

# Accessibility

---

## Learning Objectives

In this session you will learn:

- ☐ what accessibility is and why it is important for web developers
  - ☐ what the benefits of semantic HTML are
  - ☐ how to make code accessible
- 

## Accessibility

In Web Development it is important for us to enable as many people as possible to use our Web sites, even people with disabilities (auditory, visual, neurological, physical, etc.). There is a strong business case for adapting accessible design:

- We can reach a wider audience (older people, people with disabilities, people with slow internet connection).
  - Accessible design is important for SEO.
  - Accessible design makes websites more usable and improves overall user experience.
  - It is required by law in many countries to make your Web sites accessible.
- 

## Semantic HTML

Semantic HTML helps screen readers and other assistive technology get useful information on the structure of a Web site. It is also important for SEO and improves readability of our HTML.

- Use elements like **header**, **nav**, **main**, **article**, **aside**, **footer** to structure every page.
  - Use only one **h1** heading per page.
  - Do not skip heading levels (use **h1** once, then for the next level of headings use **h2**, for the structurally next level of headings use **h3**).
- 

## ARIA

ARIA is short for Accessible Rich Internet Applications. It is a set of attributes and roles that help make Web sites more accessible to people with disabilities.

### ARIA Roles

An ARIA Role is an attribute that can be added to an HTML element. They give semantic meaning to elements. There are six different categories of roles. Let us briefly look at two categories:

- Document structure Roles like **note** or **tooltip** describe the structure of a section of content.
- Widget Roles like **slider** or **menu** describe the type of element presented on a Web site.



A comprehensive list of Roles and their categories can be found in the [mdn ARIA Roles docs](#).

## ARIA States and Properties

ARIA states and properties refer to similar features. They provide specific information on elements, their state and relationship to other elements. Assistive technology like screen readers use them to present content to users.

💡 You can find a list of ARIA states and properties in the [mdn ARIA Attributes docs](#).

Two important ARIA attributes are:

- **aria-label**: Defines a label for an interactive element. Use it when the accessible name of an element is missing, for example when a button contains no text but only an icon:

```
<button aria-label="Close" onclick="...">
  <svg ...><path .../></svg>
</button>
```

- **aria-labelledby**: Identifies which element labels the element it is applied to. Some elements have a native way of referencing another element with its label (for example input elements and label elements which we will come to at a later point in time). If there is no native way to reference a labelling element use the aria-labelledby attribute:

```
<nav aria-labelledby="title">
  <h2 id="title">Products</h2>
  ...
</nav>
```

## Accessibility Quick Wins

- Use alt attributes on your images:

```

```

- Use high contrast colors so your content is shown clearly on all devices.
- Make sure all your interactive elements have an accessible name
- Use plain and simple language

---

## Resources

- [MDN: Accessibility](#)
- [The A11y Project: Checklist](#)
- [Web Accessibility Initiative: Introduction to Web Accessibility](#)
- [Web Accessibility for Designers](#)