



Commonwealth of Massachusetts

Executive Office of Technology Services and Security (EOTSS)

Accessibility Center for Education, Consultation, and Support Services (ACCESS)

Web and App Accessibility Testing Checklist

What are Accessibility Checks?

Accessibility “checks” are a series of simple manual tests that examines digital content against the Web Content Accessibility Guidelines (WCAG) version 2.1 level A and AA. Each manual accessibility check is phrased as a yes/no question; a “Yes” response means the page or content passes the check, while a “No” response means an accessibility violation is likely present and needs to be remediated. Be sure to check all pages in the site or application.

What Accessibility Checks should I perform?

The following accessibility checks are categorized by the level of skill required to perform the check. Level 1 checks are the simplest checks while levels 4-6 checks require more advanced skills and a deeper understanding of WCAG. To ensure that each check is performed correctly, Commonwealth employees should be sure to view the video resources prior to performing the accessibility checks.

Each level requires developers and testers to develop their accessibility skillset by adding more manual testing and validation. Each level includes and builds off of the items in the preceding levels (e.g., Level 2 encompasses the items in Levels 1 and 2, Level 4 encompasses the items in Levels 1, 2, 3, and 4 and so on).

- Level 1: Run automated tests and fix any violations found
- Level 2: Test with bookmarklets and contrast tool and fix any violations found
- Level 3: Manual checks of visible content and fix any violations found
- Level 4: Test with keyboard and fix any violations found
- Level 5: Test with operating system and browser settings and fix any violations found
- Level 6: Test with assistive technologies and mobile devices and fix any violations found

Note: Accompanying how-to resources will also be provided for each of these sections except for section six, please check the websites and applications accessibility testing page to explore this content.

Toolkit to install

Recommended automated tools. ANDI is a robust automated tool specific for developers but anybody can install and use. WAVE and AXE automated tools provide a listing of violations but may not be as comprehensive as the ANDI tool. It is required that web pages be tested using an automated tool with all violations fixed. Please use the tool(s) that work best for you.

- [ANDI](#) to be used by developers
- [WAVE](#) browser extension
- [AXE](#) Chrome extension

Secondary tools you can use include:

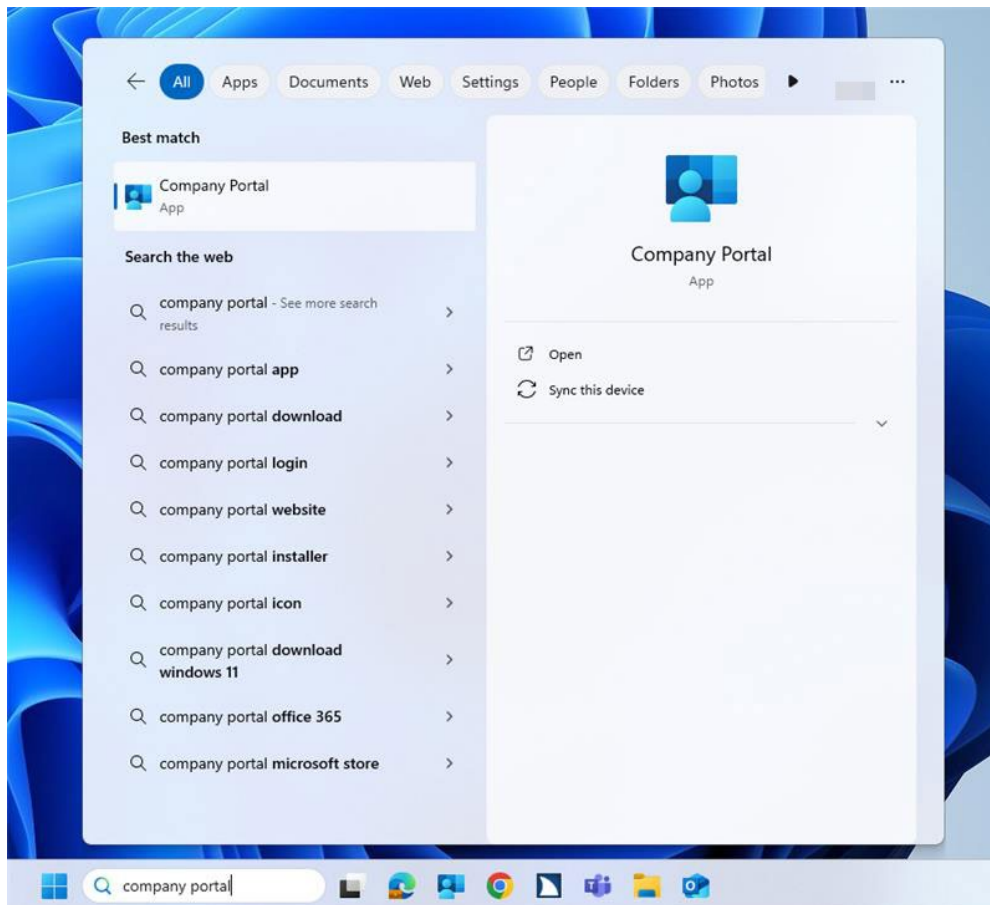
- [Lighthouse](#) Chrome extension
- [Arc Toolkit](#) Chrome extension

