

CSS Advanced

Session 1: Advanced Positioning and Visual Effects

1. Recap (Position)

- Brief review of CSS positioning (relative, absolute, fixed, sticky).
- Clarifying complex positioning scenarios and best practices.

2. Animation

- **Introduction to CSS animations:** keyframes, animation properties, and timing functions.
- Creating smooth transitions and effects.

3. Visibility

- Managing element visibility using `visibility` and `display` properties.
- Differences between `visibility: hidden` and `display: none`.

4. Overflow

- Controlling content overflow with `overflow`, `overflow-x`, and `overflow-y`.
- Handling scrolling and hidden content scenarios.

5. Transform

- Using the `transform` property for 2D and 3D transformations (translate, rotate, scale, skew).
- Combining transformations for complex visual effects.

Session 2: **Box Model Enhancements and Pseudo**

1. Recap (Box-shadow)

- Advanced use of box-shadow for realistic effects and depth.

2. Box-sizing

- Understanding **`box-sizing`** and its impact on layout.
- Practical applications of **`content-box`** and **`border-box`**.

3. Pseudo Elements (Before & After)

- Creating and styling pseudo-elements **`::before`** and **`::after`**.
- Practical examples and creative uses, such as decorative elements and icons.

4. Examples Using Before & After

- Hands-on projects to solidify understanding of pseudo-elements.



Session 3: Advanced Pseudo Classes and Interactivity

1. Pseudo Class:

- **Hover:** Creating interactive elements with hover effects.
- **Opacity:** Managing element opacity for visual emphasis.
- **Transition:** Smoothing transitions between states for a better user experience

2. Focus (Inputs)

- Styling input elements when focused, enhancing form usability.

3. Tooltips

- Designing tooltips that appear when hovering over elements.
- Positioning tooltips and managing their content and visibility.



Session 4: Layout Techniques and Responsive Design

1. Difference Between Flex and Float

- Comparing the usage, advantages, and limitations of flex and float layouts.

2. Flexbox:

- Mastering ``justify-content``, ``align-items``, ``flex-wrap``, and ``flex-direction``.
- Building responsive and flexible layouts with Flexbox.

3. Responsive Design (Media Queries):

- **Introduction to Media Queries:**

- Learn the basics of media queries and their role in responsive design.
- Understand how to use media queries to apply different styles based on screen size, orientation, and resolution.

- **Implementation:**

- Create responsive design adjustments for various breakpoints, such as mobile, tablet, and desktop views.
- Practice writing media queries to modify layout properties, font sizes, and element visibility for different devices.

Session 5: Grid Layout and Modern CSS Features

1. Grid:

- Implementing CSS Grid for advanced layouts.
- **Key properties:** ``justify-content``, ``align-items``, ``grid-template-column``, ``grid-template-row``.

2. Portfolio Header

- Designing a professional portfolio header using advanced CSS techniques.

3. CSS Variables & Introduction to Frameworks

- Utilizing CSS variables for consistent theming and styling.
- Introduction to the concept of CSS frameworks and their benefits.



Session 6: Final Project

1. Complete the Portfolio Project

- Bringing together all the advanced CSS techniques learned to create a complete portfolio webpage.
- Focus on layout, responsiveness, aesthetics, and interactivity.

