



MadCap Software

HTML5 Guide

Flare 11.1



Copyright 2015 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
7777 Fay Avenue
La Jolla, California 92037
858-320-0387
www.madcapsoftware.com

THIS GUIDE WAS CREATED USING MADCAP FLARE.

CONTENTS

CHAPTER 1 Introduction	7
CHAPTER 2 Responsive Web Design	19
Responsive Skins	20
Responsive Content	32
Viewing Responsive Output	60
CHAPTER 3 Top Navigation Output	63
Frameless Output	64
HTML5 Top Navigation Skin	65
Skin Components and Proxies	66
Skin Set to None for Target	81
Home Topic	82
Stylesheets	84
Responsive Output and Device Width Media Queries	87
Producing Top Navigation Output	93
Simple Conversion to Top Navigation Output	128
Advanced Conversion to Top Navigation Output	137
CHAPTER 4 Skin Styles	163
Interactive Live Preview	164
Style Sections and Root/Child Properties	167
Default Values in Gray	177
Medium-specific Styles	178



Tasks for HTML5 Skins—Tripane, Top Navigation, and Components	180
Main Page in HTML5 Skins	204
Headers in HTML5 Skins	208
Menus in HTML5 Skins	217
Navigation in HTML5 Skins	222
Topics in HTML5 Skins	237
Search Results in HTML5 Skins	246
Feedback in HTML5 Skins	262
Topic Toolbars in HTML5 Skin Components	270
CHAPTER 5 Other Skin Tasks	279
Changing Text Strings in HTML5 Skins	280
Showing Navigation Links in Standalone Topics	281
Specifying Browser Settings	282
Specifying Navigation Pane Settings for HTML5 and WebHelp Outputs	283
Specifying Web Toolbar Settings—HTML5 Output	287
CHAPTER 6 CSH Calls	297
What You Need to Do	298
CSH Calls for HTML5 Output—Developers	299
CHAPTER 7 More About HTML5	305
Enabling HTML5 Server-based Output	306
Installing IIS and ASP.NET	307
Setting Up an HTML5 Target	318
Configuring IIS on Production Server	320
Starting Microsoft Indexing Service	322
Starting Microsoft Windows Search	323
Enabling HTML5 Server-based Search	324
Testing HTML5 Server-based Search	325
Viewing HTML5 Server-based Output	326

Merging Output at Runtime Using HTML5	331
Including Non-XHTML Files in HTML5 Search	341
HTML5 Server-based Output Scenario	345
APPENDIX A PDF Guides	351
APPENDIX B Output Type Comparison Tables	353
General	355
Accessibility	356
Context-sensitive Help	357
Feedback Statistics and Reporting Features	358
Generated Content	359
Glossaries	362
Images	363
Indexes	364
Language Support	365
Master Pages and Page Layouts	366
Merging Output	367
Miscellaneous Features	368
Multimedia	370
Navigation Links	374
Pulse (Socially Enabled Output)	375
Search	376
Skins	382
Tables of Contents and Mini-TOCs	385
Target Settings	386
Variables	388

Introduction

HTML5 output is similar to the regular WebHelp format; however, whereas the original WebHelp format works with HTML4, this output format supports the HTML5 specification developed by the Web Hypertext Application Technology Working Group (WHATWG—<http://whatwg.org>) and the World Wide Web Consortium (W3C—<http://w3.org>). Therefore, the HTML5 format results in better markup and offers additional features not found in the WebHelp outputs.

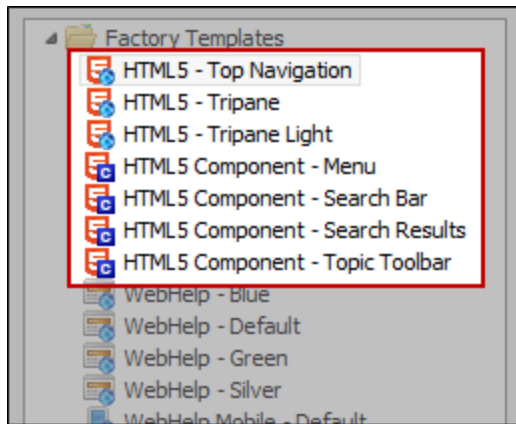
The HTML5 output consists of a collection of files that you will distribute to users. The output will be displayed in the user's Internet browser window. The main entry file has an .htm extension.

The online Help on the MadCap Software website (webhelp.madcapsoftware.com) was created with HTML5.

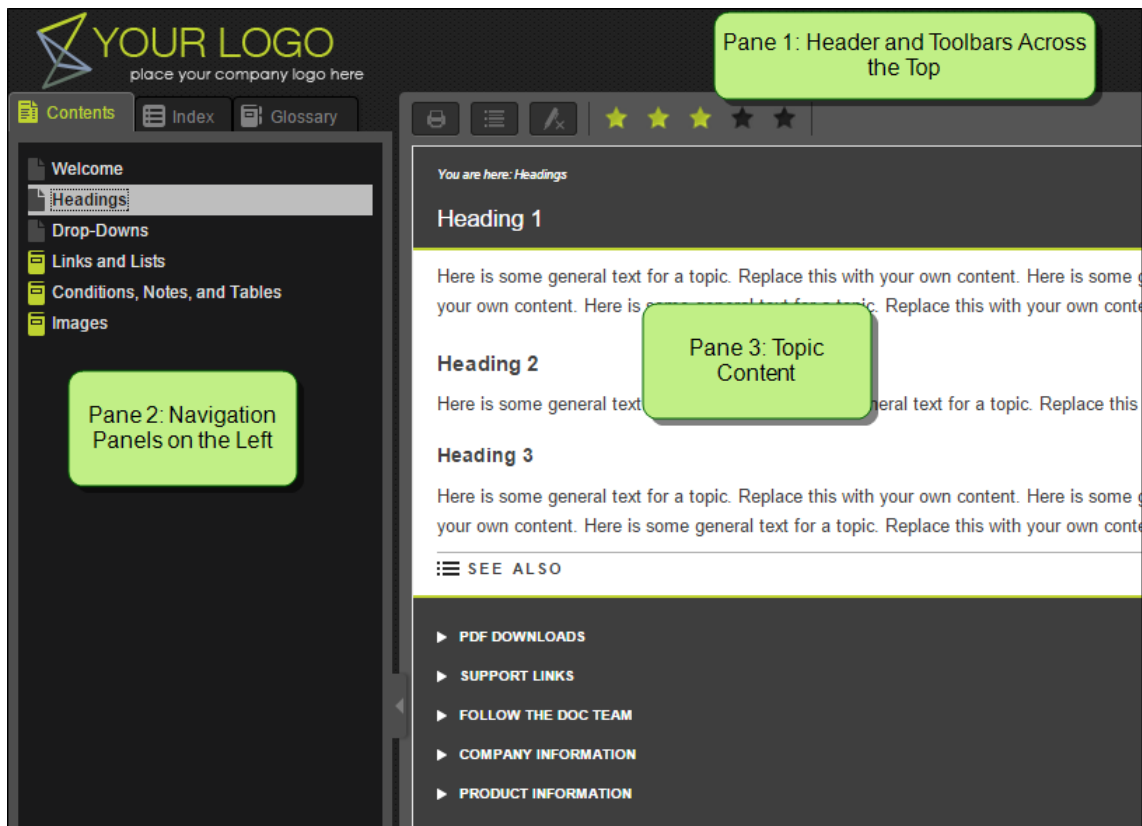


Following are some of the key features of the HTML5 format:

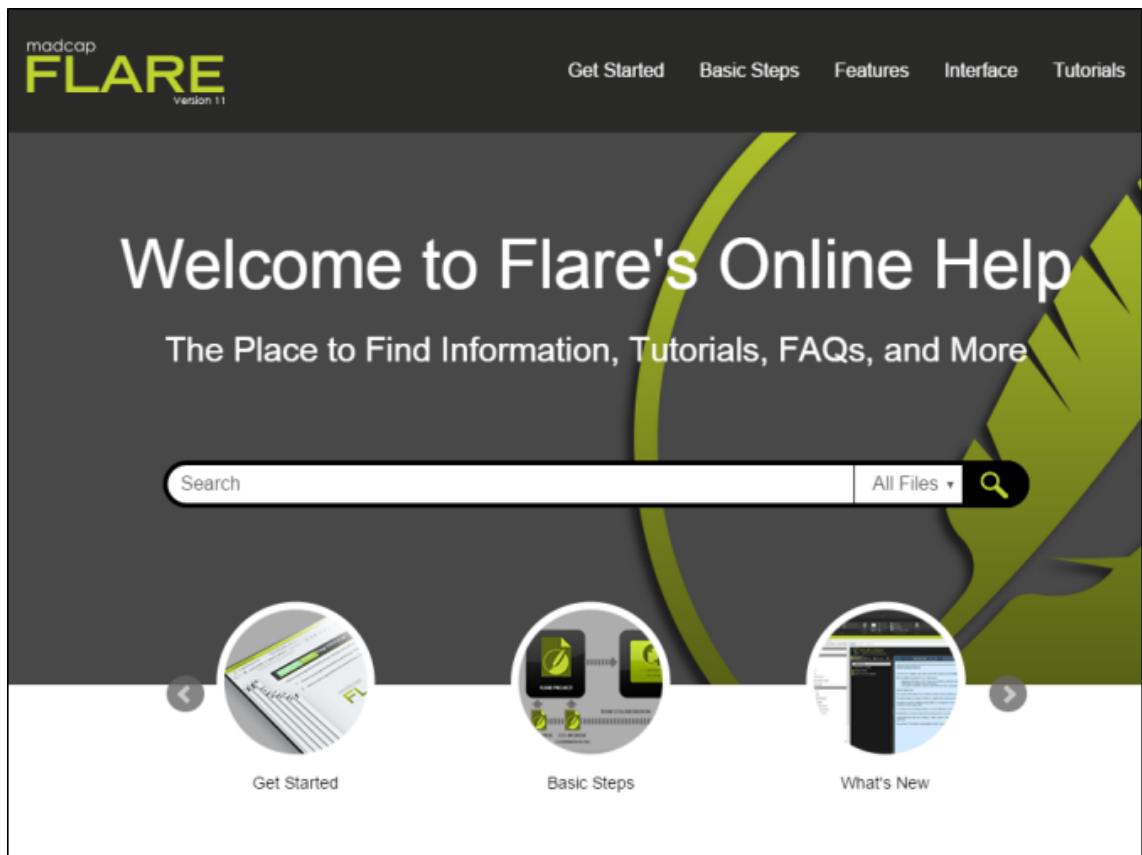
- » **Skins—Top Navigation, Tripane, and Components** Because HTML5 output is quite different from the other online formats, it requires its own skin. In fact, you have more than one skin from which to choose.



You can use a Tripane skin, which lets you generate output in a traditional format with three distinct frames (a navigation pane on the left, a toolbar and search on the top, and the main body pane).



Alternatively, you can use a Top Navigation skin. This lets you generate a more flexible, frameless output like a modern website.



You can even tell Flare not to use any skin at all for HTML5 output.

You can also use smaller skin components that work with related proxies. With these, you can add and design menus and toolbars, inserting them in your content anywhere you like. For Top Navigation and outputs without any skin, you can also create search bars and customized results pages.



Note: Unlike other output types, separate language skins are not used for HTML5 targets. Instead, a UI Text tab is located in the Skin Editor, which lets you perform all of the localization tasks that you would normally accomplish in a language skin.

- » **Frameless** Pages in HTML5 output do not use `<frameset>` or `<frame>` tags, like they do in regular WebHelp output. In addition, HTML5 Top Navigation output is completely frameless, meaning it also does not contain iframes. This means that output is much easier for web crawlers to find, thus improving search engine results.
- » **Responsive Output** Responsive web design (RWD) for HTML5 targets means that the display is adjusted automatically depending on the width of the window. Therefore, on tablets and smart phones, users will see a condensed look that is more appropriate for those devices. You can get the same effect if you shrink your browser to a smaller size.
- » **Specific Page Addresses** With the regular WebHelp output, you can only give readers a single URL path, which opens the starting page for the output. With HTML5 output, you can point end users to a specific page in the output by giving them the exact URL path. Because HTML5 output also allows for pagination in search results, you can also direct users to specific page addresses in search results.

EXAMPLE

Let's say you generate regular WebHelp output, with the primary output file called "MyOutput.htm." When you view the output with the initial topic shown (say it's called "Welcome.htm"), the URL path in the browser might look something like this.

`http://www.mycompany.com/documentation/MyOutput.htm`

And if you then click on another topic (say, "WhatsNew.htm"), the same URL path will be shown. It does not change, even though you opened a different page.

On the other hand, suppose you generate HTML5 output. The URL in the browser may look the same as the path for regular WebHelp when the output is first displayed.

`http://www.mycompany.com/documentation/MyOutput.htm`

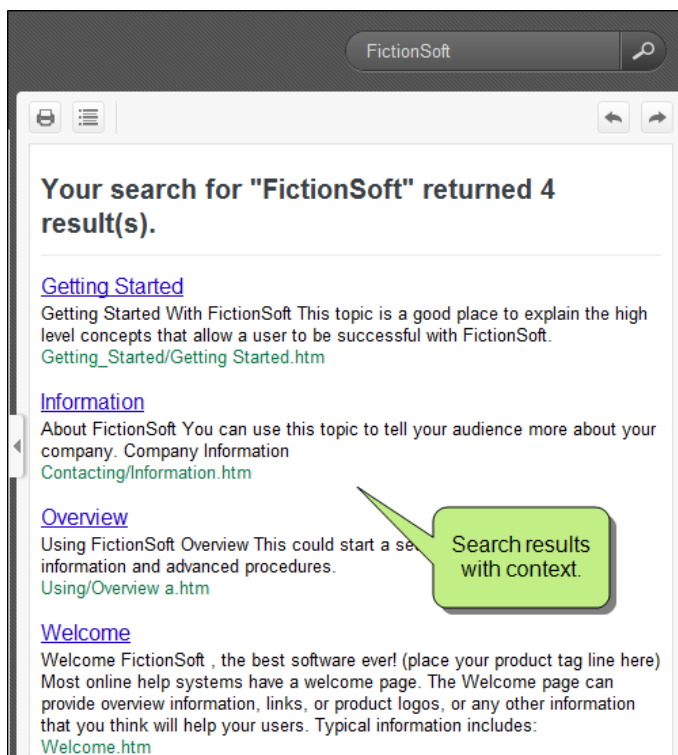
But if you click on the "WhatsNew.htm" page, the URL will change to this.

<http://www.mycompany.com/documentation/MyOutput.htm#WhatsNew.htm>

And if you click on the "Welcome.htm" page to return to it, the URL will change to this.

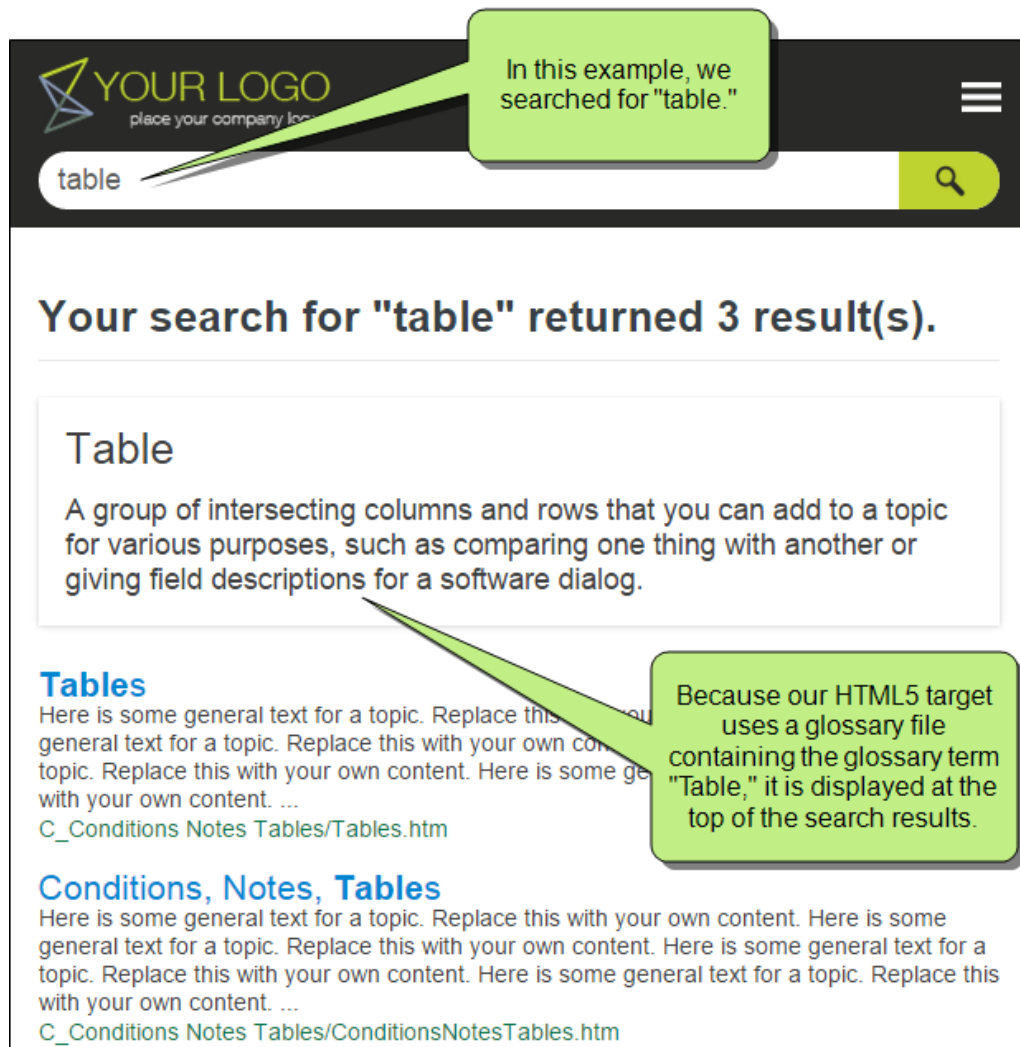
<http://www.mycompany.com/documentation/MyOutput.htm#Welcome.htm>

- » **Sitemap/Search Engine Optimization (SEO)** You can generate a sitemap when compiling your output. This helps with search engine optimization (SEO), making it easier for search indexing services (i.e., spiders, crawlers, or bots) to find your output. Therefore, the entire output is indexed and search engine results are improved.
- » **Search with Context** If end users perform a search in your output, they will see context next to each result, rather than terms only.



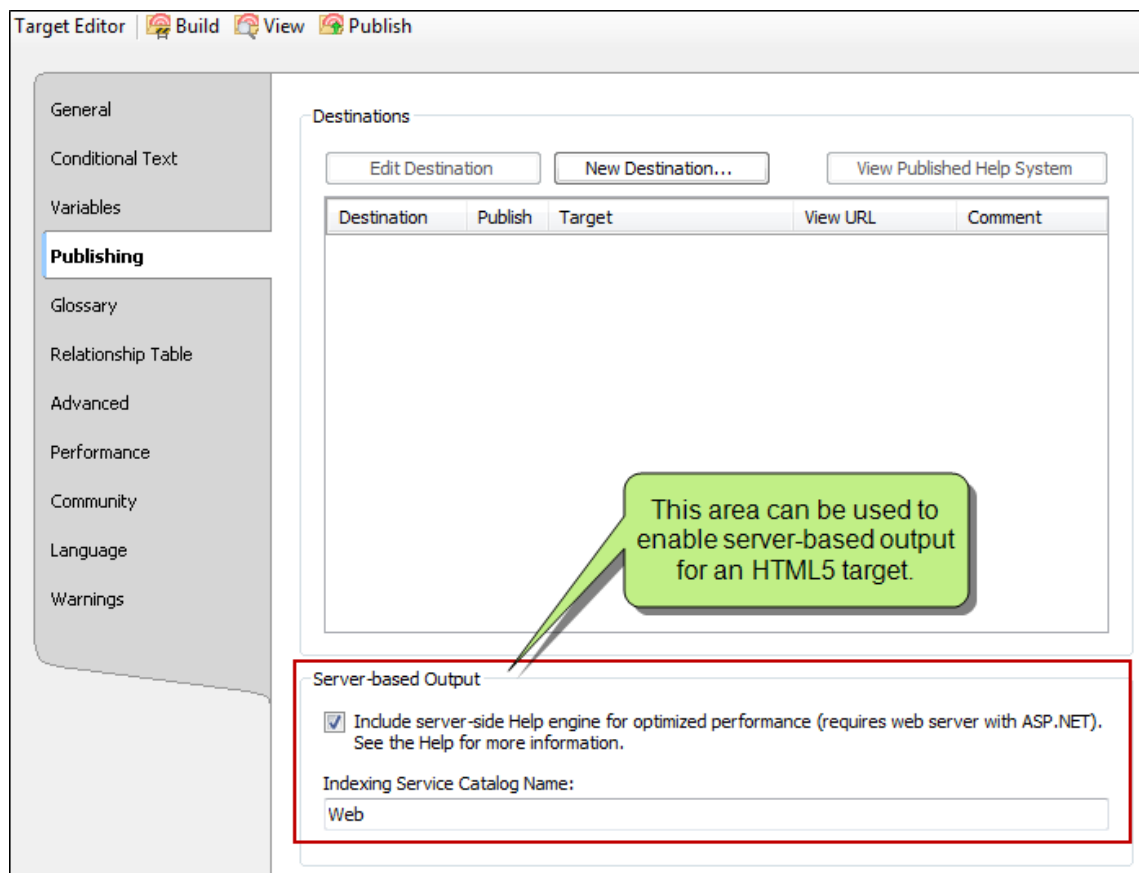
In addition, you can create topic meta descriptions, which will be used in the search results abstract. Also, the meta description itself is searchable.

- » **Glossary Terms in Search Results** For HTML5 output, search results display glossary terms if they are used as the search text.



This option to display glossary terms in search results is enabled by default. However, it can be turned off in the HTML5 Target Editor.

- » **Smoother Dynamic Effects** Features such as drop-down text, expanding text, and popups have a smoother appearance when they are activated.
- » **Server-based Output** You can create HTML5 output in its regular state, or you can select an option in the Publishing tab of the Target Editor to enable server-based functionality. This allows you to accomplish the same results as WebHelp Plus output—searching of non-XHTML content, server-side search, and automatic runtime merging. See "More About HTML5" on page 305.



Following are benefits of server-based HTML5 output:

- » **Multiple Platforms** You can publish to a machine running Windows XP, Windows Server 2003, Windows Server 2008, Windows 7, or Windows Vista.
- » **Searching of Non-XHTML Content** When end users perform a search in your online output, you can ensure that non-XHTML files (e.g. PDF, DOC, XLS) are included in that search. When you build HTML5 output, a subfolder named "AutoSearch" is created and placed in the generated output folder. You can place non-XHTML files within the published AutoSearch subfolder (whether the non-XHTML files are linked to content from your Flare project or not). When users perform a search, those non-XHTML files will also be accessible to the users. See "Including Non-XHTML Files in HTML5 Search" on page 341.
- » **Microsoft IIS Search Engine** Another benefit of generating and publishing HTML5 output to a web server is that it takes advantage of Microsoft's IIS search engine to help power your search. This is especially useful if you have a very large Help system.
- » **Automatic Runtime Merging of Flare Projects** This is an easy way to merge the output from multiple HTML5 server-based Flare targets into one Help system. These targets can be originated from the same Flare project or from different Flare projects. You simply place the output files in the correct location on the server (i.e., within your master project's AutoMerge folder). Flare then automatically merges the output from all of the targets when users access the Help. From the end user's perspective, the results are seamless, appearing as one large Help system. All of the TOCs, browse sequences, indexes, glossaries, and search capabilities for the projects are merged. See "Merging Output at Runtime Using HTML5" on page 331.



Note: The following browsers support HTML5: Internet Explorer 8 or later, Firefox 10 or later, Google Chrome 13 or later, and other browsers that support the HTML5 standard.

COMPARISON OF HTML5 TRIPANE, TOP NAVIGATION, AND SKINLESS

	Tripane	Top Navigation	Skinless
Frameless—No Framesets	✓	✓	✓
Frameless—No iframes ¹	✗	✓	✓
Glossary Terms in Search Results	✓	✓	✓
Navigation Panes—Browse Sequence, Glossary, Index, Pulse Community, TOC	✓	✗	✗
Project Merging	✓	✗	✗
Pulse Integration	✓	✓	✓
Responsive Skin Always Enabled	✗	✓	✗
Skin Component—Menu	✓	✓	✓
Skin Component—Search Bar	✗	✓	✓
Skin Component—Search Results	✗	✓	✓

	Tripane	Top Navigation	Skinless
Skin Component—Topic Toolbar	✓	✓	✓

¹If your HTML5 output is enabled with MadCap Pulse, the comments area at the bottom of topics is wrapped in an iframe. Therefore, the output is technically not entirely frameless. However, because it is only the comments that are contained in the iframe, the main topic content still retains the benefits (e.g., better search results) of frameless output.

Responsive Web Design

Responsive web design (RWD) for HTML5 targets means that the display is adjusted automatically depending on the width of the window. Therefore, on tablets and smart phones, users will see a condensed look that is more appropriate for those devices. You can get the same effect if you shrink your browser to a smaller size.

There are two areas where RWD can be applied: (1) skin and (2) content.

HTML5 Top Navigation skins are always enabled for responsive output, but you can adjust some settings on the Skin tab in the Target Editor. For Tripane output, you can enable responsive output in the Skin Editor, and you can adjust the same settings as Top Navigation output in the Target Editor. When a skin is responsive, the navigation elements are automatically adjusted depending on the size of the screen. See "Responsive Skins" on the next page.

Styles and media queries can be used to make your content responsive. This lets you present information—both its substance and structure—differently depending on the size of the screen or device. See "Responsive Content" on page 32.