Install these bookmarklets to quickly perform accessibility checks for specific page elements.

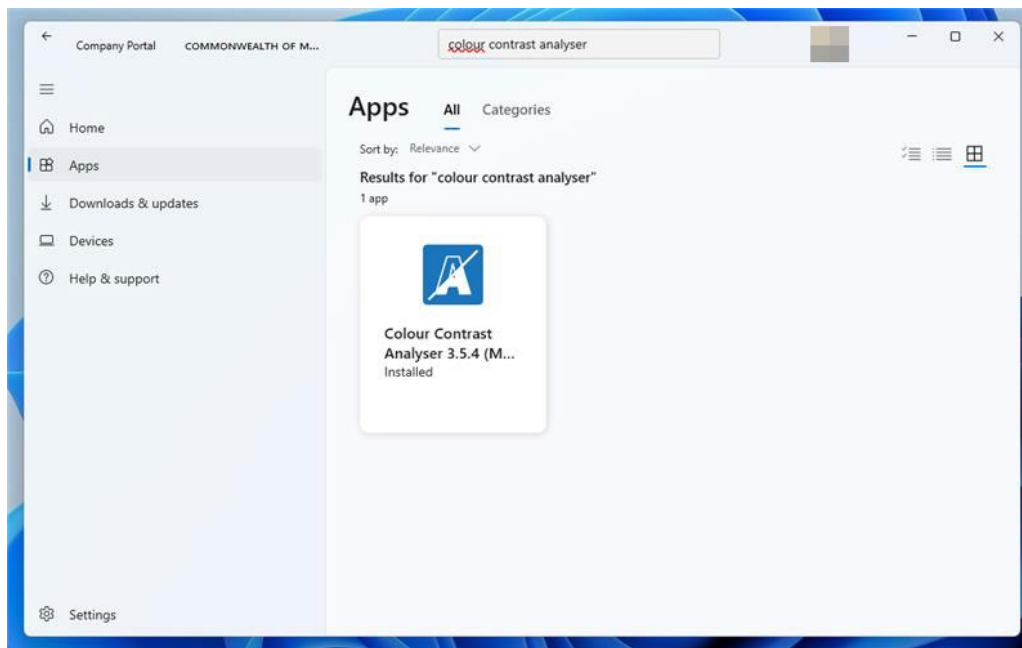
- [DigitA11Y Tablets](#) (Chrome extension) to check headings, images, language, keyboard focus, links, text spacing, ARIA, autocomplete and forms.
- [Page Title Check Bookmarklet](#)
- [Check Skip Link Bookmarklet](#)
- [Tables Bookmarklet](#)
- [Lists Bookmarklet](#)
- [Color contrast bookmarklet](#) simple tool
- More robust contrast analyzer: Colour Contrast Analyser, spelled in British English (install from the Company Portal)

Install Colour Contrast Analyser

Step 1: Go to the **Company Portal** from the Windows menu on your computer.



Step 2: Search for **Colour Contrast Analyser**, then select the app to install.



Level 1: Test with automated tools

Required to check

- ☐ Run an automated testing tool selected from the automated tools recommendations list on your pages and fix any identified violations. Please reference additional content testing resources for Mass.gov in this section.

Note: Automated testing can only catch 25%-30% of violations. Manual testing checks are also required.

