

Training Day 22 Report

Amrinder Singh

URN: 2302468 CRN: 2315013

14 July 2025

Topic: CSS Flexbox – Responsive Layout System

Today's session introduced Flexbox—a one-dimensional layout model that makes aligning and distributing space among elements intuitive and efficient. We used it to build responsive containers that adapt neatly across screen sizes.

Key Areas Covered

1. Flex Container Setup

- Applied `display: flex` to create a flex container.
- Discussed flex direction: `row`, `column`, `row-reverse`, and `column-reverse`.
- Enabled item wrapping using `flex-wrap: wrap`.

2. Item Alignment

- Horizontally aligned items using `justify-content`.
- Vertically aligned items using `align-items` and `align-content`.
- Explored values like `center`, `space-between`, `space-around`, and `stretch`.

3. Flexible Item Sizing

- Controlled item growth and shrink with `flex-grow`, `flex-shrink`, and `flex-basis`.
- Used shorthand: `flex: 1` for dynamic resizing.
- Ensured consistent spacing using `gap` and margins.

4. Flexbox Debugging Use Cases

- Visualized flex behavior using browser DevTools.
- Common use cases: navigation bars, card layouts, and image galleries.
- Compared Flexbox to traditional floats and inline-block methods.

Hands-On Activity

Built a responsive service section:

- Three horizontal feature cards with icons and text.
- Aligned items to center with equal spacing using `justify-content`.
- Converted layout to stacked columns for mobile view.

Key Takeaways

- Flexbox simplifies complex alignment and spacing tasks.
- It adapts layouts responsively across devices without extra media queries.
- Flex properties allow dynamic control over individual item behavior.
- This layout model is essential for modern web design workflows.