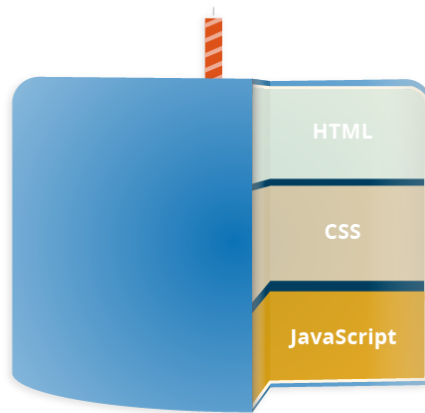
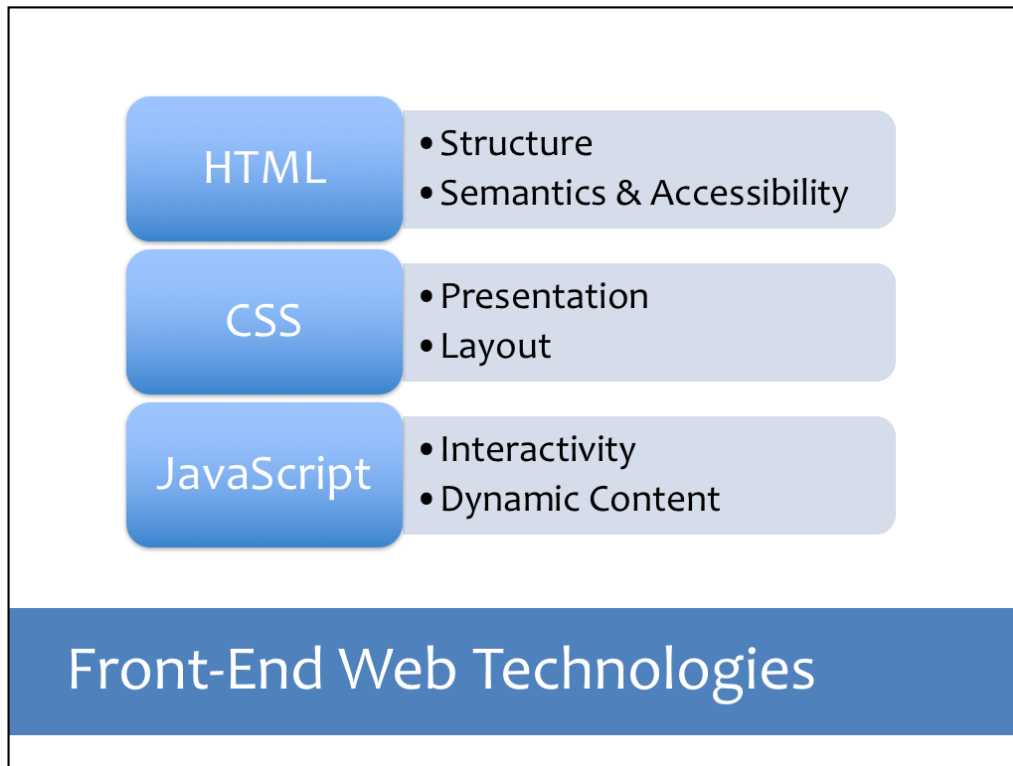


# Javascript and the DOM

Adding Interactivity to Web Pages

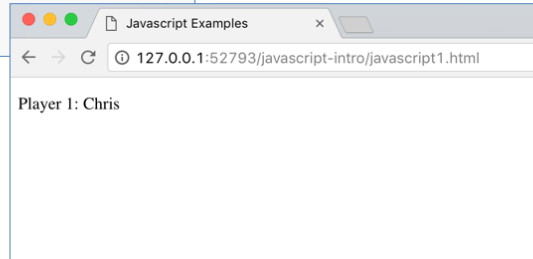


Web Front-End Technologies:  
Structure + Presentation + Interaction



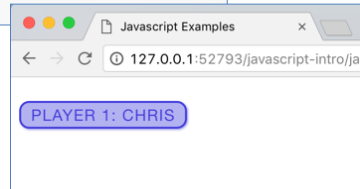
This class is mostly about front-end web technology, although we will cover some basic server-side functionality at the end of the semester.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Javascript Examples</title>
6 </head>
7 <body>
8   <p>Player 1: Chris</p>
9 </body>
10 </html>
```



## HTML (Structure)

