📝 **Internship Report**

Program: Frontend Development Internship

Organization: Prosensia

Intern: Hamza Rafique

Date: 21 July 2025

Week 5 - Day 1

Topic: Event Handling in JavaScript

📌 **Topics Covered:**

1. **Introduction to Events in JavaScript**

   o   What are browser events?

   o   Common types of events: click, mouseover, keydown, submit, etc.

2. **Event Listeners**

   o   addEventListener() method

   o   Removing event listeners with removeEventListener()

3. **Event Object**

   o   Accessing event metadata using the event parameter (event.target, event.type, etc.)

4. **Types of Event Handlers**

   o   Inline handlers (not recommended)

   o   DOM property handlers: element.onclick = function() {}

   o   addEventListener method: preferred and modern approach

5. **Event Bubbling and Capturing**

   o   Understanding the flow of events in the DOM tree

   o   Use of event.stopPropagation() and event.preventDefault()

6. **Practical Examples**

   o   Button click events

   o   Form submission handling

- Keyboard input tracking

❓ **Common Questions:**

- What is the difference between onclick and addEventListener?

- How can we stop a form from refreshing the page on submit?

- How to handle multiple events on the same element?

🔧 **Issues Faced:**

- Initially used inline event handlers which caused unexpected behavior.

- Event bubbling led to multiple functions being triggered unintentionally.

✅ **Resolutions:**

- Shifted to using addEventListener for cleaner and modular code.

- Used event.stopPropagation() to control bubbling.

- Used preventDefault() to handle form events without reloading.

🔍 **General Observations:**

- Event handling adds interactivity to static pages.

- Understanding the event flow is key to debugging complex interactions.

💡 **Suggested Improvements:**

- Practice more with nested DOM elements and complex event structures.

- Explore **event delegation** to handle events on dynamically added elements.

- Work on keyboard accessibility events for better user experience.