Training Day 3

Day 3 – 23rd June 2025

Topics Covered – Flexbox, JavaScript Operators & Conditions, Icon Frameworks

CSS Flexbox

We began the day learning about **Flexbox**, a modern CSS layout model that simplifies aligning and distributing space among items in a container.

Key properties introduced:

- display: flex: Enables Flexbox layout.
- justify-content: Aligns items horizontally (start, center, space-between, etc.).
- align-items: Aligns items vertically.
- flex-direction: Defines the axis (row/column) of layout.

It allows the creation of responsive layouts that adjust automatically to different screen sizes.

JavaScript Operators and If-Else Conditions

We were introduced to **JavaScript syntax basics**, focusing on operators and conditional logic.

- **Operators**: Used for calculations and comparisons.
  - Arithmetic: +, -, *, /
  - Comparison: ==, !=, ===, >, <
  - Logical: &&, ||, !
- **If-Else Conditions**: Helps control the flow of code based on conditions. Example:

```
let age = 18;
if (age >= 18) {
  console.log("Eligible to vote");
} else {
  console.log("Not eligible");
```

}

Icon Framework – Font Awesome

Font Awesome is a library of scalable vector icons used in web design. Instead of images, it uses simple HTML and CSS classes to display icons.

To use:

1. Add CDN link in HTML <head>.
2. Use classes like <i class="fa fa-user"></i> to display icons.

It's lightweight and improves visual clarity and design consistency.

Code Task

- Create a web page using **Flexbox** to align 3 boxes horizontally (or stack them vertically on small screens).
- Write a **JavaScript function** using if-else conditions to check age or marks and print the result.
- Add **Font Awesome icons** for contact and navigation (e.g., home, mail, user) using HTML.

Task for Tomorrow

- Deep dive into **Font Awesome** usage and categories (social, navigation, form icons).
- Build a mini navigation bar with 3 icons (Home, About, Contact).
- Practice combining **Flexbox layout** with **icons** for responsive design.