```
<style type="text/css">
  p {
    font-family: 'helvetica neue', helvetica, sans-serif;
    letter-spacing: 1px;
    text-transform: uppercase;
    text-align: center;
    border: 2px solid rgba(0, 0, 200, 0.6);
    background: rgba(0, 0, 200, 0.3);
    color: rgba(0, 0, 200, 0.6);
    box-shadow: 1px 1px 2px rgba(0, 0, 200, 0.4);
    border-radius: 10px;
    padding: 3px 10px;
    display: inline-block;
    cursor: pointer;
  }
</style>
```



## HTML + CSS (Presentation)

```
<body>
  <p>Player 1: Chris</p>

  <script>
    var para = document.querySelector('p');

    para.addEventListener('click', updateName);

    function updateName() {
      var name = prompt('Enter a new name');
      para.textContent = 'Player 1: ' + name;
    }
  </script>
</body>
```

PLAYER 1: LIZ

## HTML + CSS + JS (Interaction)

## BASIC JAVASCRIPT CONCEPTS

I don't expect you to know Javascript in this class. I do expect you to know basic programming concepts and terms, like variables, objects, expressions, arrays, and functions. I'm going to do a quick overview of concepts and terms, but you're going to need to do the short tutorials I've assigned for Thursday in order to really understand these.

```
<body>
  <p>Player 1: Chris</p>

  <script>
    var para = document.querySelector('p');

    para.addEventListener('click', updateName);

    function updateName() {
      var name = prompt('Enter a new name');
      para.textContent = 'Player 1: ' + name;
    }
  </script>
</body>
```

## JS Variables & DOM Addressing

Defining a variable by selecting a component of the document. (We'll be talking more about those components, which are part of the Document Object Model)



```
<body>
  <p>Player 1: Chris</p>

  <script>
    var para = document.querySelector('p');
    para.addEventListener('click', updateName);

    function updateName() {
      var name = prompt('Enter a new name');
      para.textContent = 'Player 1: ' + name;
    }
  </script>
</body>
```

## JS Events and Event Listeners

Defining a method for the document to “listen” for user input and act on it.

```
<body>
  <p>Player 1: Chris</p>

  <script>
    var para = document.querySelector('p');

    para.addEventListener('click', updateName);

    function updateName() {
      var name = prompt('Enter a new name');
      para.textContent = 'Player 1: ' + name;
    }
  </script>
</body>
```

## JS Functions

Creating a function

```
<body>
  <p>Player 1: Chris</p>

  <script>
    var para = document.querySelector('p');

    para.addEventListener('click', updateName);

    function updateName() {
      var name = prompt('Enter a new name');
      para.textContent = 'Player 1: ' + name;
    }
  </script>
</body>
```

## JS Expressions

Defining a variable by concatenating variables and strings.

```
<!DOCTYPE html>
<html>

<head>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>

<body>

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

</body>
</html>
```

## Adding JS to Pages: <head>

Javascript can be placed in a variety places in the document. Sometims you'll find it in the head...

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

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>

</body>
</html>
```

## Adding JS to Pages: <body>

Sometimes you'll find it in the body—typically at the end of the body, so that it loads after all of the document elements have loaded.

```
<!DOCTYPE html>  
<html>  
<body>  
  
<script src="myScript.js"></script>  
  
</body>  
</html>
```

```
<script src="https://www.w3schools.com/js/myScript1.js"></script>
```

```
<script src="/js/myScript1.js"></script>
```

## Adding JS to Pages: External

Like CSS, it can be embedded in the page, or linked to externally.

```
<button type="button" onclick="document.getElementById('demo').innerHTML = Date()">Click me to display Date and Time.</button>

<p id="demo"></p>
```

## My First JavaScript

Click me to display Date and Time.

## My First JavaScript

Click me to display Date and Time.

Sat Mar 25 2017 18:25:06 GMT-0400 (EDT)

# Adding JS to Pages: Inside Element

Calling the functionality can happen when the document loads, or based on user actions.



