

MADCAP FLARE DESKTOP 2025 r2

Accessibility

Copyright © 2025 MadCap Software. All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of MadCap Software.

MadCap Software
1660 17th Street, Suite 201
Denver, Colorado 80202
858-320-0387
www.madcapsoftware.com

THIS PDF WAS CREATED USING MADCAP FLARE.

CONTENTS

CHAPTER 1

Introduction	5
Section 508 and WCAG	6
Accessibility General Considerations	7
Accessibility Features Per Output Type	8

CHAPTER 2

Accessibility and Content	10
Alt Text and Title Attributes	11
Applying Alt Text	14
Applying Empty Alt Text to Images	21
Accessibility and Tables	23

CHAPTER 3

Accessibility and the User Interface	28
Accessibility and Keyboard Navigation	29
Accessibility and ARIA Tags	33
Accessibility and Skip to Main Content	38
Accessibility and Search Results	40
Accessibility and Scrolling	41
Accessibility and User Interface Text	44

CHAPTER 4

Accessibility and Analysis	46
Viewing Accessibility Suggestions	47
Customizing Accessibility Suggestions Shown	49
Resolving Accessibility Issues	50
Accessibility and Warnings	52

CHAPTER 5

Accessibility and Outputs	55
Accessibility and PDF Output	56
Accessibility and Eclipse Help Output	63

APPENDIX

PDFs	64
Tutorials	64
Cheat Sheets	65
User Guides	66

CHAPTER 1

Introduction

Supported In:



With Flare Desktop, you can make your documentation accessible to people with disabilities (i.e., individuals who have visual, hearing, and mobility impairments). Documentation is considered accessible when it is easier for those with disabilities to use and easier to read using assistive technologies such as Microsoft Windows Narrator, Connect Outloud, and Job Access With Speech (JAWS).

This chapter discusses the following:

Section 508 and WCAG	6
Accessibility General Considerations	7
Accessibility Features Per Output Type	8

I Section 508 and WCAG

Following are the two primary web accessibility standards:

Section 508

Section 508 In 1973 the United States government passed the Rehabilitation Act. Part of this Act is Section 508, which was amended in 1998 and requires Federal agencies to follow standards in developing, maintaining, acquiring, or using electronic and information technology to make systems accessible to people with disabilities.

See <https://www.section508.gov>.

Web Content Accessibility Guidelines (WCAG)

Web Content Accessibility Guidelines (WCAG) These guidelines are created by the Web Accessibility Initiative (WAI), as part of the World Wide Web Consortium (W3C), in an effort to make the web accessible to people with disabilities.

See <http://www.w3.org/WAI/intro/wcag.php>.

I Accessibility General Considerations

When working to make your output accessible, consider the following:

- Different output formats in Flare Desktop support different accessibility features.
- One way to make content accessible is to provide one output for most users and an entirely separate output for individuals with disabilities. In the locations where you have elements that can present accessibility issues (e.g., images, multimedia), you can provide descriptions of those elements. Then by using condition tags and different targets, you can have customized outputs for each set of users.
- You can use character-level styles when formatting text. For example, the `<strong>` tag conveys the idea that you want to emphasize the importance of some words in a paragraph.

Accessibility Features Per Output Type

Following are accessibility features (e.g., Section 508, WCAG) supported in each output type.

	HTML5	PDF	Word	Clean XHTML	Eclipse Help	EPUB	HTML Help	PPTX
Accessibility Supported	✓	✓	✗	✓	✓	✓	✓	✗
Alt Text and Title Attributes	✓	✓	✗	✓	✓	✗	✓	✗
ARIA Tags	✓	✗	✗	✗	✗	✗	✗	✗
Compiler Warnings	✓	✓	✗	✓	✓	✗	✓	✗
Empty Alt Text for Images	✓	✓	✗	✓	✓	✗	✓	✗
Keyboard Navigation	✓	✗	✗	✗	✗	✗	✗	✗
Scrolling for Toolbars and Navigation Frames (Enable)	✗	✗	✗	✗	✗	✗	✗	✗

	HTML5	PDF	Word	Clean XHTML	Eclipse Help	EPUB	HTML Help	PPTX
Search Results	✓	✗	✗	✗	✗	✗	✗	✗
Skip to Main Content	✓	✗	✗	✗	✗	✗	✗	✗
User Interface Text	✓	✗	✗	✗	✗	✗	✗	✗

CHAPTER 2

Accessibility and Content

There are multiple ways to adjust content to make it more accessible to end users.

This chapter discusses the following:

- Alt Text and Title Attributes 11
- Applying Alt Text 14
- Applying Empty Alt Text to Images 21
- Accessibility and Tables 23

I Alt Text and Title Attributes

Supported In:



You can accomplish much of the work of making your content accessible by setting **alt** and **title** attributes on various elements—such as links, images, forms, and multimedia. These two attributes, although similar, are not identical.

- **Alt** An alt attribute is intended to provide *alternative* information for elements that cannot be seen (e.g., visually impaired individuals cannot see images in your output). The alt attribute text should briefly describe what the element is about.
- **Title** A title attribute is created when you add screen tip (or tool tip) text for an element. The title attribute text should provide *additional* information about the element.

Table of Alt and Title Elements

Following are the different types of elements for which you can set alt and/or title (screen/tooltip) attributes.

Element	Alt Attribute (Alternate Text)	Title Attribute (Screen Tip)
Audio (Links and Embedded)	✗	✓
Concept Links	✓	✗
Controls (e.g., Buttons, Labels, Input Boxes)	✓	✓
Cross-References ¹	✓	✓
Drop-Down Text	✓	✗
Equations	✓	✓
Expanding Text	✓	✗
Glossary Term Links	✓	✗
Image Hyperlinks	✓	✓
Image Maps	✓	✓
Images	✓	✓

Element	Alt Attribute (Alternate Text)	Title Attribute (Screen Tip)
Keyword Links	✓	✗
Movies (Links and Embedded)	✗	✓
QR Codes	✓	✓
Related Topics Links	✓	✗
Table of Contents (TOC) Icons	✓	✗
Text Hyperlinks ¹	✓	✓
Togglers	✓	✗
Toolbar Items	✓	✓
Topic Popups	✗	✓

¹When you add a screen tip a <title> tag is created in the markup.

I Applying Alt Text

Supported In:



Alternate "alt" text provides a textual option for non-text elements on a web page, which is required as part of Section 508 compliance. See "Section 508 and WCAG" on page 6.

In Flare Desktop, you can apply alt text to many elements, including concept links, drop-down text, expanding text, images, related topics, and more.

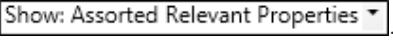
How to Apply Alt Text Through the Language Skin

1. Add a language skin, or open one if it is already created on your project.
2. In the upper-right corner of the skin under **Language Skin Targets**, select **Print and Online**.
3. Under the **Print and Online** category in your skin, scroll down to the section with the **Formats**. You may need to sort the **Name** column to list the Formats together.
4. In the **Value** column, type the text of the property you wish to modify.

Name	Property	Value
Dialog.Reply Comment	SubmitButtonLabel	
Dialog.Reply Comment	TitleLabel	
Dialog.Reply Comment	UserNameLabel	
Formats	BreadcrumbsYouAreHereText	You are here:
Formats	ConceptLinkIconAltText	Concept Link Icon
Formats	ConceptLinkText	See Also
Formats	CrossReferenceAbove	above
Formats	CrossReferenceBelow	below
Formats	CrossReferenceFormat	{paratext}
Formats	CrossReferenceOnFacingPage	on the following page
Formats	CrossReferenceOnNextPage	on the next page
Formats	CrossReferenceOnPage	on page
Formats	CrossReferenceOnPreviousPage	on the previous page
Formats	CrossReferencePrintFormat	"{paratext}" (pageref)
Formats	DropDownClosedAltText	Closed
Formats	DropDownOpenAltText	Open
Formats	ExpandingClosedAltText	Closed
Formats	ExpandingOpenAltText	Open
Formats	GlossaryTermClosedAltText	Closed
Formats	GlossaryTermOpenAltText	Open
Formats	KeywordLinkIconAltText	Keyword Link Icon
Formats	KeywordLinkText	Search Index
Formats	RelatedTopicsIconAltText	Related Topics Link Icon
Formats	RelatedTopicsText	Related Topics
Formats	TogglerClosedAltText	Closed
Formats	TogglerOpenAltText	Open
Frame.Body Comments	Label	Comments

5. Click  to save your work.

How to Apply Alt Text Through the Stylesheet

1. Open your stylesheet.
2. In the local toolbar, make sure the first button displays  **View: Advanced**. If the button displays  **View: Simplified** instead, then click it.
3. In the upper-left of the editor, make sure the drop-down field is set to .
4. Scroll down to the section with the MadCap styles. You can modify the alt text for the following:
 - MadCap|conceptLink
 - MadCap|dropDown
 - MadCap|expanding
 - MadCap|glossaryTerm
 - MadCap|keywordLink
 - MadCap|relatedTopics
 - MadCap|toggler
5. From the **Show** drop-down list on the upper-right side of the editor, select . The most relevant properties for that style are shown on the right side of the editor.
6. (Optional) You can use the toggle button in the local toolbar to show properties below in a group view  or an alphabetical view .
7. If you are using the group view, expand the **Unclassified** group.

8. Enter the alt text for each property. You can modify the following properties:

■ **mc-closed-image-alt-text**

This enables alt text when the drop-down icon is closed. This property applies to the following:

- MadCap|dropDown
- MadCap|expanding
- MadCap|glossaryTerm
- MadCap|toggler

■ **mc-open-image-alt-text**

This enables alt text when the drop-down icon is open. This property applies to the following:

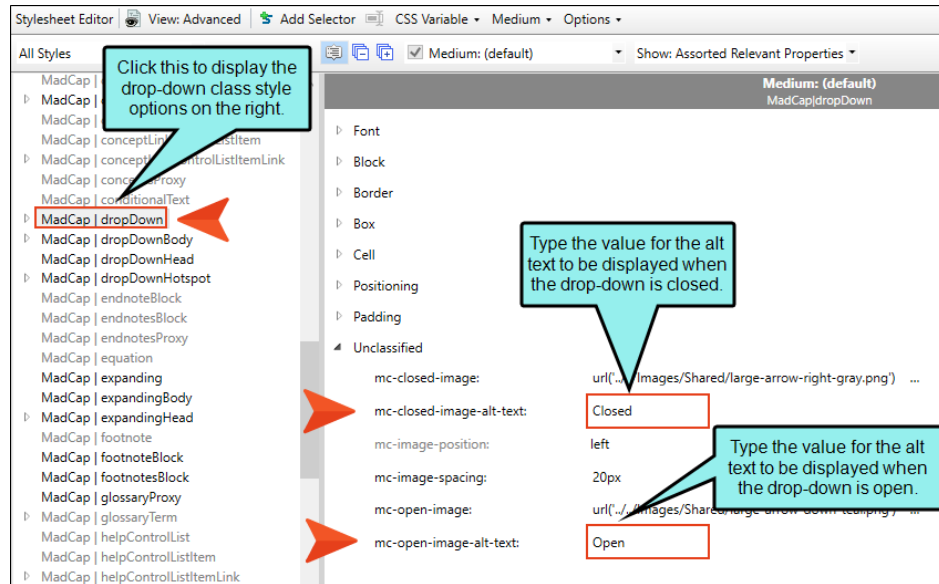
- MadCap|dropDown
- MadCap|expanding
- MadCap|glossaryTerm
- MadCap|toggler

■ **mc-image-alt-text**

This enables alt text on link icons. This property applies to the following:

