An introduction to ARIA

What does ARIA stand for?

- **A**ccessible
- **R**ich
- **I**nternet
- **A**pplications

What is ARIA?

“Accessible Rich Internet Applications **(ARIA)** is a set of attributes that define ways to make web content and web applications (especially those developed with JavaScript) more accessible to people with disabilities.”[1]

1. https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA

Examples of ARIA attributes

- **Roles** indicate what an element is or does
  e.g. `role=“search”` to indicate an element is used for searching

- **Properties** give an element extra meaning or semantics
  e.g. `aria-haspopup` to indicate an element triggers a popup

- **States** define the current condition of an element
  e.g. `aria-expanded` indicates whether an element, *or* the element it controls, is in an expanded state

The first rule of ARIA

Don’t use ARIA

…unless you have to

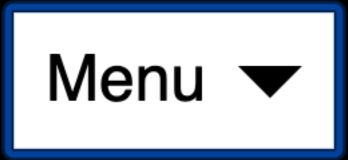
The first rule of ARIA

"If you *can* use a native HTML element or attribute with the semantics and behavior you require **already built in**, instead of re-purposing an element and adding an ARIA role, state or property to make it accessible, **then do so**."[1]

[1] https://w3c.github.io/using-aria/

Example: A menu

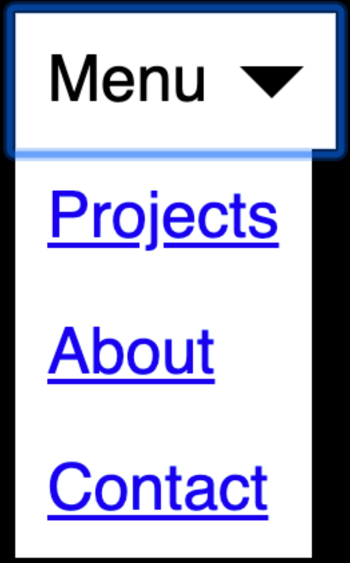
```
<button aria-haspopup="true" aria-expanded="false">
  Menu
  <svg xmlns="http://www.w3.org/2000/svg"
    viewBox="0 0 256 256" aria-hidden="true">
    <polygon points="0,64 128,192 256,64" />
  </svg>
</button>
<div role=“menu" hidden>
  <ul>
      <li><a href="/projects">Projects</a></li>
      <li><a href="/about">About</a></li>
      <li><a href="/contact">Contact</a></li>
  </ul>
</div>
```



“Menu, collapsed, popup button”

Example: A menu

```
<button aria-haspopup="true" aria-expanded="true">
  Menu
  <svg xmlns="http://www.w3.org/2000/svg"
    viewBox="0 0 256 256" aria-hidden="true">
    <polygon points="0,64 128,192 256,64" />
  </svg>
</button>
<div role="menu">
  <ul>
      <li><a href="/projects">Projects</a></li>
      <li><a href="/about">About</a></li>
      <li><a href="/contact">Contact</a></li>
  </ul>
</div>
```



“Menu, expanded, popup button”

No ARIA is better than Bad ARIA

- A role is a **promise**...

- ...a promise that the author has also incorporated JavaScript that provides the keyboard interactions expected for that role

- ARIA can both enhance and cloak existing HTML semantics

- Testing assistive technology is *essential* before using it in production

Conclusions

1. Learn to use native HTML elements effectively

2. If there isn't an HTML element that does what you need, or it isn't well supported, add a light sprinkling of ARIA

3. Test your site with a screen reader

    - On Windows, download NVDA (it’s free): www.nvaccess.org/download/

    - Voiceover on a Mac (it’s built in)