JavaScript

JavaScript in Browsers: a Very Brief Intro

A Simple Example of Embedded JavaScript

```html
<p>
    <script>
        var today = new Date();
        document.write("Current date: ");
        document.write(today.toDateString());
    </script>
</p>
<p>
    <script>
        var today = new Date();
        document.write("&copy; ");
        document.write(today.getFullYear());
        document.write(", San Joaquin Valley Town Hall")
    </script>
</p>
```

The
Output

Current date: Thu Jan 08 2015

© 2015, San Joaquin Valley Town Hall

Some JavaScript Embedded in an HTML Page

```
<head>
    ...
    <script>
        var $ = function (id) {
            return document.getElementById(id);
        }
        window.onload = function() {
            var today = new Date();
            $("date").firstChild.nodeValue =
                "Current date: " + today.toDateString();
        }
    </script>
</head>
```
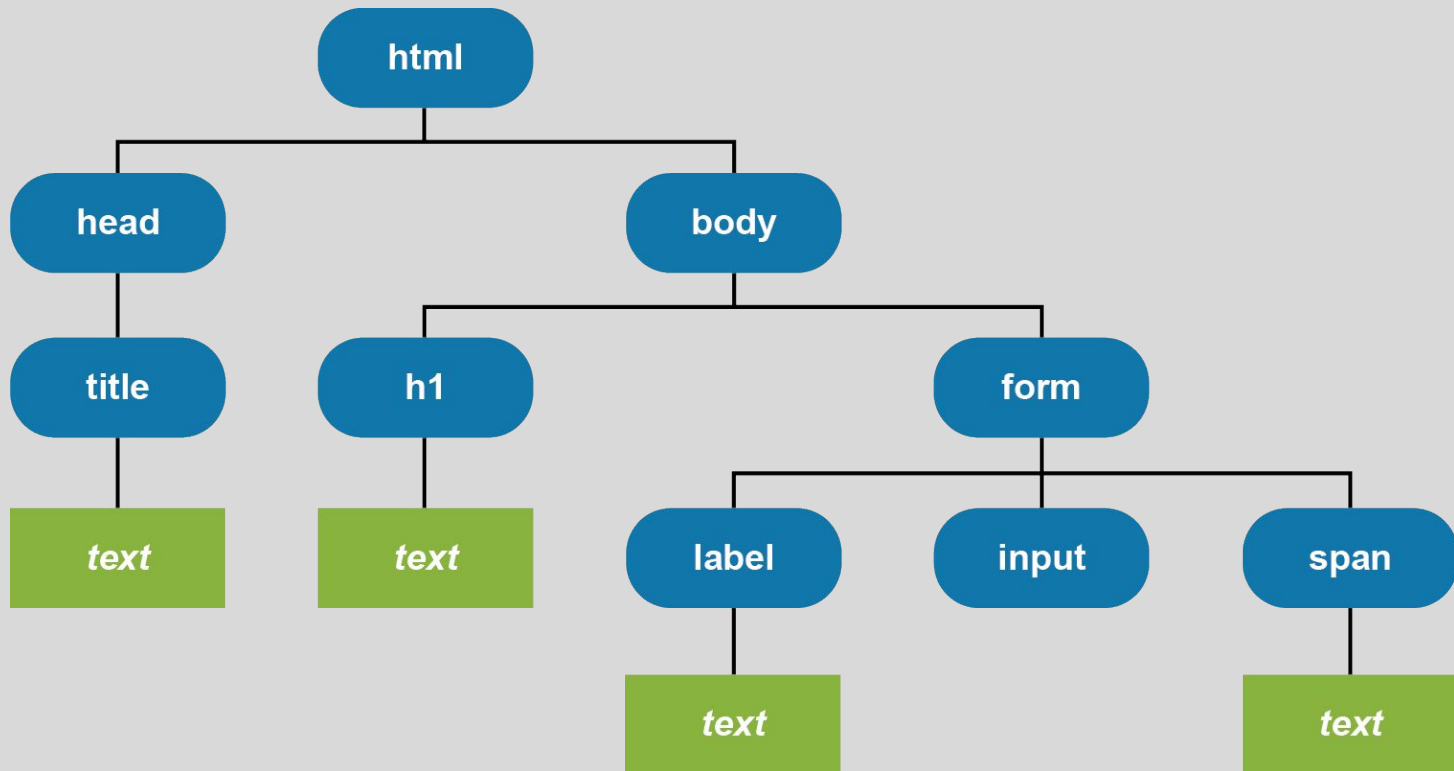
JavaScript can also be embedded in the body

