

The Document Object Model (DOM) and what it's good for

If you develop for the web, you may have heard the term “Document Object Model” or “DOM” thrown around. You might have even used it without realizing it. The DOM is an object structure that represents web pages. It’s primary use is in Javascript, though it can crop up in other places too. Because Javascript itself is just a language, it relies on the DOM to give meaning to objects such as “window” and “document”. Without the DOM, you would have a lot of cool functions at your disposal, but nothing to use them on!

Most good Javascript references also include a layout of the DOM because it is so vital to using Javascript. However, it can be useful to see the different browser “flavors” of the DOM by itself. Here are a few resources:

- [The official W3C DOM](#)
- [Gecko \(Mozilla, Firefox, Netscape\) DOM Reference](#)
- [Internet Explorer DOM Reference](#)
- [A nice cross-browser comparison of DOM feature availability](#)

As stated earlier, DOM is more than just Javascript. Sometimes it can be useful to browse the DOM tree for a web page to debug problems, as you can easily see what objects fit where. You can also quickly identify where an object’s boundaries are, and how it relates to its neighbors.

Safari comes with a built-in DOM inspector, but you have to [enable its Debug window](#) to use it. Firefox comes with a DOM inspector, but is not always included by default. The OS X Firefox includes it, but for the Windows version, you must choose “Custom Install” when installing Firefox, and check “Developer Tools” when it asks you what add-ons you want. There is a [Developer Toolbar for Internet Explorer](#) which includes (amongst other things) a DOM inspector. At the time this article was written, it was still in beta, but should be usable.

Revision #3

Created 2006-09-29 16:10:52 UTC by Rozett, Keith

Updated 2011-05-10 22:09:10 UTC by Franks, Mike