Events

HTML Events

HTML events can be triggered by various actions performed by the user or the browser. Here are some common examples:

- **Page Load**: The page has fully loaded.
- **Input Change**: An input field value is changed.
- **Button Click**: A button is clicked.
- **Mouse Events**: Mouse moves over or out of an element.
- **Keyboard Events**: A key is pressed or released.
- **Form Submission**: A form is submitted.

Handling Events with JavaScript

JavaScript provides several ways to handle these events:

1. Inline Event Handlers

You can add event handlers directly to HTML elements using attributes.

Example with single quotes:

```
<button onclick='alert("Button clicked!")'>Click Me</button>
```

Example with double quotes:

```
<button onclick="alert('Button clicked!')">Click Me</button>
```

2. Using the `this` Keyword

You can use `this` to refer to the HTML element that triggered the event.

Example:

```
<button onclick="this.innerHTML = Date()">The time is?</button>
```

3. Calling Functions from Event Handlers

It's more common to call a JavaScript function from an event handler attribute.

Example:

```
<button onclick="displayDate()">The time is?</button>

<script>
function displayDate() {
  document.getElementById("demo").innerHTML = new Date();
}
</script>
```

Common HTML Events

Here are some commonly used HTML events:

- **onchange**: An element has been changed.
- **onclick**: An element is clicked.
- **onmouseover**: The mouse pointer moves over an element.
- **onmouseout**: The mouse pointer moves out of an element.
- **onkeydown**: A key is pressed down.
- **onload**: The page has finished loading.

Event Handling in JavaScript

In addition to using HTML attributes, you can also define event handlers in JavaScript:

1. Adding Event Listeners

Using the `addEventListener` method, you can add multiple event listeners to an element.

Example:

```
<button id="myBtn">Click Me</button>

<script>
document.getElementById("myBtn").addEventListener("click", function() {
  alert("Button was clicked!");
});
</script>
```

2. Inline Event Handling

You can directly assign event handler functions to HTML element properties.

Example:

```
<button id="myBtn">Click Me</button>

<script>
document.getElementById("myBtn").onclick = function() {
  alert("Button was clicked!");
};
</script>
```