Using an External JavaScript File

```
<script src="set_date.js"></script>
```

JavaScript in a Full Web Page Using a Form

```html
<!DOCTYPE html>
<html>
<head>
    <title>Join Email List</title>
</head>
<body>
    <h1>Please join our email list</h1>
    <form id="email_form" name="email_form"
          action="join.html" method="get">
        <label for="email_address">Email Address:</label>
        <input type="text" id="email_address">
        <span id="email_error">*</span>
    </form>
</body>
</html>
```

The DOM for the web page

DOM nodes commonly used in DOM scripting

- **Element**
- **Attr**
- **Text**

Two methods of the document object

- **getElementById(**id**)**
- **write(**string**)**

Examples of document methods

```
// returns the object for the HTML element
var rateBox = document.getElementById("rate");

// writes a string into the document
document.write("Today is " + today.toDateString());
```

Three properties for scripting the DOM

- **value**
- **firstChild**
- **nodeValue**

A page that uses a JavaScript button for printing

Print the Page

Murach's JavaScript and jQuery

Section 1	A subset of JavaScript for jQuery programmers	
Chapter 1	Introduction to web development	3
Chapter 2	A starting subset of JavaScript	43
Chapter 3	How to work with objects, functions, and events	89

The JavaScript in an external file named printPage.js

```javascript
var $ = function (id) {
    // this function returns the object for the HTML element
    return document.getElementById(id);
}
var printPage = function() {
    // the event handler for the click event of the button
    window.print();
}
window.onload = function() {
    // the event handler for the onload event
    $("printButton").onclick = printPage;
}
```

HTML that includes the file in the head section

```html
<script src="printPage.js"></script>
```

HTML for the button in the body section

```html
<input type="button" id="printButton" value="Print the Page">
```

Event Handlers

JavaScript can respond to events: pieces of code that responds to events are called *event handlers*

