Setting Up an Accessible Web Page: Making Tables Accessible (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/)

Tables that have specified rows or columns with header information about that particular row or column is considered a data table. Without a header, a table can be considered a formatting table. Personally, I don’t like to use tables to format content, but it’s not *technically* wrong. In this blog post, I’ll be focusing on making data tables accessible.

**General**

The simpler the table, the better. Simple tables are more easily understood than ones that are more complex (i.e. have multiple table headers or sections). Simpler is always better, but necessity will dictate how simple your data tables can be. However, it is recommended that you do your best to simplify your data tables as much as possible to be understood by a broader audience.

**TH and the scope attribute**

Sighted users have the luxury of being able to read data tables in any orientation the data may be displayed in. Screen reader users can only hear one cell at a time from left to right, top to bottom. This can easily cause confusion, especially if the table is complex or doesn’t use the right attributes in the markup to make it accessible.

Table headings allow sighted users to designate certain table cells as headers for the column or row it belongs to.

The scope attribute is used to clarify the scope of a header cell and aid comprehension for those who use screen readers. It identifies whether the cell is a header for a row, column, group of columns, or group of rows. The scope attribute can also be used on <td> tags to denote the content of the cell is the header for a row. For example:

<table class="table table-bordered col-2-bloat">

<thead>

<tr>

<th scope="col">#</th>

<th scope="col">Address</th>

<th scope="col">GLA</th>

<th scope="col">Sales Price</th>

<th scope="col">MLS Number</th>

<th scope="col">Service</th>

</tr>

</thead>

<tbody>

<tr>

<td>1</td>

<td scope="row">380 Agate St</td>

<td>1250</td>

<td>$1,000,000</td>

<td>s709276</td>

<td>Desktop Appraisal</td>

</tr>

<tr>

<td>2</td>

<td scope="row">2160 S Coast Hwy</td>

<td>3205</td>

<td>$470,000</td>

<td>l40212</td>

<td>Desktop Appraisal</td>

</tr>

</tbody>

</table>

Sometimes the scope may not need to be on the first element of the row. As in the example above, it would aid comprehension for the screen reader to read out the street address before every value rather than the number row it is.

Penn State’s Accessibility article titled “Tables for Data in HTML” has a great example for how table headings can aid comprehension of a visually impaired user.

[picture]

Assuming the markup is correct and uses scope as we showed in the code example above, the screen reader should read out the following:

“black, Spanish: Negro

black, French: noir

black, Irish: dubh

black, Welsh: du…”

This way, by the time the screen reader reads the last element of the row, the user is still able to remember what English term is being translated.