This chapter discusses the following:

Responsive Skins	20
Responsive Content	32
Viewing Responsive Output	60



Responsive Skins

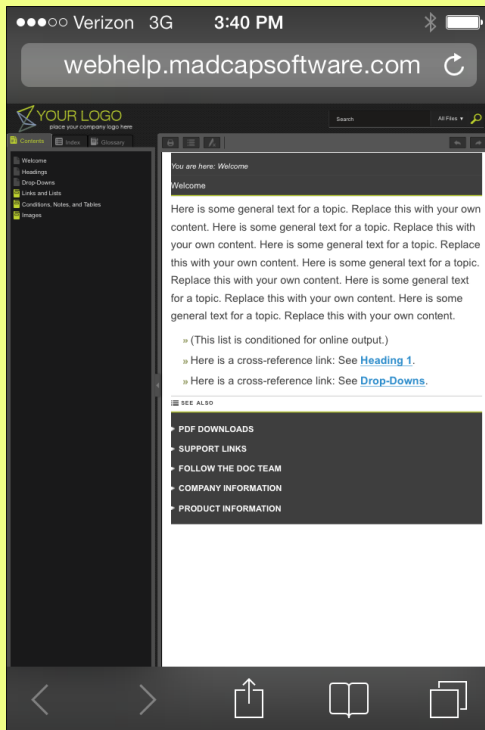
Supported In:



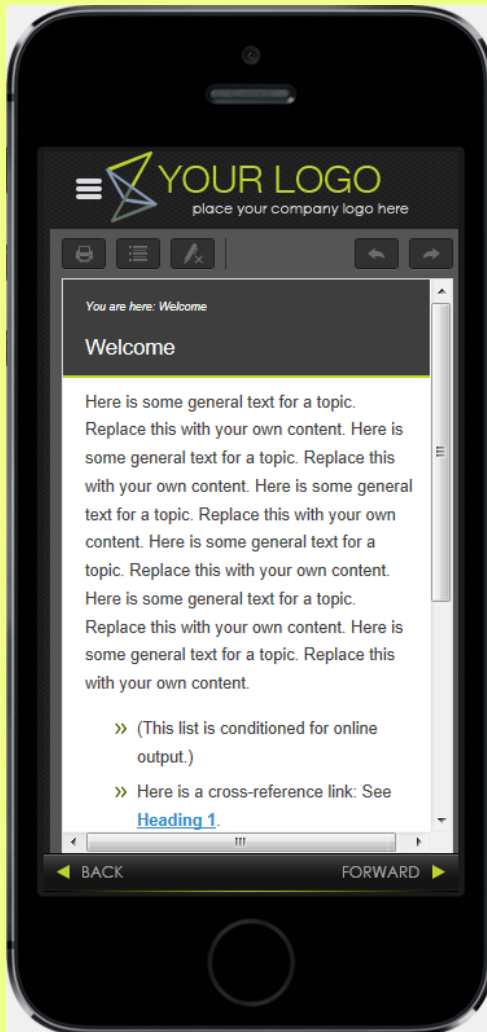
HTML5 Top Navigation skins are always enabled for responsive output, but you can adjust some settings on the Skin tab in the Target Editor. For Tripane output, you can enable responsive output in the Skin Editor, and you can adjust the same settings as Top Navigation output in the Target Editor. When a skin is responsive, the navigation elements are automatically adjusted depending on the size of the screen.

EXAMPLE

Let's say you want to create HTML5 and mobile output from your project. Without using responsive output, you would need to create two targets and two skins, and perhaps separate settings elsewhere (e.g., stylesheet mediums). If you do not create a separate target and skin for the mobile output, but instead direct people to open the full HTML5 output on their smart phones, everything seems quite small and it can be difficult to navigate through that output on a smart phone.



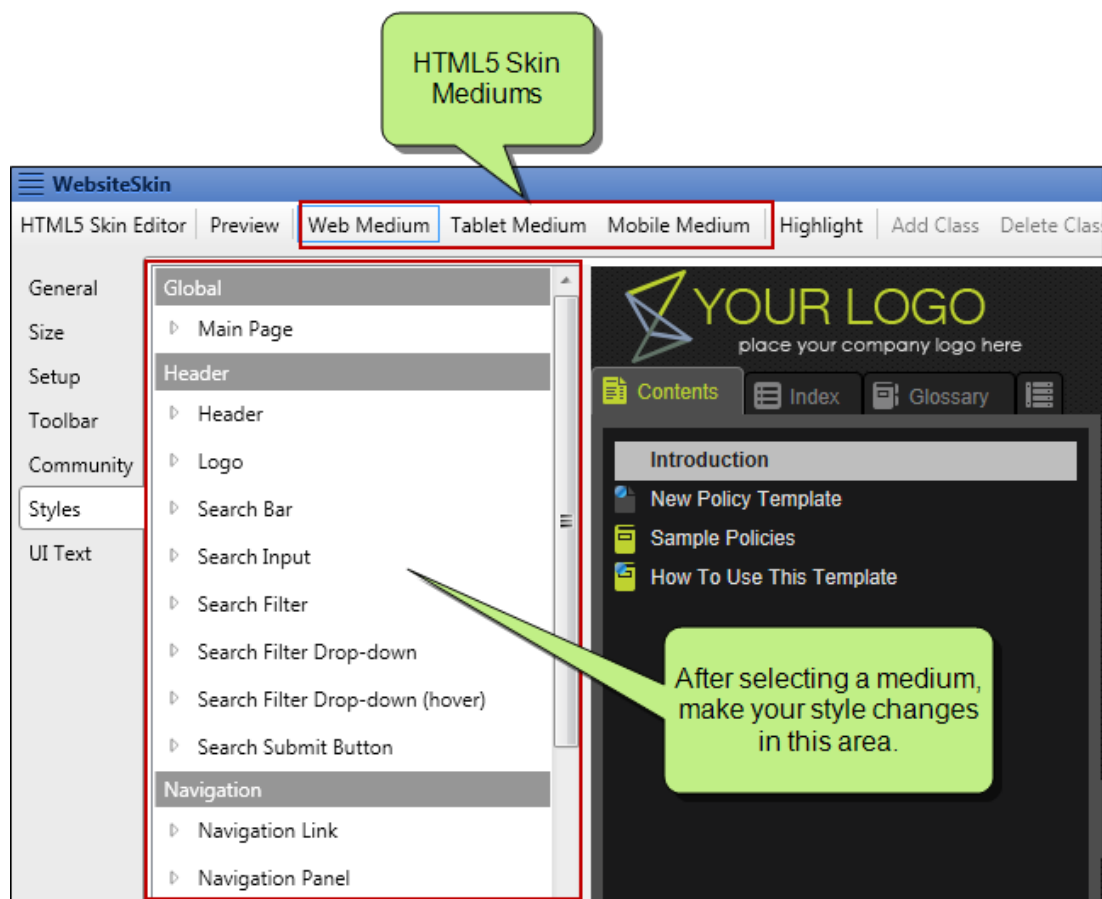
So instead, you enable responsive output in the HTML5 target. When finished, you direct end users to open the same HTML5 output, regardless of their browser or device. On a smart phone, the skin is adjusted automatically.



In the end, you accomplished essentially the same thing that you did with two targets and two skins, but you did it with only one target and skin.

Responsive Skins and Mediums

To support responsive output, HTML5 skins come equipped with three mediums—Web, Tablet, and Mobile. You can use these mediums to specify different looks when necessary for the different devices people might be using to view your output.



- » **Web** When you first open an HTML5 skin, **Web Medium** is selected. With that medium selected, any changes you make to the styles are applied to that medium. This means that when users are viewing your output on a regular browser, that's the look they will see.
- » **Tablet** If you want to change the look that users see when viewing your output on a tablet, select **Tablet Medium** and make your style changes.
- » **Mobile** And if you want to change the look that users see when viewing your output on a mobile phone, select **Mobile Medium** and make your style changes.

Similar to cascading stylesheets (CSS), there is a system of inheritance at work with skin mediums. The Web medium is the default. Most settings in that medium are inherited by the Tablet medium. In turn, settings in the Tablet medium are inherited by the Mobile medium. Therefore, if you want all of the mediums to share the same look (e.g., your company logo), you can set it once in the Web medium and it will automatically be used in all three. If you make any changes in the Tablet medium, those settings will override whatever had been inherited from the Web medium. Likewise, you can make changes in the Mobile medium, which will override any settings inherited from the Tablet medium.



Note: An exception to this inheritance is with navigation icon images. Because these icon images are often different sizes in the Web medium than in the other mediums, most navigation icon images do not pass down from the Web medium to the Tablet medium. However, the Tablet medium does inherit the icon image used for the "Mark as New" feature from the Web medium. Also, icon images are passed down from the Tablet medium to the Mobile medium.



Note: Within each medium, the icons under the TOC Entry style inherit from the icon settings in the Navigation Panel>TOC. Therefore, if no icons are set in the TOC Entry style classes in the Web medium, Flare will use the Navigation Panel>TOC icons in that medium. The same holds true for the Tablet medium; if no icons are set in the TOC Entry classes in the Tablet medium, they will be inherited from the Navigation Panel>TOC icons in the Tablet medium. And the same is the case with the Mobile medium.

How to Enable Responsive Tripane Skins

For Tripane output, you can use the following steps to enable responsive output:

1. Open an HTML5 Tripane skin.
2. Select the **Setup** tab.
3. In the **Responsive Output Settings** section, select **Enable responsive output**.
4. (Optional) Responsive output works by automatically changing the display once the viewer reaches a certain width. You can change the maximum width at which the display changes from one medium to the next. Use the following to specify responsive settings for a skin. For more information about making topic content responsive, see "Responsive Content" on page 32.
 - » **Tablet-Max-Width** Enter the number of pixels for the maximum width of a Tablet view.
 - » **Mobile-Max-Width** Enter the number of pixels for the maximum width of a Mobile view.

EXAMPLE

Let's say you keep the default settings of 1279 pixels for the Tablet maximum width and 767 pixels for the Mobile maximum width.

If you generate output and view it in a regular browser with the window maximized, you will see the skin style settings for the Web medium.

You then click and drag the browser window to reduce it. Once the width of the browser window reaches 1279 pixels of width, the display changes to show the skin style settings associated with the Tablet medium.

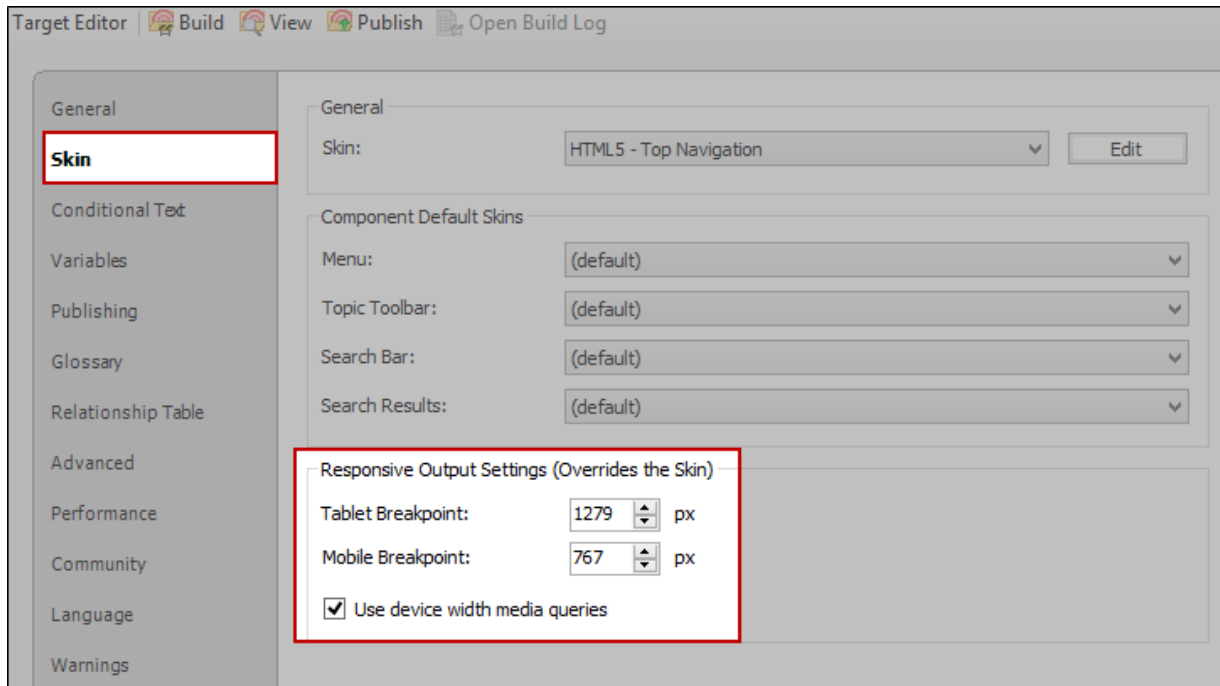
You continue to reduce the size of the browser window. Once the width reaches 767 pixels, the display changes to show the settings for the Mobile medium.

Therefore, the Web medium is named as it is because the largest size is usually meant for a regular web browser. The next size down is often used for tablets, such as iPads, so that medium is called "Tablet." And the smallest size is controlled by the Mobile medium, because a display that small is usually seen in mobile phones.

5. Click  to save your work.

Device Width Media Queries—Top Navigation and Tripane Skins

In addition to the fields in the Skin Editor, there are additional responsive output settings in the Target Editor that are available for both Top Navigation (see "Top Navigation Output" on page 63) and Tripane outputs. One setting lets you enable **device width media queries** for responsive output.



This means that the responsive nature of the skin depends on the device being used to view the output (browser, tablet, or mobile phone), rather than on merely the width of the screen.

Similar to the fields in the Skin Editor, you can set values to tell Flare at which sizes to change the display.

- » **Tablet Breakpoint** Enter the number of pixels for the maximum width of a Tablet view.
- » **Mobile Breakpoint** Enter the number of pixels for the maximum width of a Mobile (or phone) view.

EXAMPLE

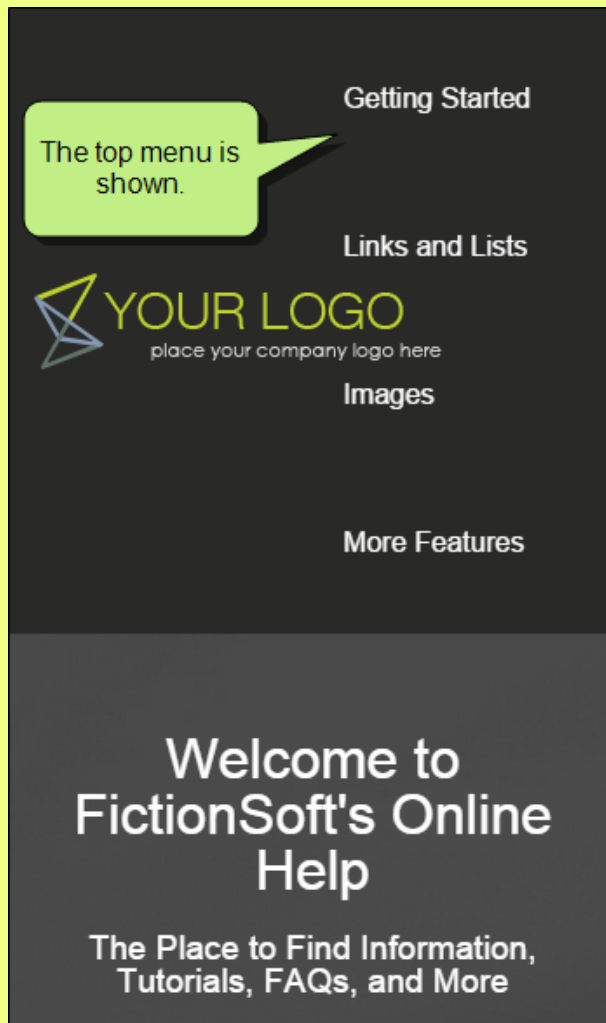
Let's say you disable **Use device width media queries**. When you view the output maximized in a browser, you will see the top menu.



When you drag the browser window, making it smaller so that its resolution is lower than the number you provided in the Tablet Breakpoint field, the display changes. Now you won't see the top menu anymore. Instead, you will see the side flyout menu. That's because the responsiveness is based on the width of the output in the browser, rather than on the width of the output in the device itself.



Now suppose you enable **Use device width media queries**. When you view the output maximized in a browser, it will look just like it did before, with the top menu visible. But now when you make the browser window smaller, the display stays the same, even if you reduce it all the way down to the mobile width settings.



But if you view that same output on an actual tablet or mobile phone, it will display with the side fly-out menu.





Tip: Even if you want to base your responsive skin on the device width, you might find it most useful to leave the "Use device width media queries" option disabled while you are still editing content. This lets you test your responsive output more easily by dragging the browser to different sizes. Then when you're ready to generate and publish your final output, enable the check box.



Tip: If you want to disable the top menu when viewed on a browser, and use only the side flyout menu that is usually reserved for tablets and mobile devices, you can set the tablet width value to a very high number.



Note: If you have a Tripane skin in your project and you enter responsive output settings in both the Skin Editor and Target Editor, the settings in the target take precedence. However, this is not true if you *have not* yet made a change to the tablet or mobile breakpoint fields in the Target Editor, but you *have* made changes to them in the Skin Editor. In that case, the numbers from the changed Skin Editor will be used.

Responsive Content

Supported In:



Styles and media queries can be used to make your content responsive. This lets you present information—both its substance and structure—differently depending on the size of the screen or device.



Note: It is technically possible to create responsive content in other formats besides HTML5, such as DotNet Help, HTML5, WebHelp, WebHelp Plus, WebHelp Mobile, and WebHelp AIR. However, you will notice that best results are achieved with HTML5 targets, and in particular, Top Navigation output. Many of the other formats are able to produce responsive content only in very basic ways.

Media Queries

Media queries let you define different style settings, depending on the features of the device that is showing the output. Therefore, media queries are central to making responsive content possible.

Similar to mediums in Flare, media queries use the `@media` rule. The difference is that mediums in Flare can be associated with targets of any format that you create in your project. On the other hand, media queries adjust the display automatically, depending on the device (e.g., browser, tablet, smart phone) and its features (e.g., width, orientation). In addition, media queries are compatible only with some online output formats, with HTML5 being the recommended target type.

In a stylesheet, a media query is often written like this:

```
@media [media type] and ([media feature])
```

To do this, you must open the stylesheet in the Internal Text Editor, because the regular user interface showing the Simplified and Advanced views does not support the addition of media queries at this time.

The media type is an older kind of specification, and many media types have been deprecated. The two most common media types you might see are screen and print.

There are several ways to write a media query and many kinds of media features. Some of the most common media features are width, max-width, and min-width.

As far as units of measurement are concerned in media queries, pixels are quite common, but there are arguments that `em` (a relative unit of measurement similar to percentage) is preferred.

EXAMPLE

Here is a common way to write a media query in a stylesheet:

```
@media screen and (max-width: 40em)
```

And just as is the case with mediums, any settings that follow within curly brackets are associated with that media query, like this:

The diagram shows a CSS code block with several callouts explaining its parts:

- Media Query:** Points to the line `@media screen and (max-width: 40em)`.
- Opening Curly Bracket:** Points to the opening curly brace `{` of the media query block.
- Closing Curly Bracket:** Points to the closing curly brace `}` of the media query block.
- Anything contained within the opening and closing curly brackets is associated with the media query.** This callout points to the styles inside the media query block: `MadCap|searchBarProxy.homeSearchBar { width: 100%; }`, `div.topichero h1 { font-size: 2em; }`, and `div.topichero h3 { font-size: 1.2em; }`.
- These styles and settings are not within the media query brackets. Therefore, they are associated with the default medium in the stylesheet.** This callout points to the styles outside the media query block: `MadCap|slideshow { border-style: none; box-shadow: none; }` and `html.HomePage { mc-master-page: url('../MasterPages/HomePage.flmsp'); }`.

```
@media screen and (max-width: 40em)
{
  MadCap|searchBarProxy.homeSearchBar
  {
    width: 100%;
  }

  div.topichero h1
  {
    font-size: 2em;
  }

  div.topichero h3
  {
    font-size: 1.2em;
  }
}

MadCap|slideshow
{
  border-style: none;
  box-shadow: none;
}

html.HomePage
{
  mc-master-page: url('../MasterPages/HomePage.flmsp');
}
```

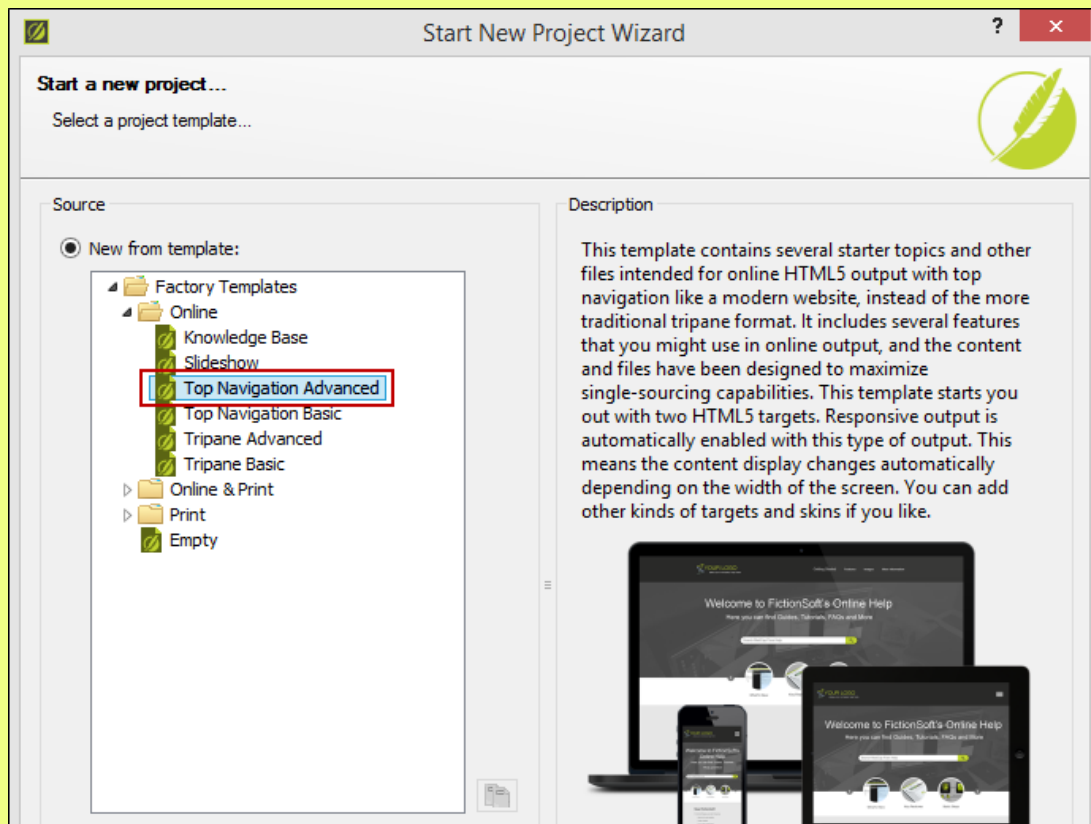
For more information about media queries, you can refer to external online resources, such as w3schools.-com (http://www.w3schools.com/cssref/css3_pr_mediaquery.asp).

Top Navigation Templates and Foundation Grids

In Flare's HTML5 Top Navigation project templates, a special method has been used to create responsive content on the Home page. As a result, sections of content are stacked on top of one another when viewed on small screens. This responsiveness was done through the use of the Foundation grid framework provided by Zurb (<http://foundation.zurb.com/>).

EXAMPLE

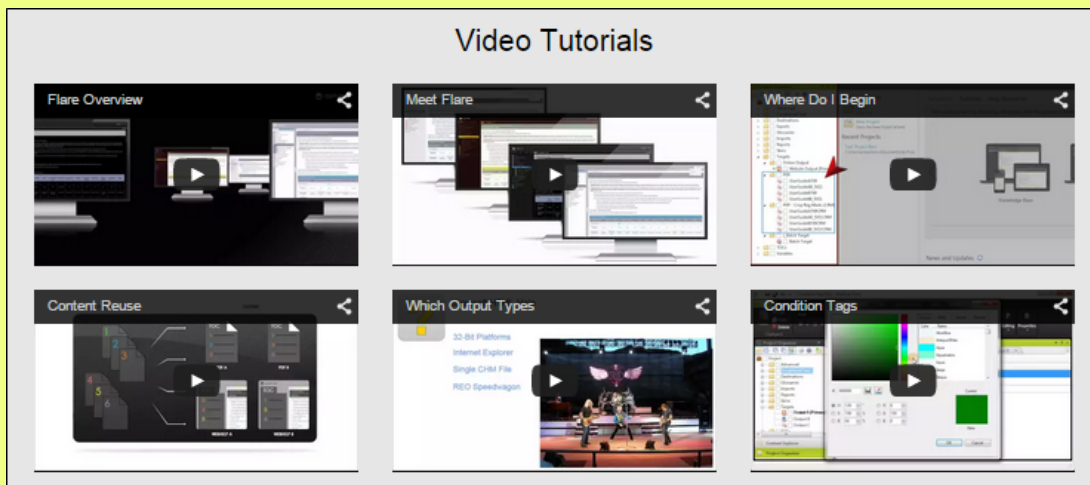
Let's say you create a new project using Flare's factory Top Navigation Advanced template.



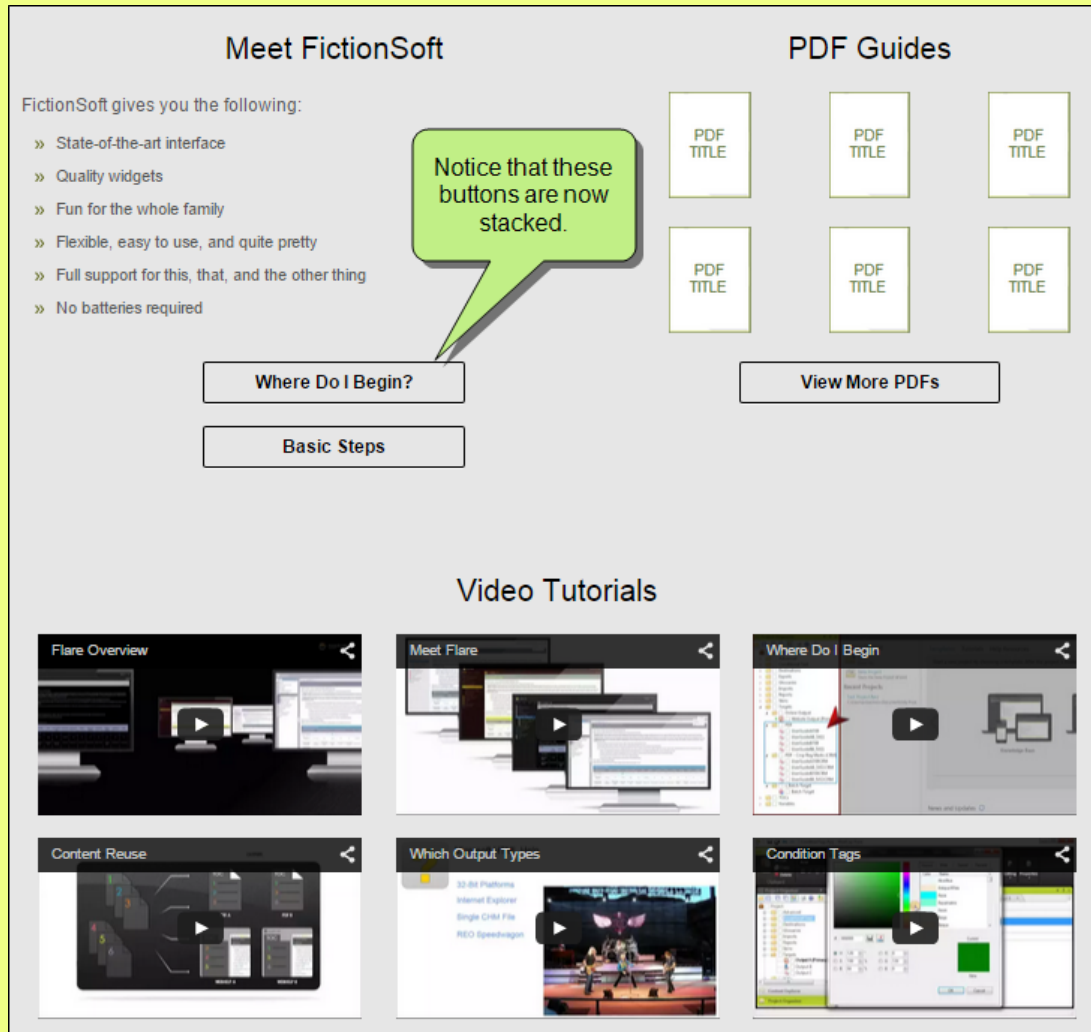
When you generate the output and view it on a large screen, you will notice a couple of compartmentalized sections on the Home page. The first section shows some text and a bullet list on the left and a series of images linking to PDF manuals on the right.



