Audit Access Expires Apr 27, 2020
You lose all access to this course, including your progress, on Apr 27, 2020. Upgrade by Apr 13, 2020 to get unlimited access to the course as long as it exists on the site. **Upgrade now**

What is internationalization?

What is internationalization?

Access to the Web for all has been a fundamental concern and goal of the W3C since the beginning. It is easy to overlook the needs of people from cultures different to your own, or who use different languages or writing systems, but you have to ensure that any content or application that you design or develop is ready to support the international features that they will need.

'Internationalization' is sometimes abbreviated to 'i18n' in English, because there are 18 characters between the 'i' and the 'n'.

The W3C Internationalization Activity works with W3C working groups and liaises with other organizations to make it possible to use Web technologies with different languages, scripts, and cultures.

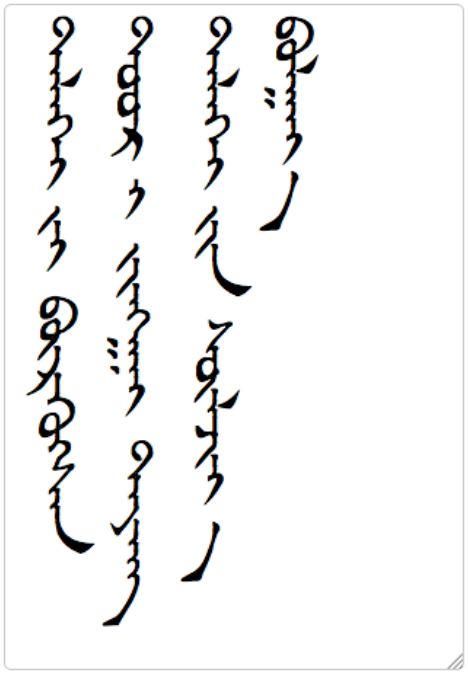
People who use non-Latin writing systems or use the Latin script for certain languages, often have specific typographic needs that differ from text in, say, English. As you learn more about CSS, you will find that it provides many features to support those needs.

Whereas HTML markup provides structure for the content of your page, CSS bring the expressive power to make the page look the way a person from particular culture would expect.

Examples

Here are some examples of things that can be done with CSS.

- It is already possible to make text run vertically in CSS for languages such as Chinese, Japanese, Korean and Mongolian. For more information see Styling vertical Chinese, Japanese, Korean and Mongolian text.

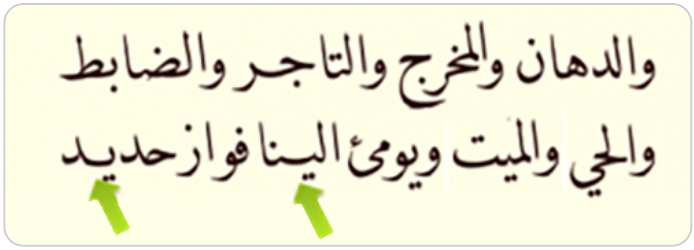


- You can also style counters for lists or chapter headings and such like according to local preferences. Here we see lists using Georgian and Japanese labels.

ა. This
ბ. list
გ. has
დ. Georgian
ე. counter
ვ. styles.

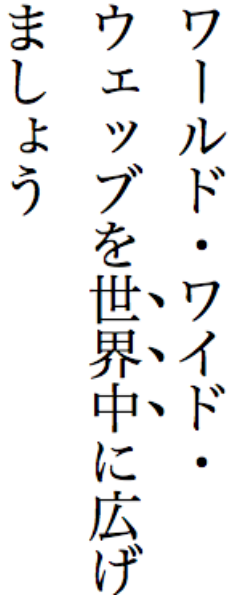
ア、This
イ、list
ウ、has
エ、Japanese
オ、katakana
カ、counter
キ、styles.

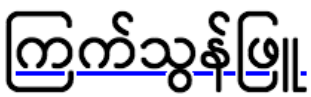
- When you want to justify text so that the lines are straight on both sides of your column, different strategies are used for different scripts. Most Western typography puts an emphasis on adjusting inter-word spaces, but Chinese doesn't use spaces between words, so you generally do *inter-character* spacing. In text written using the arabic script it is common to stretch the baseline that joins letters, or use other techniques to balance the line.



  Some scripts allow words to be hyphenated in order to improve the visual effect of a paragraph, but note that the way in which words are hyphenated depends on the language. (And in arabic script, the CSS specification requires that both parts of the word retain their joining line during hyphenation.)

- Text decoration and text style features can vary in applicability from script to script. For example, Japanese characters are fairly complicated so, rather than italicise their text for emphasis, which can make it harder to read at small sizes, they have a tradition of placing special marks alongside the emphasised text (see the middle line of the Japanese example below). Also, it may be important to avoid underlines running over descenders in some scripts, since it can obscure important marks attached to a base character, so CSS allows you to skip 'ink' as shown in the Burmese example below.





These are just a few examples. There are many more.

CSS & Language

An important point to bear in mind is that for many of these features to work as expected, you need to declare the language of the content. For example:

- Hyphenation won't work unless the content is labelled for language. This is because the way that hyphenation works, and the dictionaries it uses, are language-specific.
- If you want to convert Turkish or Azeri text to uppercase or vice versa, you will get incorrect results unless the browser knows that the text is in that language, because they have a dotted i and a non-dotted ı which do case conversion differently from European languages.
- If text wraps to a new line, by default it does so differently dependent on whether you are dealing with Chinese or Japanese.
- And we could continue...

Therefore, you should always ensure that the correct language is specified in the `lang` attribute on the `html` tag, to indicate the default language of the page. And if you have passages in another language inside the page, you should put a `lang` attribute on markup that surrounds them, too.

Localization

In addition, CSS provides tremendous help if you have to translate content from one language to another. Being able to change a single line in a style sheet to appy a change to all the pages being translated, rather than having to edit every page, saves a massive amount of time. However, this works best when you keep the distinction between semantics (markup) and presentation (styling) clear.

Don't use CSS to apply direction for bidirectional or right-to-left scripts, such as content in Arabic, Hebrew, Persian, Urdu, Divehi, etc. Use HTML markup instead.

Internationalization quick tips

1. **Language:** Always declare the default language of your page using the `lang` attribute on the `html` tag, and indicate internal language changes.
2. **Localizable styling:** Use CSS styling for the presentational aspects of your page. So that it's easy to adapt content to suit the typographic needs of the audience, keep a clear separation between styling and semantic content, and don't use 'presentational' markup.
3. **Use international features:** Use the international features provided by CSS to make your pages look natural to your audience. The more you use such features, and the more you request them, the better browsers will support them.
4. **Check your colors and styles:** Be sensitive to local preferences of your audience for things such as color, but also use of white space, two-dimensional vs. uni-directional display of information, etc.
5. **Use `start` and `end`:** Using these values, where possible, rather than `left` and `right` makes it easier to convert content between languages that use right-to-left and lef-to-right scripts.