



character encoding in XHTML

To create any web page that will display properly in whichever browser it is being viewed in - one of the most important details to define is the character encoding for the XHTML document.

At its very basic - text is a sequence of characters with semantic rules which control how the text characters flow, ie drawn from left to right or right to left, or top to bottom and so on - and how the text characters connect, ie how punctuation and accents format the body text itself or join adjacent letters. For most of the Western world this will mean a so-called Latin character set consisting of letters, lowercase and uppercase, from a to z and numbers between 0 and 9.

To display a sequence of characters as found in any text or language in a digital format - there are several details which need to be defined and specified to achieve a legible and uniform result:

- the character set
which clearly specifies the actual characters and their index within the character set, ie which position each letter takes within the overall character set (in our case, for example, the letter C comes third in the character set)
- the encoding of each character
which defines how each of the characters within the character set is encoded for digital output on screen

Together both definitions ensure that whichever software deals with the request of processing text in digital format can access each character individually and its encoding and process this to position and display the text as asked for by the user. This process allows us to work with text in various different softwares and takes place each time you use text on a computer. Once your text has been processed you can then save/store your files digitally.

It is only possible for any software to process the text digitally if both definitions for character set and encoding are present and accessible. This, of course, means that if you are distributing your text document over the internet - you will need to include the character set and encoding as well in order for the browser to interpret the body's characters correctly.

For the coded character set however the definitions are more complex and involved as they will also need to define the 'nature' of characters such as their directionality (does the text flow from left to right - or right to left). Their definition will need to include combined characters such as accented letters (which fundamentally consist of 2 parts on top of each other).

There are a number of different character sets which define and encode only a few of the ten of thousands of different characters used in the world's different languages. To allow for a universal character set to be implemented for several languages - 2 character sets were merged which became the UCS - Universal Character Set.



UCS is the document character set of all XML, XHTML and HTML documents. The UCS does have the same positions for certain characters as other character sets making it possible for the standard sets already used on the web to remain compatible and display correctly.

The UCS character set supports several different encodings. The main encoding, known as UTF-16 (UTF stands for Universal Character Set Transformation Format), stores each character in two bytes. This is the easiest encoding for software to handle and is often used when UCS text is stored in memory (e.g., by tools such as editors or browsers).

UCS also supports two encodings that use single bytes as the basic encoding unit. The first of these, known as UTF-8, represents each UCS character as a stream of one or more bytes—this encoding uses all the bits in the byte for encoding purposes. A second encoding, known as UTF-7, represents each character as one or more bytes, but uses only the seven least significant bits for encoding purposes.

The existence of different encodings can create problems when files are stored on disk or sent over the Internet, because character set information must now tag along with the data and be available to subsequent software. If this information is not available, then the next program to see the text will not know how to decode the data and convert it back into the correct characters.

In practice - this simply means it is essential to specify the character encoding for each document to ensure correct decoding of characters. In the case of XHTML - this will depend on whether the document is being served as XML or HTML document.

For an XML interpretation:

```
<?xml version="1.0" encoding="char-encoding" ?>  
  
<meta http-equiv="Content-Type"  
      content="text/subtype; charset=char-encoding" />
```

For an HTML interpretation:

```
<meta http-equiv="Content-Type"  
      content="text/html; charset=utf-8" />
```