Common events: loading a web page, clicking a button or html element

```html
<!DOCTYPE html>
<html>
<body>

<h1 onclick="changeText(this)">Click on this text!</h1>

<script>
function changeText(id) {
    id.innerHTML = "Ooops!";
}
</script>

</body>
</html>
```
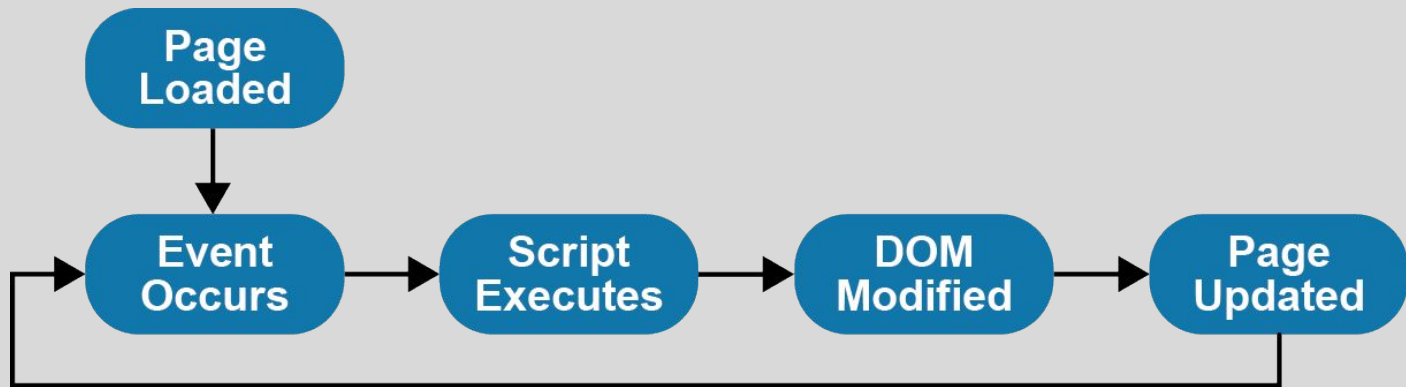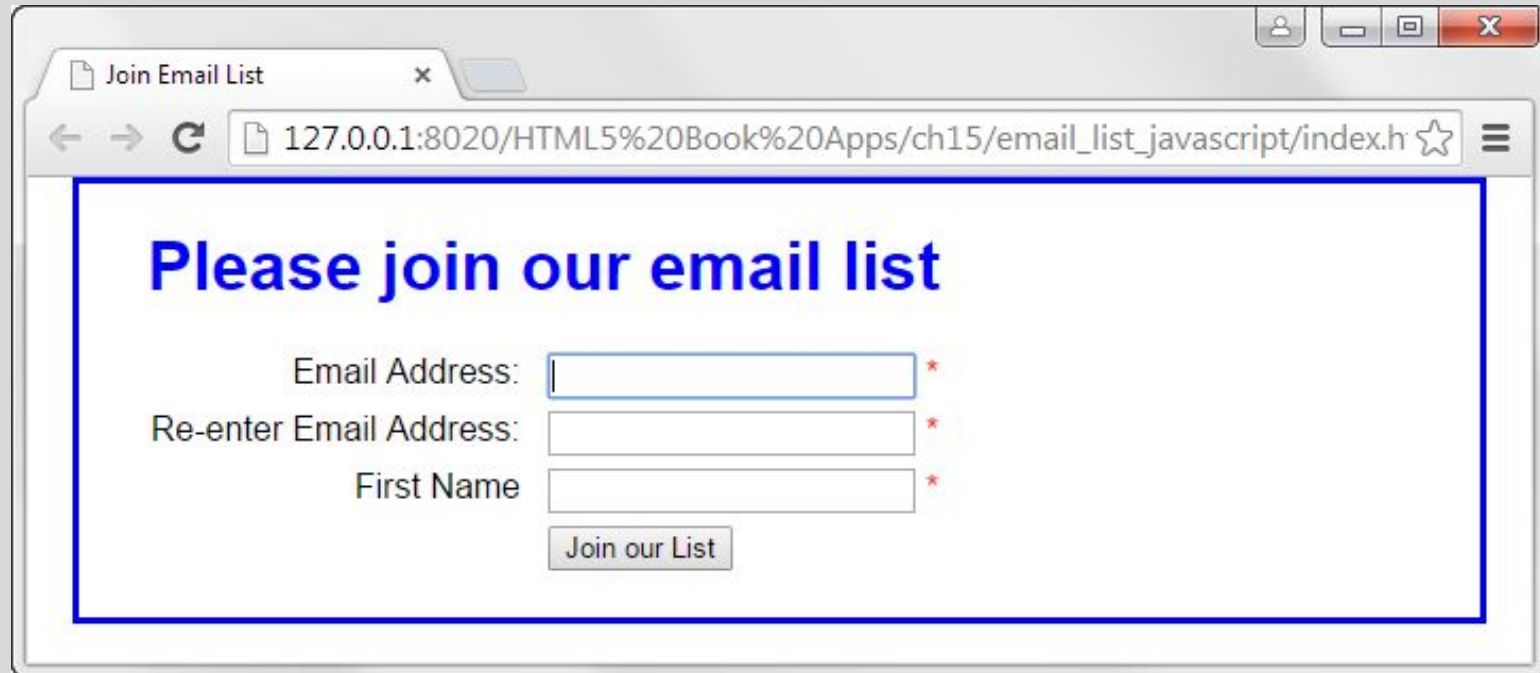
The Event Cycle

Code Example: JavaScript Error Checking with a Form

The code for the HTML file

```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Join Email List</title>
    <link rel="stylesheet" href="email_list.css">
    <script src="email_list.js"></script>
</head>
<body>
    <main>
        <h1>Please join our email list</h1>
        <form id="email_form" name="email_form"
            action="join.html" method="get">
            <label for="email_address1">
                Email Address:</label>
            <input type="text" id="email_address1"
                name="email_address1">
            <span id="email_address1_error">*</span><br>
```