- MadCap|conceptLink
- MadCap|keywordLink
- MadCap|relatedTopics

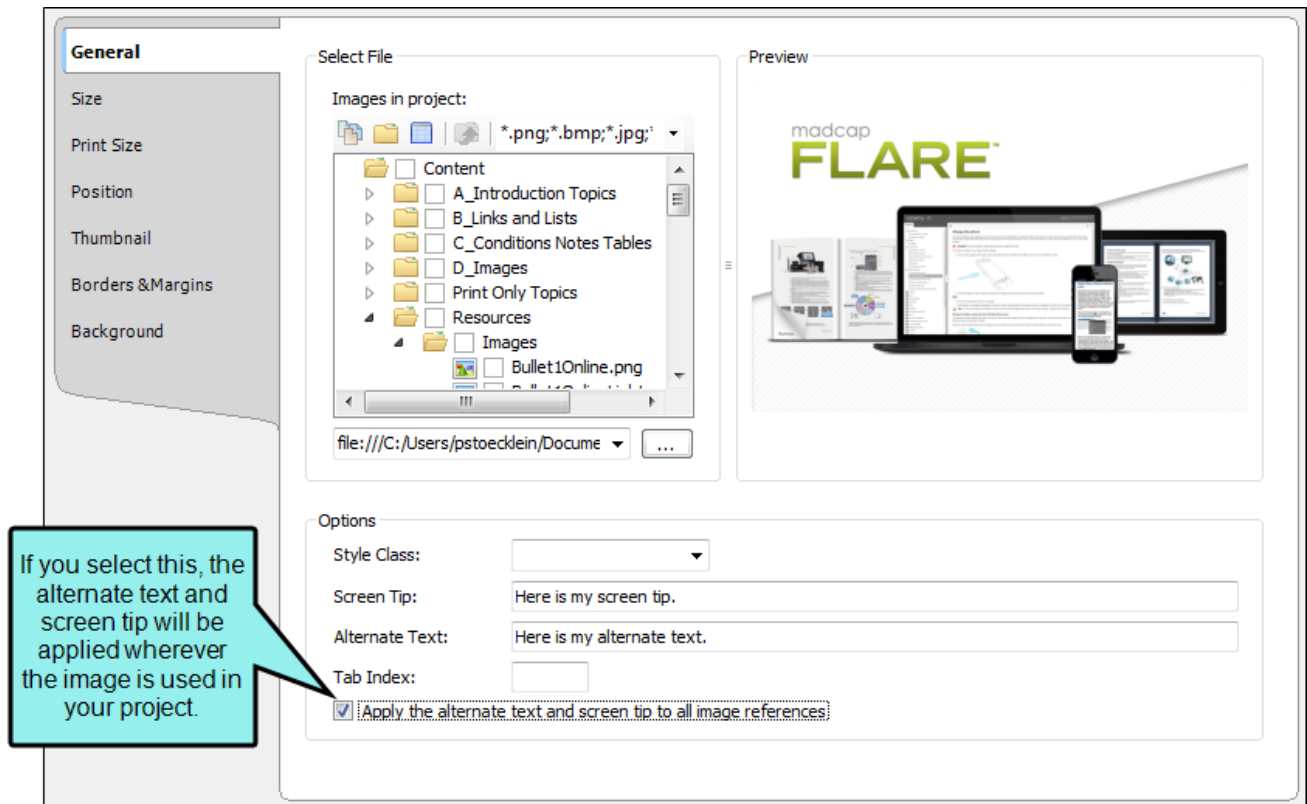
☆ **EXAMPLE** You want to modify the alt text for the drop-downs in your project. So you modify the alt text when the drop-down is open, and when it is closed.



9. Click  to save your work.

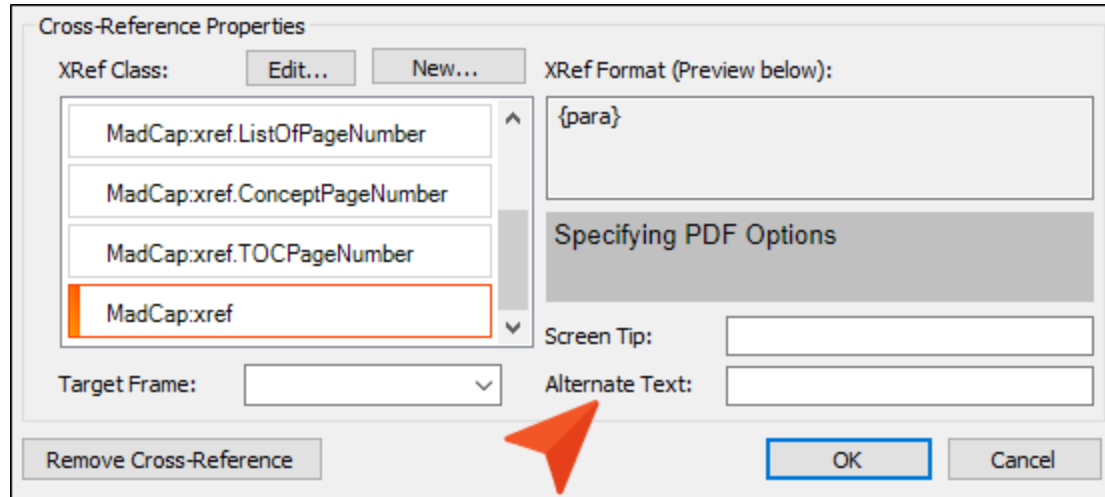
Applying Alt Text and Screen Tips to All Image References

When you insert or edit an image, you can apply alternate text or screen tips to one image at a time, or to every instance of the image throughout the project. In Image Properties, you will see a check box that lets you apply the same alternate text and screen tip to all image references.



Applying Alt Text and Screen Tips to Links

When you insert or edit a link (e.g., cross-reference, or hyperlink), you can apply alternate text or screen tips. If you open the link dialog, enter text in the appropriate field to alter its properties. See "Accessibility and PDF Output" on page 56.



What's Noteworthy?

 **NOTE** If there are alt text settings for any elements in both the stylesheet and the language skin, the stylesheet settings take precedence.

Applying Empty Alt Text to Images

Supported In:



By default, a new image does not contain an "alt" (alternate text) attribute when you insert it into a topic or snippet, unless you specifically add alt text in the Insert Image dialog. However, alt attributes on images are required for Section 508 compliance, even if the string is empty or null (alt=""). Also, by adding empty alt attributes to images, you can search the files to find the images without alternate text and then add it as needed.

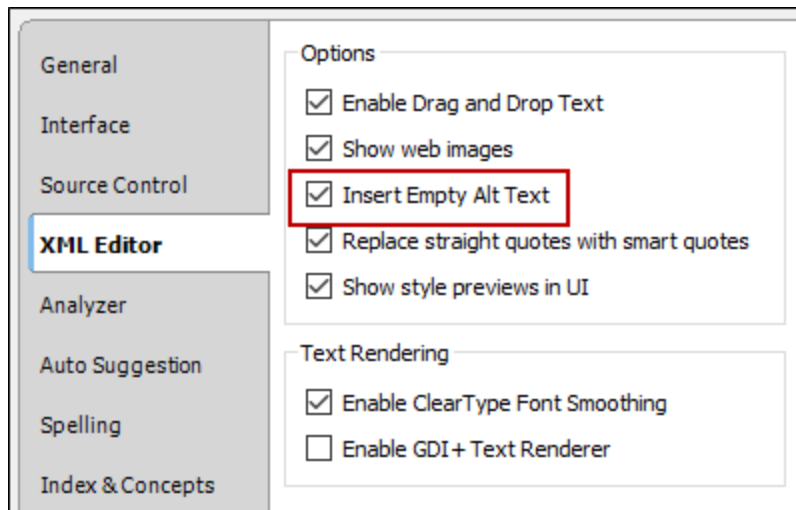
There are a couple of ways to quickly add empty alt tags to images.

How to Apply Empty Alt Text When Output is Generated

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Use empty ALT text for images that do not have ALT text**.
3. Click  to save your work.

How to Apply Empty Alt Text When Images are Inserted

1. Select **File > Options**. The Options dialog opens.
2. Select the **XML Editor** tab.
3. Select the **Insert Empty Alt Text** check box. A check mark indicates the feature is enabled.



4. Click **OK**.

Accessibility and Tables

You can make tables more accessible by performing certain tasks. Also, support for Accessible Rich Internet Applications (ARIA) tags helps make tables more accessible.

Accessibility Tasks

You can improve the accessibility of tables by including the following to each one:

- Header row
- Captions
- Summary

Table Properties

General

Table Size

Number of columns: 2

Number of rows: 3

Number of header rows: 1

Number of footer rows: 0

Table Caption

Text: My Table Caption

Side: (default)

Repeat: (default)

Continuation: (default)

Summary

Summary of my table

AutoFit Behavior

☐ AutoFit to contents

☒ AutoFit to window 100%

☐ Fixed column width: (not set)

Align

Left

Table Style

☒ Table Style: Alternate-Row-Color - C:\Users\pstoer

☐ Style Class:

Text to Table

☒ None ☐ Commas ☐ Other:

☐ Paragraphs ☐ Tabs

Preview

My Table Caption

ARIA Tags

For HTML5 output, Flare Desktop adds ARIA attributes to new tables automatically. This helps make tables more accessible in HTML5 output by letting users know where the cursor is located in the table.

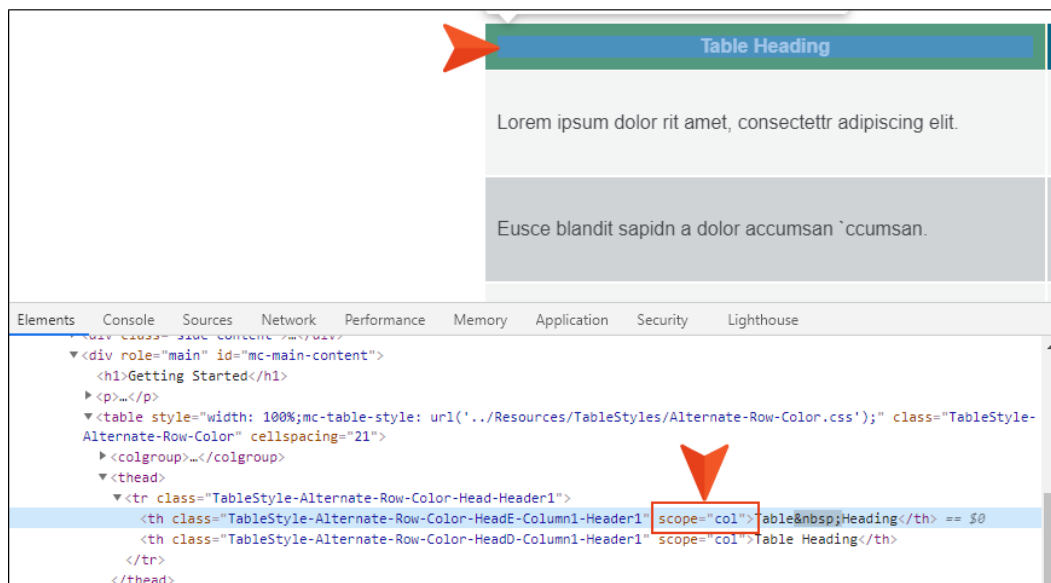
Depending on how you structure the table or table stylesheet, the following attributes will be included in the code:

- `scope="col"`
 - This displays when the content in a `<th>` tag is within a `<thead>` tag.



