

Creating and appending nodes

`document.createElement(element)` Makes a new element

Has to be added to the DOM

`node.appendChild(element)` Adds element inside a node

Controlling node insertions

`appendChild()` lacks precision

Need to insert a node anywhere in the node list

Use `insertBefore()` for surgical insertions

`newNode.insertBefore(pNode, newNode.childNodes[5])`

Cloning and removing nodes

`cloneNode()` makes a copy

you can then reposition the node

`removeChild(node)` removes the node

Has to be called from a parent node

`var newNode = myNode.cloneNode(true)` copy node and all its children

`insertLocation.insertBefore(newNode, insertLoction,childNodes[4])`

`myNode.parentNode.removeChild(myNode)`

Replacing existing nodes

`replaceChild()` replaces a node

You must call it from the parent node

Saves you the step of having to delete the original

`replaceNode.parentNode.replaceChild(myNode, replaceNode)`



By [deleted]
cheatography.com/deleted-20357/

Published 22nd October, 2015.
Last updated 22nd October, 2015.
Page 1 of 1.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>