Day 14

Today's session focused on using JavaScript to work with HTML forms. We explored how to capture user input, validate it before processing, and provide real-time feedback using the DOM and event listeners. Input validation is a critical step in ensuring the integrity and usability of form-based applications.

Accessing Form Elements:

- Forms can be accessed using document.forms or directly by id:
- const form = document.getElementById("taskForm");
- const taskInput = document.getElementById("taskInput");

Reading Input Values:

- Used .value to read current user input:
- const task = taskInput.value;
- Added visual cues using CSS classes to indicate valid/invalid input.
- Used keyup event to provide feedback as user types:
- taskInput.addEventListener("keyup", function () {
- if (this.value.length >= 5) {
- this.classList.add("valid");
- } else {
- this.classList.remove("valid");
- }
- });

Summary:

This session emphasized best practices in front-end development by making our form more user-friendly and secure. Real-time validation gave a better UX, and learning how to stop form submission using prevent Default () gave us control over what gets submitted.