Table Heading	Table Heading
Lorem ipsum dolor rit amet, consectetur adipiscing elit.	Dondc sagittis elementsum arcu, at gravida puam mollis id.
Eusce blandit sapidn a dolor accumsan `ccumsan.	Suspdndisse lectus augte, auctor in aliqua nec, fringilla id dnlor.
Integer pukvinar lacus libern, eget volutpat enil finibus non.	Aliqu`m erat volutpat.

- It tells the user that the content covers the scope of the current column.



- `scope="row"`

- This displays when the row content is within a header column. You can create this scenario by changing the `<td>` tag in a row to a `<th>` tag.

Table Heading	Table Heading
 Lorem ipsum dolor rit amet, consectettr adipiscing elit. 	Dondc sagittis elemensum arcu, at gravida puam mollis id.
Eusce blandit sapidn a dolor accumsan `ccumsan.	Suspndisse lectus augte, auctor in aliqua nec, fringilla id dnlor.
Integer pukvinar lacus libern, eget volutpat enil finibus non.	Aliqu`m erat volutpat.

- It tells the user that the content covers the scope of the current row.

p.table-text 443 × 26

Table Heading

Lorem ipsum dolor rit amet, consectettr adipiscing elit.

Eusce blandit sapidn a dolor accumsan `ccumsan.

Console Sources Network Performance Memory Application Security Lighthouse

```

<h1>Getting Started</h1>
<p>...</p>
<table style="width: 100%;mc-table-style: url('../Resources/TableStyles/Alternate-Row-Color.css');" class="TableStyle-Alternate-Row-Color" cellspacing="21">
  <colgroup>...</colgroup>
  <thead>...</thead>
  <tbody>
    <tr class="TableStyle-Alternate-Row-Color-Body-Body1">
      <th class="TableStyle-Alternate-Row-Color-BodyE-Column1-Body1" scope="row">
        <p class="table-text">Lorem ipsum dolor rit amet, consectettr adipiscing elit.</p>
      </th>
      <td class="TableStyle-Alternate-Row-Color-BodyD-Column1-Body1">...</td>
    </tr>
    <tr class="TableStyle-Alternate-Row-Color-Body-Body2">...</tr>
    <tr class="TableStyle-Alternate-Row-Color-Body-Body1">...</tr>
  </tbody>
</table>
    
```

■ role="rowheader"

- This displays when the content is in a column header, as specified by the Column Type (Header) in a table stylesheet.

TableStyle Editor | Medium: (default) | Apply Style...

General

Rows

Columns

Header

Footer

Column Styles

	Name	Type	Repeat	
▶	Column1	Header	1 column	✕
	Column2	Pattern	1 column	↑ ↓

Column Style Properties

Width: (not set)

Font: (not set)

Ruler: (not set)

Separator: Solid 2px #ffffff

Background

Color:

Image: (default) ...

Repeat: (default)

X: (default)

Preview

Column1	Column2
Body1	Body
Body2	Body
Body1	Body
Body2	Body
Footer1	Footer

- It tells the user that the content covers the scope of the current row (when it is in a <td> tag rather than a <th> tag).

The screenshot displays a web browser window with a table. The table has a header row with two 'Table Heading' cells. Below the header are three data rows with alternating background colors (light blue, light gray, and light blue). The first data row contains the text 'Lorem ipsum dolor rit amet, consectettr adipiscing elit.' in the first column. A red arrow points to this cell. Below the browser window, the developer console shows the HTML structure. The first data row is expanded, showing the first cell's HTML: `<td class="TableStyle-Alternate-Row-Color-BodyE-Column1-Body1" role="rowheader">...</td>`. A red arrow points to the `role="rowheader"` attribute.

Table Heading	Table Heading	Table Heading	Table Heading
Lorem ipsum dolor rit amet, consectettr adipiscing elit.	Dondc sagittis elemensum arcu, at gravida puam mollis id.	Dondc sagittis elemensum arcu, at gravida puam mollis id.	Dondc sagittis elemensum arcu, at gravida puam mollis id.
Eusce blandit sapidn a dolor accumsan `ccumsan.	Suspdndisse lectus augte, auctor in aliquat nec, fringilla id dnlor.	Suspdndisse lectus augte, auctor in aliquat nec, fringilla id dnlor.	Suspdndisse lectus augte, auctor in aliquat nec, fringilla id dnlor.
Integer pukvinar lacus libern,	Aliqu`m erat volutpat.	Aliqu`m erat volutpat.	Aliqu`m erat volutpat.

```

<table style="width: 100%;mc-table-style: url('../Resources/TableStyles/Alternate-Row-Color.css');
" class="TableStyle-Alternate-Row-Color" cellspacing="21">
  <colgroup>...</colgroup>
  <thead>...</thead>
  <tbody>
    <tr class="TableStyle-Alternate-Row-Color-Body-Body1">
      <td class="TableStyle-Alternate-Row-Color-BodyE-Column1-Body1" role="rowheader">
        <p class="table-text">Lorem ipsum dolor rit amet, consectettr adipiscing elit.</p>
      </td>
      <td class="TableStyle-Alternate-Row-Color-BodyE-Column2-Body1">...</td>
      <td class="TableStyle-Alternate-Row-Color-BodyE-Column2-Body1">...</td>
      <td class="TableStyle-Alternate-Row-Color-BodyD-Column2-Body1">...</td>
    </tr>
    <tr class="TableStyle-Alternate-Row-Color-Body-Body1">...</tr>
    <tr class="TableStyle-Alternate-Row-Color-Body-Body1">...</tr>
  </tbody>
</table>

```



NOTE If you have already added these kinds of attributes to existing tables by using the Internal Text Editor, Flare Desktop will not overwrite them.

Accessibility and the User Interface

The user interface produced in online output is designed to improve accessibility.

This chapter discusses the following:

- Accessibility and Keyboard Navigation29
- Accessibility and ARIA Tags33
- Accessibility and Skip to Main Content38
- Accessibility and Search Results 40
- Accessibility and Scrolling41
- Accessibility and User Interface Text44

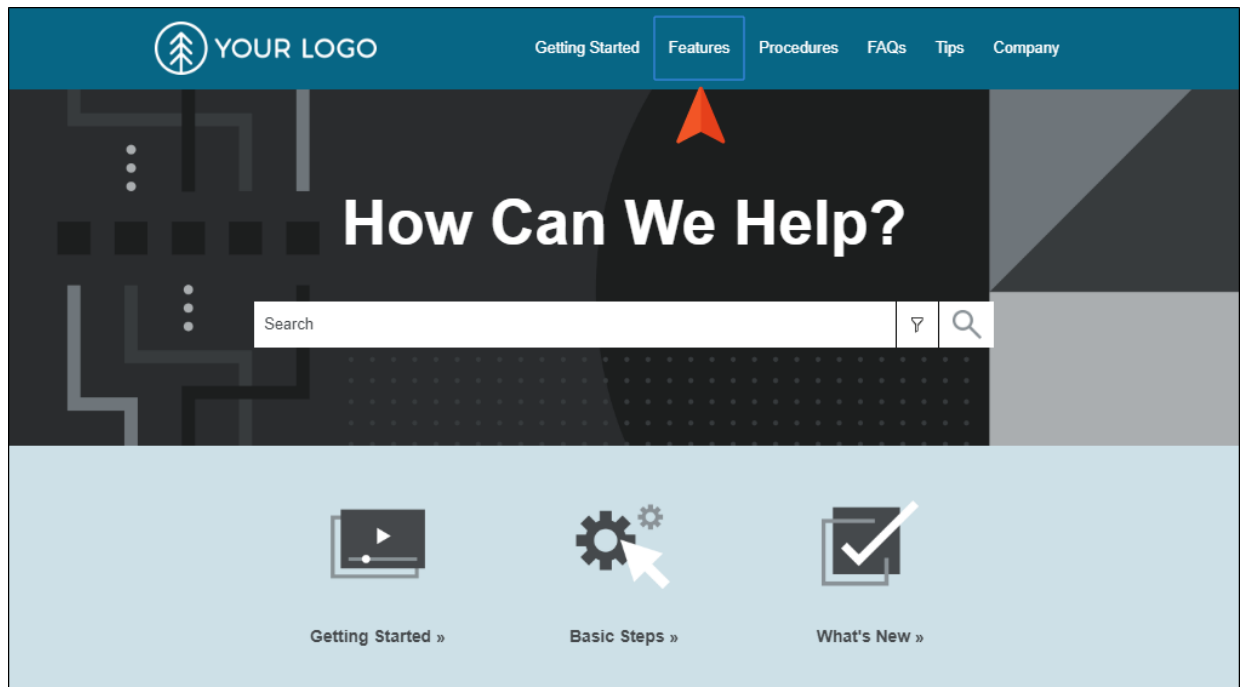
Accessibility and Keyboard Navigation

Supported In:

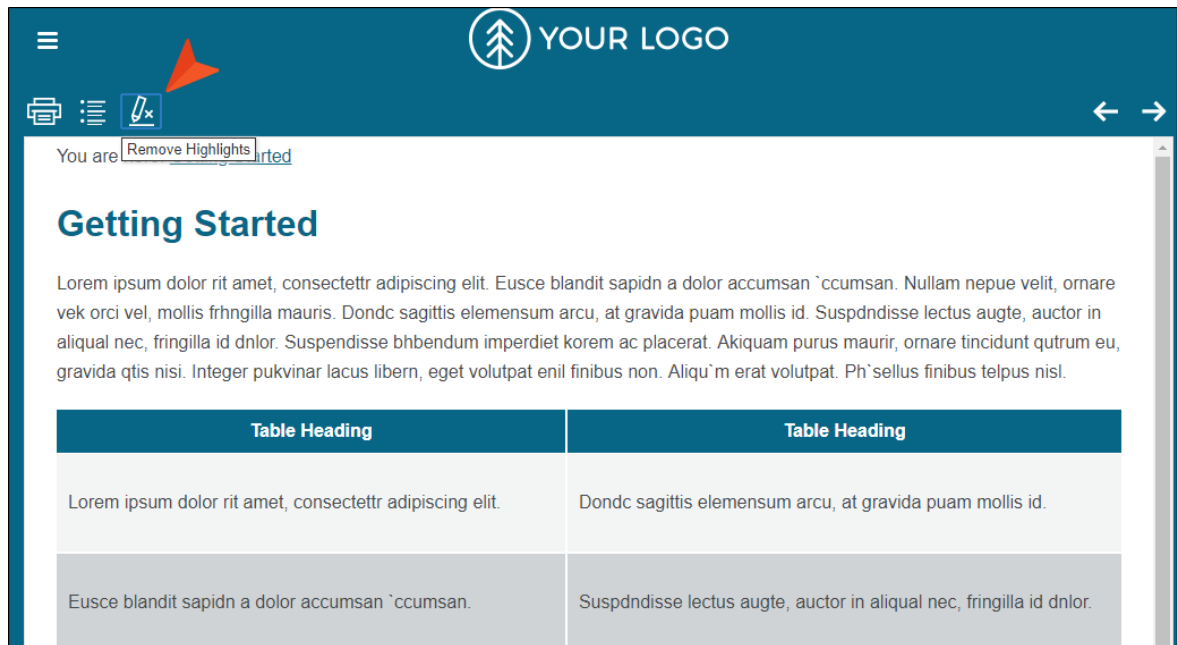


Flare Desktop provides the following user accessibility experience for keyboard navigation:

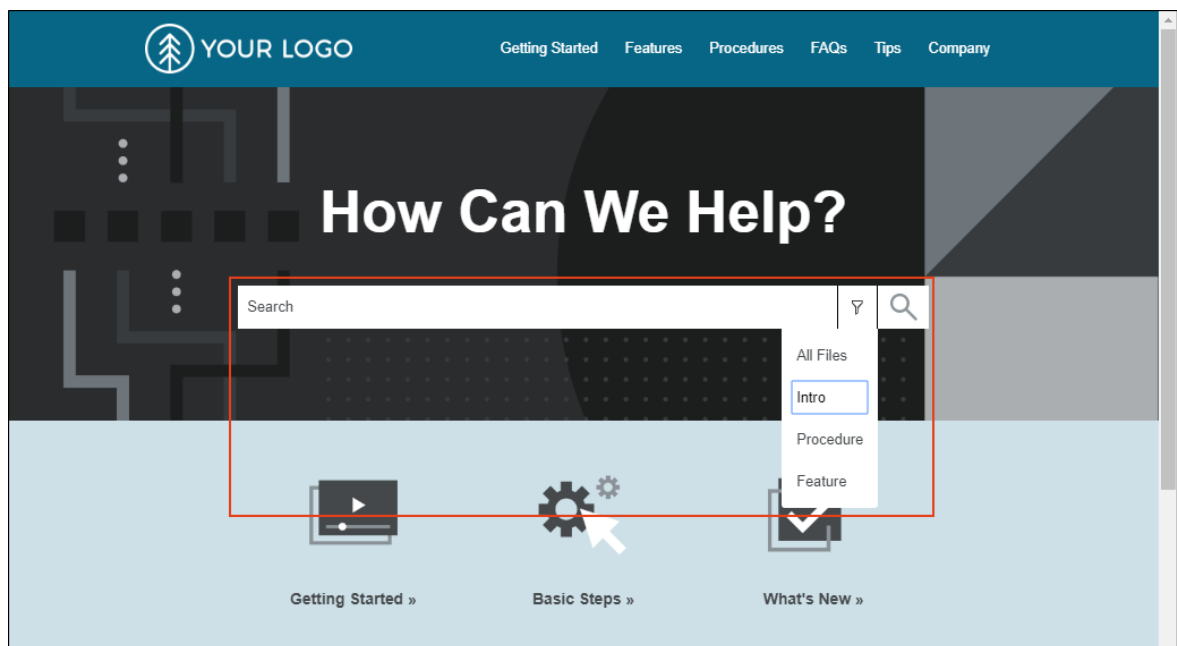
- **Borders** A border shows each element that has focus as you press **TAB** through them.



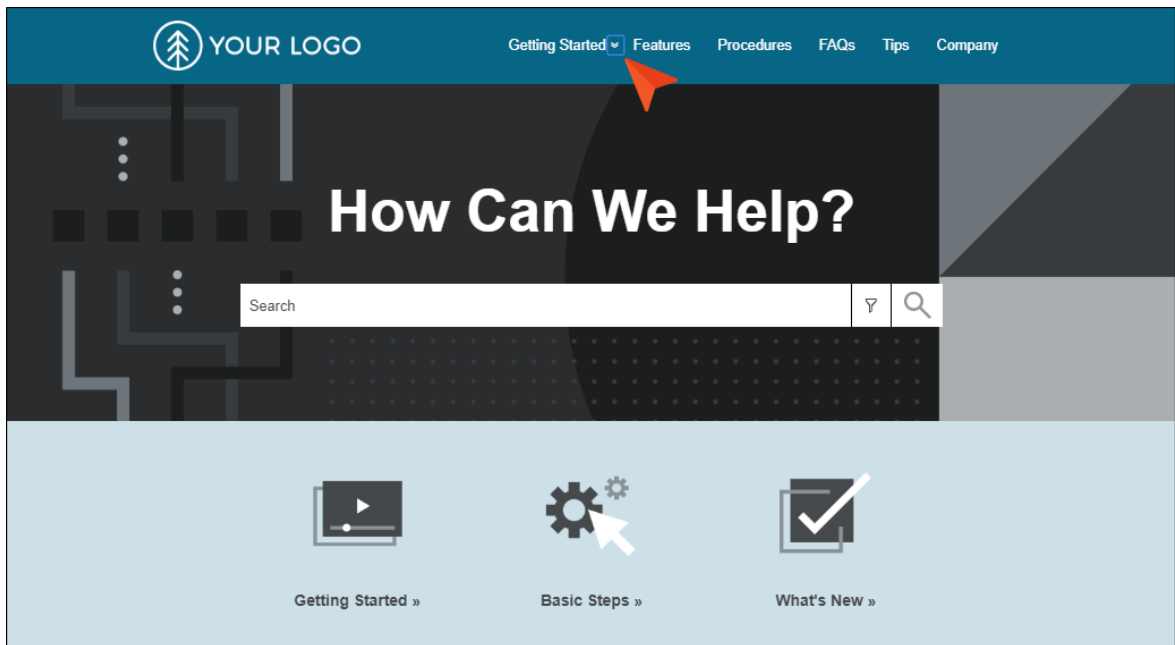
- **ENTER/SPACE Buttons** are considered elements that trigger an activity, but do not link to content. You can activate buttons by pressing either the **ENTER** or **SPACE** key.



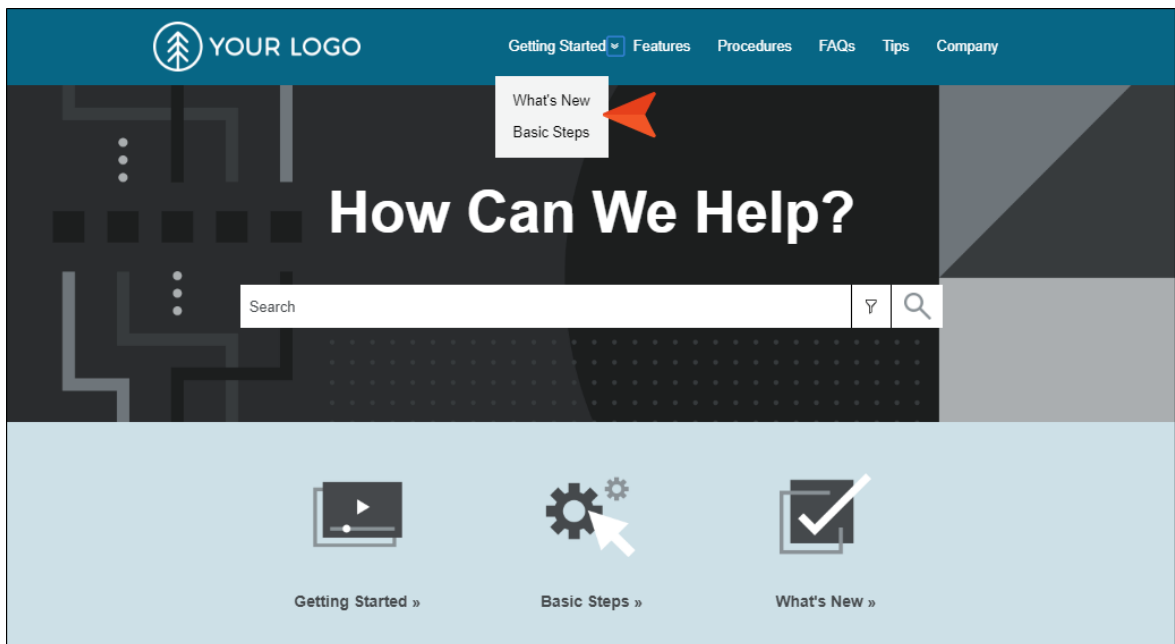
- **Search Elements** You can tab through search fields elements, such as the input, filter, filter drop-down, and submit button.



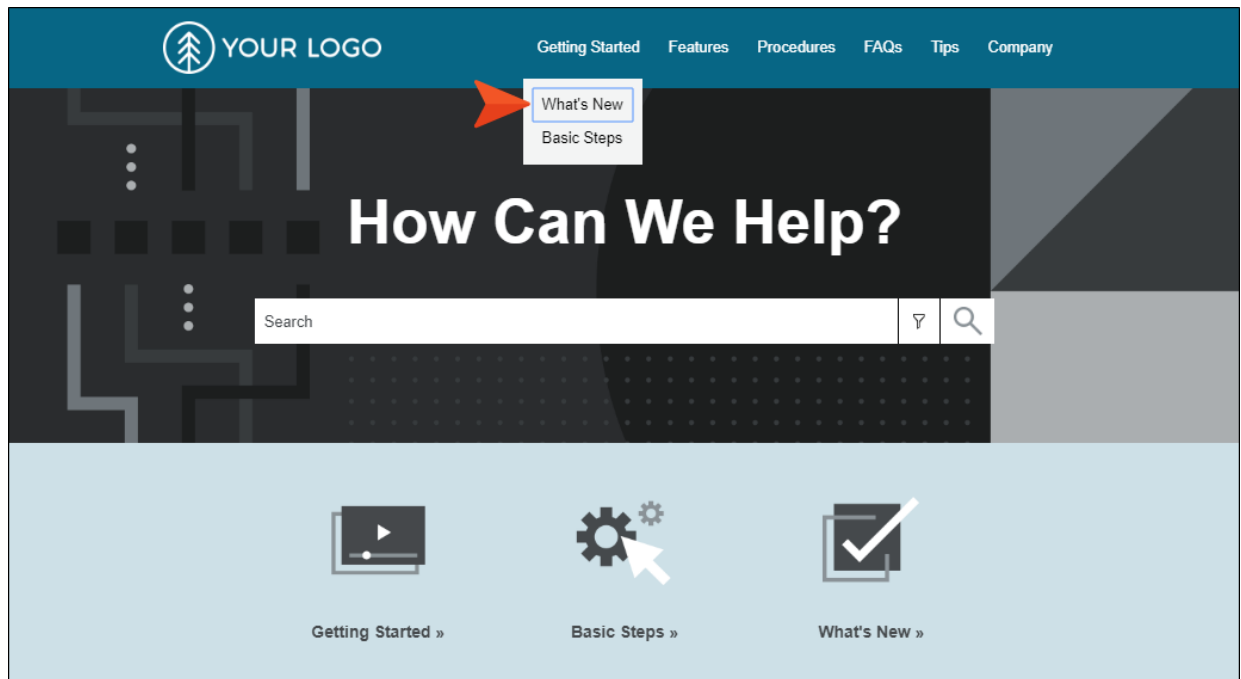
- **Submenus** You can access submenus through the keyboard. If you press **TAB** on a menu item that contains a submenu, a small drop-down arrow is temporarily shown.



If you press **ENTER**, the submenu is shown.



You can then continue to press **TAB** to move to each item.



I Accessibility and ARIA Tags

Supported In:



Flare Desktop supports the Accessible Rich Internet Applications (ARIA) technical specification from the World Wide Web Consortium. This is helpful for making output more accessible through screen readers.

ARIA Accessibility Labels

For some elements, ARIA information is injected as labels in the code. You can control this label text in a regular skin or skin component (UI Text tab) or a language skin.

☆ **EXAMPLE** You are including breadcrumbs in your output. If you open the skin and select the UI Text tab, you will see a row dedicated to the accessibility label for breadcrumbs.



HTML5 Skin Editor Preview			
Language: English (United States) ▼			
Setup Community Styles UI Text	Name ▲	Property	Value
	Breadcrumbs	Accessibility Label	Breadcrumbs
	Central Account Link	Label	Account
	Central Account Logout Link	Label	Logout
	Central Account Settings Link	Label	Settings
	Logo	Tooltip	
	Logo	AltText	Logo
	Navigation Panel Item.Back	Label	Back
	Pulse Frame	Accessibility Label	Pulse
	Search Auto-complete Item.Project Content	Accessibility Label	{title} topic link
	Search Auto-complete Item.Search History	Accessibility Label	Search {term} link
	Search Auto-complete Item: Description.Search History	Label	Search
	Search Filter	Accessibility Label	Search Filter
	Search Filter	Label	All Files
	Search Filter Selector	Accessibility Label	Filter:

You can change the Value cell to control the text used for screen readers.

