

JavaScript Workshop

October 2006 Meeting

DOM Properties

DOM Properties

- The DOM tree uses nodes to represent everything on a web page.
- You would use either `getElementById` or `getElementsByTagName` to get to a specific element in your document. Then you can use one or more of the following properties to traverse and reference individual nodes within a node tree:

childNodes

nodeName

nodeType

firstChild

lastChild

childNodes

- The `childNodes` property returns an array containing all the children of an element node in a document's node tree.
- The `hasChildNodes` method returns true or false if a specified element has any child nodes.

```
<p id = "greeting">Hello</p>
```

```
var p = document.getElementById("greeting"); ---> element node
```

```
alert(p.hasChildNodes()); ---> returns true, the child node is "Hello"
```

```
alert(p.childNodes.length); ---> returns 1
```

nodeValue

- The `nodeValue` property can be used to get or set the value of a node. Be aware that certain nodes like element nodes will return a null value since an element node has no value other than being a container for a text node or another child element node.

```
<p id = "greeting">Hello, how are you?</p>
```

```
var p = document.getElementById("greeting");
```

```
alert(p.childNodes[0].nodeValue); ---> returns "Hello, how are you?"
```

```
p.childNodes[0].nodeValue = "I'm fine." ---> changes the node value
```

firstChild and lastChild

- There is a shorthand way of writing `childNodes[0]`. Whenever you want to get the value of the first node in the `childNodes` array, you can use `firstChild`. The DOM also provides a `lastChild` property.

```
<p id = "greeting">Hello<br/>Goodbye</p>
```

```
var p = document.getElementById("greeting");
```

```
alert(p.firstChild.nodeValue); ---> returns "Hello" (text node)
```

```
alert(p.childNodes[1].nodeName); ---> returns BR (element node)
```

```
alert(p.lastChild.nodeValue); ---> returns "Goodbye" (text node)
```

Traversing a node tree

- Many elements on a page will have nested child nodes. Which means you will have to go through the different levels of child nodes in a node tree.

<i>node tree</i>	<i>node name</i>
<code><table id = "greeting"></code>	<code>---> TABLE (parent)</code>
	<code>---> TBODY (child of TABLE)</code>
<code><tr></code>	<code>---> TR (child of TBODY)</code>
<code><td></code>	<code>---> TD (child of TR)</code>
<code><h1>Hello</h1></code>	<code>---> H1 (child of TD); #text (child of H1)</code>
<code></td></code>	
<code><td></code>	<code>---> TD (child of TR)</code>
<code><h3>Bye</h3></code>	<code>---> H3 (child of TD); #text (child of H3)</code>
<code></td></code>	
<code></tr></code>	

Next meeting: Image Gallery pt. 1

- JavaScript Workshop website:
<http://www.javascriptworkshop.com>
- Any questions?
frank@javascriptworkshop.com - Frank Stepanski
- Recommended reading:
[DOM Scripting: Web Design with JavaScript](#)