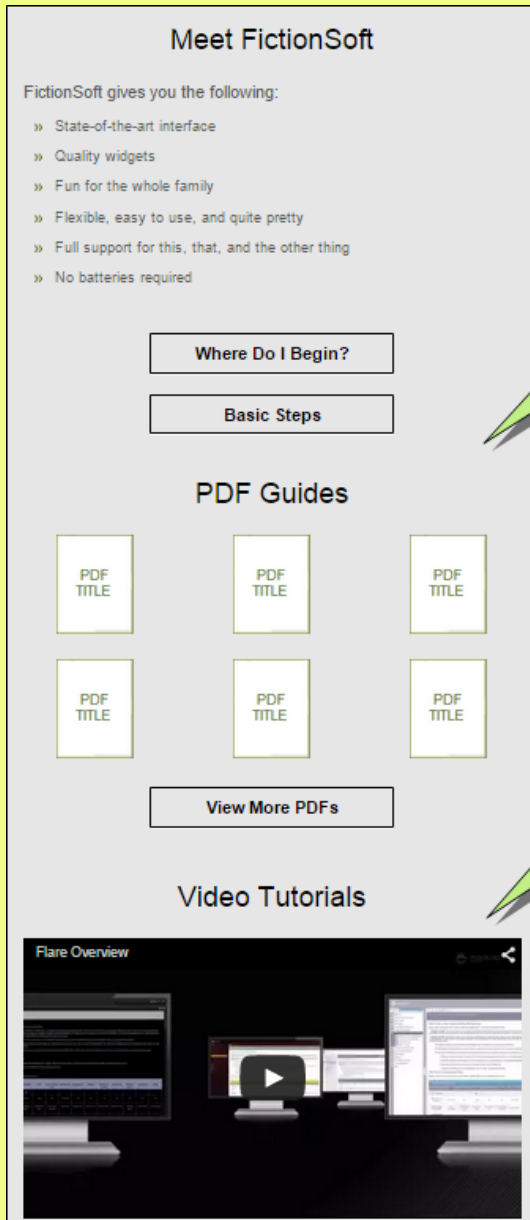
Another section shows a group of six embedded YouTube videos.



All of these sections use styles that allow for responsive output. When the Home page is viewed on a smaller tablet, the content adjusts so that it looks like this:



And when it is viewed on a small mobile device, the content adjusts again so that it looks like this:



Now the first two sections are stacked on top of one another.

And the videos are each displayed at full width.

THE FOUNDATION STYLESHEET AND GRID SYSTEM

Zurb Foundation (<http://foundation.zurb.com/>) lets you freely download files such as CSS, HTML, and java script in order to take advantage of the responsive output framework they've created. To support responsive content in Flare's Top Navigation project templates, a Foundation CSS file has been downloaded and placed in the following location where you installed Flare:

C:\Program Files (x86)\MadCap Software\MadCap Flare V11\Flare.app\Resources\
WebHelp2\Desktop\Skins\Fluid\Stylesheets\foundation.5.5.0



Note: The default installation folder for 32-bit applications on 64-bit platforms is C:\Program Files (x86).... For 32-bit platforms it is C:\Program Files....



Note: It is recommended that you do not edit the Foundation stylesheet or move it from this location in the application installation folder. However, if you want, you can make a copy of the stylesheet, place it in any Flare project, and make changes to it. Any stylesheets in your project have precedence over any stylesheets located in your Flare installation folder.



Note: There are other frameworks similar to Foundation (Bootstrap may be the most popular alternative), but we are discussing Foundation only because that is the framework used in Flare's Top Navigation project templates.

FOUNDATION MEDIA QUERIES AND STYLES

The Foundation stylesheet contains numerous media queries with various widths. Within these media queries are several style classes and settings designed to produce responsive content in a grid structure. Some of the most important style classes are as follows:

- » *row*
- » *column* and *columns*
- » *small-1* through *small-12*
- » *medium-1* through *medium-12*
- » *large-1* through *large-12*

These classes can be used within `<div>` tags in the HTML code. When you see a `<div>` tag, think of an invisible rectangle holding content.



In the previous image, the ellipsis (...) represents other tags and content that you might add within the `<div>` tag.

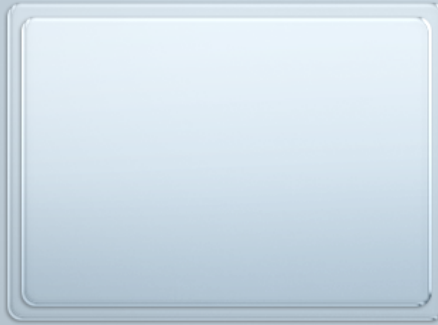


Note: When using this Foundation framework, you need to work in the text editor when editing a topic, rather than using the regular interface. Part of the reason for this is that the Foundation grid system often requires more than one style class associated with a single tag, and in this version of Flare, you can do this only by writing in the code. To display the topic code, click the **Text Editor** tab when viewing a topic.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <html xmlns:MadCap="http://www.madcapsoftware.com/Schemas/MadCap.xsd" Ma
3 <head>
4 <link href="Resources/Stylesheets/StylesForHomePage.css" rel="st
5 </head>
6 <body>
7 <div class="topichero">
8 <div class="title-text">
9 <div class="small-12 small-centered columns centered-tex
10 <h1 class="whiteheader">Welcome to <MadCap:variable
11 </h1>
12 <h3 class="whiteheader">The Place to Find Informatio
13 </div>
14 <MadCap:searchBarProxy data-mc-skin="/Project/Skins/Home
15 </div>
16 </div>
17 <div class="slideshow">
18 <MadCap:slideshow MadCap:SlidesDisplayed="3" MadCap:Adaptive
19 <MadCap:slide title="Getting Started">
20 <p><a href="A_Getting Started/Getting Started.htm"><
21 </p>
22 </MadCap:slide>
23 <MadCap:slide title="Basic Steps">
24 <p><a href="A_Getting Started/Basic Steps.htm"><img sr
25 </p>
26 </MadCap:slide>
27 <MadCap:slide title="What's New">
28 <p><a href="A_Getting Started/Whats New.htm"><img sr
29 </p>
30 </MadCap:slide>
31 <MadCap:slide title="FAQs">
```

If you nest <div> tags in the code, you will have one rectangle within another.

```
<div>  
  <div>...  
  </div>  
</div>
```



To create a Foundation grid, start by adding the *row* style class to a <div> tag.

```
<div class="row">  
</div>
```

In one or more secondary `<div>` tags, add the *column* or *column* style classes. Each time you do this, a new column is created within that row.



By using the *small*, *medium*, and *large* style classes, you can specify the width of each column, depending on the device. In the Foundation stylesheet, the *small* classes are contained within the media query intended for mobile devices, the *medium* classes are contained within the media query intended for tablets, and the *large* classes are contained within the media query intended for larger screens. Each of these numbered style classes has the same width percentage, regardless of whether it is small, medium, or large.

EXAMPLE

Small-1, medium-1, and large-1 classes each have a width setting in the stylesheet of 8.33333%.

Small-7, medium-7, and large-7 classes each have a width setting in the stylesheet of 58.33333%.

Small-12, medium-12, and large-12 classes each have a width setting in the stylesheet of 100%



Note: You may not need to use the large style classes if your intended output design for large screens is essentially the same as that for medium devices (i.e., tablets). That's because the large design will automatically inherit from the medium settings, unless you specify otherwise. In the same way, the medium design automatically inherits from the small design, unless you specify otherwise.

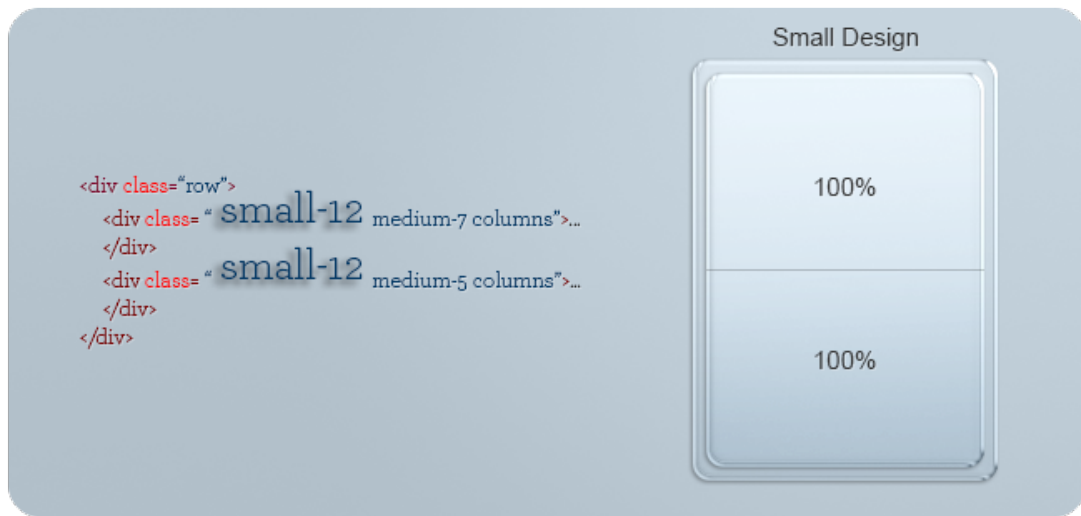
Wherever you have a `<div>` tag with a *column* or *column* class, you can add *small*, *medium*, or *large* style classes that either use 12 or add up to 12 when taken together.

```
<div class="row">  
  <div class="small-12 medium-7 columns">...  
</div>  
  <div class="small-12 medium-5 columns">...  
</div>  
</div>
```

In the image above, notice that the *medium* style classes add up to 12 ($7+5=12$). This tells Flare that on medium devices such as tablets—as well as on larger screens, since the large design inherits the medium settings—the first column displays at 58.33333% of the total width, and the second column displays at 41.66667% of the total width.



And because it is possible to have multiple columns that each have a style class with 12 (100% width)—which is most common for small designs—the output may appear as if it has multiple rows when viewed on that small screen. The truth is that there is still one row, but because you've told Flare to make each column 100% width, the next column is stacked below the one on top, giving the appearance of another row.



So when viewing a topic on a tablet or large screen, you might see columns of content side-by-side like this:



But when viewing the same topic on a smart phone, the content would get shifted so that it looks more like this:





Note: There are other style classes related to the Foundation grid system that you may also use. For more information on these classes and the grid in general, see:

<http://foundation.zurb.com/docs/components/grid.html>



Note: A couple of media queries similar to those in the Foundation stylesheet were added to the StylesForHomePage.css file in Flare's Top Navigation project templates. Within these media queries, we added a few more style settings intended for content when displayed on tablets and mobile devices. However, most of the styles created for responsive content in the Top Navigation project templates are being borrowed from the Foundation stylesheet previously mentioned.

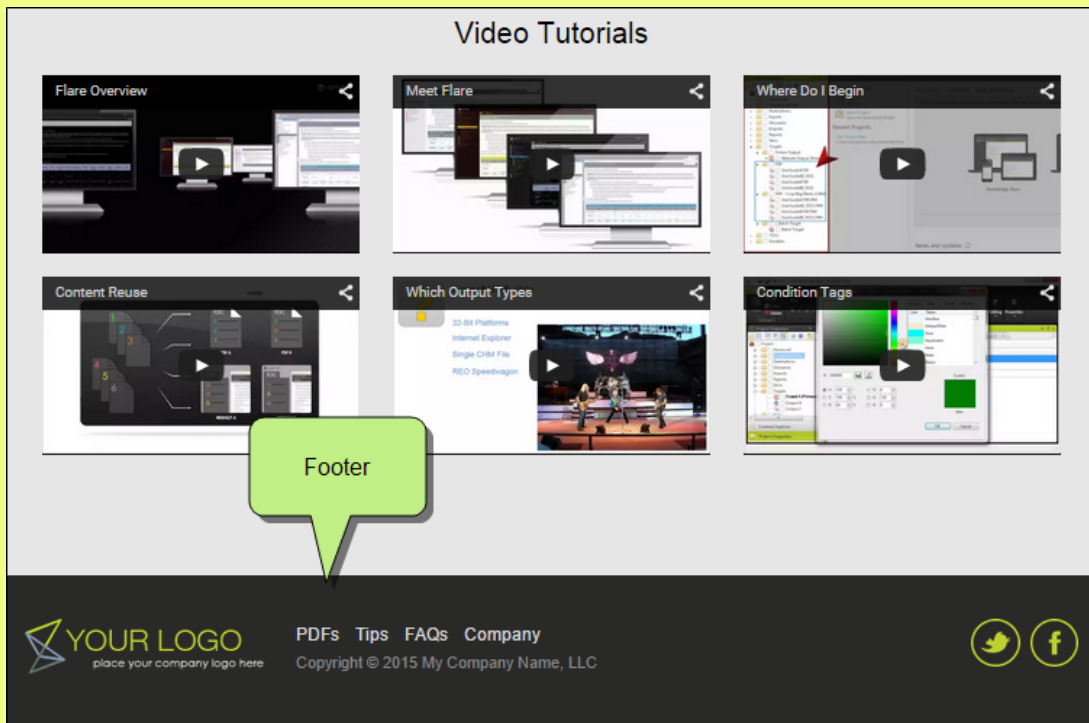
MODIFYING FOUNDATION GRIDS IN THE OUTPUT

Instead of creating a new responsive grid from scratch, you are likely to use existing grids created by others, such as those in Flare's Top Navigation templates. However, you may discover that you need to make adjustments to the grid in order to fit your content.

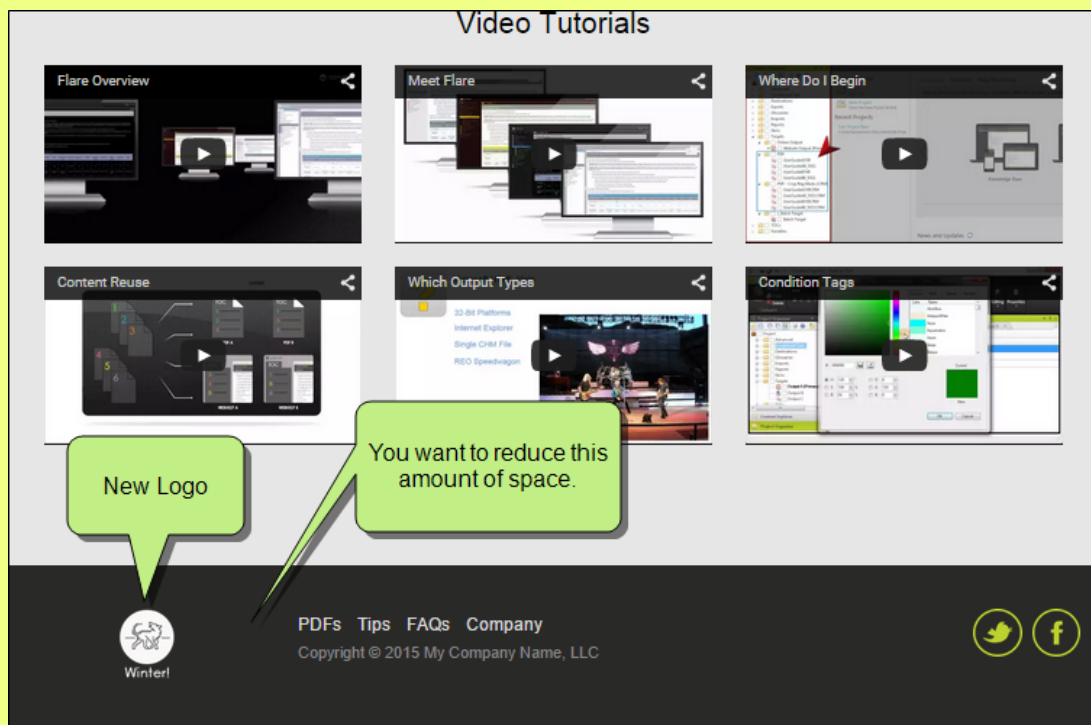
One efficient way to do this is to use the "Inspect Element" feature found in many browsers. After generating your output, you can view it in a browser such as Google Chrome. By resizing the browser window, you can see how well the content adjusts for different screen widths. If you come across an issue, you can right-click where the issue occurs and select **Inspect element** (the exact name of the option may be different, depending on your browser). A pane opens, showing the output code for the page you are viewing. If you make changes to the code in that pane, the output changes accordingly so that you can see the results. After you are satisfied with the changes, you can return to Flare and make the same changes in your topic.

EXAMPLE

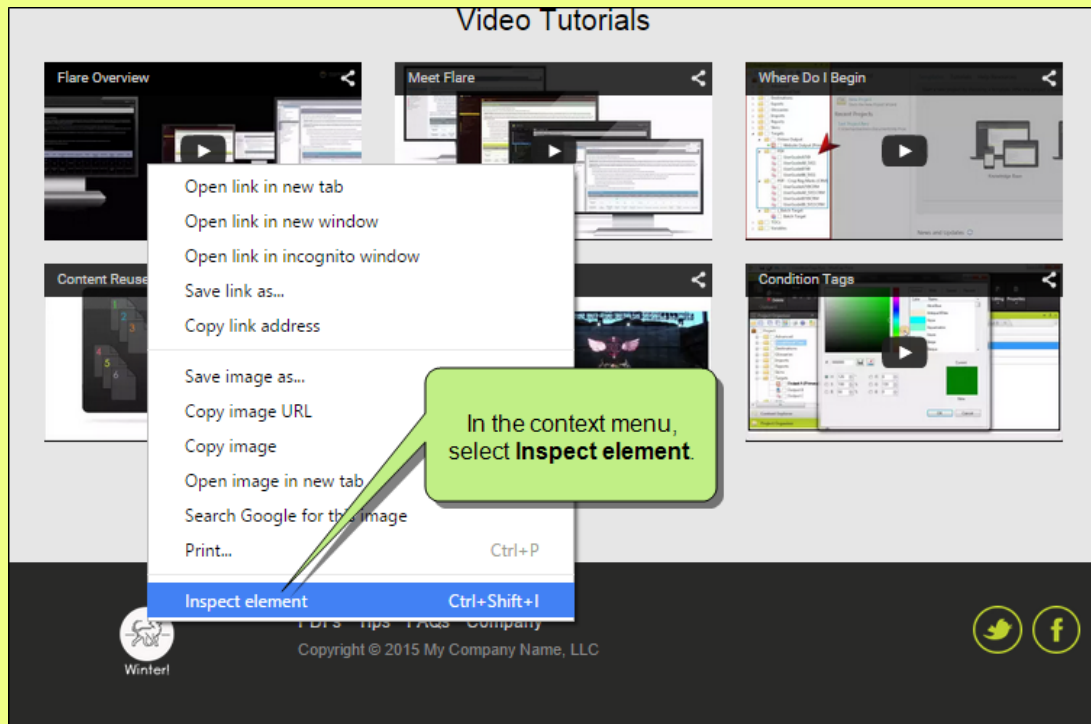
Let's say you use Flare's Top Navigation Advanced template when creating a new project. This template includes a master page with a footer that displays at the bottom of the Home page.



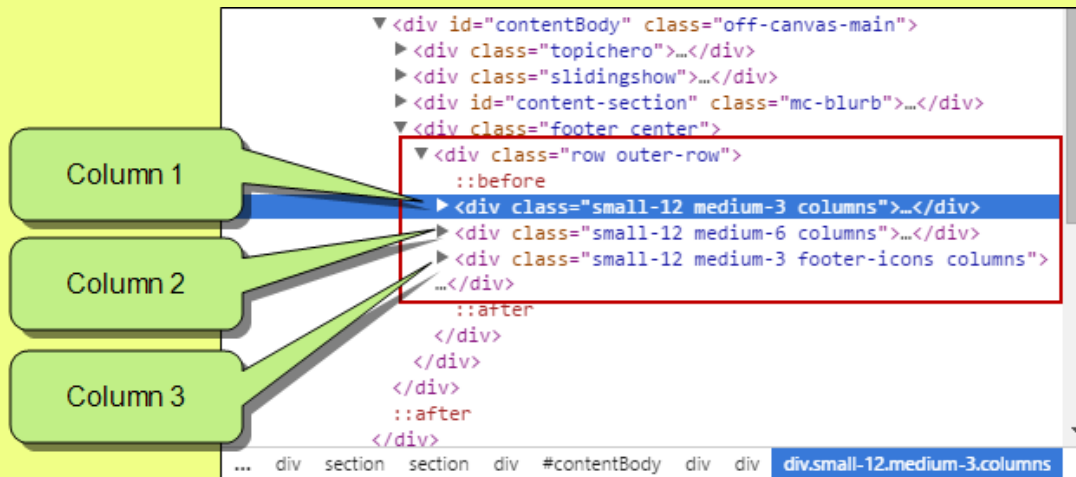
In the master page for the Home topic, you replace the generic logo with your own. After doing this, you generate and view the output in Google Chrome. When you resize the browser to a small width that would be used for a mobile device, the logo looks fine because it appears centered by itself. But when enlarging the browser width to a tablet size and larger, you notice that because your logo is much narrower than the generic one from the template, there is extra space between it and the text links after it.



So you right-click on the extra space after your logo and select **Inspect element**.



In the window pane that opens in the browser, you see that some of the code is highlighted, indicating the location where you right-clicked in the output. The code indicates that the footer area consists of three columns.



The first column uses the medium-3 style class when the output is viewed on a tablet and larger screens. This means it is set to use 25% of the width.

The diagram illustrates the relationship between Bootstrap column classes and their visual output in a footer. A code editor snippet shows the following HTML structure:

```
<div class="footer center">
  <div class="row outer-row">
    ::before
    <div class="small-12 medium-3 columns">...</div>
    <div class="small-12 medium-6 columns">...</div>
    <div class="small-12 medium-3 footer-icons columns">...</div>
  ...</div>
  ::after
</div>
</div>
::after
</div>
```

Two callouts provide context:

- A callout labeled "Column 1 Area" points to the first column in the footer layout, which contains a logo and the text "Winter!".
- A callout states: "The first column is using medium-3, or 25% of the width."

The footer layout itself consists of a dark grey bar with the following elements:

- On the left: A logo with a stylized animal head and the text "Winter!" below it.
- In the center: Navigation links "PDFs Tips FAQs Company" and a copyright notice "Copyright © 2015 My Company Name, LLC".
- On the right: Social media icons for Twitter and Facebook.

The second column uses the medium-6 style class when the output is viewed on a tablet and larger screens. This means it is set to use 50% of the width.

The second column is using medium-6, or 50% of the width.

```
<div class="footer center">
  <div class="row outer-row">
    ::before
    <div class="small-12 medium-7 columns">...</div>
    <div class="small-12 medium-6 columns">...</div>
    <div class="small-12 medium-3 footer-icons columns">
      ...</div>
    ::after
  </div>
</div>
::after
</div>
```

Column 2 Area

PDFs Tips FAQs Company
Copyright © 2015 My Company Name, LLC

The third column uses the medium-3 style class when the output is viewed on a tablet and larger screens. This means it is set to use 25% of the width.

The third column is using medium-3, or 25% of the width.

