

Practice: JavaScript and the Document Object Model

Kris Hadlock

November 01, 2011

Take your knowledge of JavaScript a bit further—into the DOM. Through the exercises in this practice, learn how to use JavaScript properties to traverse the DOM, targeting nodes, changing node values, and manipulating attributes.

About this article

These exercises and solutions supplement the developerWorks article "[JavaScript and the Document Object Model](#) (DOM). You may want to read the "JavaScript and the Document Object Model" article before working through these exercises.

The exercises in this article give you practice in:

- Traversing the DOM using a number of different JavaScript properties
- Targeting nodes
- Retrieving and changing the values of nodes and their attributes
- Manipulating the DOM
- Adding, removing, and replacing nodes using JavaScript properties

To work through these exercises, you should have an HTML editor and a working Internet connection or localhost environment in which to test your JavaScript.

Exercise 1: Traversing the DOM

Knowing how to traverse the DOM using JavaScript provides a foundation to altering an HTML page in real time. Using the HTML markup in Listing 1, perform these tasks:

1. Use the `firstChild` property to access an element.
2. Use the `lastChild` property to access an element.
3. Use the `nextSibling` property to access an element.
4. Use the `previousSibling` property to access an element.
5. Use the `parentNode` property to access an element.
6. Use the `childNodes` property to access a group of child elements.

Listing 1. HTML markup for the traversing the DOM exercise

```
<html>
<head>
  <title>JavaScript and the Document Object Model</title>
</head>
<body>
  <div id="page">
    <div id="header"><h1 id="title">Page title</h1></div>
    <div id="content">
      <h2>Content title</h2>
      <p>Some copy goes here</p>
      <p>More copy goes here</p>
    </div>
  </div>
</body>
</html>
```

Exercise 2: Targeting nodes

In [exercise 1](#), you learned how to target nodes in several ways. Continuing to use the markup in [Listing 1](#), perform the following tasks:

1. Retrieve the value of a node using `nodeValue`.
2. Change the value of a node using `nodeValue`.
3. Retrieve the value of a node attribute.
4. Change the value of a node attribute.

Exercise 3: Manipulating the DOM

Now that you know how to traverse the DOM and alter node values, the next logical step is to learn how to add, remove, and replace nodes. Perform the following tasks:

1. Use the `appendChild` method to add a node.
2. Use the `insertBefore` method to add a node.
3. Use the `removeChild` method to remove a node.
4. Use the `replaceChild` method to replace a node.

Exercise solutions

Follow these [solution steps](#) to check your work.

Related topics

- Learn more about "[JavaScript and the Document Object Model](#)" (developerWorks, July 2002).
- To learn more about the fundamentals of the JavaScript language, check out "[Get started with the JavaScript language, Part 1: JavaScript language fundamentals](#)" (developerWorks, April 2011) and "[Part 2: Events, cookies, timing, and more](#)" (developerWorks, August 2011).

© Copyright IBM Corporation 2011

(www.ibm.com/legal/copytrade.shtml)

[Trademarks](#)

(www.ibm.com/developerworks/ibm/trademarks/)