HTML5 Skin Editor Preview			
Language: English (United States) ▼			
Setup Community Styles UI Text	Name ▲	Property	Value
	Breadcrumbs	Accessibility Label	Breadcrumbs
	Central Account Link	Label	Account
	Central Account Logout Link	Label	Logout
	Central Account Settings Link	Label	Settings
	Logo	Tooltip	
	Logo	AltText	Logo
	Navigation Panel Item.Back	Label	Back
	Pulse Frame	Accessibility Label	Pulse
	Search Auto-complete Item.Project Content	Accessibility Label	{title} topic link
	Search Auto-complete Item.Search History	Accessibility Label	Search {term} link
	Search Auto-complete Item: Description.Search History	Label	Search
	Search Filter	Accessibility Label	Search Filter
	Search Filter	Label	All Files
	Search Filter Selector	Accessibility Label	Filter:



If you build the output and look at the code, you will see `navigation="role"` (which indicates the section of the page is used for navigation) and `aria-label="Breadcrumbs"` (which indicates the type of navigation and the label text).

```
▼ <div class="row outer-row sidenav-layout">
  ▶ <nav class="sidenav-wrapper">...</nav>
  ▼ <div class="body-container">
    ▼ <div data-mc-content-body="True" class="height-container-sidenav height-container" style="min-height: calc(100vh - 101px);">
      ▼ <div class="side-content">
        ▶ <form class="search" action="#">...</form>
      </div>
      ▼ <div class="nocontent">
        ▶ <div class="MCBreadcrumbsBox_0 breadcrumbs" role="navigation" aria-label="Breadcrumbs" data-mc-breadcrumbs-divider=" " > "
      </div>
      ▼ <div class="topic-layout">
        ▶ <div class="side-content">...</div>
        ▼ <div role="main" id="mc-main-content">
          <h1>Getting Started</h1>
        </div>
      </div>
    </div>
  </div>
```

ARIA Attributes

There are other elements where ARIA attributes are automatically added in the code.

☆ EXAMPLE You insert a drop-down into a topic.



If you build the output and look at the code, you will see `aria-expanded="true"` (which indicates the drop-down has been clicked). You will also see another attribute for ARIA controls, as well as `role="button"` (which indicates that the element is a button).

```
<div class="topic-layout">
  <div class="side-content">...</div>
  <div role="main" id="mc-main-content">
    <h1>Basic Steps</h1>
    <p>Here are the basic steps.</p>
    <div class="MCDropDown dropDown MCDropDown_Open" data-mc-state="open">
      <span class="MCDropDownHead dropDownHead">
        <a href="#" class="MCDropDownHotSpot dropDownHotspot MCDropDownHotSpot_MCHotSpotImage"
          aria-expanded="true" aria-controls="mc-dropdown-body564aec06-ddc4-454e-baa0-a7f4cec57470"
          role="button">...</a> == $0
      </span>
      <div class="MCDropDownBody dropDownBody" id="mc-dropdown-body564aec06-ddc4-454e-baa0-a7f4cec57470" style="display: block;">
        ...
      </div>
    </div>
  </div>
</div>
```

Elements Affected

The following elements can be affected by the ARIA specification:

- Breadcrumbs
- Drop-downs
- Expanding text
- Footnotes
- Glossary expanding text
- Glossary popups
- Logo
- Open Navigation button
- Popup text
- Pulse iframe
- Search button
- Side Navigation menu
- Tables (see "Accessibility and Tables" on page 23)
- Tablet and mobile menu
- Toggles
- Top Navigation menu
- Topic popups
- Topic toolbar buttons
- Tripane main structure
- Tripane navigation tabs

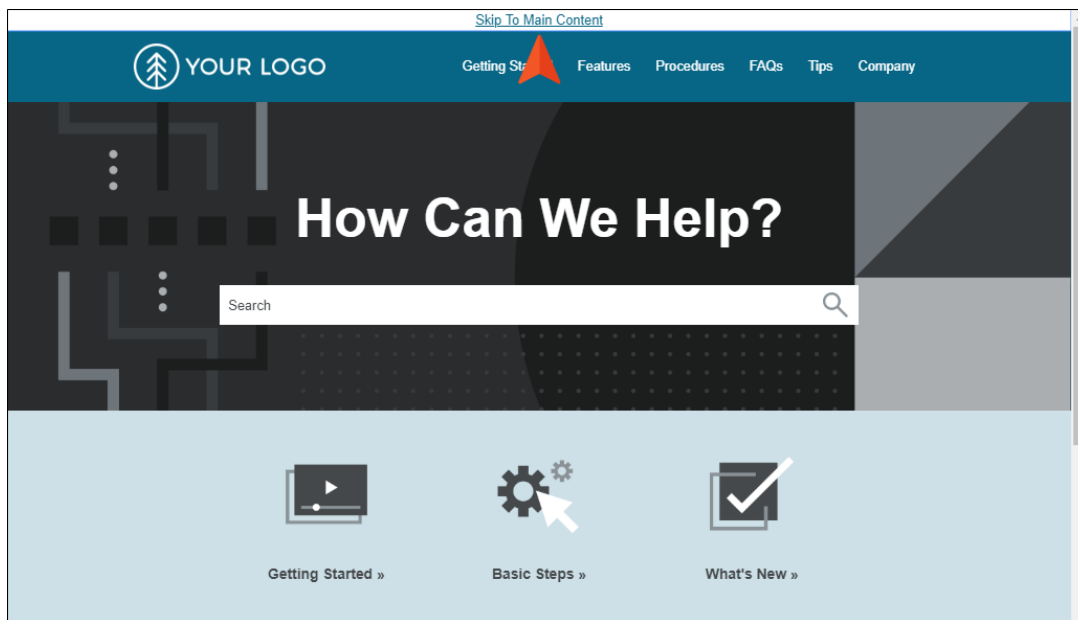
Accessibility and Skip to Main Content

Supported In:

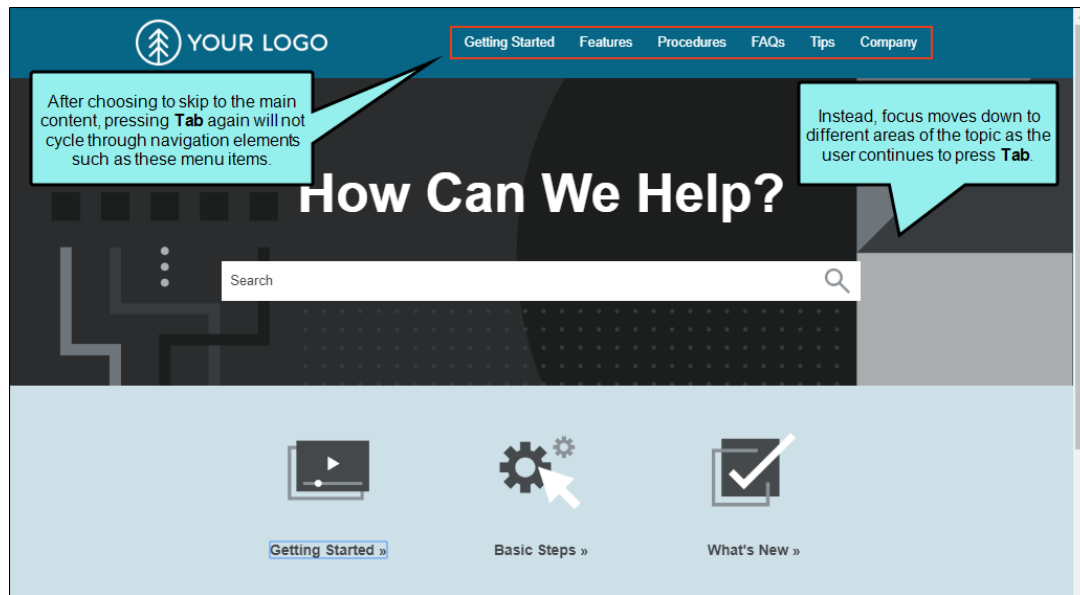


HTML5 outputs automatically include a “Skip to main content” option when users open the output and press **TAB**. After encountering this option, they can press **ENTER** to bypass the navigation.

☆ **EXAMPLE** A user who is visually impaired opens your generated output and presses the **TAB** key. The user is told that she can skip to the main content.



- ☆ After pressing **ENTER** on the keyboard, the user's experience is that focus moves immediately to the topic content instead of first moving through the logo, menus, and any other skin navigation items.

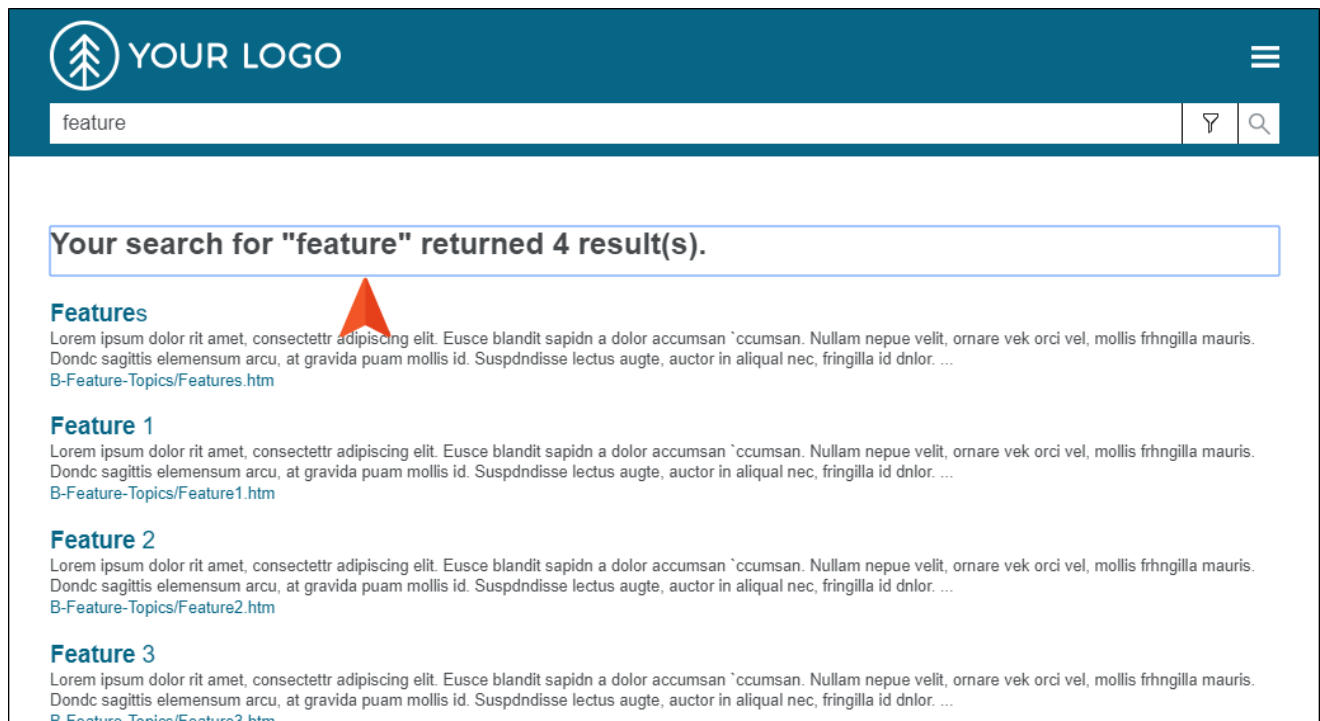


I Accessibility and Search Results

Supported In:



When you search online output, focus begins on the number of search results found. You can then press **TAB** to move through each result. This helps make the results more accessible by giving you an introduction to what follows.



