

HTML and CSS tutorial

1. Introduction

This tutorial will explain how to create web pages. To do this, you firstly need the Hypertext Markup Language (HTML). It is a simple technique to learn and you definitely do not need any programming experience. The easiest way to come to terms with HTML is probably to see it primarily as a set of codes that can be used to influence the lay-out or the typographical appearance of a text document in a webbrowser. As is stated clearly in the name of the technique, HTML is a markup language. The term 'mark up' originally stems from the publishing world. A layout artist would communicate his ideas for the typography of a manuscript by circling or underlining certain fragments of the text and by scribbling instructions for these selected words in the margins. Such shorthand instructions were referred to as 'mark up'. They typically included instructions for the font type, font size, margins, and the use of colours. Once it was marked up, the document was passed on to a typesetter or a typographer, who would carry out the instructions as they were provided. HTML codes have largely the same function. They are inserted into the text, and the internet browser will interpret these codes and perform certain actions on the basis of those codes.

Each website that can be found on the Worldwide Web makes use of some form of HTML. Most browsers allow you to inspect the underlying HTML code of web pages. In Internet Explorer, for instance, you can select "Source" from the "View" menu. Listing 1 gives an impression of what an HTML file may look like.

```
<html>
<head>
<title>Knowledge Representation in Humanities
Computing</title>
</head>
<body>
<h1>Knowledge Representation in Humanities
Computing</h1> <b>by</b> <h2>John Unsworth</h2>
  <p><i>Best viewed with Internet Explorer 5 or
higher</i></p>
<p>The assertion of this paper is that the methodology
known as knowledge representation has profound
implications for humanities computing, and through
humanities computing, has the potential to change the way
humanities scholarship is done, to change the nature of
graduate education in the humanities, and to change the
relationship between the humanities and other
professions, let alone other disciplines. ... </p>
  </body>
</html>
```

Listing 1

In the fragment above, it can be seen that HTML files basically consist of two things. Firstly, there is the text that can be read on the screen. Secondly, there are also certain codes that are given in square brackets. These codes are referred to as ‘tags’. An internet browser can do certain things to these words on the basis of these tags. The scope of such an instruction is defined by the position of the opening and closing tag. The opening tag points to the beginning of a piece of text, and the closing tag, which has a forward slash before the tag name, indicates the end of the text fragment.

2. Formatting a text

Here are a number of simple formatting rules:

- Use the <p> tag to create a paragraph.
- Use the <i> tag to render text in italics.
- The tag is used to make the text bold.
- Text headings can be created by using the <hx> tag. This “h” must always be followed by a number. The higher the number, the larger the text will appear on the screen. There are six levels of headings: <h1> to <h6>. <h1> is considered to be the most important heading, and <h6> the least important.

When listing 1 is saved as “webpage.html”, for instance, and opened in a web browser, it should look more or less as follows:

Knowledge Representation in Humanities Computing

by

John Unsworth

Best viewed with Internet Explorer 5 or higher

The assertion of this paper is that the methodology known as knowledge representation has profound implications for humanities computing, and through humanities computing, has the potential to change the way humanities scholarship is done, to change the nature of graduate education in the humanities, and to change the relationship between the humanities and other professions, let alone other disciplines. ...

Note that the tags that were used in listing 1 do indeed have the effects that were discussed.

3. Basic structure of an HTML document

For HTML files to function correctly, the text that will be printed on-line must be incorporated within a larger structure. The entire document must be wrapped

into an `<html>` container tag. More concretely, this means that an `<html>` opening tag must be given at the start, and that an `</html>` closing tag must be used at the very end of the file. The `<html>` tag can contain two other tags: `<head>` and `<body>`. These three tags `<html>`, `<head>` and `<body>` should appear in every HTML document.

The `<body>` contains the content that will be visible in the web browser. The `<head>` tag normally contains some information about the web page, such as its author, the title, and some keywords. All the texts that appear within the `<head>` will not be shown on the web page. In most browsers, the text that is given in the `<title>` tag will appear in the title bar of the browser window. A proper HTML document should also begin with a line that declares what version of HTML is used in the document. Listing 2 contains the tags that must minimally be present in proper HTML files.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head> <title>[Title of the web page]</title></head>
<body></body>
</html>
```

The use of white space, such as tabs, spaces, hard returns is not relevant. In fact, all such white space will be reduced to one single space.

4. Hyperlinks

The great advantage of the Worldwide web is that it is very easy to connect web pages. HTML enables us to make hypertexts, and these are texts that contain hyperlinks. Hyperlinks enable users to jump from one page to another by following links. To transform a certain piece of text into a link that can be clicked on, use the `<a>` tag. The 'a' stands for 'anchor'. The text that is given in between the opening and closing tags of the `<a>` tag is used as the caption for the link.

To link to a page on another website you need to give the full web address in a so-called attribute. An attribute may be seen as an additional instruction that can be used within a tag to modify or refine its behaviour. Attributes can only be used in opening tags. They consist of a property and a value. A link to the English homepage of Leiden University looks as follows:

```
This a link to <a href="http://www.leiden.edu">the
website of Leiden University</a>.
```

As can be seen, the property name is given after the tag name. Property and value are separated by the equals sign ("="). The value of the attribute must be given in

single or double quotation marks. In the <a> tag, the URL must be given in an attribute named 'href'.

5. Images

Image files can be included on a webpage by using the tag. Like the <a>, the tag needs to be used with an attribute. The value of this attribute must be the name of the image file that is to be included. If you provide only the filename, this assumes that the file is stored in the same directory as the image. If this is not the case, a full path to this image must be given. An image called "logo.jpg" can be included on the page using the code below:

```

```

If no further attributes are used, this will have the effect that the image will be added in its original dimensions. Of course, it may be the case that the image is too large to fit on the page. In that case, you may want to resize the image by using the height and width attributes. If no measurement is given, the numbers that are given will be interpreted as the number of pixels.

