

The SCRIPT Tag

The `script` element

All JavaScript, when placed in an HTML document, needs to be within a `script` element.

A `script` element is used to link to an external JavaScript file, or to contain inline scripting (script snippets in the HTML file). A `script` element to link to an external JavaScript file looks like:

```
<script src="script.js"></script>
```

while a `script` element that contains inline JavaScript looks like:

```
<script>
  // JavaScript code here
</script>
```

Inline scripting has the advantage that both your HTML and your JavaScript are in one file, which is convenient for quick development and testing. Having your JavaScript in a separate file is recommended for JavaScript functions that can potentially be used in more than one page, and also to separate content from behaviour.

Inline JavaScript

Using inline JavaScript allows you to easily work with HTML and JavaScript within the same page. This is commonly used for temporarily testing out some ideas, and in situations where the script code is specific to that one page.

```
<script>
  // JavaScript code here
</script>
```

Inline HTML comment markers

The inline HTML comments are to prevent older browsers that do not understand the `script` element from displaying the script code in plain text.

```
<script>
  <!--
  // JavaScript code here
  //-->
</script>
```

Older browsers that do not understand the `script` element will interpret the entire content of the `script` element above as one single HTML comment, beginning with "`<!--`" and ending with "`--`".

-->", effectively ignoring the script completely. If the HTML comment was not there, the entire script would be displayed in plain text to the user by these browsers.

Current browsers that know about the `script` element will ignore the *first* line of a `script` element, if it starts with "`<!--`". In the above case, the first line of the actual JavaScript code is therefore the line "`// JavaScript code here`".

The last line of the script, "`// -->`", is a one line JavaScript comment that prevents the HTML end comment tag "`-->`" from being interpreted as JavaScript.

The use of comment markers is rarely required nowadays, as the browsers that do not recognise the `script` element are virtually non-existent. These early browsers were Mosaic, Netscape 1, and Internet Explorer 2. From Netscape 2.0 in December 1995 and Internet Explorer 3.0 in August 1996 onward, those browsers were able to interpret JavaScript.^[1] Any modern browser that doesn't support JavaScript will still recognize the `<script>` tag and not display it to the user.

Inline XHTML JavaScript

In XHTML, the method is somewhat different:

```
<script>
  // <![CDATA[
  // [Todo] JavaScript code here!
  // ]]>
</script>
```

Note that the `<![CDATA[` tag is commented out. The `//` prevents the browser from mistakenly interpreting the `<![CDATA[` as a JavaScript statement. (That would be a syntax error).

Linking to external scripts

JavaScript is commonly stored in a file so that it may be used by many web pages on your site. This method is considered a best practice. This is because it separates a page's behavior (JavaScript) from its content (HTML), and it makes it easier to update code. If several pages all link to the same JavaScript file, you only have to change the code in one place.

Add `src="script.js"` to the opening `script` tag. This means that the JavaScript code for this page will be located in a file called "script.js" that is in the same directory as the web page. If the JavaScript file is located somewhere else, you must change the `src` attribute to that path. For example, if your JavaScript file is named "script.js" and is located in a directory called "js", your `src` would be "js/script.js".

Location of script elements

The `script` element may appear almost anywhere within the HTML file.

A standard location is within the `head` element. Placement within the `body` however is allowed.

```
<!DOCTYPE html>
<html>
<head>
  <title>Web page title</title>
  <script></script>
```

```
</head>
<body>
<!-- HTML code here -->
</body>
</html>
```

There are however some best practices for speeding up your web site [\[2\]](#) suggested by the Yahoo! Developer Network that specify a different placement for script tags: to put scripts at the bottom, just before the </body> tag. This speeds up downloading, and also allows for direct manipulation of the DOM while the page is loading. It is also a good practice to separate HTML documents from CSS code for easier management.

```
<!DOCTYPE html>
<html>
<head>
  <title>Web page title</title>
</head>
<body>
<!-- HTML code here -->
<script src="script.js"></script>
</body>
</html>
```

