**Accessibility Training**

1. **Getting Started**
2. To test your feature/component you can download a chrome extension called Chromevox. [Download](https://chrome.google.com/webstore/detail/chromevox-classic-extensi/kgejglhpjiefppelpmljglcjbhoiplfn?hl=en-GB#:~:text=The%20ChromeVox%20screen%20reader%20is,like%20HTML5%2C%20CSS%20and%20Javascript.&text=ChromeVox%20is%20just%20one%20of%20the%20Google%20Accessibility%20extensions.)
3. You can see a demonstration on how Chromevox and screen readers work. It also touches on accessibility as a whole. [View](https://www.youtube.com/watch?v=x18vEEfpK3g)
4. Here is a video on how VO for Mac works. Jaws works pretty much in the same way, so it is worth watching to gain insight as to how users with screen readers interact with the browser. [View](https://www.youtube.com/watch?v=5R-6WvAihms&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=7)
5. Another screen reader to have a look at is NVDA. This is a free screen reader if you would like to use this. The navigation with this screen reader is a little different to VO – Here is a video to see how it works - [View](https://www.youtube.com/watch?v=Jao3s_CwdRU&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=9) / [Download](https://www.nvaccess.org/download/)

**2. Introduction**

It is important that you get to know what is expected from the user and use correct syntax when developing functionality. Below you will find some readings for the bases of how the keyboard should be used as well as some important WAI-ARIA information.

1. <https://www.w3.org/TR/wai-aria-practices/#read_me_first>
2. <https://webaim.org/techniques/keyboard/>
3. **Design Patterns and Widget**. This should be bookmarked and used as reference as there are numerous elements in this section going in to great detail regarding best practice, WAI-ARIA roles, states, properties, keyboard navigation and examples for each. [View](https://www.w3.org/TR/wai-aria-practices/#aria_ex)

**3 Focus**

In this section we will be learning about focus and what focus is. It is important to know that when testing focus, you will also need to test the screen readers’ focus and not only the browser’s focus.

Jaws and VO have their own way of navigating (Ctrl + Option/Alt + Right Arrow) and the focus may differ to that of the browser if the settings are not synced. As developers we will need to test that both ways of navigating and focusing on elements work correctly.

* Here is a short video on focus. [View](https://www.youtube.com/watch?v=EFv9ubbZLKw&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=3)
* Here is a video on controlling focus and tab index. [View](https://www.youtube.com/watch?v=Pe0Ce1WtnUM&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=4)
* Please read this section [Developing a Keyboard Interface](https://www.w3.org/TR/wai-aria-practices/#keyboard)
* [Tabindex attribute](https://webaim.org/techniques/keyboard/tabindex).

**3.1 Roving Tabindex**

To give the user an efficient experience while using the keyboard, it a good idea to use Roving tabindex. This is when complex controls are treated as a single tab stop and then with the use of the arrow keys, they can interact with elements within the control.

Another example would be when opening a dialog, the focus would be on the first interactive item in the dialog, and once closing the dialog, focus should return to the element that opened it.

The last example is when deleting an item in a list, instead of focus just disappearing, it should rather focus on the next item in the list.

This video explains the process in more detail. [View](https://www.youtube.com/watch?v=uCIC2LNt0bk&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=6)

1. **Page Structure, Landmark Regions**

Semantic structure is important, especially for screen readers, as they act as shortcuts to navigate through the page using these semantics as guidelines. Ie bypassing the header and content and going straight to the side nav or navigating the site through the heading tags.

<https://webaim.org/techniques/semanticstructure/#nature>

<https://www.w3.org/TR/wai-aria-practices/#aria_landmark>

Here is an video on landmarks - [View](https://www.youtube.com/watch?v=vAAzdi1xuUY&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=18)

**4.1 How to check accessibility**

This video gives a brief view on how to check for accessibility. It also touches on how page structure can be navigated with a screen reader - [View](https://www.youtube.com/watch?v=cOmehxAU_4s&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=11)

1. **Labeling and ARIA**

Labels are an important factor when using controls or input fields as this is what is read by screen readers to describe the element. The label can often be seen as the “name” attribute in certain cases but do the same thing.

The label can either read from the content between the opening and closing tag eg <button>Save</button> where the label/name would read **Save,** or it could read the label for an input field.

Below are some ways in which items can be labelled.

* 1. **Label**

Here is a video on labeling elements - [View](https://www.youtube.com/watch?v=8dCUzOiMRy4&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=12)

Here is a video on labeling custom elements - [View](https://www.youtube.com/watch?v=7a6bLXw2lqQ&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=24)

Here is documentation on naming and labeling elements - [View](https://www.w3.org/TR/wai-aria-practices/#accessible_names)

* 1. **Aria**

As mentioned in the [intro](https://www.w3.org/TR/wai-aria-practices/#aria_ex), each custom element has a best practice and a section for the Aria attributes you can use for your custom element. Make sure that you use the correct attributes as no ARIA is better than bad ARIA

Here is a video explaining this more - [View](https://www.youtube.com/watch?v=88tfx3jLV_M&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=14)

Here is another video going a little further - [View](https://www.youtube.com/watch?v=88tfx3jLV_M&list=PLNYkxOF6rcICWx0C9LVWWVqvHlYJyqw7g&index=14)

1. **Some Food for thought**

<https://www.youtube.com/watch?v=TomOQYxFnrU>

<https://www.youtube.com/watch?time_continue=1&v=OUDV1gqs9GA>