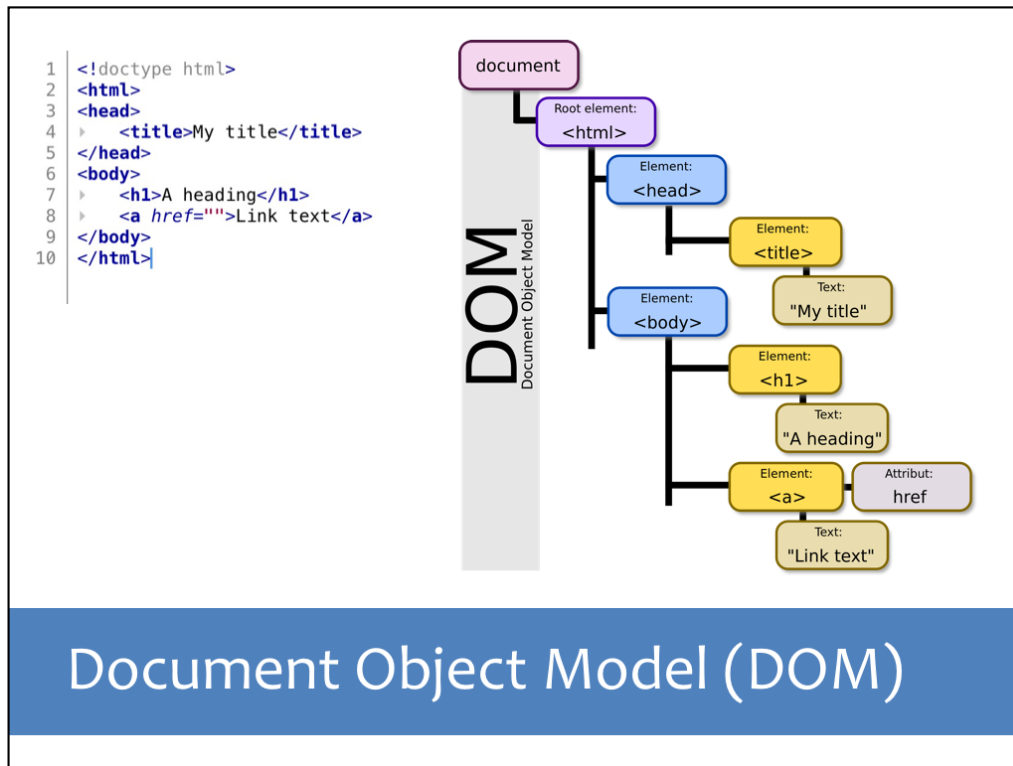
For Thursday, you'll do this portion of the W3C's Javascript tutorial—from "JS HOME" to "JS Objects." You should already be familiar with the basic concepts of variables, objects, operators, and objects from your introductory programming class(es). This will show how they work in Javascript.



- **ECMAScript:** A language standard maintained by ECMA International. Usually used to refer to the standard itself.
- **JavaScript (JS):** Commonly used name for implementations of the ECMAScript standard. This term isn't tied to a particular version of the standard.
- **ECMAScript 5 (ES5):** The 5th edition of ECMAScript, standardized in 2009. Implemented in all modern browsers
- **ECMAScript 6 (ES6)/ ECMAScript 2015 (ES2015):** The 6th edition of ECMAScript, standardized in 2015. Partially implemented in most modern browsers.
- **ECMAScript 2016 (ES2016):** The expected 7th edition

## Javascript (JS) vs ECMAScript (ES)

You will see the terms ES5 and ES6 used a lot in job descriptions and in discussions of Javascript coding. It's important to understand the relationship between JS and ES.