The code for the HTML file (cont.)

```html
            <label for="email_address2">
                Re-enter Email Address:</label>
            <input type="text" id="email_address2"
                name="email_address2">
            <span id="email_address2_error">*</span><br>

            <label for="first_name">First Name</label>
            <input type="text" id="first_name"
                name="first_name">
            <span id="first_name_error">*</span><br>

            <label> </label>
            <input type="button" id="join_list"
                value="Join our List">
        </form>
    </main>
</body>
</html>
```
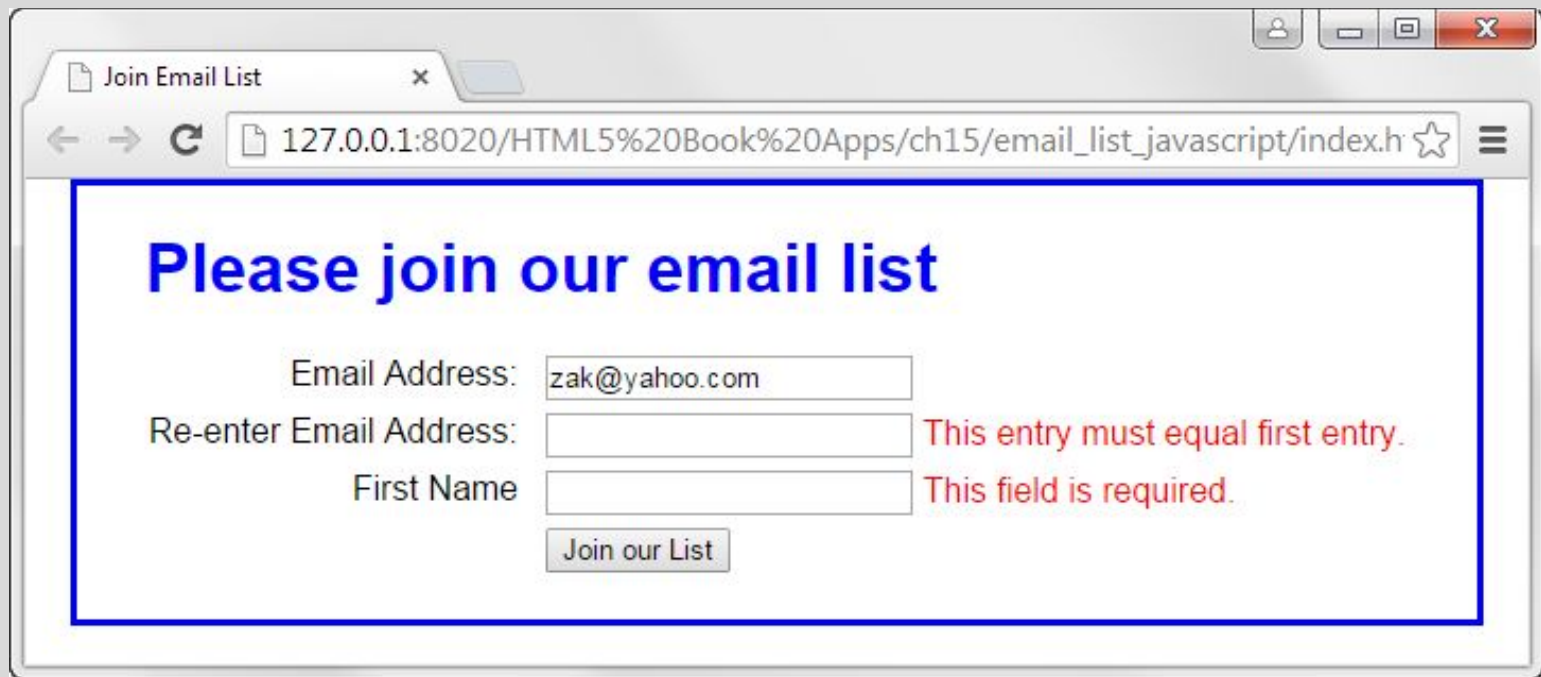
What Happens When No Input Is Entered

We'll use JavaScript to implement this behavior

The JavaScript Code for the Email Form

```javascript
var $ = function (id) {
    return document.getElementById(id);
}
var joinList = function () {
    var emailAddress1 = $("email_address1").value;
    var emailAddress2 = $("email_address2").value;
    var isValid = true;

    if (emailAddress1 == "") {
        $("email_address1_error").firstChild.nodeValue =
            "This field is required.";
        isValid = false;
    }
    else { $("email_address1_error").firstChild.nodeValue = ""; }
```

```javascript
    if (emailAddress1 !== emailAddress2) {
        $("email_address2_error").firstChild.nodeValue =
            "This entry must equal first entry.";
        isValid = false;
    }
    else { $("email_address2_error").firstChild.nodeValue = ""; }

    if ($("first_name").value == "") {
        $("first_name_error").firstChild.nodeValue =
            "This field is required.";
        isValid = false;
    }
    else { $("first_name_error").firstChild.nodeValue = ""; }

    if (isValid) {
        // submit the form if all entries are valid
        $("email_form").submit();
    }
}
window.onload = function () {
    $("join_list").onclick = joinList;
}
```