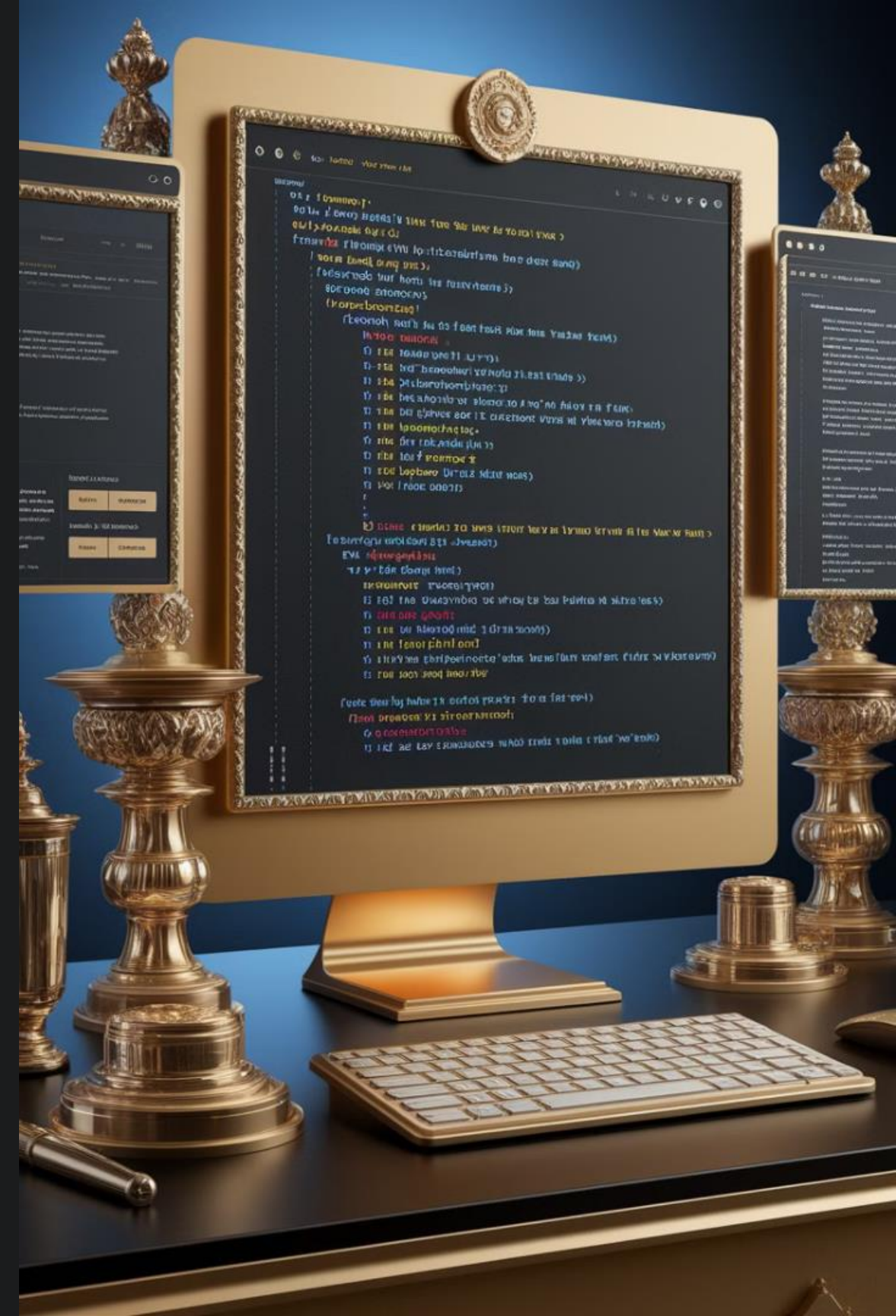




JavaScript and Forms & Browser Interaction

by : Asst. Lect. Ali Al-khawaja



Working with Forms in JavaScript

Introduction to Forms

Forms are used to collect user input on web pages.