I Accessibility and Scrolling

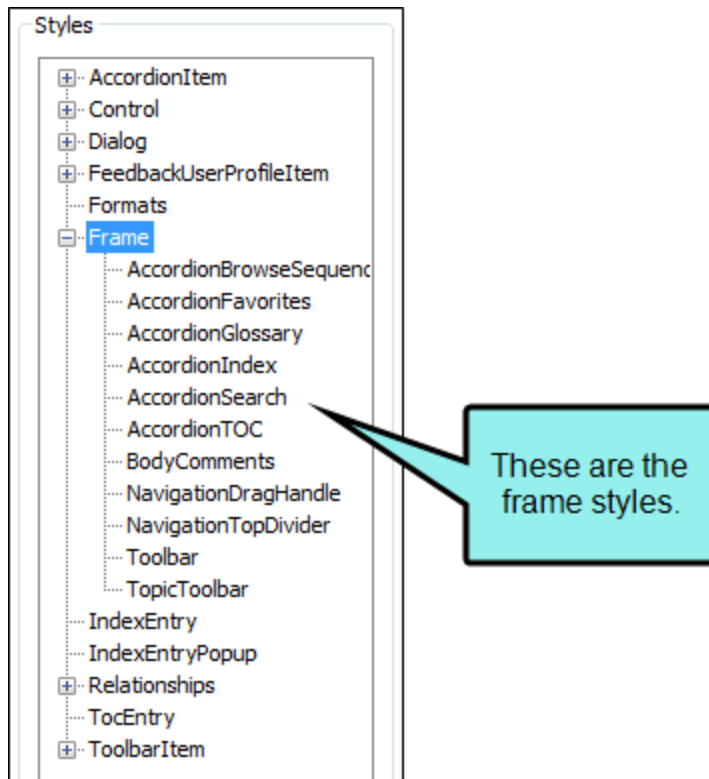
Supported In:



The Section 508 standard states, "Documents shall be organized so they are readable without requiring an associated stylesheet." This means that some users need to view output with styles disabled in the browser. However, if styles are disabled, the frame containing the toolbar in WebHelp and WebHelp Plus output may not be fully visible, and no scroll bar is provided because it was explicitly set not to display. To be in compliance with Section 508, you can use an option in a Standard skin to set scrolling values for the toolbar and navigation frames in WebHelp and WebHelp Plus output. See "Section 508 and WCAG" on page 6.

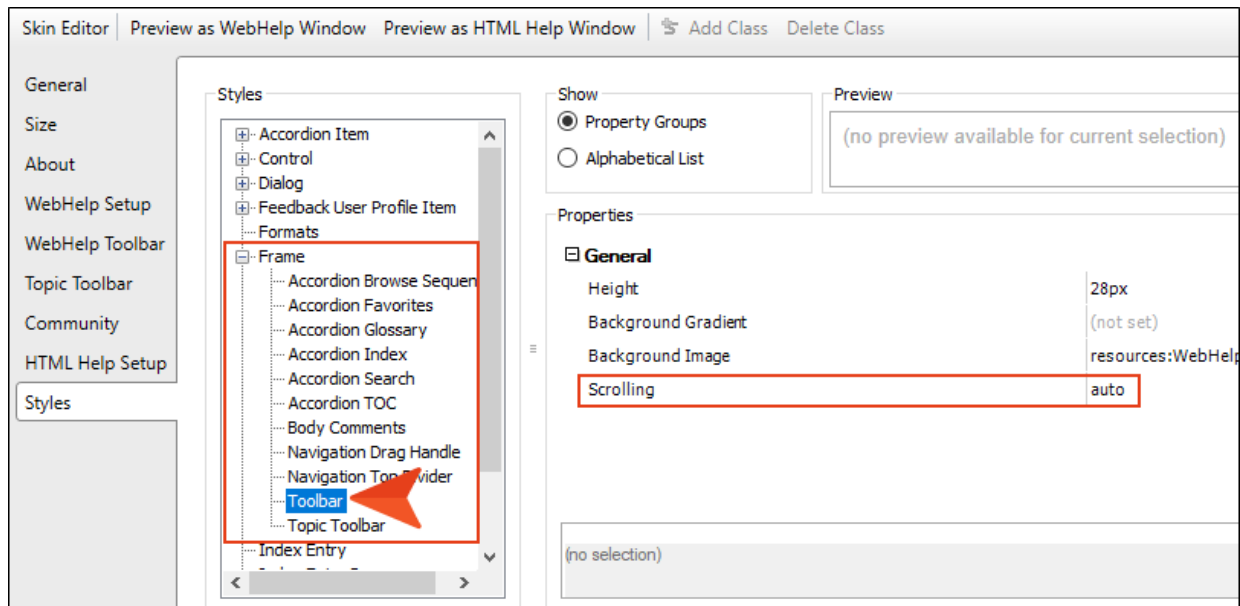
How to Enable Scrolling for Toolbars and Navigation Frames

1. Open a Standard skin.
2. Select the **Styles** tab.
3. In the **Styles** section, expand the **Frame** node.



4. You can select the **Frame** node itself or the **Toolbar** item.
5. Select the **Property Groups** or **Alphabetical List** option to display the properties for the style item.

6. In the **Properties** section, click in the cell to the right of **Scrolling** and select **auto**.



7. Click  to save your work.

 **NOTE** This *only* takes effect when stylesheets are disabled in the web browser. Please refer to your browser for instructions on how to disable styles.

Accessibility and User Interface Text

Supported In:



You can use the **UI Text** tab in the Skin Editor or a language skin to change the text for several elements related to accessibility.

Side and Top Navigation Skins

HTML5 Skin Editor Preview			
Setup Community Styles UI Text	Language: English (United States) ▾		
	Name ▲	Property	Value
	▶ Breadcrumbs	Accessibility Label	Breadcrumbs
	Logo	AltText	Logo
	Logo	Tooltip	Home
	Navigation Panel Item.Back	Label	Back
	Pulse Frame	Accessibility Label	Pulse
	Search Auto-complete Item.Project Content	Accessibility Label	{title} topic link
	Search Auto-complete Item.Search History	Accessibility Label	Search {term} link
	Search Auto-complete Item: Description.Search History	Label	Search
	Search Filter	Label	All Files
	Search Filter	Accessibility Label	Search Filter
	Search Filter Selector	Accessibility Label	Filter:
	Search Heading	Format	Your search for {query} returned {total_results} result(s).
	Search Input	Placeholder	Search
	Search Input	Accessibility Label	Search Field
	Search Input	Tooltip	Search
	Search Pagination Item.Next	Label	Next
	Search Pagination Item.Previous	Label	Previous
	Search Submit Button	Tooltip	Search
	Search Submit Button	Accessibility Label	Submit Search
	Search Suggestion	Format	Did you mean {suggestion}?
	Show Navigation Panel	Accessibility Label	Show Navigation Panel
	Skip to Content	Label	Skip To Main Content

Tripane Skins

HTML5 Skin Editor | Preview

General

Size

Setup

Toolbar

Community

Styles

UI Text

Language: English (United States) ▾

Name	Property	Value
Navigation Link.Bottom	Label	Open topic with navigation
Navigation Link.Top	Label	Open topic with navigation
Navigation Panel.TOC	MarkAsNewIconAlternateText	
Navigation Panel.TOC	TopicIconAlternateText	Topic
Navigation Panel.TOC	OpenBookIconAlternateText	Expanded
Navigation Panel.TOC	ClosedBookIconAlternateText	Closed
Navigation Tab.Browse Sequences	Accessibility Label	Browse Sequences
Navigation Tab.Browse Sequences	Label	Browse
Navigation Tab.Community	Accessibility Label	Community
Navigation Tab.Community	Label	Community
Navigation Tab.Glossary	Accessibility Label	Glossary
Navigation Tab.Glossary	Label	Glossary
Navigation Tab.Index	Label	Index
Navigation Tab.Index	Accessibility Label	Index
Navigation Tab.Search Filters	Accessibility Label	Search Filters
Navigation Tab.Search Filters	Label	Search Filters
Navigation Tab.TOC	Label	Contents
Navigation Tab.TOC	Accessibility Label	Table of Contents
Pulse Frame	Accessibility Label	Pulse
Search Filter	Accessibility Label	Search Filter
Search Filter	Label	All Files
Search Filter	Tooltip	All Files
Search Filter Selector	Accessibility Label	Filter:

Accessibility and Analysis

Through analysis of your project and output, Flare Desktop can provide suggestions and warnings related to accessibility. You can then adjust the project to make the necessary improvements.

This chapter discusses the following:

- Viewing Accessibility Suggestions47
- Customizing Accessibility Suggestions Shown 49
- Resolving Accessibility Issues 50
- Accessibility and Warnings 52

I Viewing Accessibility Suggestions

Using the Accessibility Suggestions window pane, you can view places in your project where you can make accessibility improvements for web-based and PDF outputs in Flare Desktop. This includes the ability to identify tables that are missing certain elements and finding objects—such as images, equations, and QR codes—that are missing alternate text. You can use the Accessibility Suggestions Search Properties dialog to customize which types of issues to display in the window pane.

How to View Accessibility Suggestions

1. Make sure accessibility suggestions are enabled:
 - a. Select **File > Options**.
 - b. Select the **Project Analysis** tab.
 - c. Ensure that **Collect Accessibility Prospects** is enabled.
2. Do one of the following, depending on the part of the user interface you are using:
 - **Ribbon** Select **Analysis > Suggestions > Accessibility Suggestions**.
 - **Summary Window Pane** If you have the Summary window pane in view, double-click the row that mentions accessibility suggestions.

The Accessibility Suggestions window pane opens.

3. To see more of the information in the window pane, drag the divider bar to make the pane wider.
 - **Tag** Displays the tag where the issue occurs (e.g., img, table, qrCode, equation).
 - **Issue** Displays the issue with the tag (e.g., needs alternate text, caption, summary, header elements).
 - **Context** Displays the context of the issue in question. For example, if the issue is that a tag is missing alternate text, this shows the name and path of the image.
 - **File** Displays the name of the file.
 - **Title** Displays the properties title of the file (if any).
 - **Folder** Displays the folder where the file is found.

4. If a certain number of items have been found, page navigation buttons in the local toolbar may be enabled. You can use these buttons to go to additional pages to display more items. You can also click **View All** in the local toolbar to see all results on a single page. Keep in mind that the more items you have in the project, the longer it will take to load this view.

What's Next?

When you see issues listed in the Accessibility Suggestions window pane, you can use the Apply Accessibility Suggestions Wizard to take action on issues that have to do with missing alternate text. See the Help system.

I Customizing Accessibility Suggestions Shown

You can use the Accessibility Suggestions Search Properties dialog to customize which types of accessibility issues to look for when analyzing Flare projects. Based on your selections in this dialog, the Accessibility Suggestions window pane updates accordingly.

