` are used to create the two-column layout, with the text content on the left and the user image on the right.
- **Skills Section (.skills-container):** This also uses a two-column flex layout to separate the descriptive text on the left from the grid of skill icons on the right.
- **Project Cards (.project-content):** Inside each project card, `display: flex;` and `flex-direction: column;` are used to stack the project's heading, description, skills, and buttons vertically.

4. Advanced Positioning for Layered Effects

- One of the most visually interesting aspects of this project is the use of layered elements, which is achieved with the CSS `position` property.
- **Creating a Positioning Context:** The main .hero-section and the individual .project-cards are given `position: relative;`. This is a crucial step because it establishes them as a "positioning context" for any child elements that will be positioned absolutely.
- **Absolute Positioning (position: absolute;):** This is used extensively to create decorative and layered effects that are removed from the normal document flow:
 - **Faded Text:** The large, faded text in the background of the hero and footer sections is achieved by giving it `position: absolute;` and using

bottom and left properties to place it precisely. A lower z-index or its placement in the HTML ensures it sits behind other content.

- **Decorative Icons:** All the animated icons in the hero section (dots, cube, circle, zigzag) are absolutely positioned. This allows them to be placed at specific coordinates around the user's image, creating a dynamic and visually rich composition.
- **Project Numbers:** The large numbers (01, 02, etc.) on the project cards are also absolutely positioned, allowing them to be placed outside the main content area of the card for a stylish, overlapping effect.