DOM and Event Listeners

Learning Objectives

- Understand what the DOM is
- Understand what the Event Listeners are
- Understand how to use Event Listeners

Material

DOM

The Document Object Model (DOM) is a programming interface which allows JavaScript running in the browser to read and manipulate the webpage it is embedded in. The DOM can be accessed by JavaScript using the `document` object.

Accessing DOM Elements

Often, we want to get a reference to a particular HTML element within our JavaScript. Two useful DOM methods for achieving this are: `document.getElementById` and `document.querySelector`:

```
    <form action="/todos" method="POST">
        <label for="task-input">Task</label>
        <input name="task" id="task-input" required>
        <label for="due-input">Due</label>
        <input name="due" id="submit" type="date">
        <button type="submit">Submit</button>
    </form>
    <script>
        const taskInput = document.getElementById("task-input");
        const dateInput = document.querySelector("input[type='date']");
    </script>
```

Once we have access to a DOM element, we can read its values or manipulate it. For example, `taskInput.value` would give us the current text in the task input box.

Event Listeners

Event listeners can be used to run JavaScript functions in response to user interactions with the web-page. Event listeners are added to a particular DOM element and subscribe to a particular event type (e.g. “click”, “keydown” etc.).

```
element.addEventListener(event, function);
```

When the event occurs, the event handler callback is run.

```
<label>
  Count: <span id="count">0</span>
</label>
<button>Increment</button>
```

```
const countEl = document.getElementById("count");
const button = document.querySelector("button");
button.addEventListener("click", () => {
  let currentCount = parseInt(countEl.innerText);
  currentCount++;
  countEl.innerText = currentCount;
});
```

The example above increments the count value whenever a user clicks the “Increment” button.

Core Assignments

Calculator

Create a calculator using HTML and JavaScript. The calculator should have two number inputs and an add button. When the user clicks the add button, the calculator should sum the numbers in the two inputs and display the answer to the user.

Extension Assignments

Calculator++

Extend your calculator so users can choose to add, subtract, divide or multiply. You may find an HTML `<select>` element useful.

Additional Resources

- MDN: Introduction to the DOM
- W3schools DOM
- W3schools Event Listeners