```
<div class="footer center">
  <div class="row outer-row">
    ::before
    <div class="small-12 medium-3 columns">...</div>
    <div class="small-12 medium-3 columns">...</div>
    <div class="small-12 medium-3 footer-icons columns">...</div>
    ::after
  </div>
</div>
::after
</div>
```

Column 3 Area

The screenshot shows a web browser displaying a footer. On the left is a circular logo with a stylized 'W' and the text 'Winterl' below it. In the center are links for 'PDFs', 'Tips', 'FAQs', and 'Company', followed by the copyright notice 'Copyright © 2015 My Company Name, LLC'. On the right are two circular social media icons for Twitter and Facebook. A red vertical line separates the footer content from the social media icons.

By changing the class in the first column to medium-1 and the class in the second column to medium-8, you are telling Flare to set the first column to 8.33333% width and the second column to 66.66667% width. As a result, the first column becomes much narrower and the text links in the second column appear to slide to the left.

The first column was changed to medium-1, and the second column was changed to medium-8.

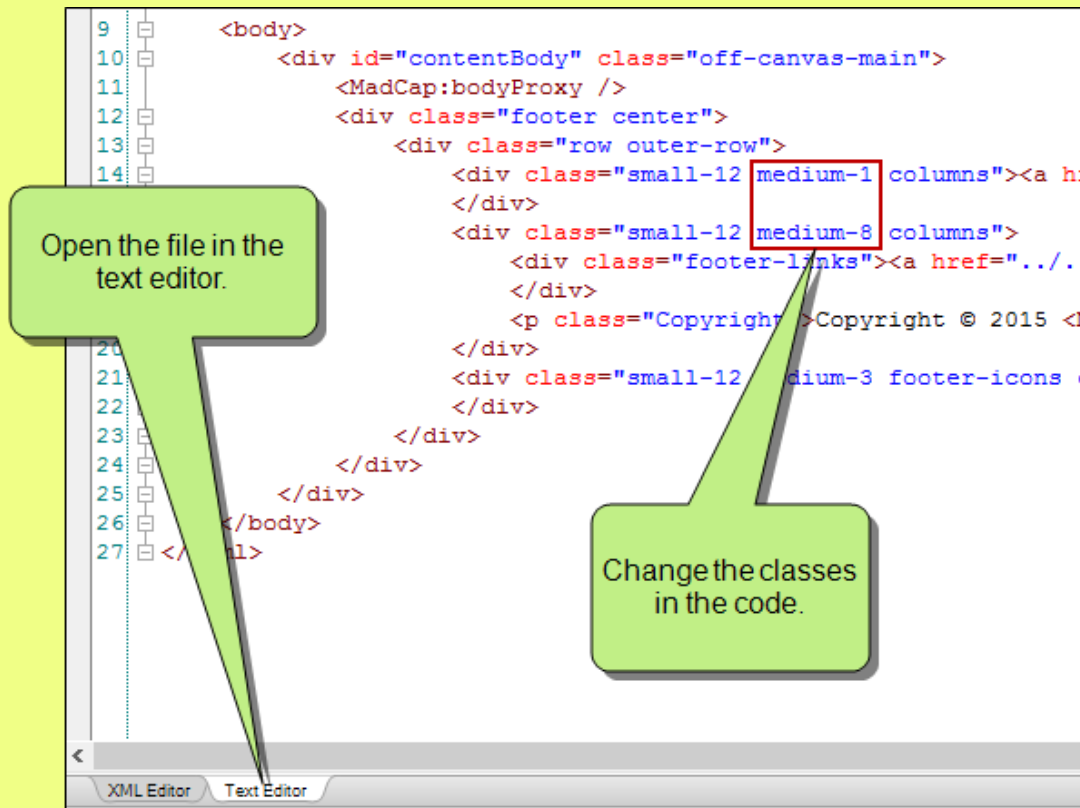
```
<div class="row outer-row">
  ::before
  <div class="small-12 medium-1 columns">...</div>
  <div class="small-12 medium-8 columns">...</div>
  <div class="small-12 medium-3 footer-icons columns">
    ...</div>
  ::after
</div>
</div>
::after
</div>
```

Now there is much less space between the logo and the text links.

PDFs Tips FAQs Company
Copyright © 2015 My Company Name, LLC

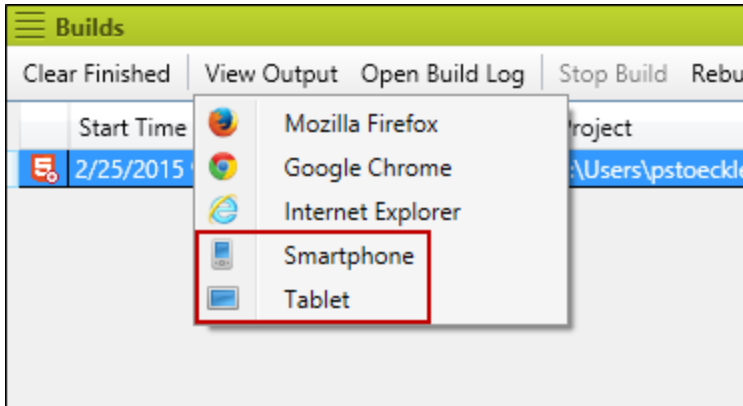
Twitter Facebook

Now that you know the correct code, you return to your Flare master page and change the HTML accordingly. (If this content had been located in a topic, you would make the change in the Flare topic instead of the master page.)



Viewing Responsive Output

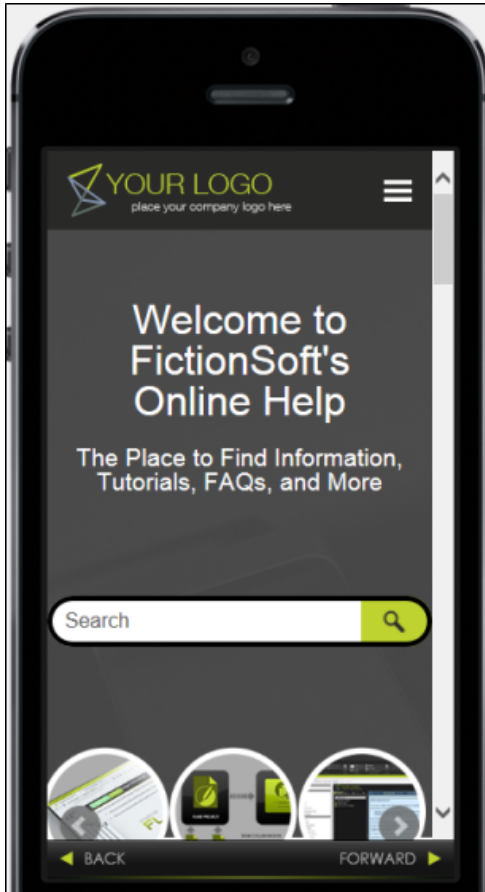
When you finish generating responsive output, the Builds window pane includes view options not only for seeing output on a full browser output, but also on a tablet or smart phone.



If you select the Tablet option, the view displays like this:



If you select the Smartphone option, the view displays like this:



Top Navigation Output

If you generate an HTML5 target, you can create output with top navigation like a modern website. This is possible thanks to a new Top Navigation skin type and other related features. You can even set your target not to use any skin at all, relying on smaller skin components to provide menus, search, and toolbar features.

This is in contrast to the more traditional Tripane output, which includes a toolbar pane at the top, navigation panels on the left, and a main content pane.

This chapter discusses the following:

Frameless Output	64
HTML5 Top Navigation Skin	65
Skin Components and Proxies	66
Skin Set to None for Target	81
Home Topic	82
Stylesheets	84
Responsive Output and Device Width Media Queries	87
Producing Top Navigation Output	93
Simple Conversion to Top Navigation Output	128
Advanced Conversion to Top Navigation Output	137



Frameless Output

HTML5 Top Navigation output is frameless. This has the following benefits:

- » **Better Search Engine Optimization** Top Navigation means better search engine optimization (SEO). This is thanks in part to the absence of iframes. In addition, the output is not dynamically loaded in div tags, but rather the content is flattened, which makes it easier for web crawlers to locate.
- » **Navigation Displays with External Search Results** Top Navigation provides a better experience with external searches and navigation. For example, if you have Tripane output and click on a Google search result for a specific page, that page opens without the surrounding navigation (e.g., TOC) included in that Help system. But with Top Navigation output, that same page would display with its intended navigation.
- » **Improved Scrolling and Zoom in Mobile Devices** When Top Navigation output is viewed on a mobile device, scrolling and zoom features are typically better than they are for Tripane output.

HTML5 Top Navigation Skin

A Top Navigation skin is the primary element involved in creating HTML5 output with navigation (i.e., menu and search bar) at the top of topic pages. Aside from the obvious structural characteristics, a Top Navigation skin is different from an HTML5 Tripane skin in the following ways:

- » **Fewer Tabs in the Skin Editor** When you open a Top Navigation skin, you will notice that it does not have the General, Size, or Toolbar tabs. Those tabs contain fields and features that are pertinent only to Tripane output. For example, with Top Navigation output you do not select navigation elements to include (e.g., TOC, glossary, index), because those types of elements are more prominent in Tripane output. Instead, Top Navigation output puts a premium on search and menu items instead.
- » **Responsive Output Always Enabled** You do not need to turn responsive output on or off, because it is always enabled for Top Navigation skins. However, there are some settings you can provide for responsive output on the Skin tab of the Target Editor. For more details, see "Responsive Output and Device Width Media Queries" on page 87.
- » **Fewer and Different Styles** Because the Top Navigation skin has fewer elements in it, there are fewer fields in the Styles tab of the Skin Editor. Also, there are some styles for menus that are unique to the Top Navigation skin.
- » **Fewer UI Text Fields** Again, with fewer elements involved, there are fewer fields to be concerned about in the UI Text tab.

You can add a Top Navigation skin in the same way that you add other kinds of skins to a project.



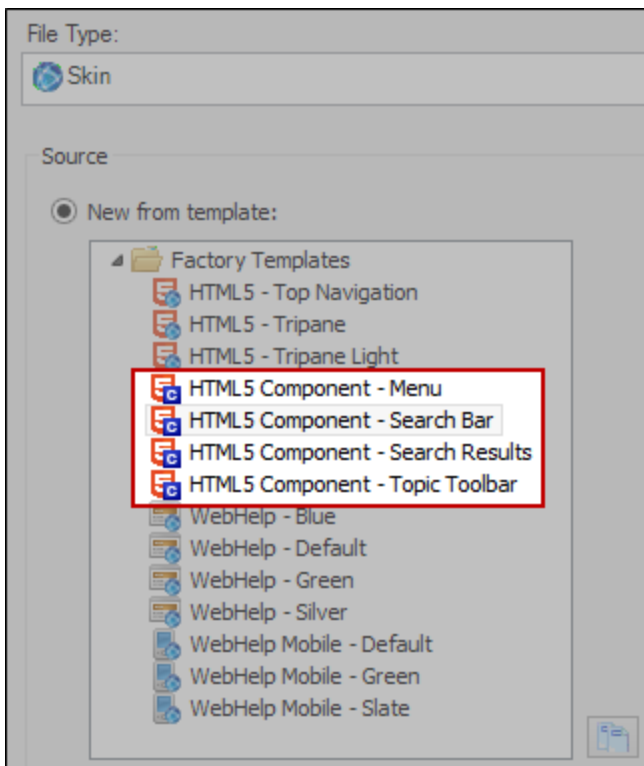
Note: Flare's HTML5 Top Navigation skin does not support project merging.

Skin Components and Proxies

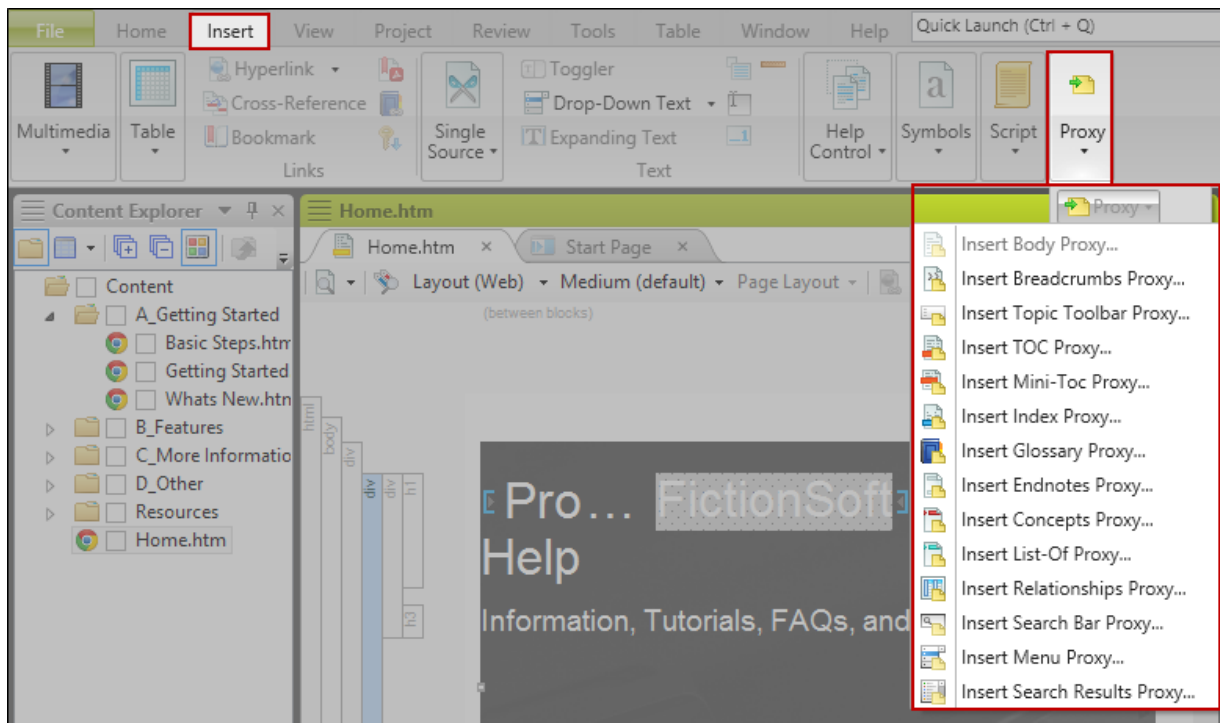
Even if you use a full HTML5 skin to create either Top Navigation or Tripane output, you can also use smaller skin components and related proxies for them. Depending on the type of skin, this allows you to include and design additional search elements, menus, and toolbars in various locations in your output.

A proxy is the element that actually generates the search element, menu, or toolbar when you build output. The related skin component is used to provide a look for it. The proxy is always necessary to generate the desired element, but a skin component is optional. If you do not add a particular type of skin component to your project, Flare provides a default design.

Skin components can be added to a project in the same way that you would add full skins. In the Project Organizer, right-click on the **Skins** folder and from the context menu select **Add Skin**. Then choose the kind of skin component you want to add (Menu, Search Bar, Search Results, or Topic Toolbar).



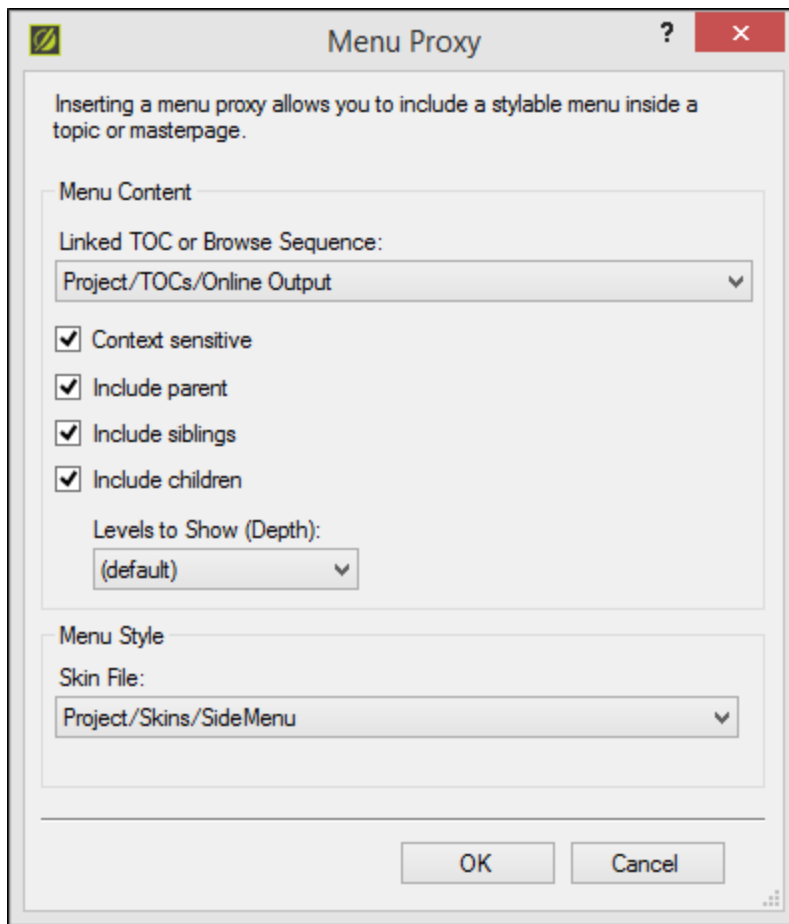
You can add a proxy by clicking in the content file where you want it to be placed. Then from the **Insert** ribbon, select **Proxy>Insert [Name of Proxy]**.



The following skin components and related proxies are commonly used for HTML5 Top Navigation output, but some of them can also be used in Tripane output.

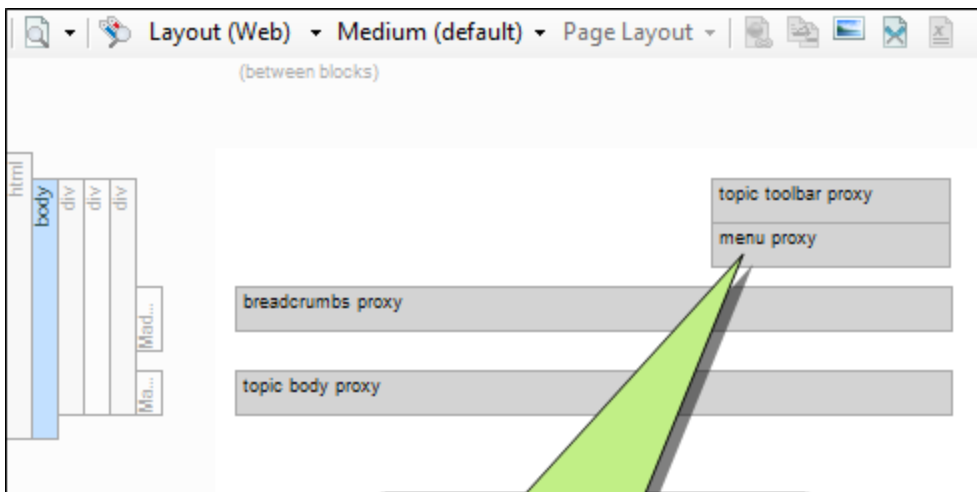
MENU

When you insert a Menu proxy, the Menu Proxy dialog opens.

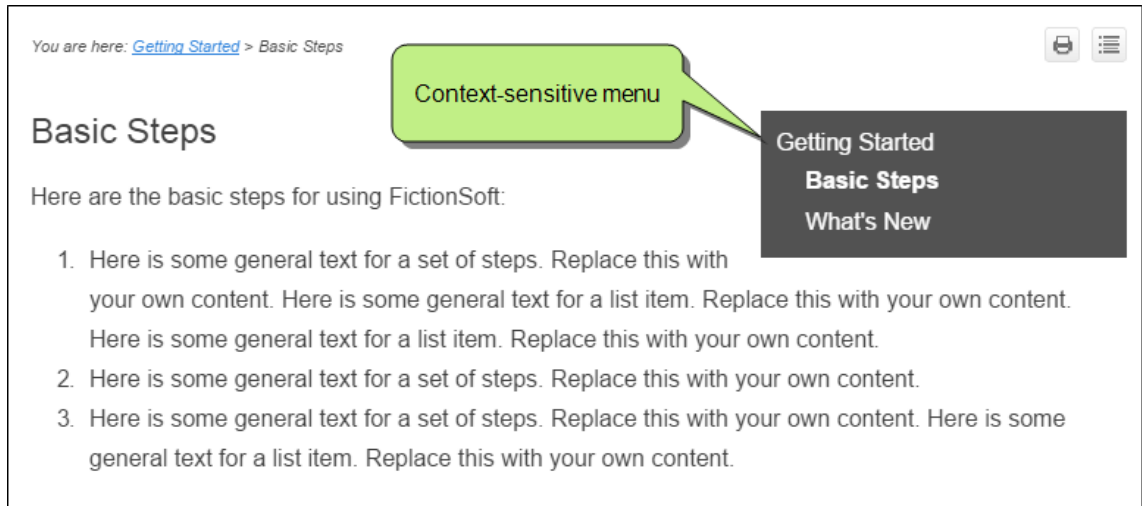


You can select the following options:

- » **Linked TOC or Browse Sequence** If you have more than one TOC file in your project, you can select the one that the Menu proxy should use. It is most common to base a menu on a TOC, but you can also select a browse sequence.
- » **Context sensitive** Select this check box if you want the menu to show only closely related entries in the TOC. In Flare's Top Navigation project templates, this kind of proxy was inserted into a master page to create a side menu for most of the topics in the output.

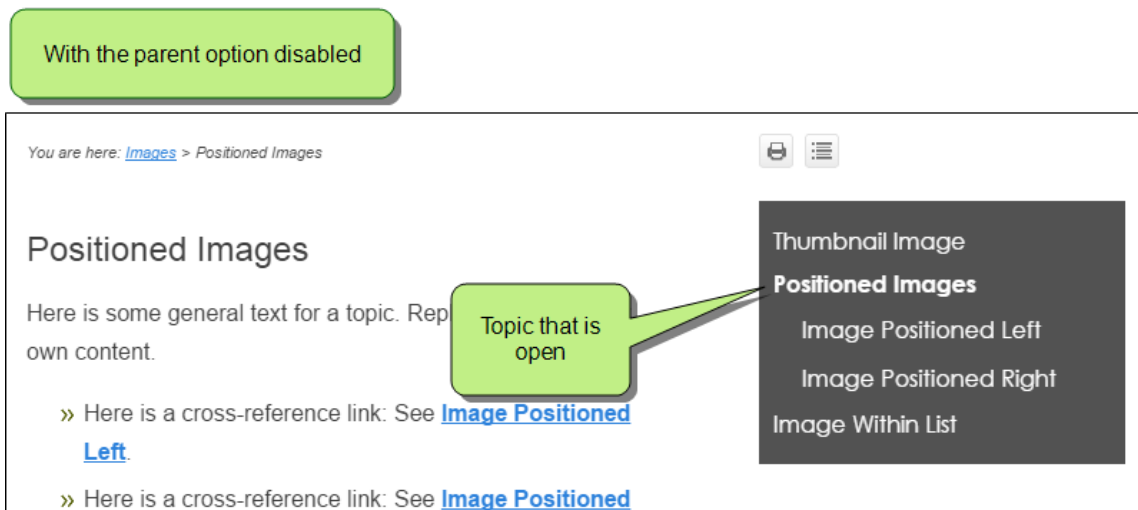
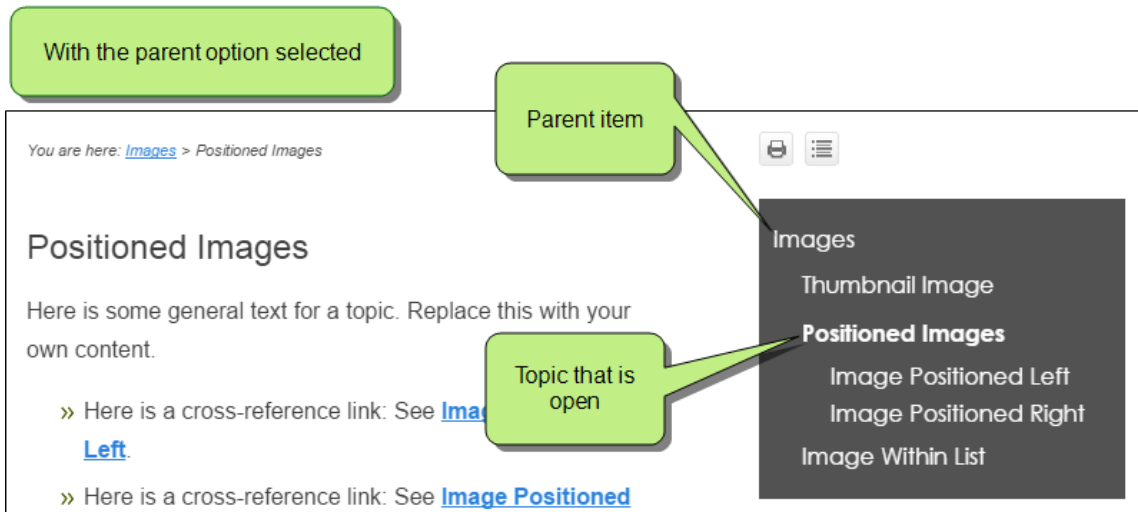


This Menu proxy is set to be context-sensitive. It has been styled to display on the right side of topics with content wrapping under it.



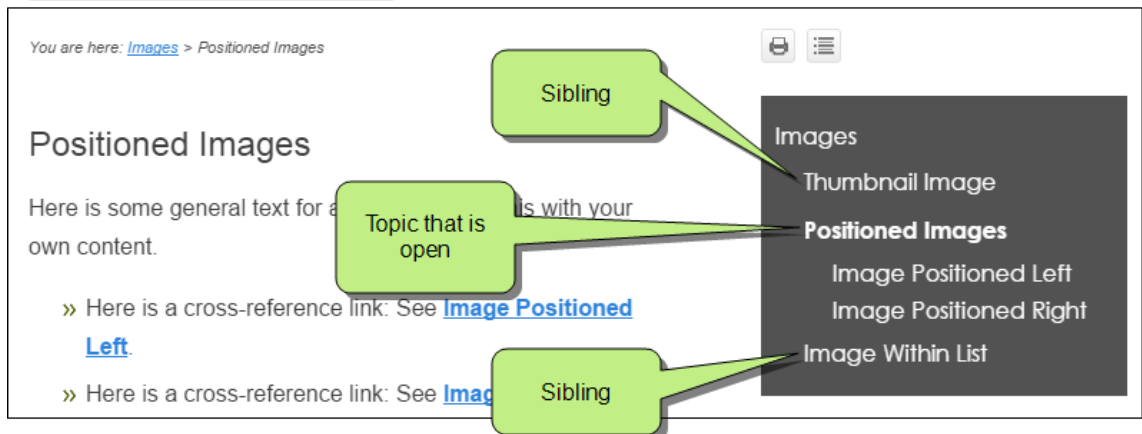
If you do not make the menu context-sensitive, it displays everything in the TOC (depending on the depth level you select).