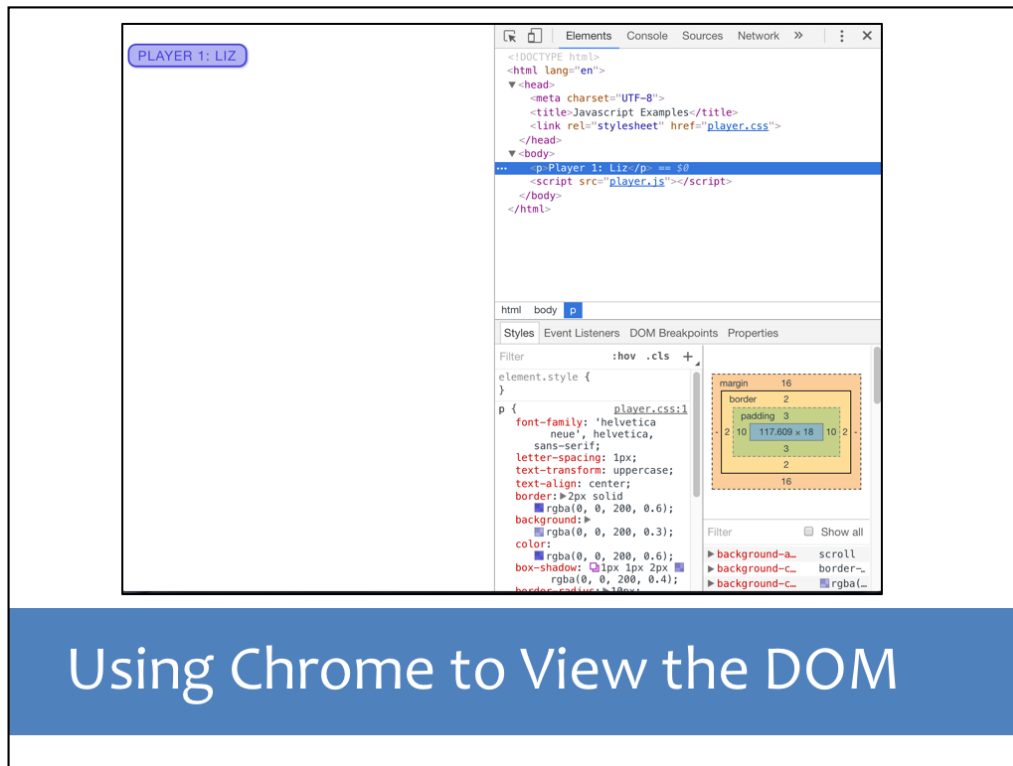


This is the document object model—the DOM. It's an application programming interface to a web page.

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5     <meta charset="UTF-8">
6     <title>Javascript Examples</title>
7     <link rel="stylesheet" href="player.css">
8 </head>
9
10 <body>
11     <p>Player 1: Chris</p>
12
13     <script src="player.js"></script>
14 </body>
15 |
16 </html>
17
```

## Using Chrome to View the DOM

Here's the source of the player button example I used before. (Bring it up in an HTML editor and/or browser, show function again, view source.)



## Using Chrome to View the DOM

If we use Chrome's developer tools to view the DOM of the page, it looks a lot like your source HTML, but there are two differences:

- 1) If you left out required tags that the browser needs for laying out the page, they'll be added in here.
- 2) Any changes to your original HTML resulting from scripts will show up here

w3schools.com

THE WORLD'S LARGEST WEB DEVELOPER SITE

[HTML](#) [CSS](#) [JAVASCRIPT](#) [SQL](#) [PHP](#) [BOOTSTRAP](#) [MORE ▾](#) [REFERENCES ▾](#) [EXAMPLES ▾](#) [🔍](#)

JS HTML DOM

DOM Intro

DOM Methods

DOM Document

DOM Elements

DOM HTML

DOM CSS

DOM Animations

DOM Events

DOM EventListener

DOM Navigation

DOM Nodes

DOM NodeList

## JavaScript HTML DOM

[< Previous](#)[Next >](#)

With the HTML DOM, JavaScript can access and change all the elements of an HTML document.

### The HTML DOM (Document Object Model)

When a web page is loaded, the browser creates a **Document Object Model** of the page.

The **HTML DOM** model is constructed as a tree of **Objects**:

Again, Doing is Better Than Listening!

Do the first six pages of the DOM tutorial on W3Schools—DOM Intro through DOM CSS. Finish before Thursday's class!

- Roles & Responsibilities
  - Discuss roles & responsibilities as a group today
  - Today's in-class exercise: Update the document in Github repo with *your own* responsibilities
- Structure Document
  - Content inventory and content diagram
  - Due Thursday (you'll have feedback by Saturday)

Group Project Time