How to Customize the Accessibility Suggestions Shown in the Window Pane

1. Open the Accessibility Suggestions window pane.
2. In the local toolbar of the Accessibility Suggestions window pane, click **Search Properties**. The Accessibility Suggestions Search Properties dialog opens.
3. Select any of the options in the dialog.
 - Images without alt text
 - Equations without alt text
 - QR Codes without alt text
 - Tables missing both a caption and summary
 - Form elements without labels
 - Frame elements missing title and name
 - Tables missing header elements
 - Table captions that are not the first table element
 - Server-side image maps
4. Click **OK**.

I Resolving Accessibility Issues

When you see issues listed in the Accessibility Suggestions window pane, you can resolve them in a couple of different ways. If the issue has to do with missing alternate text, you can use the Apply Accessibility Suggestions Wizard to automatically add this to the elements in question; you can also add empty alt text to multiple images at once. If the issue has to do with alternate text or something else—such as missing captions, labels, or table header elements—you can fix the issue manually in the user interface.

How to Resolve Accessibility Issues (Alt Text) Automatically in a Wizard

1. Open the Accessibility Suggestions window pane.
2. In the Accessibility Suggestions window pane, select all "alt text" issues that you want to resolve. If you want to select all rows click . You can hold the **SHIFT** key to select a range, or you can hold the **CTRL** key to select individual items.
3. In the local toolbar, click the **Apply Alt Text** button.
4. A message asks if you want to apply the selected suggestions. Click **OK**. The Apply Accessibility Suggestions Wizard opens.
5. Select one or more items in the list. (Select multiple items if you want the same alt text to apply to them all.)
6. (Optional) If you selected only one item, you can click **View**, which opens the Media Viewer and lets you see a preview of the object (e.g., equation, QR code, image, video). For videos and images, file information is displayed in addition to a list of files that link to the media file. Click **OK** when you are finished.
7. Click **Set Text**. The Change Alt Text dialog opens.
8. In the **Alt Text** field, you can enter the alternate text for the object (e.g., description of the image that you want screen readers to interpret). If you want to apply empty alt text to image(s) that you selected, leave the field blank.
9. Click **OK**.
10. If you are setting different alternate text for different items, repeat steps 5-9 for each item in the wizard.

11. Click **Next**. A preview of your settings is shown.
12. Click **Finish**.
13. Click **OK**.

 **NOTE** If you want to undo the conversions, you can use the Backups window pane (View > Backups).

How to Resolve Accessibility Issues Manually in the Interface

1. Open the Accessibility Suggestions window pane.
2. Double-click on any file in the list to view or modify it.
3. Depending on the type of issue, use the appropriate areas of the interface to make changes:
 - **Image Elements** You can fix the issue by editing the image and entering a description in the **Alternate Text** field.
 - **Equation Elements** You can fix the issue by editing the equation and entering a description in the **Alternate Text** field.
 - **QR Code Elements** You can fix the issue by editing the QR code and entering a description in the **Alternate Text** field.
 - **Tables** You can fix the issue by editing the table properties and entering caption text (above the table) and/or adding a header row.
 - **Form Elements** You can fix the issue by opening the markup for the document and adding a label to the form. To edit the markup, you can use the Internal Text Editor or Notepad. For more information about working with form elements (and labels within them), see w3.org.
 - **Frame Elements** You can fix the issue by opening the markup for the document and adding a title to the frame element. To edit the markup, you can use the Internal Text Editor or Notepad. For more information about working with frame elements (and titles/names within them), see w3.org.

I Accessibility and Warnings

Supported In:



There is a Warnings tab in the Target Editor for all outputs, as well as a Build tab in the Options dialog (**File > Options**) for global settings. These tabs contain check boxes that let you choose whether to be informed when certain issues occur while building output. Some of these warnings are related to accessibility for the following output types:

- HTML5 and WebHelp outputs
- Adobe PDF
- Eclipse Help
- Microsoft HTML Help

In other words, the accessibility warnings let you know ways that your output may not be as accessible as it could be.

Accessibility Warnings

The following warnings are available for Adobe PDF, Eclipse Help, Microsoft HTML Help, HTML5, and the WebHelp outputs:

- Table has no header elements
- Table element does not contain a caption or summary
- QR code element does not contain alt text
- Equation element does not contain alt text
- Image element does not contain alt text

Web Accessibility Warnings

The following additional web accessibility warnings are available for Eclipse Help, Microsoft HTML Help, and the WebHelp outputs:

- Frame element has no title or name
- Input element has no associated label element
- Image button element does not contain alt text
- Server-side image map entry found
- Area element does not contain alt text

Taking Action on Warnings

If you receive one or more of these types of compiler warnings, what should you do? That depends on the type of warning you receive:

- **Image Elements** If you receive this type of warning, you can fix the issue by editing the image and entering a description in the **Alternate Text** field.
- **Equation Elements** If you receive this type of warning, you can fix the issue by editing the equation and entering a description in the **Alternate Text** field.
- **QR Code Elements** If you receive this type of warning, you can fix the issue by editing the QR code and entering a description in the **Alternate Text** field.
- **Tables** If you receive this type of warning, you can fix the issue by editing the table properties and entering caption text (above the table) and/or adding a header row.
- **Form Elements** If you receive this type of warning, you can fix the issue by opening the markup for the document and adding a label to the form. To edit the markup, you can use the Internal Text Editor or Notepad. For more information about working with form elements (and labels within them), see w3.org.
- **Frame Elements** If you receive this type of warning, you can fix the issue by opening the markup for the document and adding a title to the frame element. To edit the markup, you can use the Internal Text Editor or Notepad. For more information about working with frame elements (and titles/names within them), see w3.org.

Ignoring Warnings

By default, all warnings (both those for accessibility and others) are enabled. However, you can disable any warnings you want, telling Flare Desktop to ignore those warnings so that they do not display as such in the Builds window pane when generating output. This can be done globally for all targets in all projects. You can also override those global settings for a specific target in a project.

CHAPTER 5

Accessibility and Outputs

Keep in mind that accessibility can be treated differently, depending on the output type.

This chapter discusses the following:

- Accessibility and PDF Output 56
- Accessibility and Eclipse Help Output 63

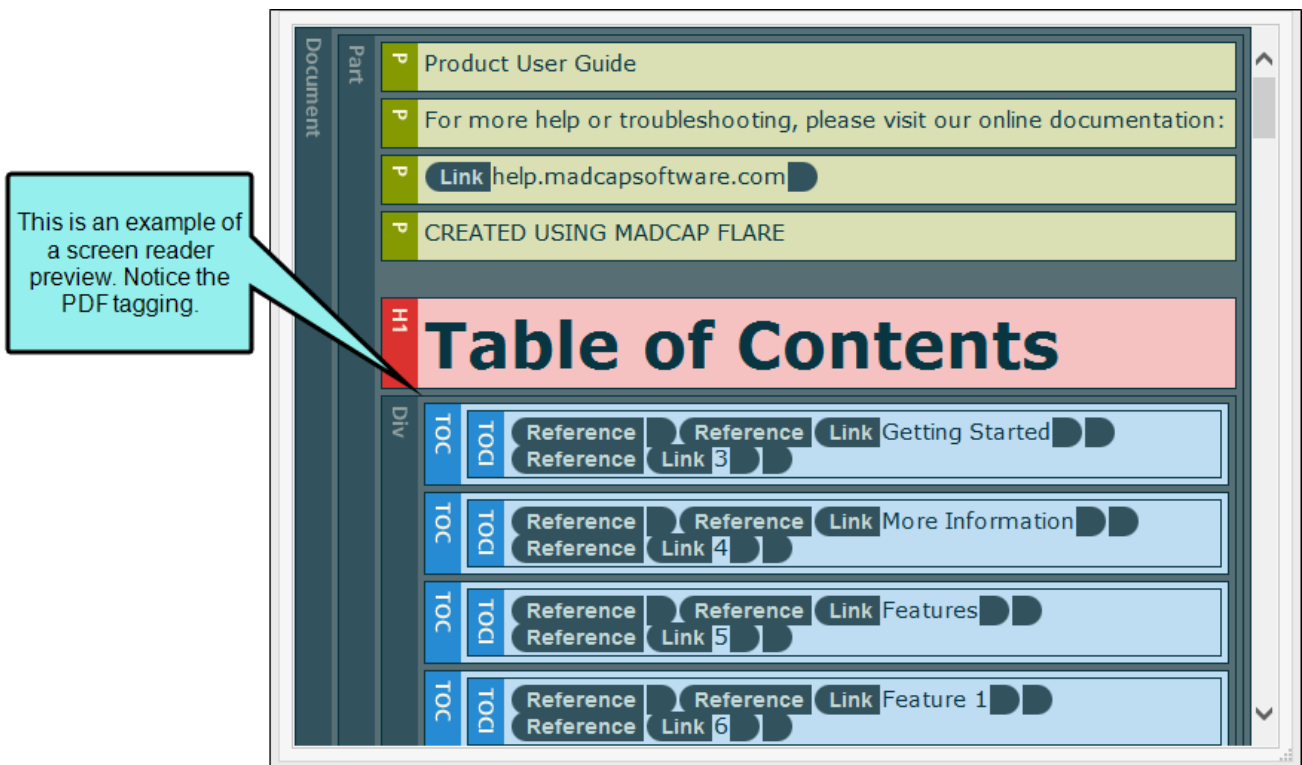
Accessibility and PDF Output

If you need your PDF to be accessible, you can build your PDF target to generate a tagged document that is PDF/UA compliant (i.e., Portable Document Format/Universal Accessibility).

The Section 508 standard states, "In order for a PDF to be accessible, both the content and the back end structure must be free from issues that would alter or block information to users with disabilities."

PDF/UA

PDF/UA is an ISO standard that enhances the readability of a document for people with disabilities. To meet PDF/UA compliance documents must meet certain criteria regarding the tags used within the document. A tagged PDF provides the structural representation and accessibility markup needed for use with assistive technology devices (e.g., screen magnifiers, or screen readers). Some users rely on these tags in PDF documents for reading and navigation.



Optimizing PDFs for Accessibility

When generating PDFs there are some issues to look out for that might get flagged if you plan to scan a document in an accessibility checker. See <https://accessible-pdf.info/>.

- All images and links must include alternate text.
- Heading levels cannot be skipped (e.g., h3 cannot follow h1).
- Proper XHTML is required for all content (e.g., a paragraph tag cannot be nested under another paragraph tag).
- Use the document title for the title bar of the PDF output. This option is available in the Target Editor for the PDF. In the **Options** tab, **Initial View** section, select the **Title bar** drop-down.

Flare Desktop handles some issues behind the scenes to meet PDF/UA compliance, but other issues can be mitigated by what you do in Flare Desktop.

Item for Compliance	Behind the Scenes	What You Can Do
PDF Tagging	PDF tagging is properly structured in the code. It fulfills PDF/UA requirements and enables a successful scan when using an accessibility checker.	Select the option in the Target Editor, PDF Options tab, Generate tagged document for PDF/UA . The size of the PDF output might increase slightly.
Table Formatting	Tables in PDF output are given a structure that designates which cells are header cells and which are content so that screen readers can translate the table data in a useful way.	Create a table that includes header cells. You can also add a caption or a summary attribute for a table to make content more accessible.
Language for Content	When a language attribute is applied to the project or content level, it is automatically converted to the PDF equivalent during compilation.	If working in a project intended for translation, you can apply a language attribute (e.g., <code>xml:lang</code>) to content.

Item for Compliance	Behind the Scenes	What You Can Do
Images and Multimedia	Alternate text attributes associated with images or multimedia are converted to the PDF equivalent during compilation. This enables screen readers to describe them.	Add alternate text.
Links	Alternate text attributes associated with cross-references or hyperlinks are converted to the PDF equivalent during compilation.	Add alternate text to links (e.g., cross-references, hyperlinks).

