Lab 11: Accessible Custom Components & widgets

In modern web development, custom UI components are widely used, offering unique interactions and features. However, accessibility is often overlooked during their creation. Your objective is to develop accessible UI components to ensure inclusivity and usability for all users.

# Objectives

In this lab you will:

* Employ accessible patterns to identify accessibility requirements for a custom component
* Integrate ARIA and keyboard operability to make custom UI component accessibility

# Lab Submission

On Brightspace, provide the URL to your published page on Github and answer the 2 questions.

* Make sure the link goes directly to the html page where I can test the page in a browser

# Question: Building accordion component

In provided zip file: accordion.html, accordion.JS, accordion.CSS

## Instructions

* Analyze the code accordion.html
* Review the ARIA authoring practices – [Accordion pattern](https://www.w3.org/WAI/ARIA/apg/patterns/accordion/examples/accordion/)
* Identify the missing keyboard interaction and ARIA
* Fix it
* Test with screen reader before and keyboard only
* **Publish you code on Github**
* **Answer questions**
  + What is the keyboard interaction missing?

The accordion items were not focusable, causing problems for keyboard users to navigate through them with the Tab key.

* + What is the ARIA missing?

**aria-expanded**: The initial code did not have the aria-expanded attribute on the accordion buttons. This ARIA will announce whether a collapsible or expandable element (like an accordion) is expanded or collapsed when using NVDA.

**aria-hidden="true" for Icon:** The initial code did not include aria-hidden="true" on the accordion icon. It is used to indicate that an element should be ignored by screen readers, meaning it is not part of the accessibility tree.

**Aria-labelledby:** The initial code did not include aria-labelledby on the content panels. The aria-labelledby attribute links the content panel to its associated button (the accordion header).

**Aria-controls:** Each accordion button should include an aria-controls attribute that references the id of the content panel it governs, establishing a clear connection between the button and the panel.