

# Accessibility Tutorial for Developers

## Objective

- Create a website in HTML, CSS, and JavaScript and deploy it for free on Netlify
- The website should teach developers how to improve the accessibility of the sites they make
- The website should itself exemplify accessibility best practices

## Site Overview

The website should contain the following pages:

- Introduction
- Interactive tutorial pages for the following topics:
  - Text Resizing
  - Keyboard Navigation
  - Semantic HTML
  - ARIA
  - Colour Contrast
- References / Useful Links
- Link to downloadable project brief (included as a footer)

## Explanation of Page Order Logic

- By beginning with 'Text Resizing,' users can ensure that the font size is optimised for the remainder of the tutorial
- By following immediately with 'Keyboard Navigation,' users can subsequently complete the rest of the tutorial without the mouse, should they wish.
- 'Semantic HTML' is best introduced before 'ARIA' because, in general, it is better to exhaust the possibilities of semantic HTML before resorting to ARIA roles and attributes.
- 'Colour Contrast' makes for a playful, memorable ending to the tutorial - although the site will deploy good colour contrast, this section will introduce a toggle feature where users can experience a less optimal colour scheme.

## Individual Page Details

- Introduction
  - Briefly introduce the user to the concept of accessibility and explain how to use the site.
- Text Resizing
  - Briefly explain the importance of text resizing and how to implement it.
  - Include '+' and '-' buttons that allow users to dynamically change the font size of the tutorial text.
  - Users should be able to activate the buttons through the keyboard.
  - Ensure that the remainder of the page responds such that changes in font size do not cause the page to look messy.
- Keyboard Navigation
  - Briefly explain how to navigate with a keyboard and how to ensure this is a feature of any site.
  - Include a short video demonstration so that mobile users (who cannot access a keyboard) can witness the keyboard navigation in action.
  - Set a specific challenge that will allow the user to test their skills (eg, "See if you can navigate to the next page of this tutorial by using the keyboard").
- Semantic HTML
  - Briefly explain the concept of HTML with examples.
  - Include a 'multiple-choice' question which tests the user's knowledge of semantic HTML tags.
- ARIA
  - Briefly explain ARIA, how it differs from semantic HTML, and when to use it.

- Include a 'fill-in-the-blank' task which tests the user's knowledge of ARIA roles and attributes.
- Colour Contrast
  - Briefly explain the importance of colour contrast.
  - Include a toggle feature where users can experience a different (less optimal) contrast.
- References / Useful Links
  - For each topic, provide an online reference and/or at least one useful link for further learning.

## **Timeline**

- 10/11/12
  - Begin writing the text content for each page
- 11/01/24
  - Finish writing the text content for each page
  - Create video demonstration for 'Keyboard Navigation'
  - Complete wireframe on Figma
  - Collect references and useful links
- 12/01/24
  - Write the code for the website, testing on the local server
  - Test the accessibility using <https://www.accessibilitychecker.org/>
- 13/01/24
  - Deploy the site on Netlify

Anthony Fort  
10/01/24