

ACCESSIBLE JAVASCRIPT

Javascript and Assistive Technologies

JavaScript accessibility is important to people with the **following disabilities**:

- Blindness
- Low vision
- Motor impairments
- Cognitive impairments

Many of these groups of people **may rely on some sort of assistive technology.**

Keyboard-only

Many of these technologies are **usually controlled with the keyboard**, rather than with a mouse or pointing-device.

There are many people who are **simply
unable to use a mouse or trackpad.**

Therefore all interactive JavaScript functionality **must be accessible using keyboard-only.**

Text

These technologies represent all
information as **structured text**.

Therefore, JavaScript functionality **must take a form that can be interpreted as text.**

Accessibility principles relating to JavaScript

Accessibility principles **specific to**
JavaScript include, but are not limited to:

All non-text content has a text equivalent:

- Visual information should have a text equivalent.
- Interactive content should have a descriptive label.

All functionality is operable with the keyboard:

- Ensure all functions are able to be used using keyboard only.
- Pair mouse actions with equivalent keyboard actions.

Information and structure can be programmatically determined:

- Use standard DOM functions to create scripted content.
- Do not use scripting to emulate native functionality.

Timed activity can be controlled:

- Provide user control over time-limits.
- Provide a pause functionality for animated content.

A logical focus order is always maintained:

- Insert dynamic content into the DOM directly after its trigger.
- Add multiple interactive components in a logical order.

Functionality that relies on JavaScript is not present without it:

- Elements that only trigger scripting should themselves be added by scripting.
- Use progressive enhancement to trigger popups from normal links.