- » **Include parent** If you have selected the **Context sensitive** option, you can select this check box to include the parent TOC item in the menu.

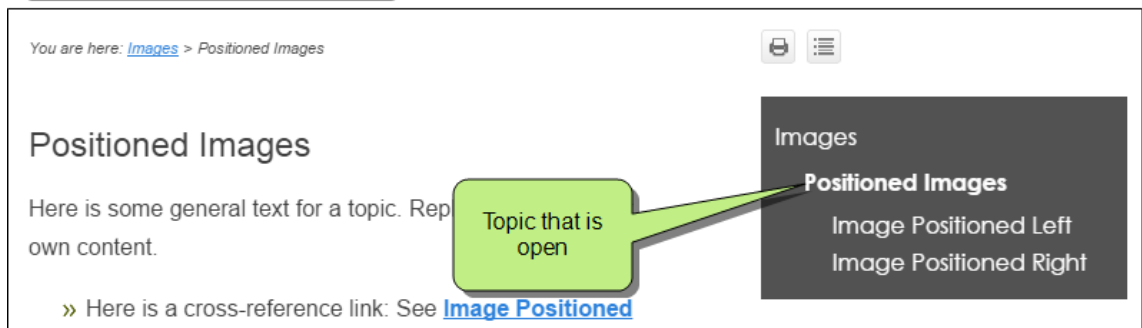


- » **Include siblings** If you have selected the **Context sensitive** option, you can select this check box to include TOC items in the menu that are on the same level as the open topic.

With the siblings option selected

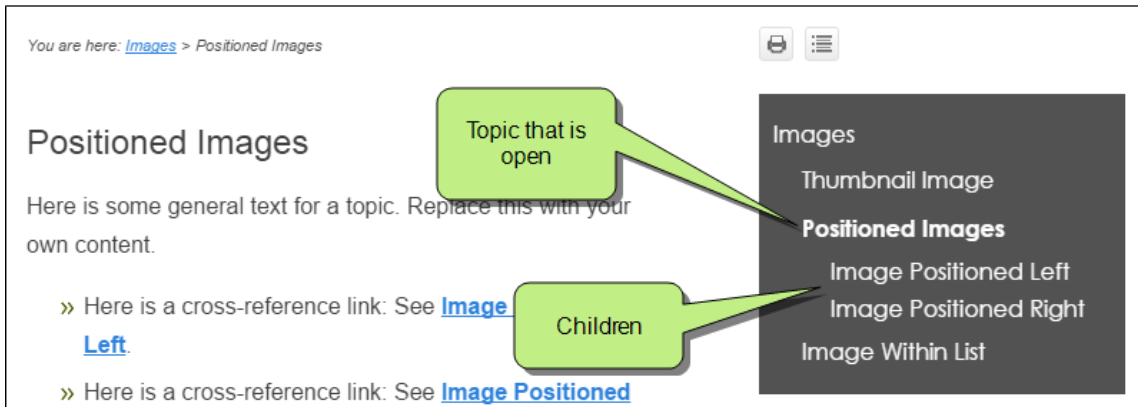


With the siblings option disabled



- » **Include children** If you have selected the **Context sensitive** option, you can select this check box to include TOC items in the menu that are children of the topic that is open.

With the children option selected



With the children option disabled



- » **Levels to Show (Depth)** This lets you choose how many levels of items deep in the TOC to include in the menu. If the **Context sensitive** option is disabled, this refers to the depth level overall for the TOC. If both the **Context sensitive** and the **Include children** options are enabled, it refers to the number of levels under the topic that is open.
- » **Skin File** If you have added a Menu skin component to your project and want to use it to control the look of the menu, you can select it from this field.

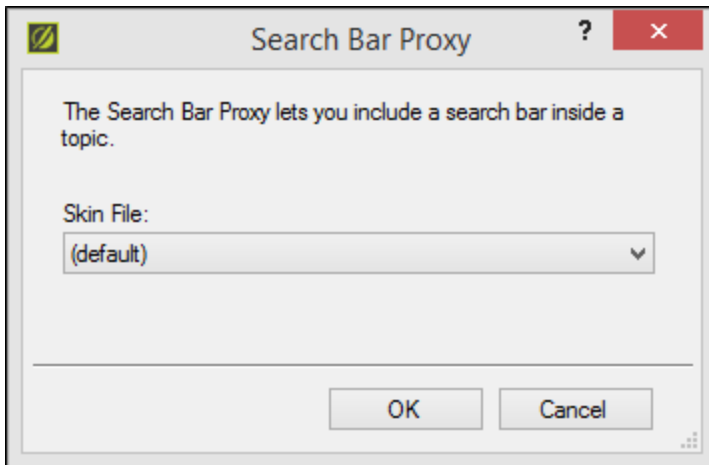
If you do not select a Menu skin component in this field, Flare uses the first one it finds in your project (if one exists). Otherwise, Flare provides a default design.



Note: For HTML5 Tripane output, the Menu proxy and skin component will not work for merged projects or linking to external Help systems. Merging projects is not supported in Top Navigation output at all.

SEARCH BAR

When you insert a Search Bar proxy, the Search Bar Proxy dialog opens.



You can select the following:

- » **Skin File** If you have added a search bar skin component to your project and want to use it to control the look of the search bar, you can select it from this field.

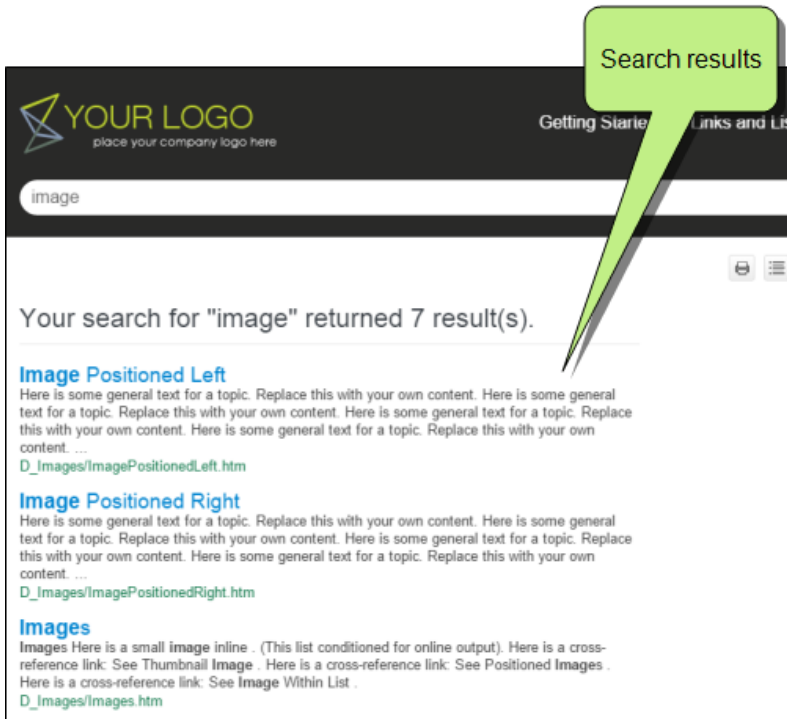
If you do not select a Search Bar skin component in this field, Flare uses the first one it finds in your project (if one exists). Otherwise, Flare provides a default design.



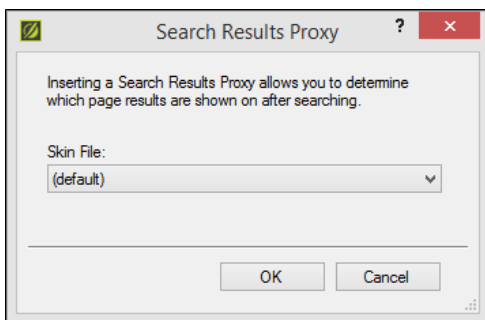
Note: The Search Bar proxy and skin component are not supported in HTML5 Tripane output.

SEARCH RESULTS

The Search Results proxy works with a Search Results skin component to provide a place to display results of an end user's search.



When you insert this kind of proxy, the Search Results Proxy dialog opens.



You can select the following:

- » **Skin File** A skin component lets you control the look of generated search results. If you have added multiple skin components to your project, you can use this field to select the one to associate with this proxy. You can then edit that skin component to change its appearance.



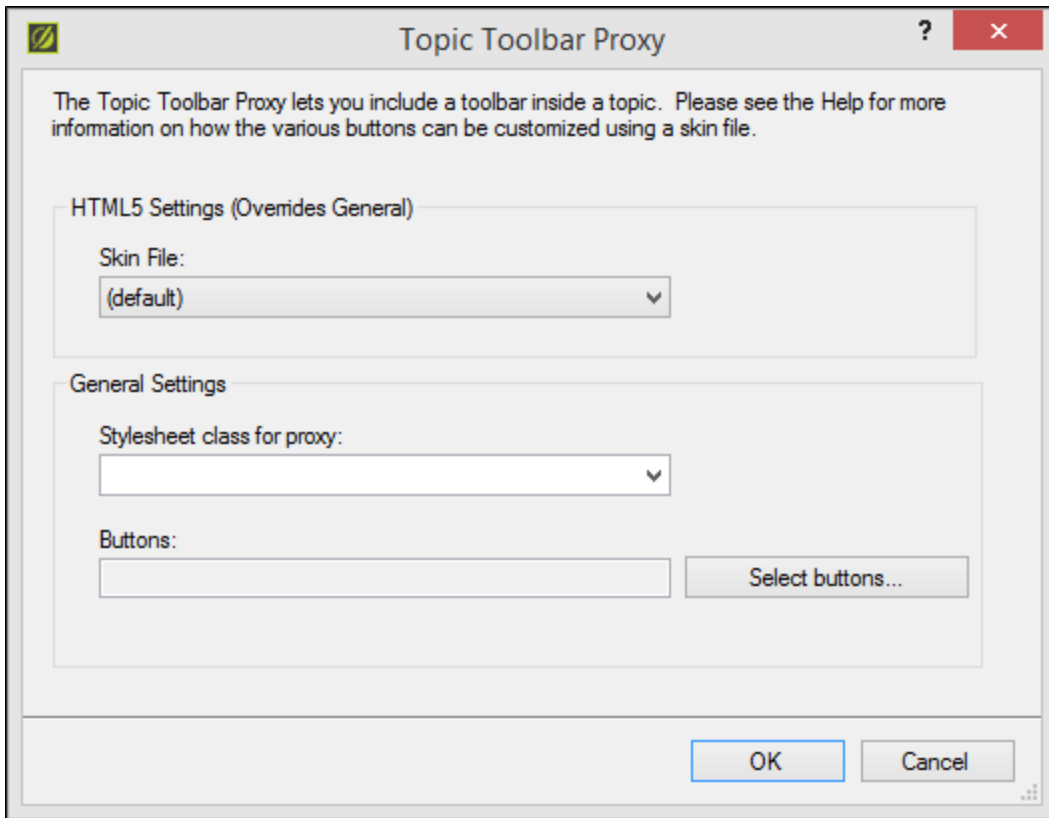
Warning: Do not insert a Search Results proxy into a master page. Insert it only into topics.



Note: The Search Results proxy and skin component are not supported in HTML5 Tripane output.

TOPIC TOOLBAR

When you insert a Topic Toolbar proxy, the Topic Toolbar Proxy dialog opens.



Topic toolbars can be inserted into outputs other than HTML5. However, the Topic Toolbar skin component is supported in HTML5 only. Therefore, the Topic Toolbar Proxy dialog is split into two sections. The HTML5 Settings area pertains only to HTML5 output. The General Settings area pertains to all of the outputs that support topic toolbars, including HTML5. Anything that is set in the HTML5 Settings area overrides what is set in the General Settings area, including the buttons that are selected for the toolbar.

You can select the following options:

- » **Skin File** If you have added an HTML5 Topic Toolbar skin component to your project and want to use it to control the look of the toolbar, you can select it from this field. From the Skin Editor, you can select the buttons to be included in the toolbar. Alternatively, you can select buttons from the Buttons field below.

If you do not select a Topic Toolbar component in this field, Flare uses the first one it finds in your project (if one exists). Otherwise, Flare provides a default design.

- » **Stylesheet class for proxy** You can select a class to affect the look of the entire toolbar. This is an alternative, or supplement, to editing the skin component in the Skin Editor. However, for HTML5 output, using a skin component is the most common method for designing the look of the toolbar.

You might create and use a proxy style class, for example, if you want to add a border around the toolbar. If you do not select a class from this field, the generated toolbar will use the style settings from the parent `MadCap|topicToolbarProxy` style. You have the option of creating a class for this proxy style in the Stylesheet Editor. To do this, select the **MadCap|topicToolbarProxy** style and click **Add Class** to create a class. The class will then be available from this field.

- » **Buttons** You have the option of selecting buttons for a toolbar in the Skin Editor or by using this field. You can click **Select buttons** to open a dialog, then select the buttons to include in the toolbar. For HTML5 outputs, the Topic Toolbar proxy will use whatever settings are specified in a Topic Toolbar skin component (if you have added one to your project), overriding any buttons you may have selected directly in the proxy. If you have not associated a Topic Toolbar skin component with the proxy, Flare will just use the first one it finds in your project. However, for outputs using Standard and Mobile skins, the settings in the proxy take precedence over anything you may have set on the Toolbar tab in the Skin Editor.



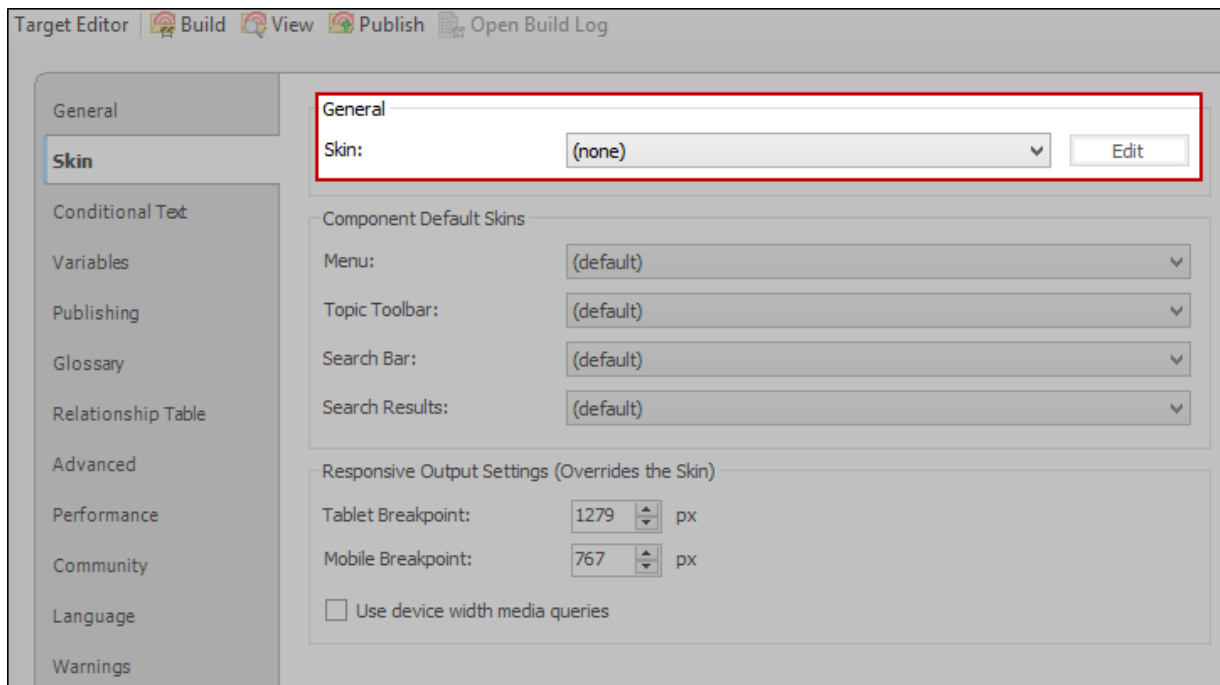
Note: HTML5 Top Navigation output does not support the Next Topic, Previous Topic, and Current Topic Index buttons. Otherwise, it supports the same buttons as HTML5 Tripane output.



Note: You are not limited to one skin component of each type, although that is the most common situation. If you want, you can use multiple skin components of any type. If this is the case, you can associate a "master" skin component with a target. When you do this, the skin component you choose will always be used for any proxy of that same type that you insert for that target, unless you override it by associating a different skin component with a specific proxy that you've inserted.

Skin Set to None for Target

With HTML5 output, you have another choice besides Tripane and Top Navigation output. You can also tell Flare not to use a full skin at all. To do this, open the Target Editor, select the **Skin** tab, and in the **Skin** field select **none**.



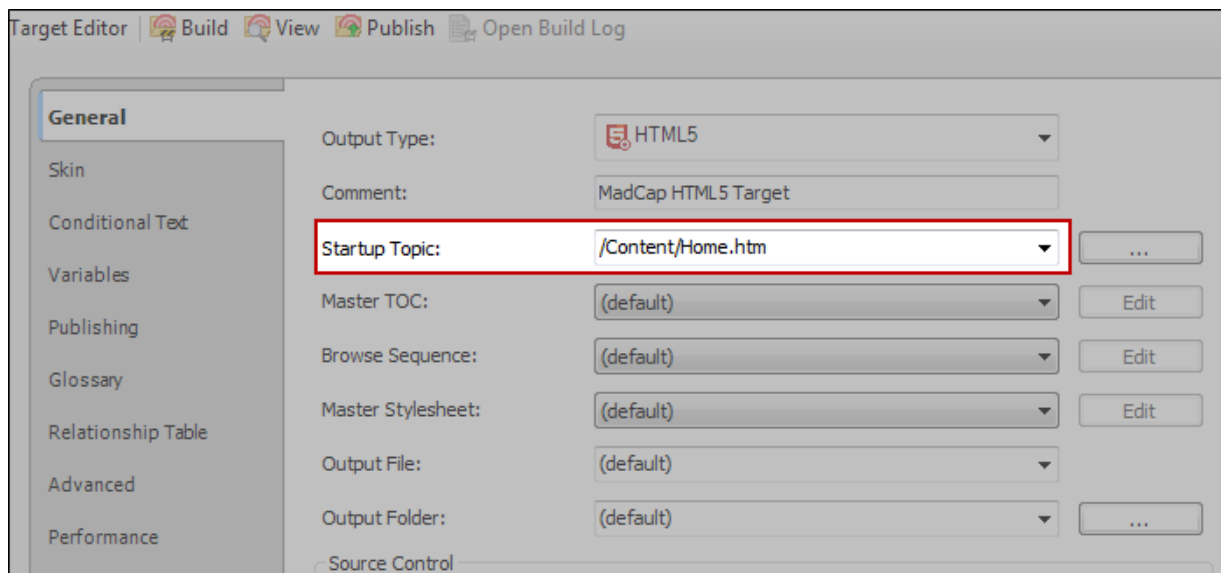
To provide navigation and search for that kind of output, you can simply insert proxies (Menu, Search Bar, Topic Toolbar) and add their related skin components to format them.

Theoretically, you could have HTML5 output without either a full skin or components, but that would be rare. In most cases, you will want to include skin elements of some kind so that end users can more easily find information in your output.

Home Topic

A Home topic is the first page an end user sees when opening your Top Navigation output. However, unlike Tripane output, you usually do not add this topic to your TOC, although you certainly can if you want. Instead, it is standard practice to link to this page from the logo you provide in your Top Navigation skin. See "How to Produce Top Navigation Output" on page 105.

A Home topic is the one that you specify as the startup topic in the Target Editor.



The Home page is just a regular topic, but because a Top Navigation skin is designed to resemble a modern website, you may want this topic to stand out with a different appearance. You might accomplish this in the following ways, all of which are optional:

- » **Unique Search Bar** The Top Navigation skin is designed to show a search bar at the top of all topics. However, you might want to emphasize the search bar on your Home page, especially since search is the most popular and effective way to find specific information.

First, you can open your stylesheet in the Internal Text Editor and enter the following to hide the top search bar that is included with the skin:

```
.row.nav-search
{
    display: none;
}
```

Then you can insert your own Search Bar proxy in a more prominent place in the Home topic. See "Skin Components and Proxies" on page 66.

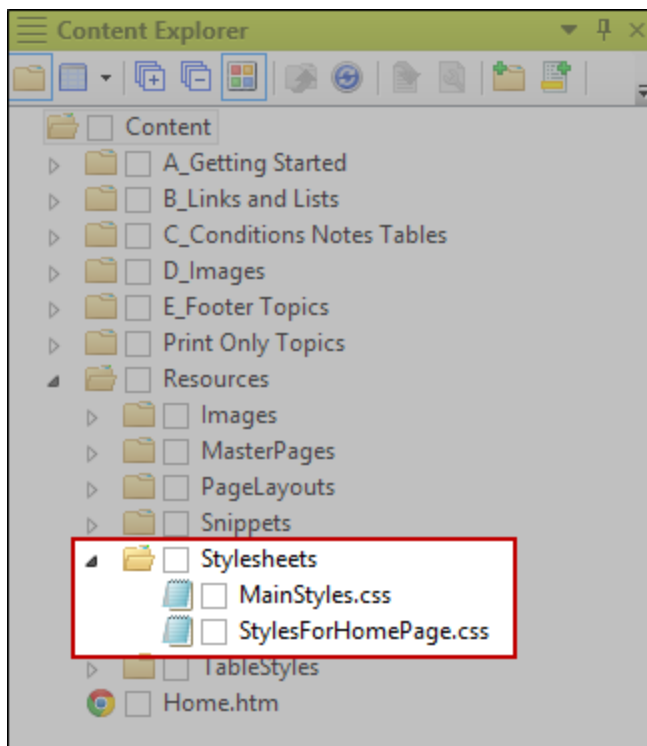
- » **Unique Stylesheet** It is not mandatory that you use multiple stylesheets for your Top Navigation output. However, you may find it easier to make your Home page unique if it has its own styles in its own stylesheet. See "Stylesheets" on the next page.
- » **Unique Background Image** One easy way to make a Home page stand out from the other topics is to place a background image (or watermark) on it. In the world of web design, this is often referred to as a "hero image." You can set a background image in a stylesheet. It is recommended that you use a hero image with a width of 1903 pixels if possible; this size helps to account for even the largest of monitors.
- » **Unique Master Page** By using one master page for your Home topic and another for the rest of your topics, it is easier to give them different looks, as well as different headers and footers.

For an example of how you might make a Home page unique in these ways, see "Advanced Conversion to Top Navigation Output" on page 137.

Stylesheets

Stylesheets are central to any type of output you generate from Flare, and Top Navigation is no exception. In fact, they may be even more important in Top Navigation output in order to achieve the type of modern website look that you want.

If you look at Flare's Top Navigation project templates, you will notice that multiple stylesheets are used—one for the Home page and another for the rest of the topics.



This was done because the Home page has such a unique look compared with the rest of the output, and therefore it required some unique style settings. In particular, several settings are in place to make the content responsive, adapting to the different size screens on which it might be displayed. This is purely optional. A fancy Home page with responsive content is not required for Top Navigation output. But it does work nicely with Top Navigation output and helps to showcase what is possible. In our case, we used some features from ZURB's Foundation framework (see <http://foundation.zurb.com/>), applying them to

content on our Home page. For more details about these styles and how you can make changes to them in Flare's Top Navigation project templates, see "Responsive Content" on page 32.

It would take too long to try to describe each style and property used in the Top Navigation project templates, but here are a few points to consider:

- » **Multiple Stylesheets Not Mandatory** Just because Flare's Top Navigation templates use multiple stylesheets, this does not mean that you need to. You can follow a relatively simple process, performing a few tasks to turn your existing project into one that has Top Navigation output. Then you can simply use your existing stylesheet and styles to make your content (including the Home page) look the way you want, just as you would for any other type of output. For an example of how this might work, see "Simple Conversion to Top Navigation Output" on page 128.
- » **Application Stylesheets** Another thing to keep in mind is that Flare relies not only on stylesheets that you add within your projects, but also on some application stylesheets. By this we mean that there are several stylesheets that can be found within the application folder where you installed Flare. Your project inherits the style definitions that are written in those external application stylesheets. But anything you set in your project stylesheet takes precedence over the same styles that might be found in an application stylesheet. In order to create the kind of look and responsiveness that you see in the Top Navigation templates, certain styles are being borrowed from one particular application stylesheet called "Foundation." So if you look in the topics and master pages found in Flare's Top Navigation templates and you aren't sure where a style is coming from, there's a good chance it is being inherited from the Foundation application stylesheet. For more about this subject, see "Responsive Content" on page 32.
- » **Borrow Styles from the Flare Templates** Chances are probably pretty good that you want to spend more of your time doing actual writing and less time trying to figure out how to style everything, especially complex designs. One solution is to borrow the styles that are used in Flare's Top Navigation templates by importing certain files into your existing project. For an example of how this might work, see "Advanced Conversion to Top Navigation Output" on page 137.
- » **Media Queries—Different Style Settings for Various Screen Sizes** Media queries can be used in your stylesheet to account for how content might shift when viewed on screens of different sizes. If you look at the stylesheet used for the Home page in Flare's Top Navigation project templates, you might notice a couple of lines that look like mediums. These are actually a bit different than mediums; they are called "media queries." One media query is named "only screen and (max-width: 64.063em." This media query contains style settings intended for medium-sized screens, such as tablets. There is another media query called "screen and (max-width: 40em)." This medium contains

style settings for small screens, such as smart phones. And then there are many other media queries that can be found in the Foundation application stylesheet. For an example of how this might work, see "Advanced Conversion to Top Navigation Output" on page 137.

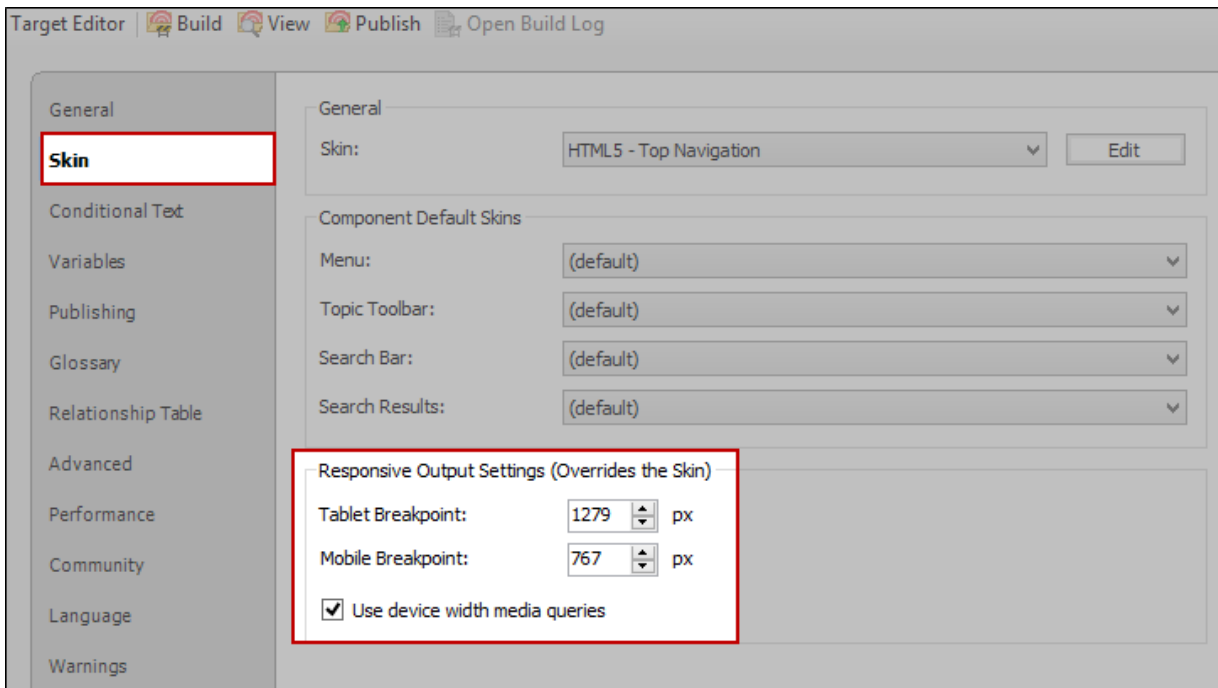
For more information see the online Help or the *Flare Styles Guide*.

Responsive Output and Device Width Media Queries

Responsive output works by automatically changing the display once the viewer reaches a certain width. You can change the maximum width at which the display changes from one medium to the next. Responsiveness is possible for both HTML5 skins and topic content. But whereas making a skin responsive requires little or no effort, making content responsive usually takes more manual effort that involves styles and media queries.

With Tripane skins, you can enable or disable responsive output on the Setup tab in the Skin Editor. But for Top Navigation skins, you will notice that the responsive output section is not included in the Skin Editor. That's because responsive output is always enabled for Top Navigation output.

In addition to the fields in the Skin Editor, there are additional responsive output settings in the Target Editor that are available for both Top Navigation and Tripane outputs. One setting lets you enable **device width media queries** for responsive output.



This means that the responsive nature of the skin depends on the device being used to view the output (browser, tablet, or mobile phone), rather than on merely the width of the screen.

Similar to the fields in the Skin Editor, you can set values to tell Flare at which sizes to change the display.

- » **Tablet Breakpoint** Enter the number of pixels for the maximum width of a Tablet view.
- » **Mobile Breakpoint** Enter the number of pixels for the maximum width of a Mobile (or phone) view.

EXAMPLE

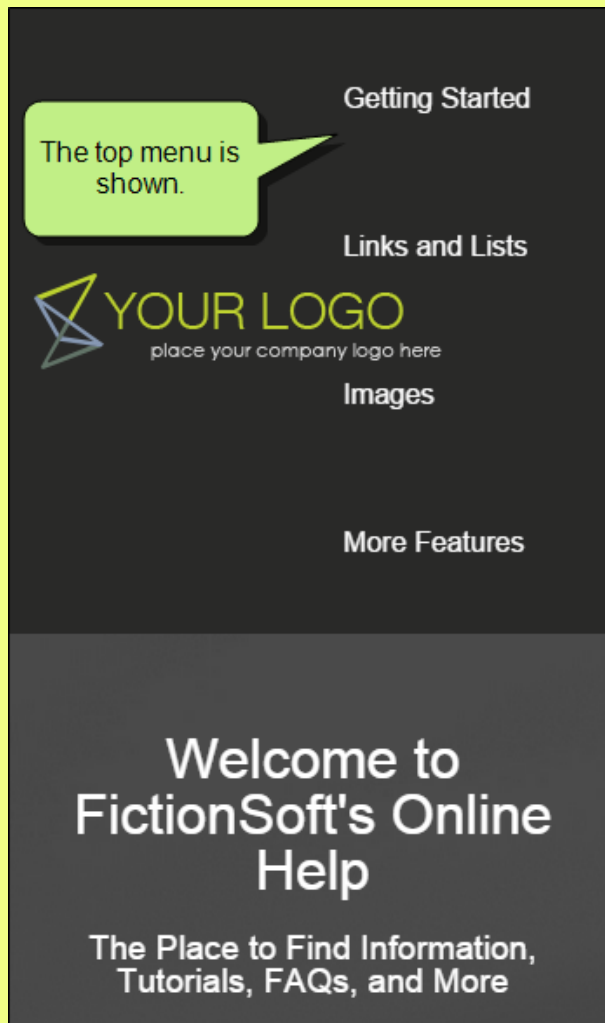
Let's say you disable **Use device width media queries**. When you view the output maximized in a browser, you will see the top menu.



When you drag the browser window, making it smaller so that its resolution is lower than the number you provided in the Tablet Breakpoint field, the display changes. Now you won't see the top menu anymore. Instead, you will see the side flyout menu. That's because the responsiveness is based on the width of the output in the browser, rather than on the width of the output in the device itself.



Now suppose you enable **Use device width media queries**. When you view the output maximized in a browser, it will look just like it did before, with the top menu visible. But now when you make the browser window smaller, the display stays the same, even if you reduce it all the way down to the mobile width settings.



But if you view that same output on an actual tablet or mobile phone, it will display with the side fly-out menu.





Tip: Even if you want to base your responsive skin on the device width, you might find it most useful to leave the "Use device width media queries" option disabled while you are still editing content. This lets you test your responsive output more easily by dragging the browser to different sizes. Then when you're ready to generate and publish your final output, enable the check box.



Tip: If you want to disable the top menu when viewed on a browser, and use only the side flyout menu that is usually reserved for tablets and mobile devices, you can set the tablet width value to a very high number.



Note: If you have a Tripane skin in your project and you enter responsive output settings in both the Skin Editor and Target Editor, the settings in the target take precedence. However, this is not true if you *have not* yet made a change to the tablet or mobile breakpoint fields in the Target Editor, but you *have* made changes to them in the Skin Editor. In that case, the numbers from the changed Skin Editor will be used.

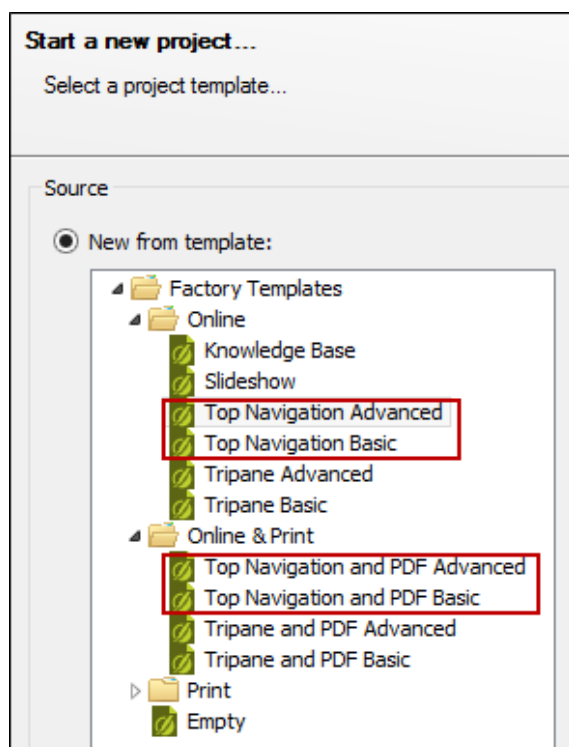
Producing Top Navigation Output

There can be just a few steps or many steps when producing Top Navigation output. It all depends on how much or how little you want to do. Notice that most of the steps described below are optional. You might decide to follow just a few of the steps, or you might complete all of them for the maximum amount of customization.

OPTIONS FOR CREATING TOP NAVIGATION OUTPUT

You can approach Top Navigation output in three basic ways:

- » **Create New Project Using Flare Top Navigation Template** The easiest way to create this kind of output is to use one of Flare's HTML5 Top Navigation templates when you create a new project, making adjustments as necessary and replacing the content with your own. If you go this route, you may find yourself following few, if any, of the steps below.





Note: Notice that the templates for the more traditional online outputs have also been renamed to "Tripane."

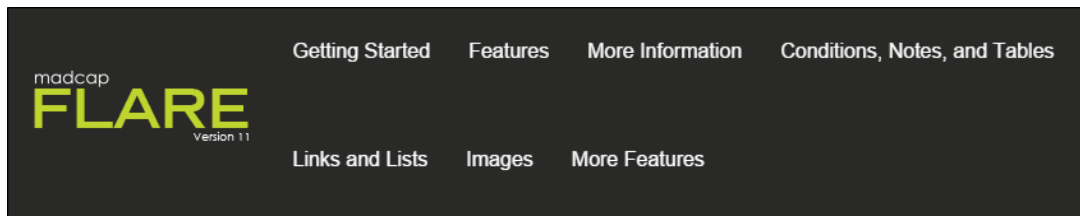
- » **Convert Existing Project** A second option is to create this kind of output in an existing project all by yourself, adding and modifying the elements described below. This option gives you the most flexibility, and it can be quite simple. On the other hand, if you want to create elaborate topic pages that work really well with responsive design, this approach can require more skill and knowledge of CSS. If you elect this option, follow the steps below, completing as many or as few of the optional tasks as you want. For an example of how this might work, see "Simple Conversion to Top Navigation Output" on page 128.
- » **Convert Existing Project by Incorporating Pieces from a Flare Template** Finally, you can use a combination of the first two methods. You can create a small project from one of Flare's templates and then import or copy various pieces from it into your existing Flare project as necessary. For instance, from the template you might want to import the Home page, as well as the stylesheet and master page that go with it, but otherwise you plan to do much of the work using the pieces that are already inside your existing project. For an example of how this might work, see "Advanced Conversion to Top Navigation Output" on page 137.

BEFORE YOU BEGIN

Before you start creating Top Navigation output for an HTML5 target, consider the following information and tips. Much of the work and time involved with Top Navigation output actually has to do with planning and preparation, especially if you are working with existing content.

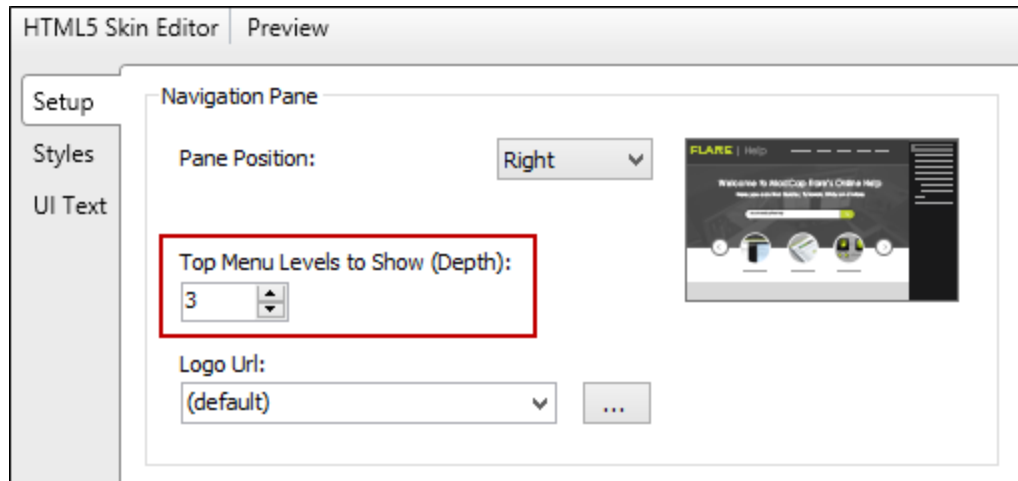
» **Limit the Number of Menu Items** For Top Navigation output, the menu at the top of pages is based on the structure and contents of your TOC file in the Project Organizer. But this menu has a design that emulates a modern website, not a traditional Help system. Therefore, you should try to limit the number of TOC books and entries under them. Following are a few ways to deal with this issue.

» **Keep First-Level Items Few and the Text Short** When your first-level TOC items are more in number than the width of your content can handle, they wrap around to the next line.