JavaScript can read, validate, and process form data dynamically.

Example of a Simple Form:

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

Selecting Form Elements

Access Form Elements

Use JavaScript to access form elements using `getElementById()` or `querySelector()`.

Example:

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

<script>
let nameInput = document.getElementById("name");
alert(nameInput.value);
console.log(nameInput.value); // Get user input
</script>
```

Handling Form Submission

Add Event Listener

Use event listeners to handle form submission without reloading the page.

Prevent Default Behavior

Use `event.preventDefault()` to stop the form from submitting normally.

Process Form Data

Access input values and perform actions with the data.

```
<form id="myForm">
  <input type="text" id="name" placeholder="Enter your name">
  <button type="submit">Submit</button>
</form>
document.getElementById("myForm").addEventListener("submit", function(event) {
  event.preventDefault(); // Prevent page reload
  let name = document.getElementById("name").value;
  alert("Hello, " + name + "!");
});
```

Validating User Input

1

Check User Input

Use JavaScript to check user input before submitting the form.

2

Display Error Messages

Alert users about invalid inputs.

3

Prevent Submission

Stop the form from submitting if validation fails.

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

```
<script>
document.getElementById("myForm").addEventListener("submit", function(event) {
  let name = document.getElementById("name").value;
  if (name === "") {
    alert("Name cannot be empty!");
    event.preventDefault();
  }
});
</script>
```

Interacting with the Browser

1 Browser Interaction with JavaScript

JavaScript can control browser behavior, such as:

