Setting Up An Accessible Web Page: The Doctype and HTML Tags (A UX Accessibility Blog Post)

This UX Accessibility Blog Series, “Setting Up An Accessible Web Page,” will walk you through the basic elements of an accessible web page in hopes that you can take what you’ve learned an apply it to more complex pages. This blog series is loosely based on Heydon Pickering’s book [“Inclusive Design Patterns: Coding Accessibility Into Web Design.”](https://www.smashingmagazine.com/inclusive-design-patterns/)

**The Doctype Declaration**

The Doctype declaration tells the web browser what version of HTML your web page is written in so that the browser can render it correctly. Even though your page may be complex and have a lot of different bells and whistles, it’s still a document (hence the “doc” part). Documents are just an interface to deliver content to your users. Leaving out this little <!DOCTYPE html> tag can cause the browser to render the content in a lot of strange ways, aptly named “quirks mode” (https://developer.mozilla.org/en-US/docs/Quirks\_Mode\_and\_Standards\_Mode).

Any time you’ve got some elements rendering weirdly, just check to make sure <!DOCTYPE html> is the very first line on your page before hunting for the solution via CSS or trying to compare your implementation to that on Ensemble.

**<html lang=”en”>**

The lang attribute of the <html> element tells the browser what “human” language the website/document is written in. It’s frequently left out of web pages by developers, but very important. This little attribute helps all of your users fill out forms on the page. The browser takes the page language to determine what dictionary to use to compare text areas against so that any spelling errors that the user makes are evident prior to form submission. Also, the page language helps any third-party translator translate to your user’s native language and helps search engines easily index your website.

Not having a page language declared or declaring the wrong page language could cause a screen reader to use the wrong accent or pronunciation of words. As Heydon Pickering says in his book “Inclusive Design Patterns,” “if <html lang=”en”> is present but the text is actually in French, you’d hear a voice profile called Jack doing a bad impression of French, rather than a French profile called Jaques using authentic French pronunciation.” Computer Braille display users also benefit from having the correct page language set since even though, for example, the French and English alphabets are fairly similar, in Braille, there are many more differences between English and French. Therefore, not only is it important to declare a language, but it’s equally as important to declare the *correct* language.

Declaring a page language also helps browsers choose and render system fonts with appropriate character sets. This is more important in languages like Chinese, but can be equally as important for English, Portuguese, etc. Having your content render in fonts that don’t have the correct character sets is certainly inaccessible and not at all inclusive.

Sometimes, you’ll have quotes or blocks of text that may be in a different language from that of your page language. To switch page languages for a paragraph, just put your lang attribute in the paragraph tag you’re using for content in a different language. Take, for example, your page content was in English, but you had a paragraph that was in Portuguese:

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Page Language Example</title>

</head>

<body>

<p>This paragraph is in English.</p>

<p lang=”pt”>Este parágrafo está em português.</p> (taken from Google Translate)

</body>

</html>

Not only does the lang attribute do all of this work for you, but it’s so easy. Just add it to the html element of all of your pages. If your page uses a layout, it’s even easier. There’s really no excuse to leave this out.