



Coding & Data Skills for Communicators

Dr. Cindy Royal

Texas State University

School of Journalism and Mass Communication

The Document Object Model

The Document Object Model (DOM) is an application programming interface (API) for valid HTML documents. It defines the logical structure of documents and the way a document is accessed and manipulated. It's not exactly HTML, but it is the HTML that is parsed by the browser and turned into the DOM.

With the Document Object Model, programmers can build documents, navigate their structure, and add, modify, or delete elements and content. The DOM can be viewed as a tree structure. It contains a root element and any number of subelements and their corresponding contents, which might be their text value or the HTML code they contain.

Manipulating the DOM

One of the most common uses of JavaScript with HTML pages is for manipulating the DOM. You have a site with which you want the user to interact. You want them to type in their name, so you can use it to address them in the site, or you want them to select a dropdown to move them to another display of content.

To manipulate the DOM, you typically need an HTML form and you provide an action that, when called, changes the DOM in the desired manner.

Using a Form with JavaScript

In the example below, the form includes a list of browsers, using the `<select>` and `<option>` elements we discussed in the HTML Forms and Form Processing Tutorial. The JavaScript function "favBrowser()" is called on the form's "onchange" event. That executes the script that changes the innerHTML of the span with the id "favorite" that is in the paragraph that displays on the page.

The script does a few things. It establishes a variable named myFav to hold the "myList" element. Then it changes the innerHTML of the id "favorite" in the paragraph to the selected text from the dropdown that is now represented by the variable myFav.

Using the variable myFav just made the code below is a little more streamlined. Don't worry if this doesn't make complete sense yet. It will when we discuss JavaScript variables and functions in more detail. For now, simply appreciate what can be done with forms and JavaScript to manipulate the content of a Web page.

Create a Web page that contains the code below and see how it functions.

Example:

```
<!DOCTYPE html>
<html>
<head>
<script>
    function favBrowser()
    {
        var myFav=document.getElementById("myList");
        document.getElementById("favorite").innerHTML=myFav.options
        [myFav.selectedIndex].text;}
</script>
</head>

<body>
<form>
Select your favorite browser:
<select id="myList" onchange="favBrowser()">
    <option></option>
    <option>Google Chrome</option>
    <option>Firefox</option>
    <option>Internet Explorer</option>
    <option>Safari</option>
    <option>Opera</option>
</select>
</form>
<p>Your favorite browser is: <span id="favorite"></span> </p>

</body>
</html>
```