2 Opening new windows/tabs

Create new browser windows or tabs programmatically.

3 Displaying alerts & confirmations

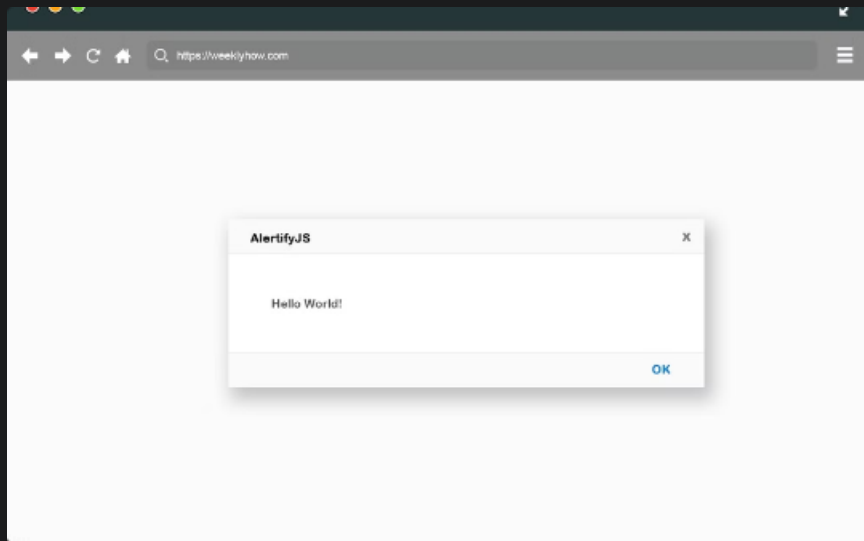
Show dialog boxes to interact with users.

4 Storing data locally

Save information in the browser for later use.

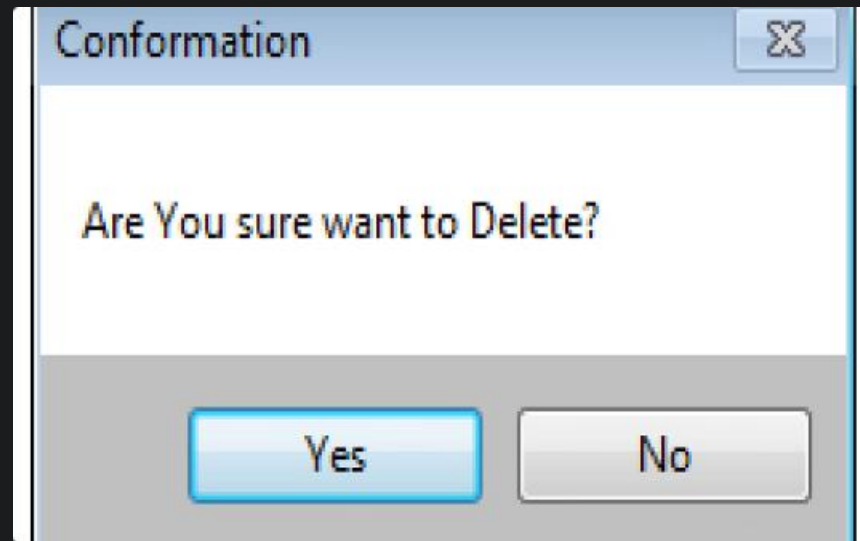


Alert, Confirm, and Prompt



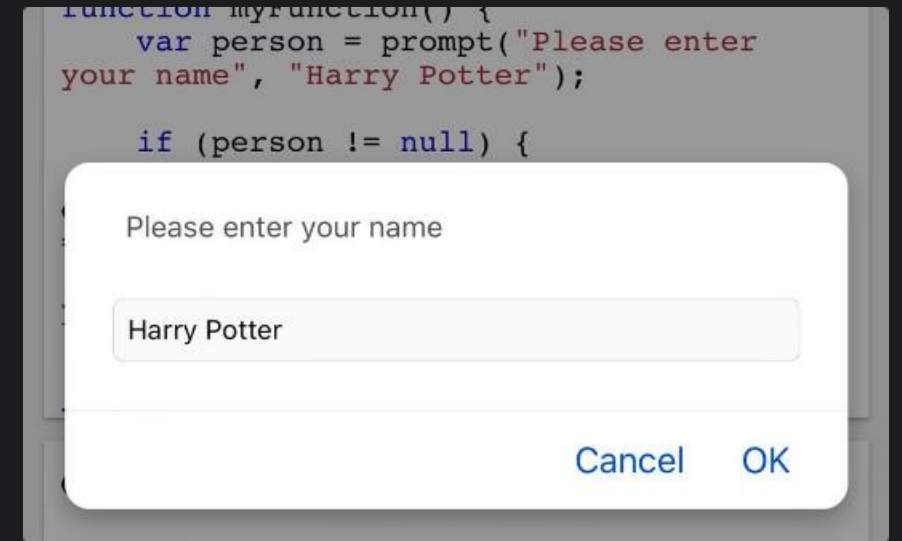
Alert

Simple message: `alert("Welcome to the site!");`



Confirm

Boolean response: `let confirmAction = confirm("Do you want to continue?"); console.log(confirmAction); // true or false`



Prompt

Text input: `let userName = prompt("Enter your name:"); console.log(userName);`

Example 1

```
<h2>Item List</h2>
<ul>
  <li id="item1">Item 1 <button onclick="deleteItem('item1')">Delete</button></li>
  <li id="item2">Item 2 <button onclick="deleteItem('item2')">Delete</button></li>
  <li id="item3">Item 3 <button onclick="deleteItem('item3')">Delete</button></li>
</ul>

<script>
  function deleteItem(itemId) {
    let confirmation = confirm("Are you sure you want to delete this item?");

    if (confirmation) {
      let item = document.getElementById(itemId);
      if (item) {
        item.remove(); // Remove the item from the page
        alert("Item deleted successfully!");
      }
    } else {
      alert("Deletion canceled!");
    }
  }
</script>
```

Example 2

```
<body>
  <button onclick="askName()">Click me</button>

  <script>
    function askName() {
      let userName = prompt("What is your name?");
      if (userName !== null) {
        alert("Hello, " + userName + "!");
      } else {
        alert("User canceled the input.");
      }
    }
  </script>
</body>
```

Working with localStorage and sessionStorage

localStorage

Saves data permanently (until manually cleared).

