# Accessible Rich Internet Applications (ARIA)

## What is it for?

ARIA stands for 'Accessible Rich Internet Applications'. ARIA is a set of roles and attributes that enable the web to be more accessible for users with disabilities. ARIA adds to HTML. It lets common web app interactions be passed to assistive technologies when there is no other way to do this.

There are some rules, set out by World Wide Web Consortium (W3C), about how ARIA should be used. They are as follows:

1. If a native HTML element or attribute already has the meaning and behaviour you need, use it. Don't take a different element and add an ARIA role, state or property to make it accessible.
2. Do not change native semantics unless you really have to.
3. All interactive ARIA controls must be usable with the keyboard.
4. Do not use `role="presentation"` or `aria-hidden="true"` on a focusable element. Using either of these on a focusable element will result in some users focusing on 'nothing'.
5. All interactive elements must have an accessible name (W3C definition).

**NOTE**: Not using ARIA is always better than using ARIA badly. Bad use of ARIA can completely mess up a website for users of assistive technology. ARIA attributes should only be used as a last resort when semantic elements can't achieve the desired result.

The note below is adapted from MDN Web Docs: ARIA:

> Note: There is a saying, "No ARIA is better than bad ARIA." In WebAIM's survey of over one million home pages, they found that home pages with ARIA present had more detected errors on average than those without ARIA. While ARIA is designed to make web pages more accessible, if used incorrectly, it can do more harm than good.

# How to support it

There are many different ARIA attributes. The two most commonly used ones are below.

## aria-label & aria-labelledby

- Adds an accessible label to elements that do not have visible text. For example, a search button with only an icon should include an `aria-label` clarifying it.
- Use it on interactive elements, like `buttons`, and on landmarks, dialogs, tables and elements with role="img". Don't use it on elements that have no role, like a plain div or span, or on text elements like p, because screen readers may ignore it there.

## aria-hidden

- Hides elements from screen readers without removing them visually.
- This is ideal for decorative or redundant elements.

ARIA roles add meaning to content. This lets assistive technologies (such as screen readers) present the content in the way users expect. These should be avoided when there is semantic HTML that can provide the same meaning (e.g. don't use `role=listitem`, and instead use an `<li>`).

# Useful links

- W3C - Rules of ARIA
- mdn web docs - ARIA
- Accessibility essentials every front end developer should know
- List of ARIA attributes
- List of ARIA roles
- What I Wish Someone Told Me When I Was Getting Into ARIA