This will work, but it looks cleaner to have a single row of menu items at the top. So you can limit the number of first-level books and items in your TOC file. Also, if you keep the text for those items relatively short, you can fit more of them in a single row. We recommend keeping the number of first-level menu items to five or fewer. This may require some reorganization of your TOC file.

- » **Set the Menu Depth Level** It is a best practice to try to have no more than three levels of menus (the root menu and two submenus) at the top. This is the default setting on the Setup tab of the Skin Editor.

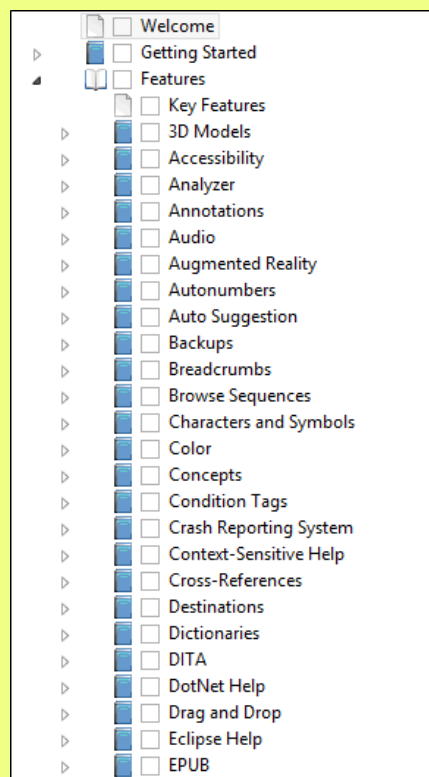


Having too many submenus extending from the top menu can be overwhelming.

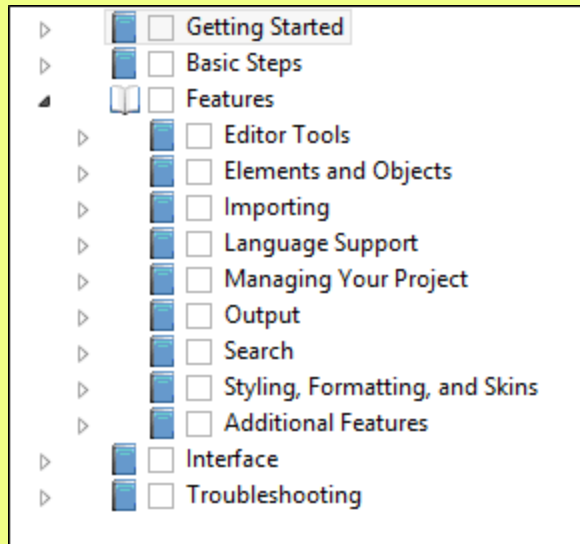
- » **Restructure the TOC** Restructuring your TOC might be where you spend most of your time in preparing for Top Navigation output. It is a good idea to reorganize longer lists of books and entries in your TOC file, limiting the number of items under a book to around 10 or fewer.

EXAMPLE

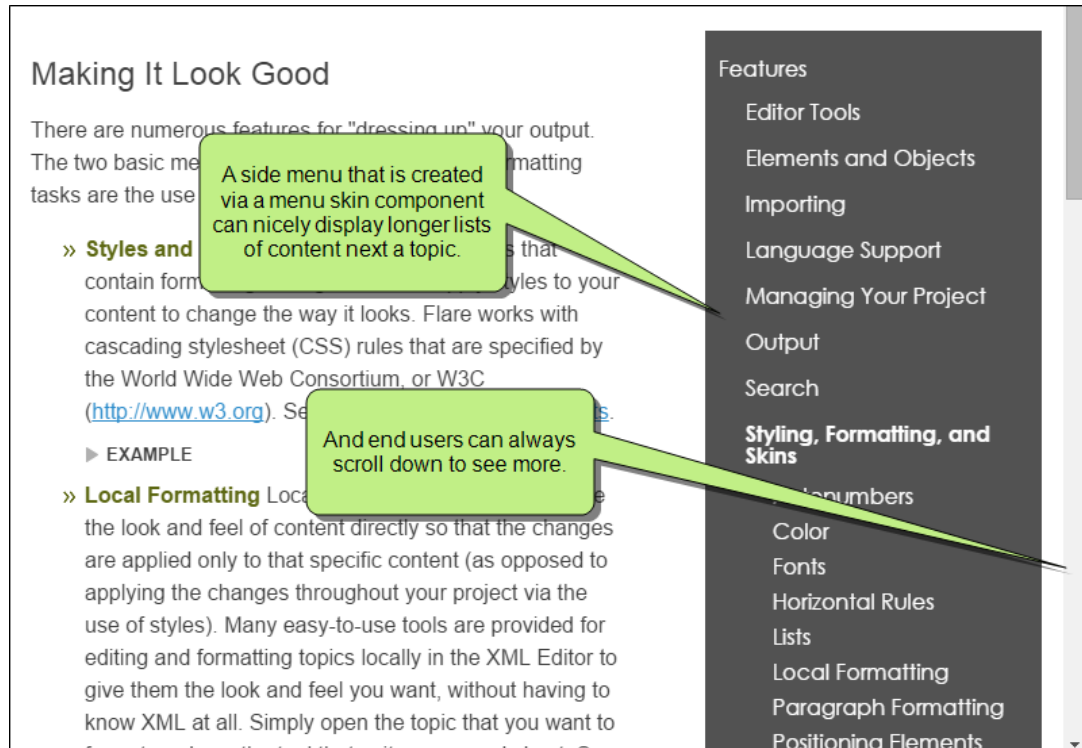
Let's say you have a section of your TOC that contains lots of books and entries at the same level, like this:



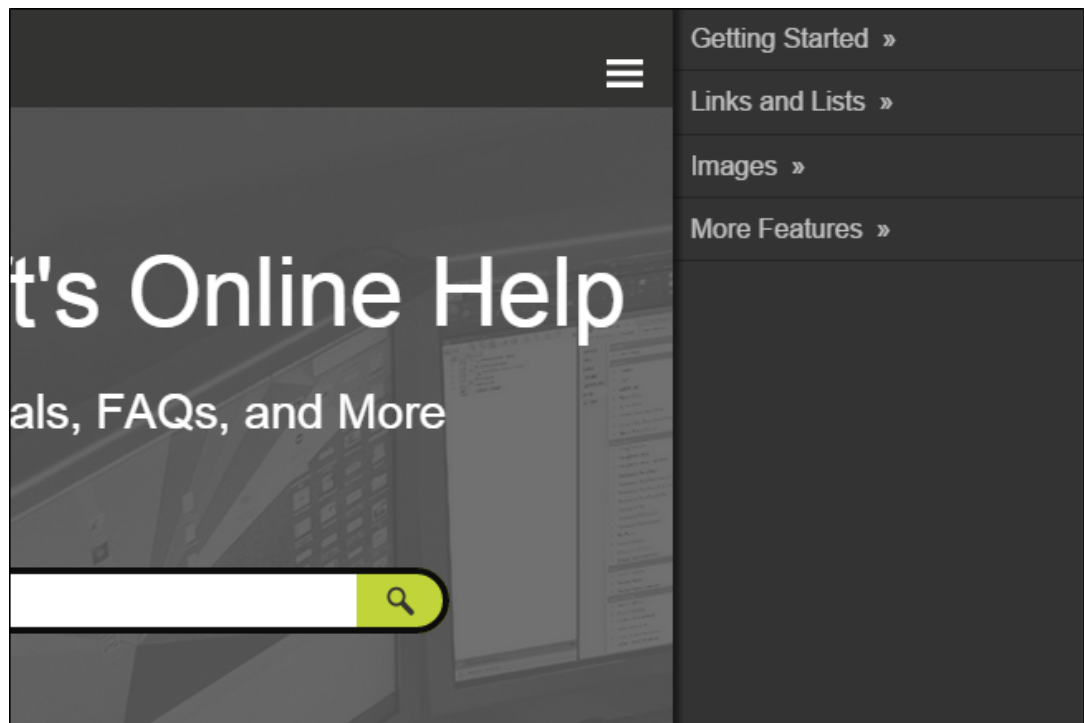
For a better result in Top Navigation output, you might restructure it so that it looks more like this:



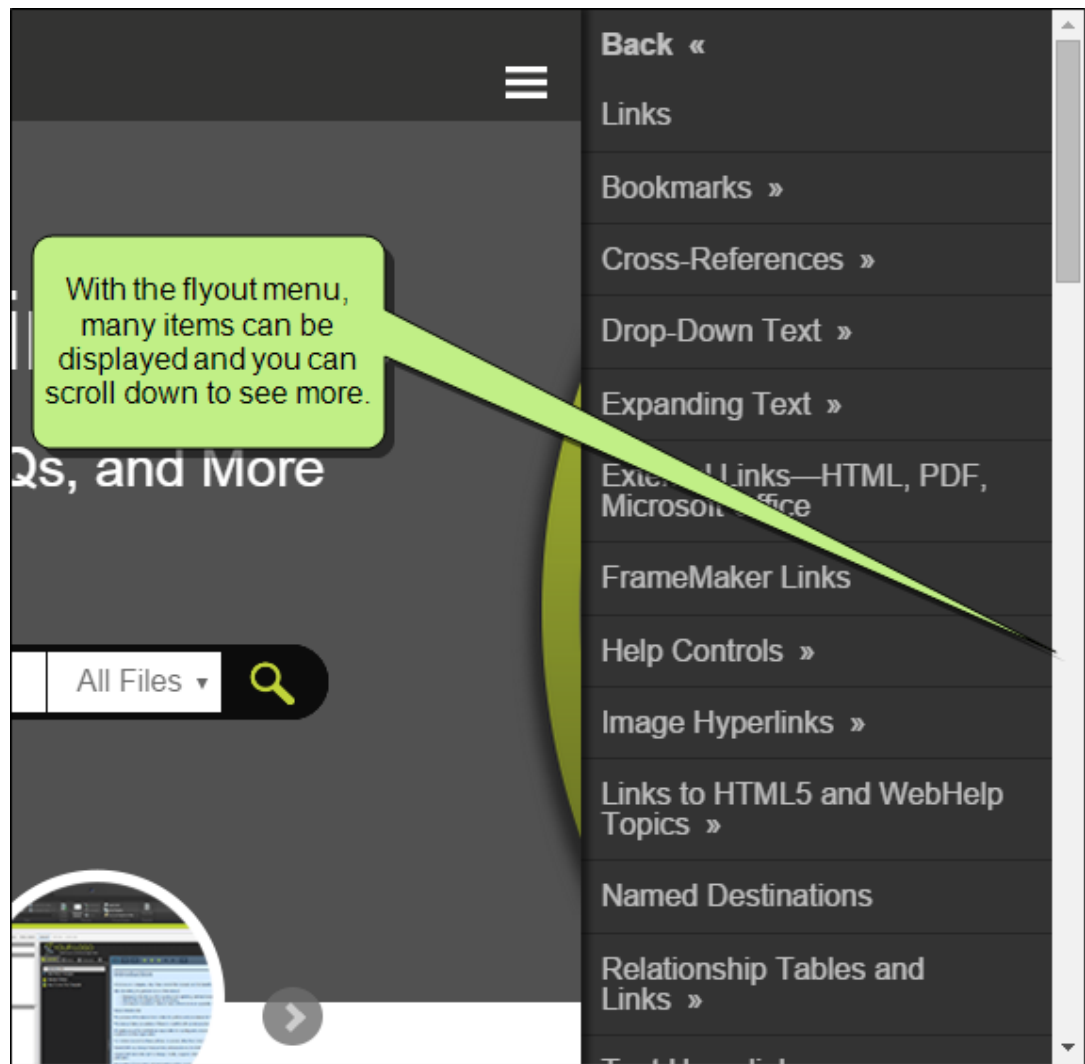
Of course, by using smaller fonts, more items can fit on the screen, but you should still try to limit the number of items in order to prevent them from disappearing off the edge of a smaller monitor. This is true at least for items that are displayed in the top menu. For books that are at deeper levels of the TOC (e.g., level 4 and beyond), it is somewhat more acceptable to allow longer lists of TOC items because a context-sensitive side menu is better able to display long lists.



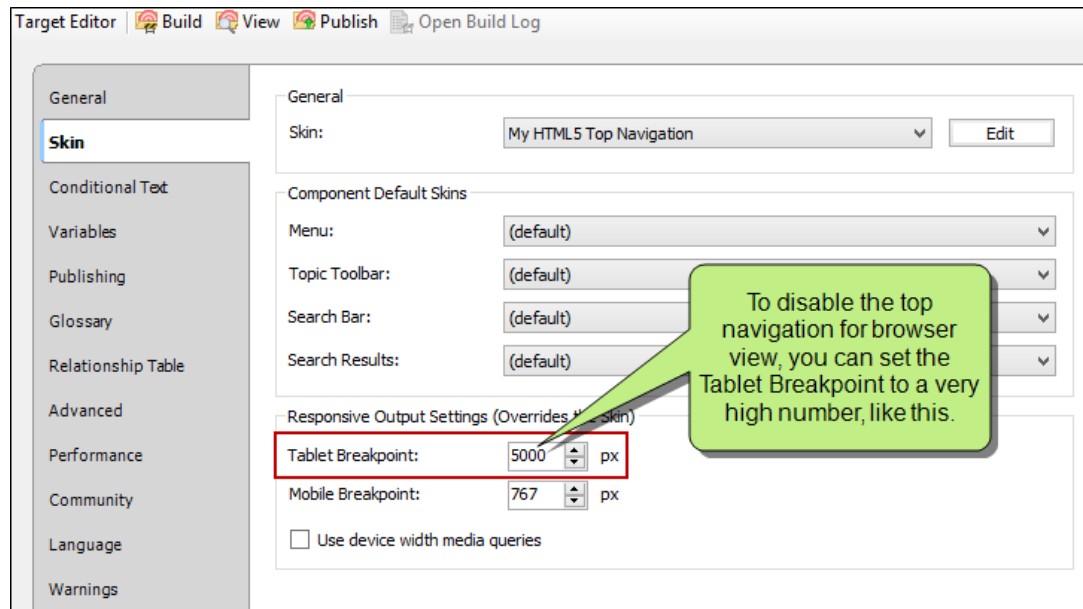
- » **Remove Items from the TOC** When looking at your TOC file, you might find that you have several topics that do not need to be included in it. Perhaps you decide to keep only the most important topics in the TOC, removing the others. When you generate your output, the most important items will be accessible in the top menu and context-sensitive menus (if you include them). As for the lesser topics you removed, end users can still find those by using search or from links found in other topics, which is what they will most likely do anyway.
- » **Turn Off Top Navigation** Although this kind of output is called "Top Navigation," it is designed to work responsively, changing its layout if it is being viewed on smaller device such as a tablet or smart phone. When this occurs, the top menu is replaced with a flyout menu on the side.



This kind of layout is better able to display longer lists of TOC items that would not look as good in a top menu.



If you would like this kind of layout for larger browser windows—as well as for smaller tablets and mobile devices—you can turn off the top navigation altogether via the responsive output settings. To do this, open the Target Editor, select the **Skin** tab, and set the **Tablet Breakpoint** to a very high number. Doing this displays the output on extremely large monitors the same way that it looks on small tablets (i.e., with the side flyout menu instead of the top navigation menu). This workaround is probably the easiest way to deal with a long TOC without having to make changes to it.



For more information about responsive output for Top Navigation, see "Responsive Output and Device Width Media Queries" on page 87.

- » **Stylesheets at Which Level?** Do you have a master stylesheet set at the project or target level? If so, you might want to rethink that for Top Navigation output. It is not mandatory that you use multiple stylesheets for Top Navigation output, but it can make certain tasks easier. In Flare's Top Navigation templates, you will notice that two stylesheets are used—one for the Home page and another for the rest of the topics. If you decide to use multiple stylesheets as well, you might need to remove any links to master stylesheets that you have in the Project Properties dialog or in the Target Editor. If you have a master stylesheet specified, you cannot link individual topics and master pages to different stylesheets, and that's what you will probably need to do if you elect to use more than one stylesheet for the output.



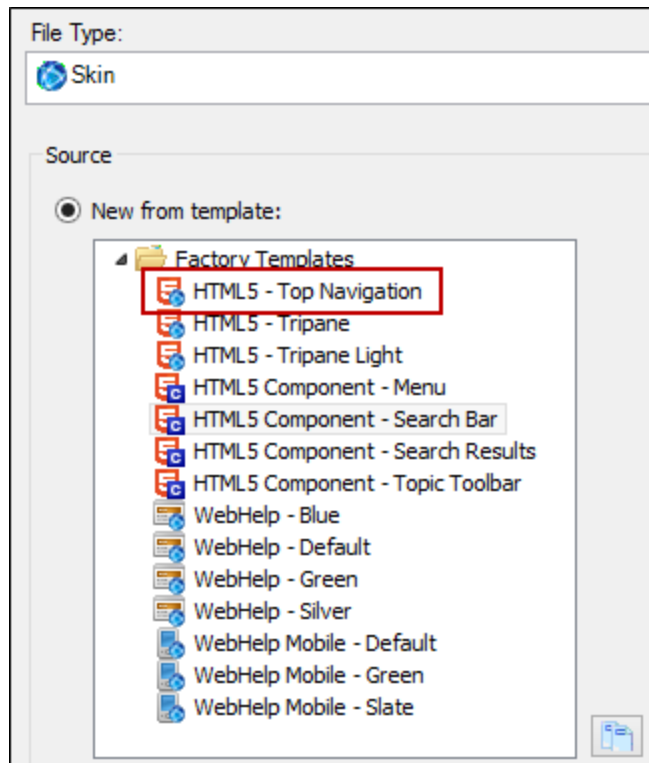
Note: It actually is possible to use multiple stylesheets and have one set at the project or target level as a master stylesheet. To do this, open that master stylesheet and use the **Options** drop-down button in the local toolbar to link it to other stylesheets that you plan to use as well. This means you would not have to associate stylesheets with files individually. However, please be aware that you must be careful when linking stylesheets because there may be certain settings from one stylesheet that you may not want to use for all of your topics. For more information and an example on linking stylesheets, see the online Help.

- » **Avoid Duplicate File Names When Importing** You might decide to create a new project from one of Flare's Top Navigation templates and then import some of the files from it to your existing project. If so, you should first make sure that you do not have files with the same name in your existing project. You probably don't want a file from the template to overwrite existing files in your project. For an example of this process, see "Advanced Conversion to Top Navigation Output" on page 137.

- » **Consider the Width of Content** If you decide to use Flare's project templates as a basis for your new Top Navigation output, you will notice that we've made the content area somewhat narrow on the left side of the context-sensitive menu, with content wrapping under the menu. It's a nice look, but it also means that you might experience issues if you have extra wide content, such as big tables. You probably won't have an issue if the wide content in question wraps under the topic menu. But if you have a long topic menu on a particular page due to the number of related links in the TOC, you could have a challenge displaying the wide content properly. If this is the case, consider either changing your content so it does not require so much horizontal space or making the content area wider. In Flare's Top Navigation project templates, you will notice several div tags in the "Other-Topics.flmsp" master page. These styles are coming from one of Flare's application stylesheets, and they are used to control, among other things, the size of the content display. So you might add the `div.content` style class to your stylesheet and make changes to it in order to override the settings from the application stylesheet.
- » **Watch for Expanding Text Next to Side Menus** In Flare's Top Navigation project templates, you'll notice that we've included a context-sensitive menu that appears to the right of most topics. In most cases, content displays nicely to the left of this menu and wraps under it once it reaches that point. However, if you have expanding text effects to the left of a side menu such as this, the text might not fill in to the left of the menu when it is expanded in the output. Instead, it leaves a gap of space that is the height of the side menu, with expanded text appearing only once it reaches the bottom of the menu. This is due to the way expanding text is designed. You have a couple of options if this takes place in your output. First, you can unbind the expanding text effects. Second, you can create a drop-down, placing your expanding text items within it. The expanding text will then adhere to the container created by the drop-down effect, ignoring the menu next to it.

HOW TO PRODUCE TOP NAVIGATION OUTPUT

1. Add a Top Navigation skin to your project (**Project>New>Add Skin**).



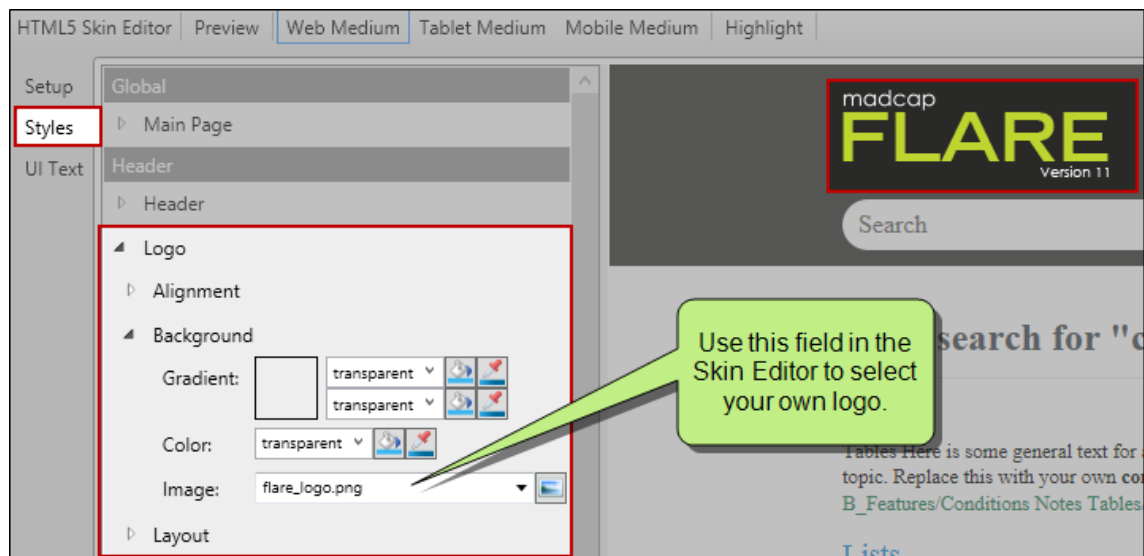
For more details about this kind of skin, see "HTML5 Top Navigation Skin" on page 65.

2. (Optional) You can edit the Top Navigation skin and its styles, just like you can edit other types of skins.

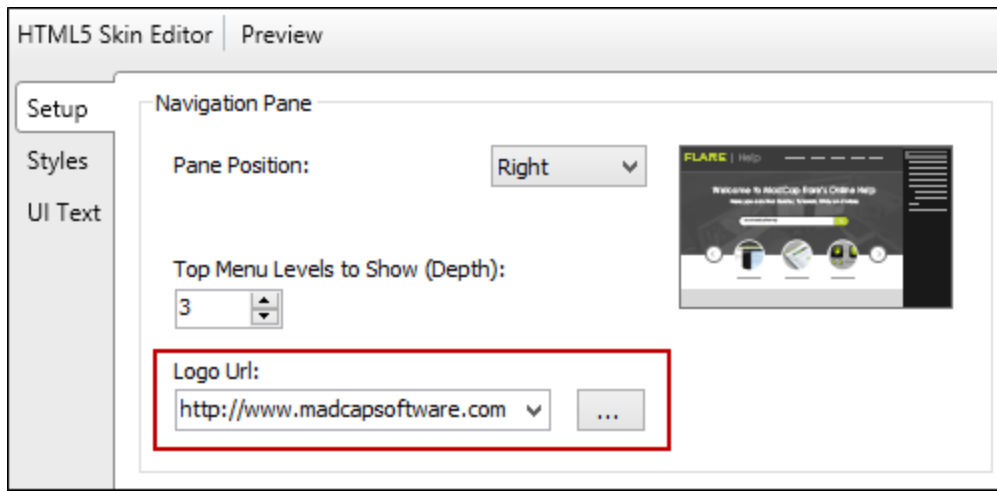
Following are a few of the more common adjustments that are made in skins:

LOGO

On the **Styles** tab of the Skin Editor, you can replace the generic logo with your own.

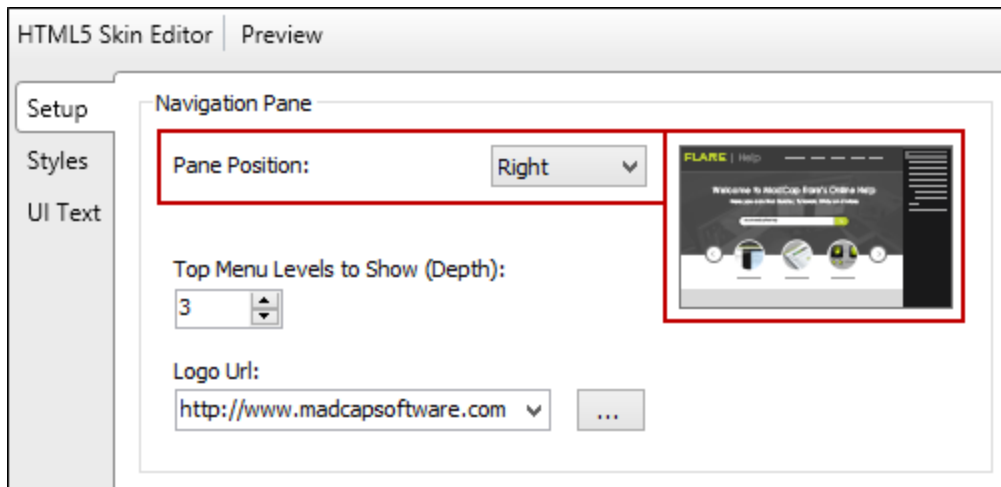


Whatever image you use for your logo, it is automatically set to link to the topic that you've set as the **Startup Topic** on the **General** tab of the Target Editor. However, you can select a different topic or even enter the URL to your company's website instead (remember to include `http://` at the beginning of the path if you link to a website). This can be done on the **Setup** tab of the Skin Editor.



PANE POSITION

On the **Setup** tab of the Skin Editor, you can position the pane either on the **Left** or **Right**. This is the flyout menu pane that is seen on the side of the output when it is being viewed on a tablet or mobile device, replacing the top menu.



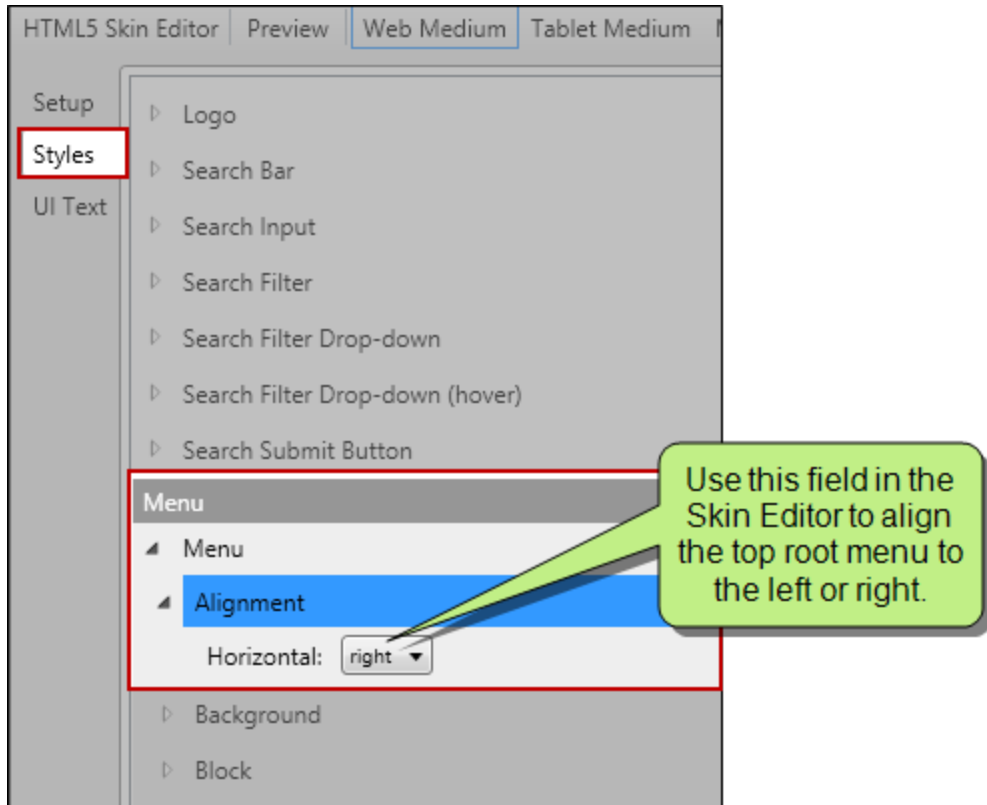
TOP MENU DEPTH

On the **Setup** tab of the Skin Editor, you can specify how many levels of your TOC items are included in the top menu navigation. The default is 3. It is recommended that you avoid including too many depth levels in the top navigation.

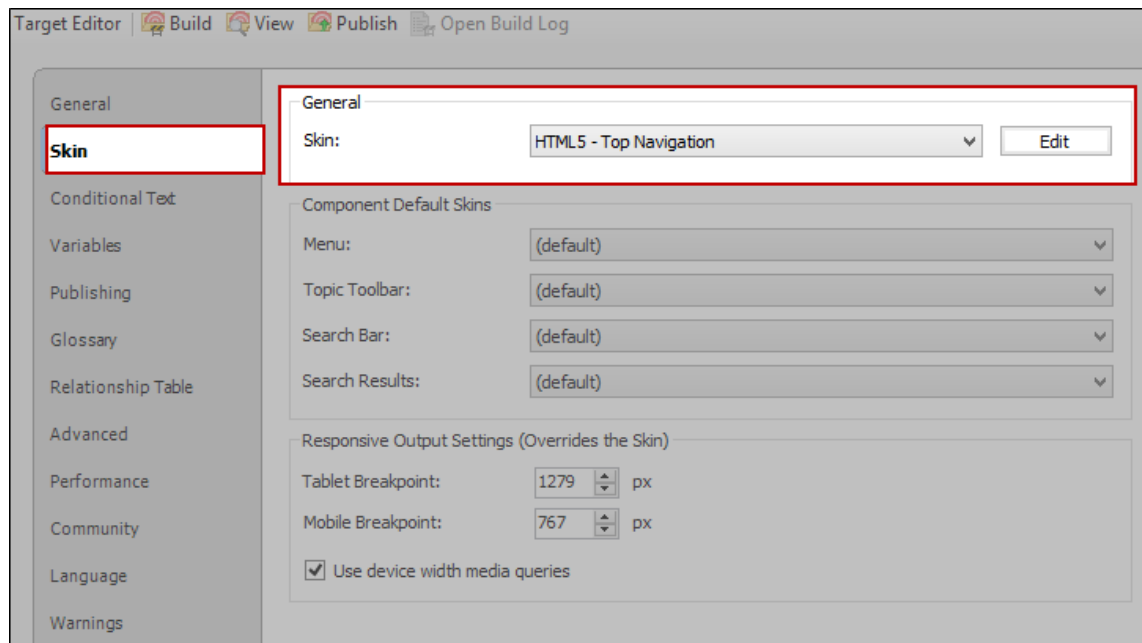


MENU ALIGNMENT

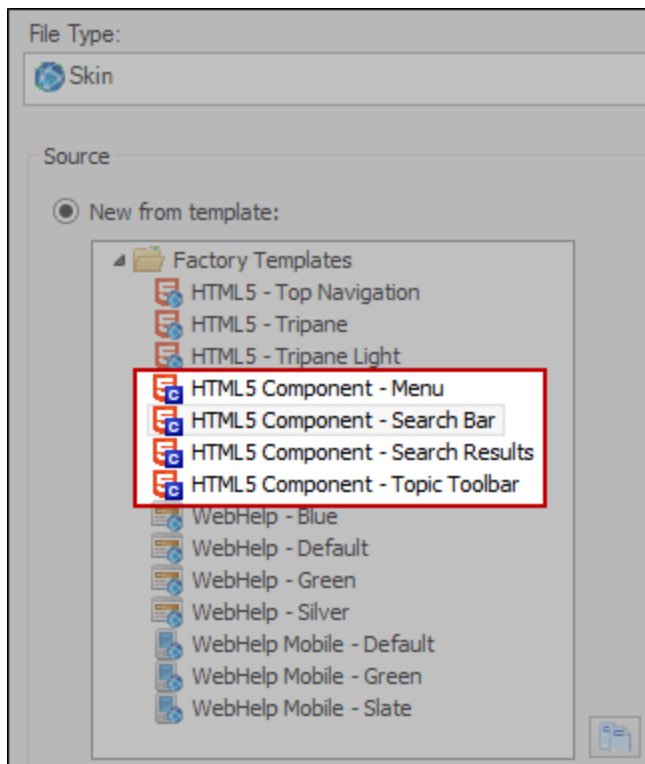
On the **Styles** tab of the Skin Editor, you can align the top root menu items to the right or to the left.



3. Open your HTML5 target and on the **Skin** tab, associate the Top Navigation skin with it.



4. (Optional) The Top Navigation skin includes a menu and search bar at the top of topics, but if you want to add elements such as these directly in topics or master pages, you can add small HTML5 skin components to your project to include special menus, search elements, and toolbars. This is the same process as adding a regular skin (**Project>New>Add Skin**), except you are adding an individual component.



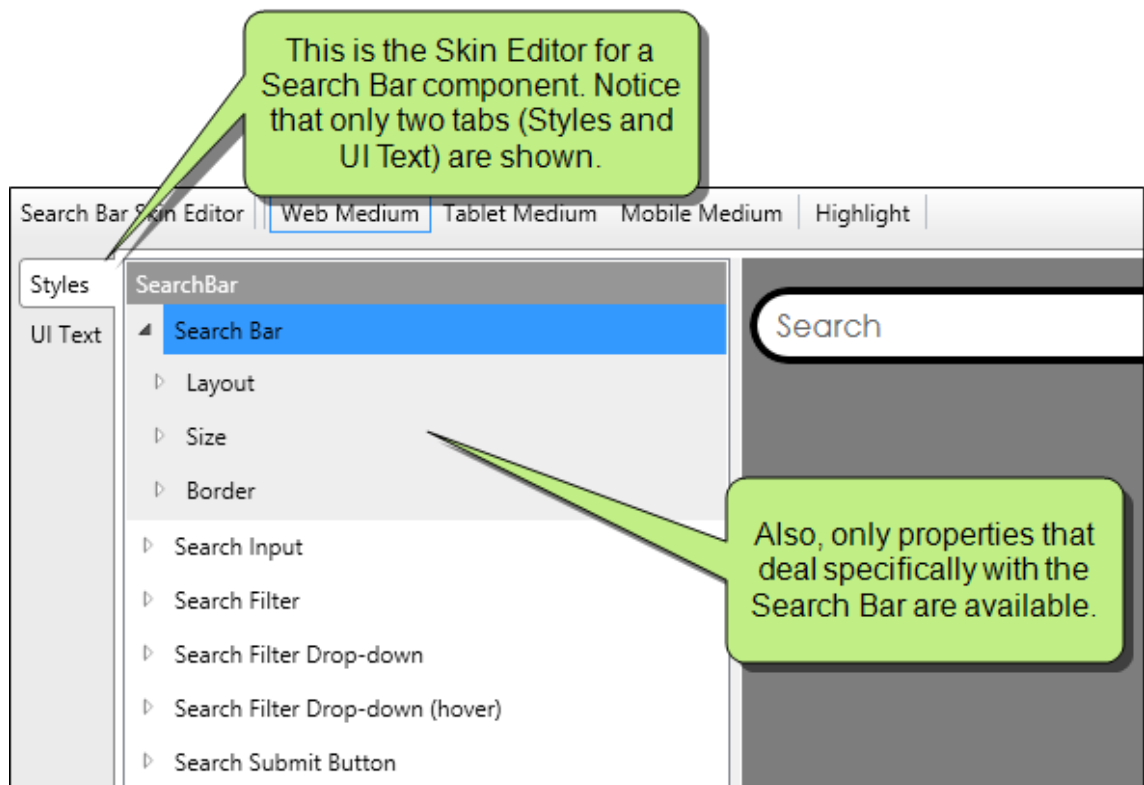
EXAMPLE

In Flare's advanced Top Navigation project templates, we've added Search Bar, Menu, and Topic Toolbar components. The Search Bar skin component is used for a prominent search bar within the Home topic. The Menu skin component is used for a context-sensitive side menu that displays to the right of topics, showing links to other topics that are next to them in the TOC. And the Topic Toolbar skin component is used to display toolbar buttons above the context-sensitive menu, allowing end users to expand or collapse content in the topic, as well as send it to a printer.

Actually, you can add features such as these in your output without adding skin components. You can do this by inserting the appropriate proxies (see step 11). The skin components are used to *create a custom look* for the elements, while their related proxies are used to *perform the actual generation* of the elements. So although adding skin components is an optional step, it is likely that you will want to do it if you decide to add navigation proxies.

For more details about these elements, see "Skin Components and Proxies" on page 66.

5. (Optional) In the same way that you can edit a regular skin, you can edit individual skin components. When you open a component to edit it, you will notice that the Skin Editor is slimmed down to show only the properties and fields related to that component.

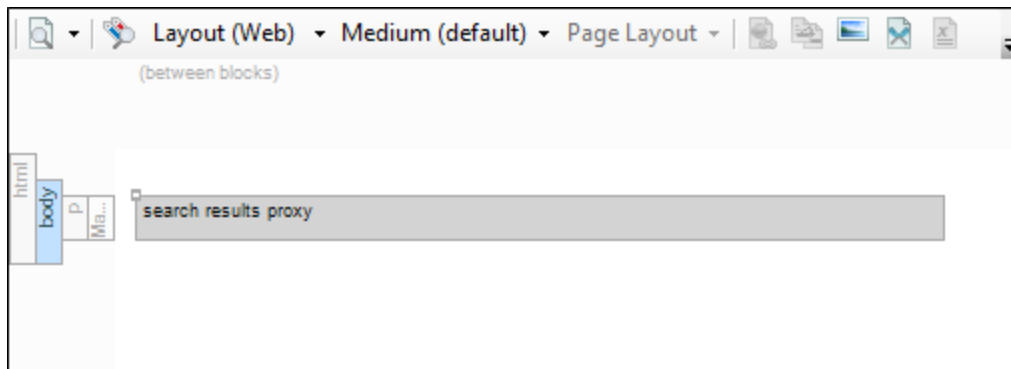


6. Create a Home topic. This is the first topic shown when users open the output. It is just a regular topic, but you might decide to design it to stand out from the rest of the topics. Make sure to set this topic as your startup topic. You can do this by opening the Target Editor, selecting the **General** tab, and selecting the topic in the **Startup Topic** field.

For more details, see "Home Topic" on page 82.

7. (Optional) You can create other topics, including regular topics that contain the bulk of your content.

You can also create a special topic to hold generated search results. A Top Navigation skin already includes a search bar, and you do not need to do anything else to incorporate search into your output. When an end user searches for text, Flare displays the results in its default format. However, if you want to be able to customize your own search results page, you can create a topic specifically for that purpose and insert a Search Results proxy into it (**Insert>Proxy>Insert Search Results Proxy**). This works with the Search Results skin component that you can add to a project (see step 4).



8. Add topics to a TOC file.