```

```

7. HTML lists

Lists are useful when you need to enumerate certain items. HTML supports bulleted and numbered lists. The first kind of list can be created with the element. Individual items must be given within the and tags. The principle of the bulleted list is illustrated in listing 3.

```
<p>Here is a bulleted list:  
<ul>  
<li>Item 1</li>  
<li>Item 2</li>  
<li>Item 3</li>  
</ul>  
</p>
```

When you use the tag in stead of , the items in the list will be numbered.

8. Tables

Numerical or other data can be presented neatly by using an HTML table. A table is a collection of rows, and rows in turn consist of a number of columns. A table is created with the `<table>` tag. Rows can be added by making use of the `<tr>` tag (“table row”), and columns are created with the `<td>` tag (“table data”).

Here is a simple example of an HTML table.

```
<table>
<tr>
<td>Row 1, column 1</td>
<td>Row 1, column 2</td></tr>
<tr>
<td> Row 2, column 1</td>
<td>Row 2, column 2 </td></tr>
</table>
```

8. Cascading stylesheets

When you want to develop a professional layout, HTML alone is often not enough. Tags such as `` and `<i>` allow you to specify some aspects of the typography, but the type of formatting that HTML makes possible is rather limited. More detailed rules for the page’s typography can be given by making use of Cascading Stylesheets (CSS). Basically, a CSS file is a list of tag names, followed by certain rules for these tags. In CSS terminology, the tag name that is given is referred to as the selector. It “selects” the tag that needs to be formatted. The rules must all be given within curly brackets. The following is a simple example of a CSS rule:

```
p { font-family: Arial; }
```

This line specifies that all the `<p>` tags in the document must be rendered in the Arial font. All CSS rules consist of a property, followed by a value. Property and value are separated by a colon. Each rule must end in a semi-colon.

Some rules may need to be applied to the entire document, and not just to one particular tag. Rules for the general ‘look and feel’ of the whole page can best be given after the ‘body’ selector:

```
body { margin-left: 10%; margin-right: 10%; font-family:
Arial }
```

The body tag contains all the texts that are visible on the webpage. This rule for `<body>` will be applied not only to the `<body>` tag itself, but also to all the tags

that are contained within <body>. The stylesheets that can be created with CSS are ‘cascading’ because the rules that are given for a tag on a high level are automatically transferred to tags on a lower level. Here is a list of commonly used text properties.

```
font-size
/* can be given in pixels, e.g. 14px */

text-align: left | center | right | justify ;

font-family

/* The font type that you specify may not be available. For
this reason, it is best to give multiple font families. If the
first font type can not be used, the second font in the list
then be chosen instead. Font names that consist of more than
one word must be given in double quotes, e.g. "Arial MS
Unicode" */

font-style: italic | oblique ;

font-weight: bold ; display: block | inline | none ;color

/* Use predefined colours such as "black", "blue", "red",
"silver", or use hexadecimal RGB values */
```

To actually apply the CSS rules to an HTML document, a <link> tag must be given in the <head> of the HTML file. This <link> tag must have an ‘href’ attribute which contains the name of the stylesheet. In addition, two other attributes must be present. Firstly, the ‘text’ attribute, with the value “text/css”, and secondly, the ‘rel’ attribute with value “stylesheet”. If the CSS Stylesheet is saved in a file named “stylesheet.css”, for example, the HTML file can be connected to this CSS file as follows:

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

Sometimes, one tag may need to be presented differently in different sections of the web page. For example, your page may contain a menu on the left-hand side of the screen, and you may want the <p> tags in this menu to look different from the <p> tags outside of this menu. If you want the text in the menu to appear in blue, for instance, you can create a so-called class for this special <p> tag. Classes are created as follows:

```
p.menu { color: "blue"; }
```

The term that follows the stop is the name of the class. The class name can be used in the HTML document as a 'class' attribute:

```
<p class="menu">This text will appear in blue</p>
```

9. CSS and hyperlinks

CSS can also be used to specify the appearance of hyperlinks. For instance, you can change the colour of links, and you can specify whether or not the link texts will be underlined. Hyperlinks on a website can have different states. There may be links that the user has not yet clicked on, and there may be links that have already been visited. It is also possible to specify what the link must look like when the user's mouse hovers over the link text, and what it must look like while the new page is loading (an "active" link). The CSS rules below demonstrate the various ways in which the appearance of hyperlinks can be influenced.

```
a { color: #0c2577; text-decoration: none; }
/* new links */

a:visited { color: #660000; text-decoration: none; }
/* visited links */

a:hover { color: black; text-decoration: underline; }
/* mouse-over */

a:active { color: black ; text-decoration: underline ; }
/* active links*/
```