Background Story

As Odessa's reputation for crafting interactive web experiences grows, her classmates turn to her for a special project—a feedback form for their class activities. This task pushes Odessa to work with user input, forms, and real-time validation, an essential part of building user-friendly software.

Creating an online feedback form requires Odessa to capture user input effectively and ensure that the data submitted is valid. Through this project, she experiences firsthand the importance of creating a seamless user experience and handling user interactions gracefully.

With her classmates eagerly waiting for the online feedback form, Odessa dives into the world of inputs, forms, submit events, and accessing input values using JavaScript. This project feels like a significant step towards building software that serves a real purpose and solves real-world problems.

Theory & Lecture Content

When working with interactive forms in web development, understanding how to capture user input and handle form submissions is crucial. Here are some key topics Odessa will explore:

Input Elements

Input elements in HTML allow users to enter data that can be submitted through a form. Various types like text, number, email, etc., cater to different data requirements.

Forms

HTML forms provide a way to collect user input. They consist of input fields, submit buttons, and can be styled and validated using CSS and JavaScript.

Submit Events

The `submit` event is triggered when a form is submitted, either by clicking a submit button or pressing Enter within an input field. This event allows us to validate user input and perform actions before data is sent.

.value Property

The `.value` property of an input element allows us to access the current value entered by the user. This value can be read or updated using JavaScript.

Exercises

Exercise 1

Problem Statement

Create a form with an input field for the user's name and a submit button. When the form is submitted, display an alert with the message "Feedback submitted successfully!".

Starter Code (HTML)

```html
<form id="feedbackForm">
  <input type="text" id="nameInput" placeholder="Enter your name" />
  <button type="submit">Submit</button>
</form>
```

Full Solution Code (JavaScript)

```javascript
const feedbackForm = document.querySelector("#feedbackForm");

feedbackForm.addEventListener("submit", function (event) {
  event.preventDefault();
  alert("Feedback submitted successfully!");
});
```

Exercise 2

Problem Statement

Create real-time validation for an email input field. Display an error message if the entered email is invalid (does not contain '@').

Starter Code (JavaScript)

```javascript
const emailInput = document.querySelector("#emailInput");

emailInput.addEventListener("input", function () {
  const email = emailInput.value;
  if (!email.includes("@")) {
    emailInput.setCustomValidity("Invalid email address");
  } else {
    emailInput.setCustomValidity("");
  }
});
```

Exercise 3

Problem Statement

Capture the user's feedback from a textarea input field when the form is submitted. Log the feedback message to the console.

Starter Code (HTML)

```html
<form id="feedbackForm">
  <textarea id="feedbackInput" placeholder="Enter your feedback"></textarea>
  <button type="submit">Submit</button>
</form>
```

Full Solution Code (JavaScript)

```javascript
const feedbackForm = document.querySelector("#feedbackForm");
const feedbackInput = document.querySelector("#feedbackInput");

feedbackForm.addEventListener("submit", function (event) {
  event.preventDefault();
  const feedbackMessage = feedbackInput.value;
  console.log("Feedback:", feedbackMessage);
});
```

Test Cases

Due to the interactive nature of form submission and real-time validation, manual testing is recommended for these exercises.

Closing Story

As Odessa finishes the online feedback form for her classmates, she takes a moment to reflect on how far she has come in her web development journey. Capturing user input, designing interactive forms, and handling submit events have become second nature to her.

The satisfaction of building software that serves a practical purpose fills Odessa with pride and excitement for the possibilities ahead. With each project like the feedback form, she solidifies her skills and grows closer to her dream of becoming a full-stack developer and startup founder.

Join Odessa in the upcoming lessons as she delves deeper into advanced topics like client-side validation, AJAX form submissions, and responsive design, further shaping her path towards mastery in web development.