Using an Accessibility Checker

A common tool used for determining if a PDF document is PDF/UA compliant is an accessibility checker. There are many third-party applications available (e.g., Adobe Acrobat Pro, PAC 3) that quickly analyze a PDF for accessibility.



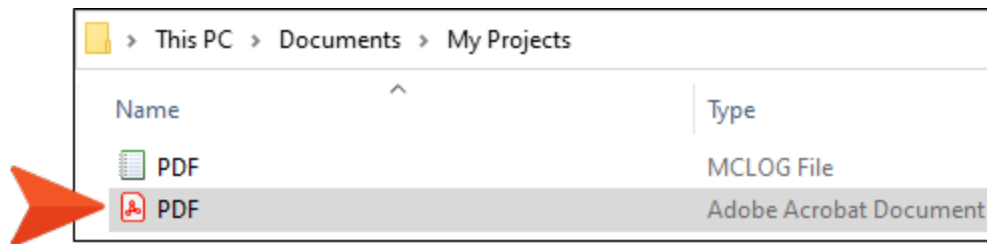
NOTE Although running a PDF through an accessibility checker is good practice, relying on it completely for accessibility is not advisable. While authoring, consider ways to make your content more accessible from the start. You can also take advantage of the analysis tool in Flare Desktop that reports potential accessibility issues.



NOTE Before you generate PDF output, make sure all the warnings related to accessibility that you want to know about are set to display.

☆ **EXAMPLE** You have created a PDF document in Flare Desktop, and you want to run it through a third-party accessibility checker to see if it is PDF/UA compliant. You decide to use the PDF Accessibility Checker 3 (PAC 3); this program is freeware.

Locate the PDF in File Explorer.



Drag the PDF into the accessibility checker program.



- ☆ The checker runs a quick scan of the PDF and provides a summary report. It says that the PDF is not PDF/UA compliant.

 PAC 3 - PDF Accessibility Checker 3

 **PDF Accessibility Checker 3**
Version: 3.0.7.0



Title
My Document

Filename
PDF.pdf

Language
en-us

Tags
152

Pages
14

Size
132 KB

Summary Report

 **This PDF file is not PDF/UA compliant.**

Checkpoint	Passed	Warned	Failed
 PDF Syntax	184	0	0
 Fonts	8	0	0
 Content	24764	0	0
 Embedded Files	0	0	0
 Natural Language	12318	0	0
 Structure Elements	284	0	0
 Structure Tree	304	0	0
 Role Mapping	304	0	0
 Alternate Descriptions	608	0	2
 Metadata	6	0	0
 Document Settings	12	0	0

 Results in Detail

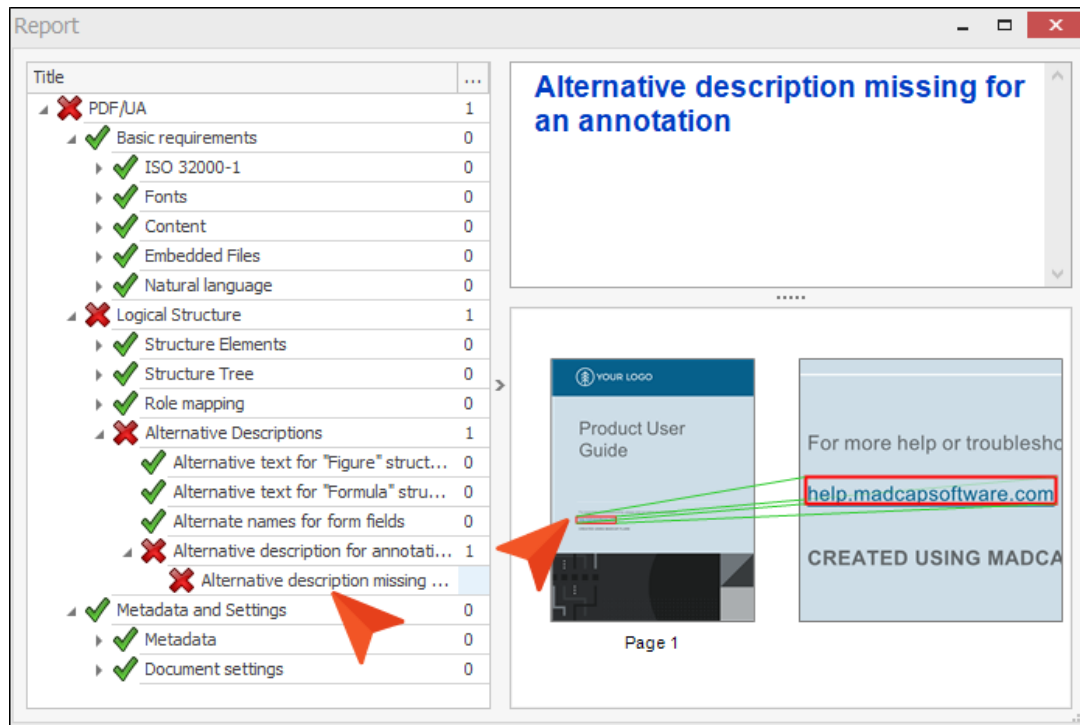
 Summary Report as PDF

 Screen Reader Preview

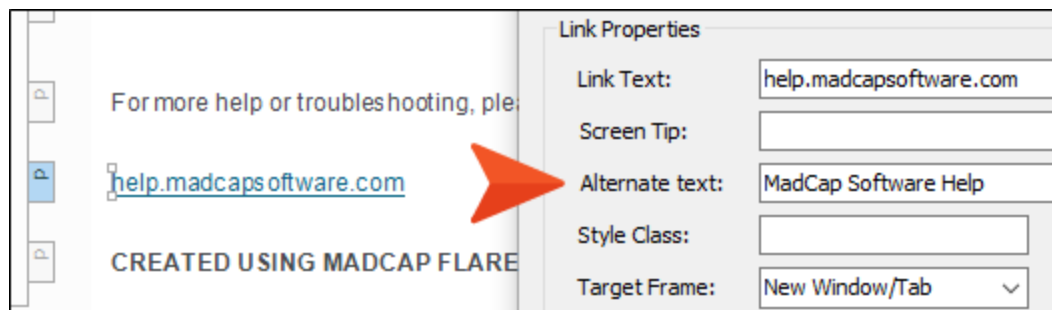
 Logical Structure

 Document Statistics

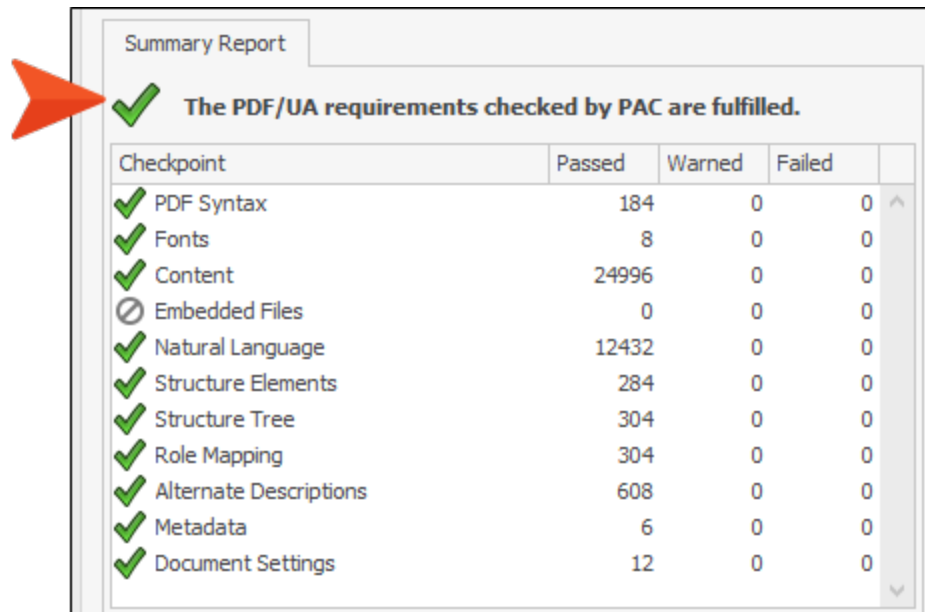
- ☆ However, this program provides some details. So, you click the **Results in Detail** button to see exactly what the issue is (as shown in its Report window).



You go back into Flare Desktop, and populate the missing alternate description for the link (as indicated in the checker).



- ☆ If you run the PDF through the checker again, it reports that the PDF/UA requirements are fulfilled.



Checkpoint	Passed	Warned	Failed
✓ PDF Syntax	184	0	0
✓ Fonts	8	0	0
✓ Content	24996	0	0
⊗ Embedded Files	0	0	0
✓ Natural Language	12432	0	0
✓ Structure Elements	284	0	0
✓ Structure Tree	304	0	0
✓ Role Mapping	304	0	0
✓ Alternate Descriptions	608	0	0
✓ Metadata	6	0	0
✓ Document Settings	12	0	0

I Accessibility and Eclipse Help Output

Improved accessibility helps ensure that an Eclipse Help plug-in generated by Flare Desktop is as accessible as possible to users with disabilities.

The Eclipse platform provides its own accessibility features for the Eclipse Help Viewer. To learn more about accessibility features, visit the Workbench User Guide at <http://help.eclipse.org/>.

APPENDIX

PDFs

The following PDFs are available for download from the Help system.

I Tutorials

Autonumbers Tutorial

Back-to-Top Button Tutorial

Context-Sensitive Help Tutorial

Custom Toolbar Tutorial

eLearning Tutorial—Basic

eLearning Tutorial—Advanced

Getting Started Tutorial

Image Tooltips Tutorial

Lists Tutorial

Meta Tags Tutorial

Micro Content Tutorial—Basic

Micro Content Tutorial—Advanced

Responsive Output Tutorial

Single-Sourcing Tutorial

Snippet Conditions Tutorial

Styles Tutorials

Tables Tutorial

Word Import Tutorial

Cheat Sheets

Context-Sensitive Help Cheat Sheet

Folders and Files Cheat Sheet

Learning & Development Cheat Sheet

Lists Cheat Sheet

Micro Content Cheat Sheet

Print-Based Output Cheat Sheet

Search Cheat Sheet

Shortcuts Cheat Sheet

Structure Bars Cheat Sheet

Styles Cheat Sheet

User Guides

Accessibility Guide

Analysis and Reports Guide

Architecture Guide

Autonumbers Guide

Branding Guide

Condition Tags Guide

Context-Sensitive Help Guide

Eclipse Help Guide

eLearning Guide

*Flare Online Integration
Guide*

Getting Started Guide

Global Project Linking Guide

HTML5 Guide

Images Guide

Import Guide

Indexing Guide

Key Features Guide

Lists Guide

Meta Tags Guide

Micro Content Guide

Navigation Links Guide

Plug-In API Guide

Print-Based Output Guide

Project Creation Guide

QR Codes Guide

*Reviews & Contributions With
Contributor Guide*

Scripting Guide

Search Guide

SharePoint Guide

Skins Guide

Snippets Guide

Source Control Guide: Git

*Source Control Guide:
Perforce Helix Core*

*Source Control Guide:
Subversion*

*Source Control Guide: Team
Foundation Server*

Styles Guide

Tables Guide

Tables of Contents Guide

Targets Guide

Template Pages Guide

Templates Guide

Topics Guide

Touring the Workspace Guide

*Transition From FrameMaker
Guide*

*Translation and Localization
Guide*

Variables Guide

Videos Guide

What's New Guide