```
localStorage.setItem("username", "John Doe");  
console.log(localStorage.getItem("username")); // Output: John Doe
```

sessionStorage

Saves data for the session (deleted when the tab is closed).

```
sessionStorage.setItem("sessionData", sessionInfo);
```

Removing Data

```
localStorage.removeItem("username");
```

Example 3

```
<h1>Using localStorage and sessionStorage</h1>

<!-- Buttons to demonstrate localStorage and sessionStorage -->

<button onclick="saveToLocalStorage()">Save to localStorage</button>

<button onclick="saveToSessionStorage()">Save to sessionStorage</button>

<button onclick="displayStoredData()">Display Stored Data</button>

<button onclick="clearLocalStorage()">Clear localStorage</button>

<button onclick="clearSessionStorage()">Clear sessionStorage</button>
```

Example 3

```
<script>
    // Save data to localStorage
    function saveToLocalStorage() {
        let name = prompt("Enter your name to save in localStorage:");
        if (name !== null && name !== "") {
            localStorage.setItem("userName", name);
            alert("Name saved to localStorage!");
        } else {
            alert("Name cannot be empty.");
        }
    }

    // Save data to sessionStorage
    function saveToSessionStorage() {
        let sessionInfo = prompt("Enter some data to save in sessionStorage:");
        if (sessionInfo !== null && sessionInfo !== "") {
            sessionStorage.setItem("sessionData", sessionInfo);
            alert("Data saved to sessionStorage!");
        } else {
            alert("Data cannot be empty.");
        }
    }
}
```

Example 3

```
// Display stored data from localStorage and sessionStorage

function displayStoredData() {
    let storedName = localStorage.getItem("userName");
    let storedSessionInfo = sessionStorage.getItem("sessionData");

    if (storedName) {
        alert("Name from localStorage: " + storedName);
    } else {
        alert("No name stored in localStorage.");
    }

    if (storedSessionInfo) {
        alert("Data from sessionStorage: " + storedSessionInfo);
    } else {
        alert("No data stored in sessionStorage.");
    }
}
```

Example 3

```
// Clear data from localStorage

function clearLocalStorage() {
    localStorage.removeItem("userName");
    alert("Data removed from localStorage.");
}

// Clear data from sessionStorage

function clearSessionStorage() {
    sessionStorage.removeItem("sessionData");
    alert("Data removed from sessionStorage.");
}
</script>
```

Redirecting Users with JavaScript

Use `window.location` to change pages dynamically.

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

<script>
document.getElementById("myForm").addEventListener("submit", function(event)
{
  event.preventDefault();
  window.location.href = "www.Google.com";
});
</script>
```

Exercise 1 – Validate Email Before Submission

```
<form id="emailForm">
  <input type="email" id="email" placeholder="Enter email">
  <button type="submit">Submit</button>
</form>

<script>
document.getElementById("emailForm").addEventListener("submit", function(event) {
  let email = document.getElementById("email").value;
  if (!email.includes("@")) {
    alert("Invalid email!");
    event.preventDefault();
  }
});
</script>
```

Exercise 2 – Save Username in Local Storage

```
<input type="text" id="username" placeholder="Enter your name">
<button onclick="saveName()">Save</button>
<p id="display"></p>

<script>
function saveName() {
    let name = document.getElementById("username").value;
    localStorage.setItem("user", name);
    document.getElementById("display").textContent = "Saved: " + name;
}

window.onload = function() {
    let savedName = localStorage.getItem("user");
    if (savedName) {
        document.getElementById("display").textContent = "Welcome back, " + savedName;
    }
};
</script>
```

Think you..

Any Questions ??