- ① [How to test with the WAVE testing tool](#)
- ① [Instructions for how to use the WAVE testing tool](#)
- ① [How to test with the Axe testing tool](#)
- ① [How to test with the ARC toolkit tool](#)
- ① [Lighthouse for Chrome instructions](#)

Apps and web pages external to Mass.gov

- ☐ Was an automated checker tool used for applications and web pages? (select a tool from the list of recommended tools)

Mass.gov

- ☐ Was the Editoria11y automated tool used to check Mass.gov content before publishing?

Accessibility Checks for Mass.gov can be found in the [Mass.gov Knowledge Base](#).

- [Creating accessible content](#)
- [Setting language for content to improve accessibility](#)
- [Live accessibility checking: Using Editoria11y on your Mass.gov pages](#)

- ☐ Were any violations found fixed?

Level 2: Test with bookmarklets and contrast tool

Required to check

The level 2 accessibility checks can be performed using accessibility bookmarklets. Add each Bookmarklet to your browsers bookmark bar. Install the bookmarklets using the links provided in the toolkit section.

Note: Can also use the web developer code inspector browser tool by right clicking or pressing shift plus F10 on any page and selecting inspect to review the code for each of these checks. Manual testing will need to be performed to validate bookmarklet results.

Check using [headings bookmarklet](#) or by using code inspection.

- ☐ Is there only one Heading 1 present?
- ☐ Are all headings correctly ordered and nested properly?
- ① [Learn about the basics of accessibility testing for headings](#)

Check using [images bookmarklet](#) or by using code inspection.

- ☐ Do all images have meaningful alt text and images irrelevant to the content set as decorative?
- ① [Learn about types of images for accessibility](#)
- ① [Learn about the basics of accessibility testing for images \(alternative text\)](#)

Check using [lists bookmarklet](#).

- ☐ Are all lists that are ordered contain numbers or unordered contain bullets with no nested list items?
- ① [Learn about the basics of accessibility testing for lists](#)

Check using [tables bookmarklet](#) or by using code inspection.

- ☐ Do all tables have a header row and/or a header column?
- ① [Learn about the basics of accessibility testing for tables](#)

Check using [page title bookmarklet](#) or by using code inspection (title tag)

- ☐ Do all pages have a page title that is descriptive located in the document head?
- ① [Learn about the basics of accessibility testing for page titles](#)

Check using [language bookmarklet](#) or by using code inspection

- ☐ Do all pages have the correct language tagged in the document head?

① [Learn about the basics of accessibility testing for headings](#)

Check using contrast analyzer tool or [color contrast bookmarklet](#) (note that the recommended automated scanning tools also contain color contrast checkers)

- ☐ Does the color contrast between text and background meet 4.5:1 for text from 12 pts-17 pts?
- ☐ Does the color contrast between text and background meet 3:1 for text 18 pts or larger?
- ☐ Does the color contrast between graphics and background and within the graphics meet 3:1?

① [How to use color contrast checker tools including Colour Contrast Analyser](#)

① [Accessible color contrast requirements](#)

Check using the [text spacing toggle](#) on [DigitA11y Tubelets](#).

- ☐ Does the content respond to text spacing with the bookmarklet enabled?
- ☐ Is any content cut off, truncated, overlapping, or missing?

Level 3: Visible content checks

Required to check

Performing visible content checks allows for verification of accuracy for page elements to supplement and go beyond any violations discovered by an automated tool. The following manual accessibility checks can be performed using visual inspection of page content and code to locate and fix violations.

Headings

- ☐ Are headings clear and descriptive?

① [Learn about the basics of accessibility testing for headings](#)

Fonts and readability

- ☐ Is plain language used, avoiding jargon, and spelling out abbreviations the first time they're used?
- ☐ Is directional or sensory language avoided? (e.g., “the image to the right” or “the blue icon”)?
- ☐ Are emojis or other non-standard symbols avoided for bulleted and numbered lists?
- ☐ Is the page title in the HTML document head descriptive and corresponds with the heading one topic?

① [How to format text content for accessibility](#)

Links

- ☐ Is all hyperlink text unique and descriptive?

① [Learn about the basics of accessibility testing for links](#)

Color and meaning

- ☐ Is the use of color alone to indicate meaning avoided?

Images

- ☐ Do all images have meaningful alt text and images irrelevant to the content set as decorative?
- ☐ Are images of text avoided?
- ☐ Are text alternatives provided for complex charts and graphs?

