

Towards more simplicity

Changes to elements in HTML5 go towards a greater simplicity. In this section, we will look at some examples highlighting these improvements, including:

- the new doctype definition;
- the fact that the "type" attribute of elements such as `<link>` or `<script>` are now optional;
- the syntax constraints that have been relaxed;
- the new structural elements that have been added, etc.

A MINIMAL HTML5 DOCUMENT

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Page Title</title>
  <link rel="stylesheet" href="style.css">
  <script src="script.js"></script>
</head>
<body>
10. ... <!-- The rest is content -->
  </body>
</html>
```

Let's compare it to the HTML4 minimal document below

(source: <http://www.sitepoint.com/a-minimal-html-document/>). Differences have been highlighted in red:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
```

```
"http://www.w3.org/TR/html4/strict.dtd">
<html lang="en">
<head>
  <meta http-equiv="content-type" content="text/html" charset="utf-8">
  <title>title</title>
  <link rel="stylesheet" type="text/css" href="style.css">
  <script type="text/javascript" src="script.js"></script>
</head>
<body>
10. ...
</body>
</html>
```

SIMPLER CHARACTER SET DEFINITION

One word about the `<meta charset="utf-8">` at line 4 in the HTML5 version: it is really a good practice to declare the character set of your document to protect against [a serious security risk](#). For more details, please refer to the "Why Internationalization is important" section in the Course Intro chapter.

NO MORE COMPLICATED DOCTYPE DEFINITIONS

The "Doctype", often called DTD (Document Type Declaration) is used by tools such as HTML validators (i.e. [the W3C validator](#)), and specifies the rules used by an HTML or an XHTML page. These rules are contained in special documents called "Document Type Definitions" (also abbreviated as DTD), written in a language that may seem a bit barbaric to humans (they are intended to be read by software), and hosted by W3C.

DTDs are not used by current Web browsers to validate the structure of an HTML page, as they "read" the code without using the DTD to decipher it, using only "rules" contained in their own "HTML engine", but it is still preferable to indicate the doctype as modern browsers have several rendering engines that are chosen depending on the doctype.

Old HTML1 Web pages will not be rendered the same way as new HTML5 pages, since in

the 90's some of them were written by hand and may contain errors, embedded HTML elements, etc.

With HTML4, doctype definitions looked like this: `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"`

`"http://www.w3.org/TR/html4/loose.dtd">`, and this was even more complicated as one had to choose between three different possibilities (doctype could be transitional, strict, or frameset). Most of the time, the doctype definition was copied and pasted from one document to another and was nearly impossible to memorize.

With HTML5, there is only one way to indicate the doctype, and it's so simple there is no reason to forget it:

```
<!doctype html>
```

THE "TYPE" ATTRIBUTE IS OPTIONAL

With a `rel="stylesheet"` attribute, it is no longer necessary to indicate `type="text/css"` (from [the specification](#): "the default type for resources given by the `stylesheet` keyword is `text/css`."

The "type" attribute is not needed in HTML5, and even old browsers will use `text/css` as the default type for stylesheets today. So, either way, you can omit the "type" attribute altogether and use:

```
<link href="file.css" rel="stylesheet"/>
```

instead of:

```
<link href="file.css" rel="stylesheet" type="text/css"/>
```

We will not go into detail about the `<link>` element, but the fact that the `type` attribute

is becoming optional shows the current direction taken by HTML5: towards greater simplicity.

Please see how to include a JavaScript file in our page:

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

Here again, the type attribute has been omitted. Just as a reminder, the old way to do the same thing:

```
<script type="text/javascript" src="script.js"></script>
```

MORE FLEXIBLE SYNTAX CONSTRAINTS

If you look at the "minimal document" example, or at other examples in this course, you won't find a lot of differences compared to the same code in XHTML: attribute values are surrounded by quotes, all elements are written in lower case, etc. This is because we are used to writing this way, but HTML5 supports also a simplified syntax:

- Thanks to HTML5, you can omit quotes (not always, but most of the time) or use uppercase, lowercase or a mix of the two.
- Lots of elements do not need a closing tag anymore: ``, `</dt>`, `</dd>`, `</tr>`, `</th>`, `</td>`, `</thead>`, `</tfoot>`, `</tbody>`, `</option>`, `</optgroup>`, `</p>` (in most cases), `</head>`, `</body>` and `</html>`. Older browsers often add closing tags automatically at render time. We recommend however closing tags that would naturally be closed: the ones that delimit a particular zone in the document.
- Attribute values only need to be quoted if they contain spaces or some non-alphanumeric characters, instead of writing `<link rel="stylesheet" href="style.css">`, we could have used `<link rel=stylesheet href=style.css>` instead. However, for

compatibility with older browsers, it might be wiser to still use quotes...