① [Learn about types of images for accessibility](#)

- ① [Learn about the basics of accessibility testing for images \(alternative text\)](#)

Tables

- ☐ Are tables avoided for visual formatting of content?
- ☐ Does each table have a descriptive caption or associated title?
- ☐ Are all tables simple with no use of merged, split, nested cells or multiple column header rows?

- ① [Learn about the basics of accessibility testing for tables](#)

Forms

- ☐ Are clear completion and submission instructions including identification of required fields for forms included at the top of every form?
- ☐ Do all form fields have appropriate visible labels?
- ☐ Do required form fields have a * or the word required to indicate their required state?
- ☐ Are form errors clear and descriptive letting the user know how to fix a problem?
- ☐ Can form fields be completed using the browser autocomplete feature (if applicable)?

- ① [Learn about the basics of accessibility testing for forms](#)

Media including audio and video

- ☐ Are flashing or blinking elements avoided?
- ☐ Is sound autoplay disabled on page load for videos and audio content?
- ☐ Does all [embedded or linked multimedia](#) contain closed captions, audio description when necessary, and a transcript for video and audio content?

- ① [Learn about the basics of accessibility testing for video captions](#)

- ① [Learn about the basics of accessibility testing for video transcripts](#)

Level 4: Test with keyboard

Required to check

Keyboard testing is vital to ensure that keyboard only users and assistive technology users can navigate and interact with digital content.

Note: The 2 recommended bookmarklets provided to assist with keyboard testing should not be the only method of testing. Testing using the keyboard will provide a more accurate check to find violations.

Note: Testing keyboard navigation, focus visibility and element interaction must also be performed when the browser is enlarged to 200% and 400%.

Note: Testing keyboard navigation, focus visibility and element interaction must also be performed when switching to Windows high contrast dark theme.

Check using the [Skip Link Bookmarklet](#).

- ☐ Is there a skip to main content link at the top of the page?
- ☐ Does pressing enter on the skip to content link place focus in the main body of the page?

① [Learn about what a skip to content link is and how to test](#)

Check using the [Visible Keyboard Focus Check Bookmarklet](#).

Press tab to move forward and shift tab to move backwards tracking the visible outline

- ☐ Is keyboard focus visible with an outline around the element such as a link, button or form field?
- ☐ Do focusable elements receive focus in a meaningful order left to right, top to bottom?
- ☐ Are all interactive components actionable using just a keyboard?

① [How to test with only the keyboard](#)

Keyboard interaction instructions

- Space/enter to activate buttons
- Enter to activate links, menu items, combo box selection
- Space to activate checkboxes
- Arrows to navigate combo boxes or dropdowns, menus, and select radio buttons
- Type for text fields
- Escape to close modals, menus, popups

Level 5: Test with operating system and browser settings

Reserved for and strongly recommended for an accessibility subject matter expert to test.

Switch operating system [contrast theme setting](#) on Windows to dark theme.

- ☐ Is the text and image content clear and visible in high contrast mode or dark theme?

① [How to switch from standard theme to high contrast theme and perform the accessibility test](#)

To increase the text size using Chrome and Edge browsers, press the control and plus key until 200% is reached, press the control plus minus key to decrease text size. Press control plus the zero key to return to the default text size.

To increase text size in Safari browser on Mac, press command plus the plus key and command plus minus to decrease the size.

Enlarge the text to 200% in the browser (check using multiple browsers)

- ☐ Is all content responsive and visible at 200% browser enlargement with nothing cut off, truncated, overlapping, or missing?

Set the browser viewport to 1280 and enlarge browser window to 400%

- ☐ Does content respond by reflowing into a single-column layout when enlarged to 400%?
- ☐ Is all content visible at 400% browser enlargement with nothing cut off, truncated, overlapping, or missing?

① [How to test using zoom for 200% text enlargement and 400% browser enlargement](#)

Level 6: Test with additional assistive technologies and mobile devices

Reserved for and strongly recommended for an accessibility subject matter expert to test

- ☐ Can all content be interacted with using dictation software?
- ☐ Can content be recognized and interacted with using keyboard and screen reader plus Braille display?
- ☐ Can all content be interacted with using a switch device?
- ☐ Can all content be interacted with using eye tracking software?
- ☐ Does content respond with appropriate sizing, orientation changes and visibility when viewed on a mobile device?

Document Change Control

Version No.	Revised by	Effective Date	Description of Changes
1.0	ACCESS Team	August 1, 2025	Initial document