The Top Navigation skin uses the structure and contents of your TOC to populate the menu that appears at the top of topics. In addition, Flare's Top Navigation templates include a master page with a special Menu proxy inserted into it. This proxy creates context-sensitive menus that are displayed on the side of content, displaying links for topics that are located in the same TOC book (as well as the parent and child TOC topics).

TOC Editor

Note 1: You can quickly open a linked topic or file
Note 2: You can build the TOC by dragging topics

- Getting Started
 - Basic Steps
 - What's New
- Links and Lists
 - Links and Footnotes
 - Lists
- Images
 - Thumbnail Image
 - Positioned Images
 - Image Positioned Left
 - Image Positioned Right
 - Image Within List
- More Features
 - Conditions
 - Notes, Examples, and Blockquotes
 - Tables

Notice that the first-level books in the TOC file (Getting Started, Links and Lists, Images, and More Features)...

... are automatically shown as menu items at the top. When you hover over one of these, the TOC items under it are shown in a submenu.

In the example below, the topic "Positioned Images" is open.

This context-sensitive menu shows the TOC items in the same book as the open topic.

Submenus are shown when users hover over the top root menu.

YOUR LOGO
place your company logo here

Getting Started Links and Lists Images More Features

Search

You are here: [Images](#) > [Positioned Images](#)

Positioned Images

Here is some general text for a topic. Replace this with your own content.

- » Here is a cross-reference link: See [Image Positioned Left](#).
- » Here is a cross-reference link: See [Image Positioned Right](#).

Thumbnail Image
Positioned Images ▶
Image Within List
Image Positioned Left
Image Positioned Right

Images
Thumbnail Image
Positioned Images
Image Positioned Left
Image Positioned Right
Image Within List



Tip: Although Flare lets you create books in the TOC file that do not link to anything (i.e., merely using the book to organize the TOC), it is a best practice for Top Navigation output to make sure that all TOC books and items are linked to something.

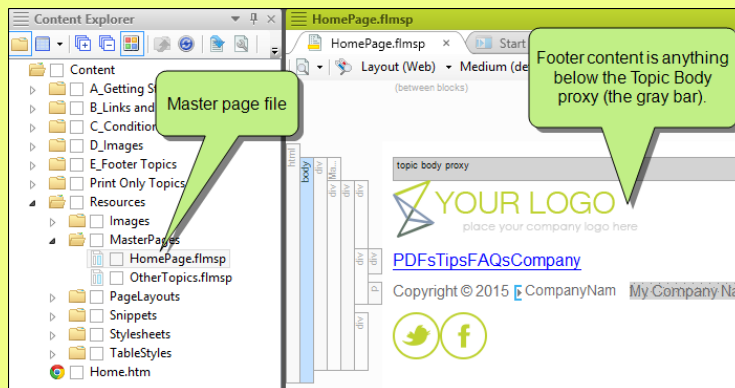


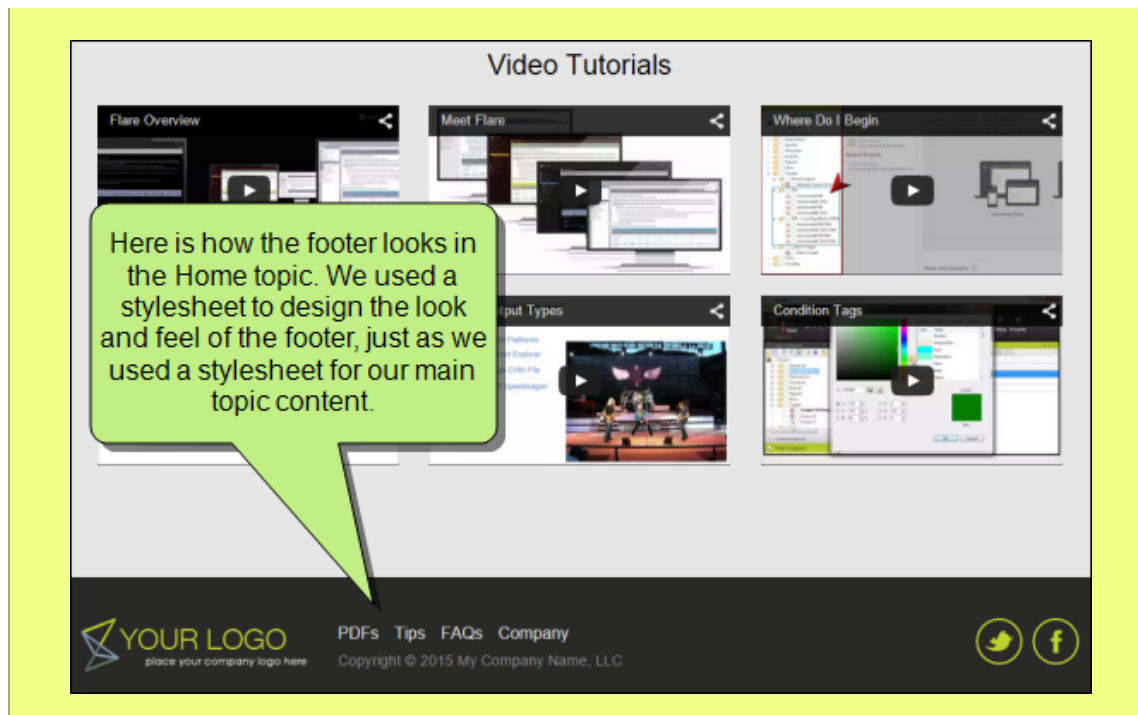
Tip: It is standard practice in web design to not include your Home page as a menu text link. The default behavior in Top Navigation output is to link the logo to the Home page, so it is not necessary to add this topic to your TOC.

9. (Optional) It is not mandatory to create any master pages in order to have Top Navigation output. However, master pages can be particularly useful, especially if you want the same content to automatically show up at the bottom or top of topics. They can also be useful if you want your Home page's design to be much different from that of the rest of your topics. So if you would like to incorporate one or more master pages, create them.

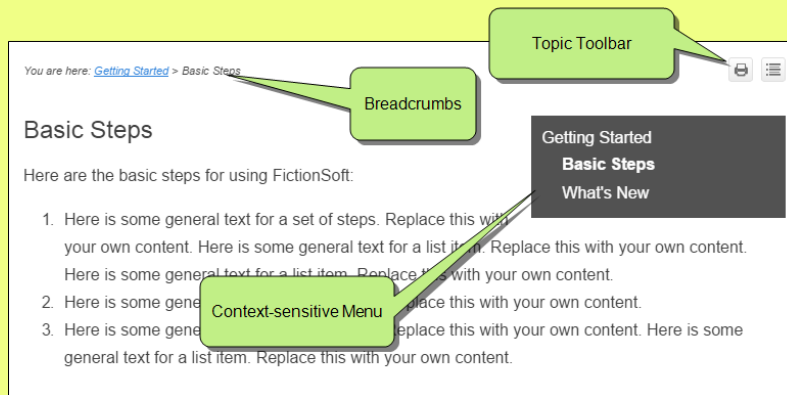
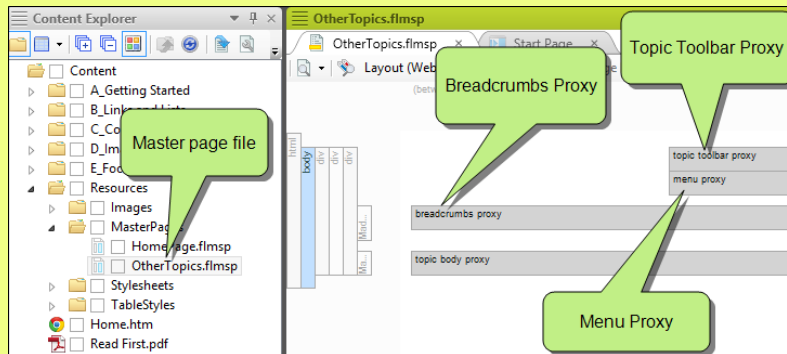
EXAMPLE

In Flare's advanced Top Navigation templates, we wanted to show footer content at the bottom of the Home page, so we created a master page just for that topic, and we added footer content under the Topic Body proxy.





Then on the rest of the topics we wanted breadcrumbs to be displayed above the topic content. So we created a second master page for all of those topics and inserted a Topic Toolbar proxy, a context-sensitive Menu proxy, and a Breadcrumbs proxy above the Topic Body proxy. Even though the Menu proxy was added above the Topic Body proxy, it was styled to display to the right of the topic content, with content wrapping under it.

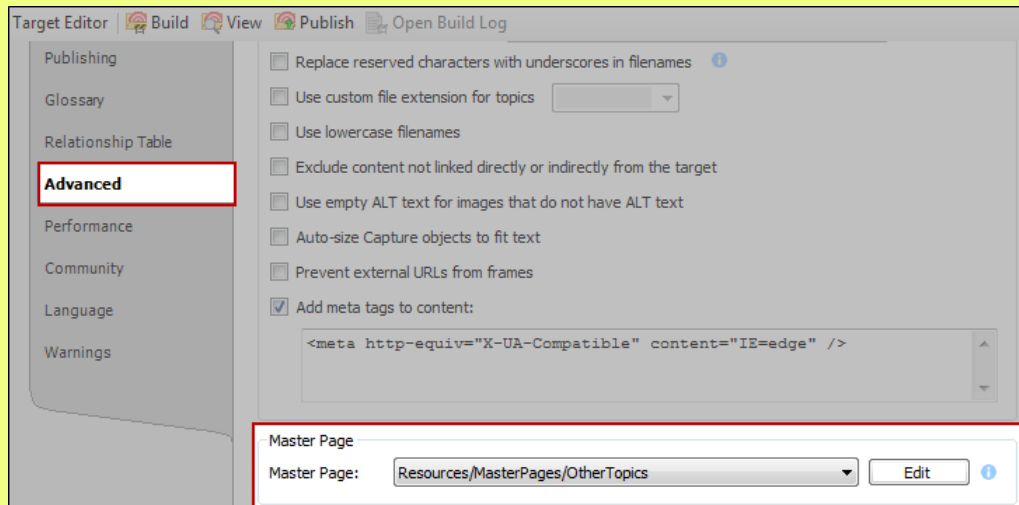


10. (Optional) You can select a master page on the **Advanced** tab of your HTML5 target.

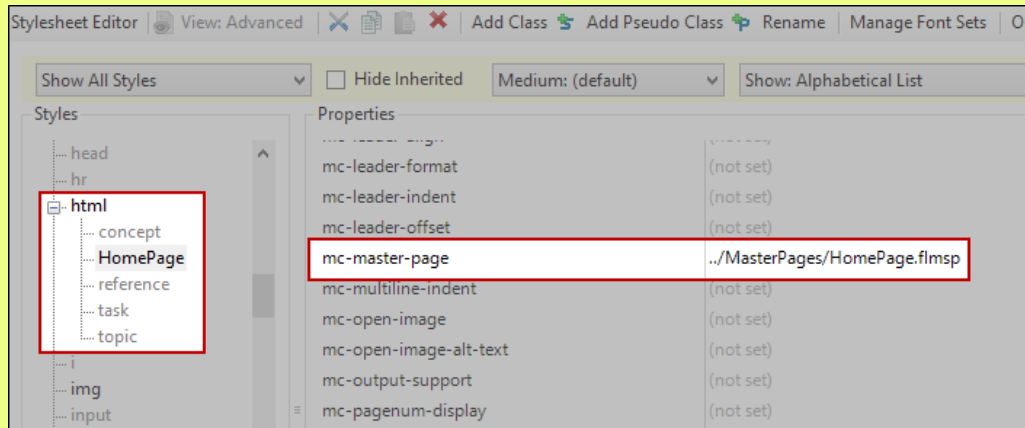
This tells Flare that, unless otherwise directed, all topics will use the master page you selected. But if you want to use a second master page, you can point to it from the html style (or a class of it) in a stylesheet. In that case, the master page set in the target is typically the one that the majority of your topics will use. But for the other topic(s) using a different master page, you can override the target setting by pointing to the stylesheet and the html class you created.

EXAMPLE

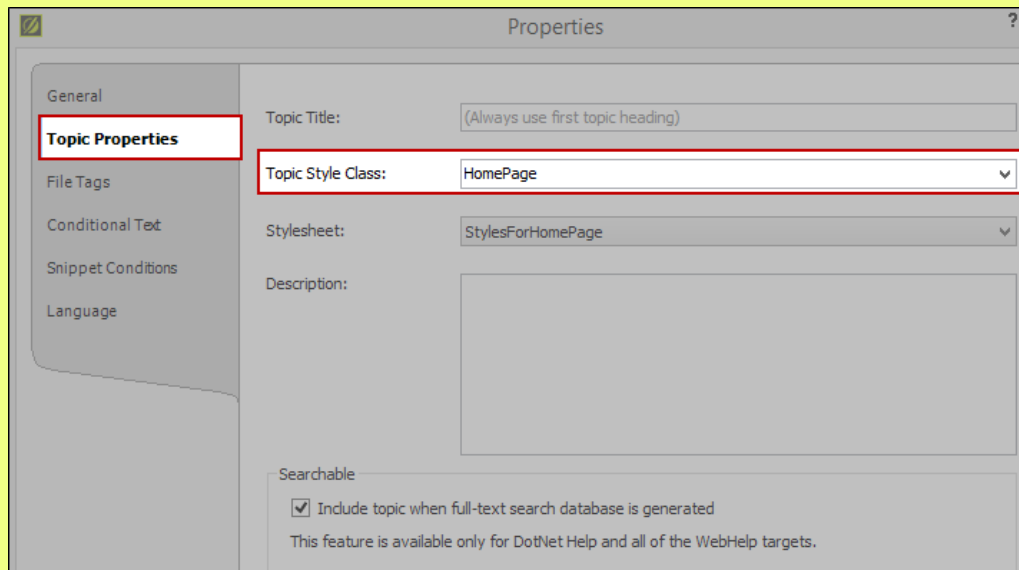
In this example, the master page that we want to use for most of our pages in the output is named "OtherTopics." So on the **Advanced** tab of the Target Editor, we selected it.



In the stylesheet used for our Home page topic, we created a class of the **html** style and named it **HomePage**. We then navigated to the **mc-master-page** property and associated it with the second master page.



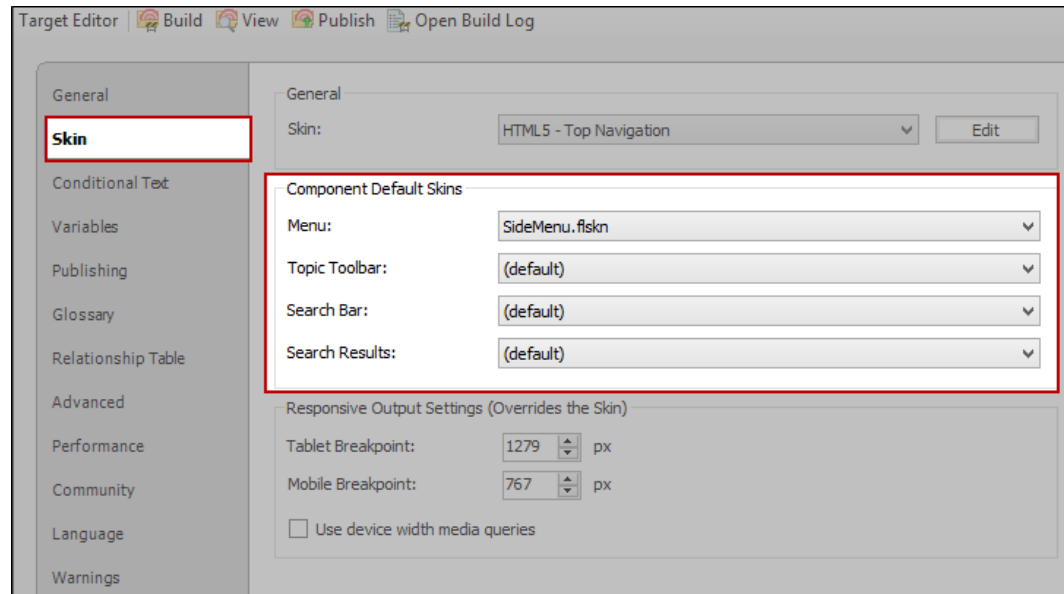
Finally, we opened the Properties dialog for the Home topic (open the topic, then select **File>Properties**). On the **Topic Properties** tab, we selected the **HomePage** style class that we created. Now this topic will use the "HomePage" master page instead of the other one for the rest of the topics.



11. (Optional) You have the option of inserting a variety of proxies (**Insert>Proxy>[Name of Proxy]**) into your master page(s) and topics. This includes proxies that correspond to the different kinds of skin components you can add to your project (Menu, Search Bar, Search Results, Topic Toolbar). For more details, see "Skin Components and Proxies" on page 66.

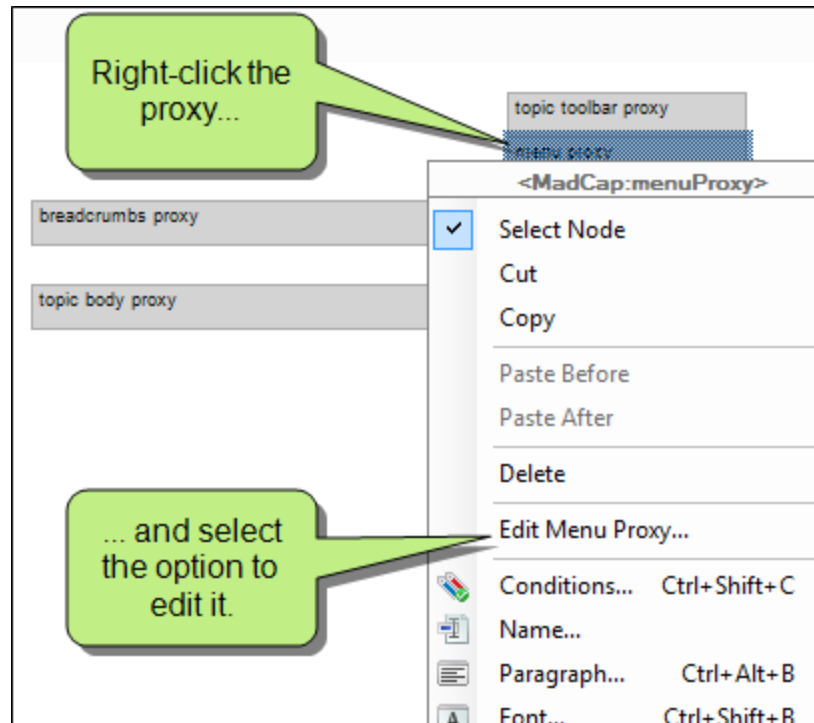
12. (Optional) If you have added skin components to your project, you can associate them with an entire target or with individual proxies. If one skin component is associated with the target and a different one is associated with a proxy, the one associated with the proxy has precedence.

» **Target** To associate a skin component with an HTML5 target, open the **Skin** tab in the Target Editor, then make your selection in the **Component Default Skins** section



It is not necessary to select anything in these fields, especially if you have only one type of a particular skin component (e.g., one Menu component, one Topic Toolbar component). But if you have more than one type of a skin component (e.g., two Menu components), you can choose one of them as the default for all of the topics in the target. Then for the other component(s) of that same type, you can override the target setting by pointing to it in the relevant proxy (see below).

- » **Proxies** To associate a skin component with a proxy, open the master page or topic where you've inserted the proxy. Then right-click the proxy and use the dialog to choose the skin component. Some proxies have additional settings, such as TOC depth for Menu proxies.



Here is an example of a dialog for setting options in a Menu proxy.

Menu Proxy

Inserting a menu proxy allows you to include a stylable menu inside a topic or masterpage.

Menu Content

Linked TOC or Browse Sequence:
(default)

☒ Context sensitive
☒ Include parent
☒ Include siblings
☒ Include children

Levels to Show (Depth):
(default)

Menu Style

Skin File:
Project/Skins/SideMenu

Stylesheet class:

For more information about associating skin components with proxies, see "Skin Components and Proxies" on page 66.

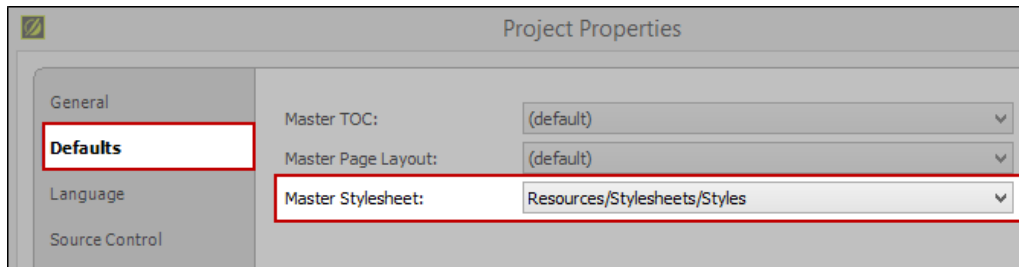


Note: It's possible that you will not need skin components in your project at all. If you have not added a skin component (e.g., a Menu component) but you insert a proxy related to it (e.g., a Menu proxy), Flare will just use the default design from the application.

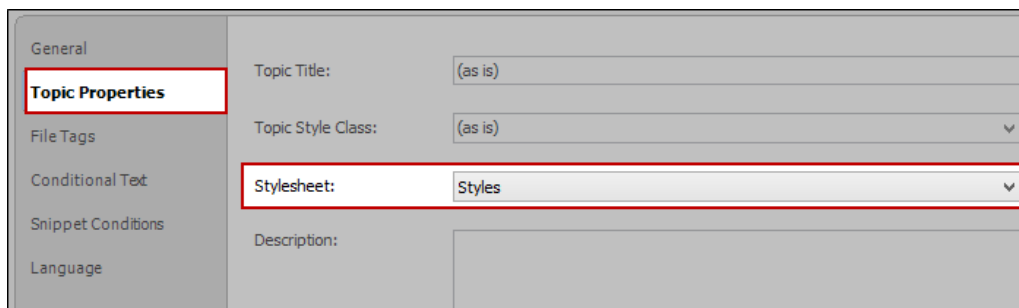
13. Create and edit stylesheets.

For more information about using stylesheets for Top Navigation output, see "Stylesheets" on page 84.

14. If you are using a single master stylesheet, you can associate it with a project or a target so that it is automatically used by all topics.



Alternatively, you can associate stylesheets with individual topics and master pages. Use this option if you have multiple stylesheets for your output. One way to do this for topics is to open the File List window pane, select all the topics that you want to associate with a particular stylesheet, and in the **Home** ribbon click **Properties**. Then on the **Topic Properties** tab, select the stylesheet.



For master pages (and snippets too), you cannot use the File List window pane approach. Instead, you need to open the file, then on the **Home** ribbon click **Stylesheet Links**. You can then associate the file with one or more stylesheets.



Note: If you set a master stylesheet at the project or target level, you cannot also set a stylesheet on individual topics or master pages. You should choose one method or the other.



Note: It is most important to link topics and master pages to stylesheets. It is less important to link snippets to stylesheets. That's because snippets will inherit the styles of the topic where they are inserted. However, if you are used to a master stylesheet being linked at the project level, you are also used to seeing snippets with the appropriate styles already shown when you open the snippets. But if you then switch to a workflow where stylesheets are linked to individual files, your snippets are not automatically linked to the stylesheet just because the topic is. Again, it is not vital that you do this, but you probably want to associate snippets with stylesheets as you edit them so that you can more clearly see what you're doing.

15. (Optional) Responsive output is automatically enabled for Top Navigation skins. But you can set a few additional options on the **Skin** tab of the Target Editor.
 - » **Tablet Breakpoint** Enter the number of pixels for the maximum width of a Tablet view.
 - » **Mobile Breakpoint** Enter the number of pixels for the maximum width of a Mobile (or phone) view.
 - » **Use device with media queries** Select this option if you want to base the responsive output not merely on the width of the output display, but on its width in the actual device (browser, tablet, or mobile). In other words, if you do not have this option selected and you view the output on a full browser, you can see the tablet and mobile layouts simply by reducing the size of the browser window. But if you select this option and you reduce the width of the browser, the layout will not change to the tablet or mobile formats.

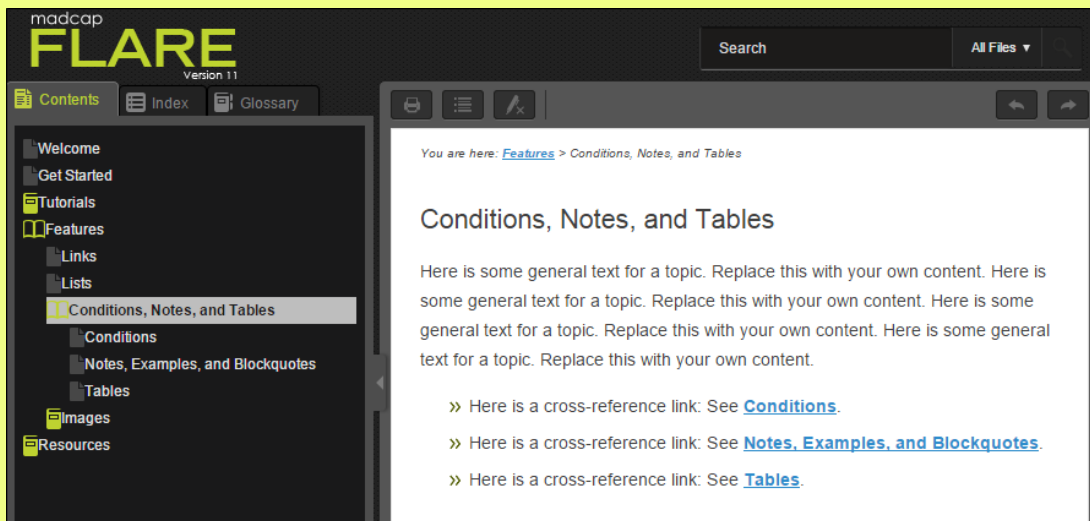
For more details, see "Responsive Output and Device Width Media Queries" on page 87.

Simple Conversion to Top Navigation Output

Following is an example of a simple conversion of a Flare project from Tripane to Top Navigation output. We show how you can create and add just a few pieces to achieve a similar look that is seen in Flare's Top Navigation Advanced template project. The biggest difference is that we are not attempting to create an elaborate Home page similar to the one found in the template. Also, while we are adding a context-sensitive menu, it is not as fancy as the one in the template.

EXAMPLE

Let's say you have a Flare project with a Tripane format that looks something like this in the output:

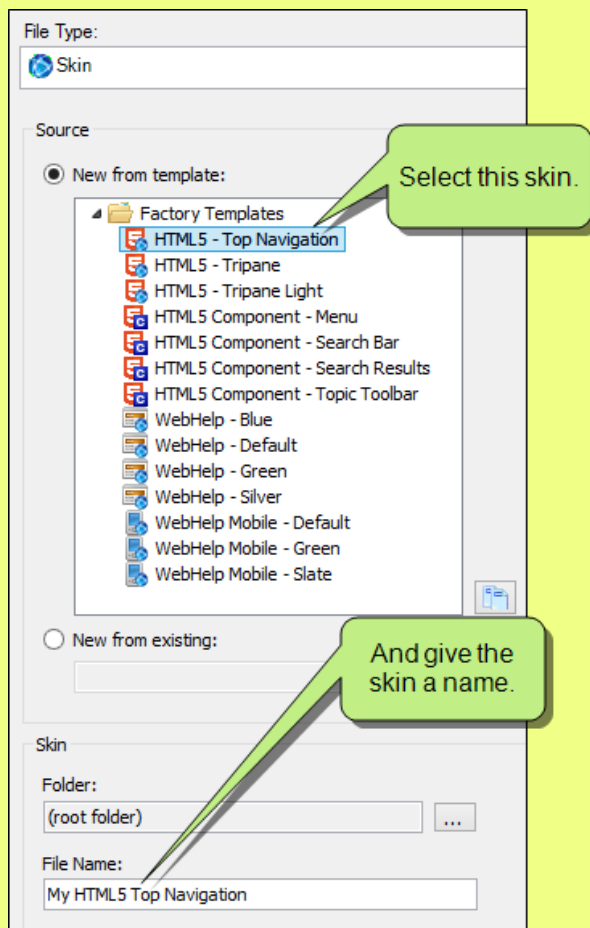


You want to convert to HTML5 Top Navigation output using as few steps as possible. For this example, let's say you want to accomplish the following:


- » Add the navigation elements (search and menu) at the top of pages.
- » Use your company logo.
- » Add a context-sensitive menu on the same page as each topic.
- » Use only the single stylesheet you already have in your project.

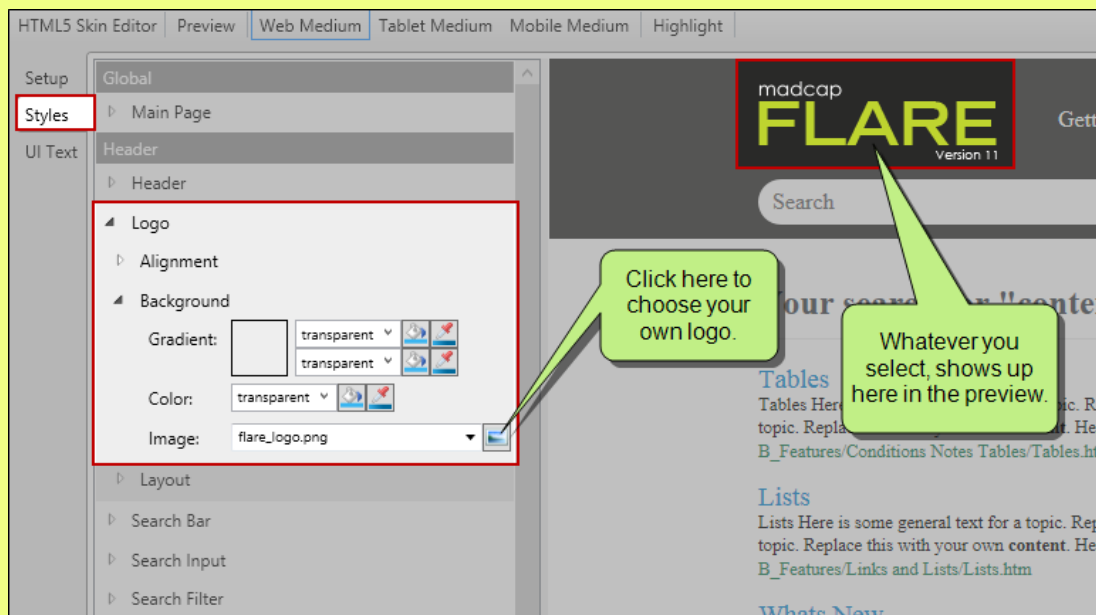
Let's say you've already completed some of the preliminary tasks to make sure your output will work well with the new format (see "Before You Begin" on page 95). You've restructured your TOC to avoid very long vertical lists of items, and you've even removed some TOC items that you decide are not necessary.

First, you select **Project>New>Add Skin**. In the Add File dialog you select the **HTML5 - Top Navigation** factory template, give it a name, and add it.

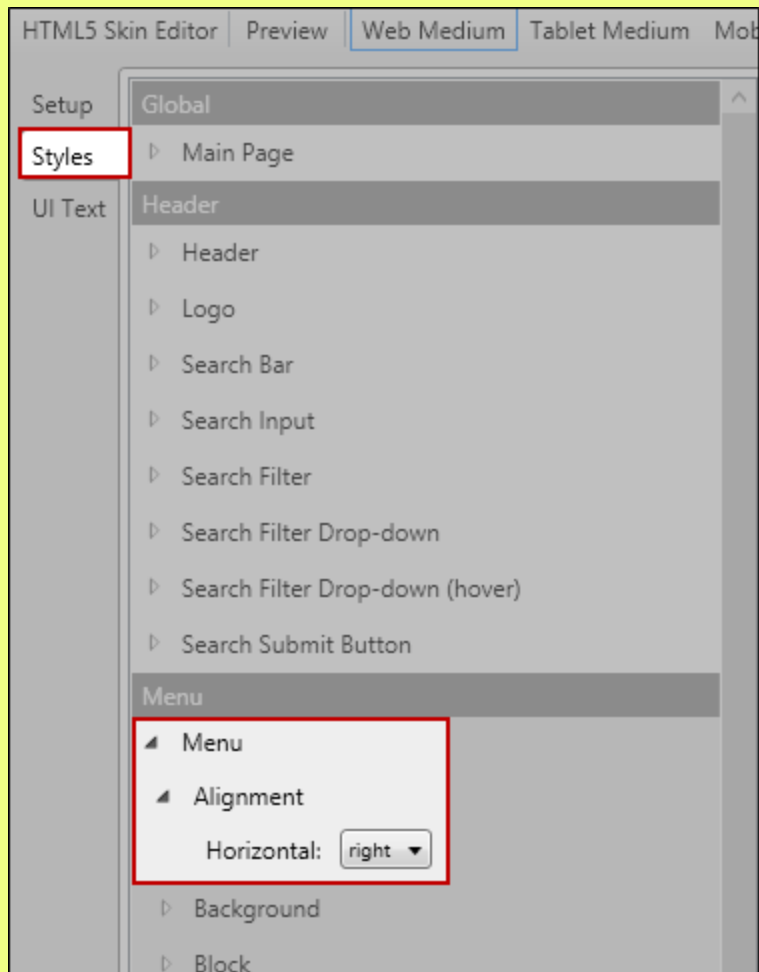


In the Skin Editor that opens as a result, you look at the fields on the **Setup** tab and decide to keep all of the settings. The flyout pane (for responsive output) will be positioned on the right, the depth of menu items from the top menu will be limited to the first three, and the logo will be linked to your Home (startup) topic (the default setting).

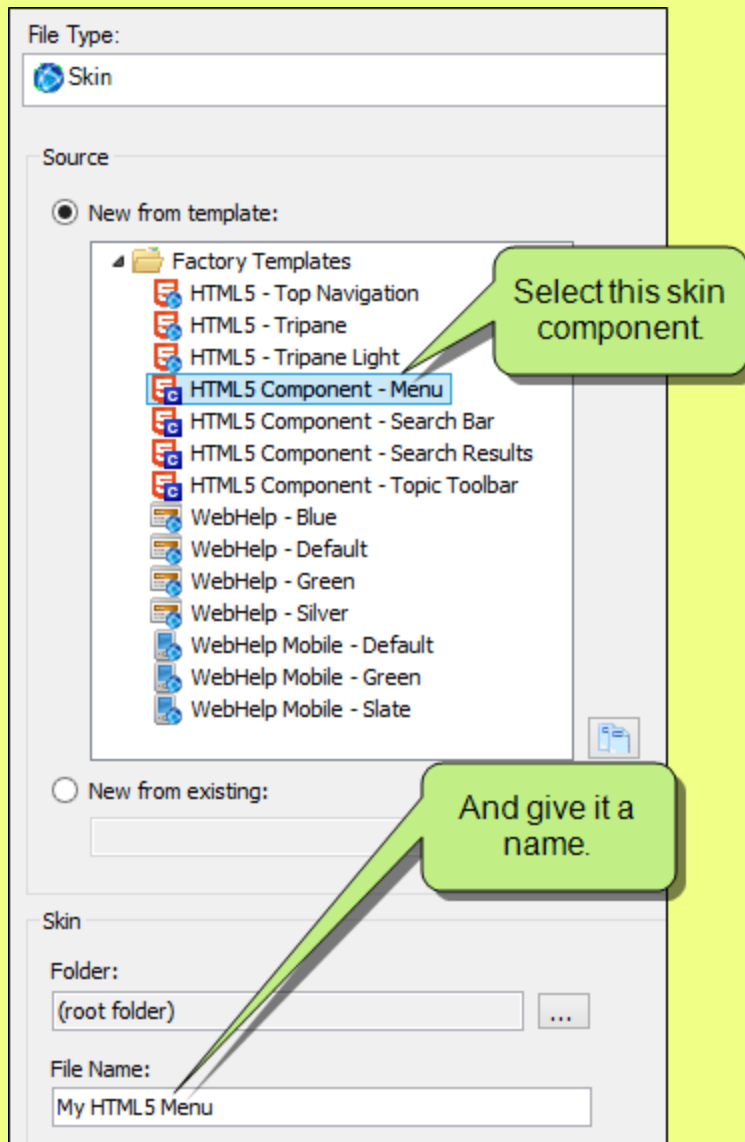
However, you need to change the logo image. So you select the **Styles** tab and expand **Logo>Background**. You click  and select the logo image.



You also decide to align your new top menu to the right. So in the **Styles** tab you expand **Menu>A-**
lignment and set the **Horizontal** field to **right**.

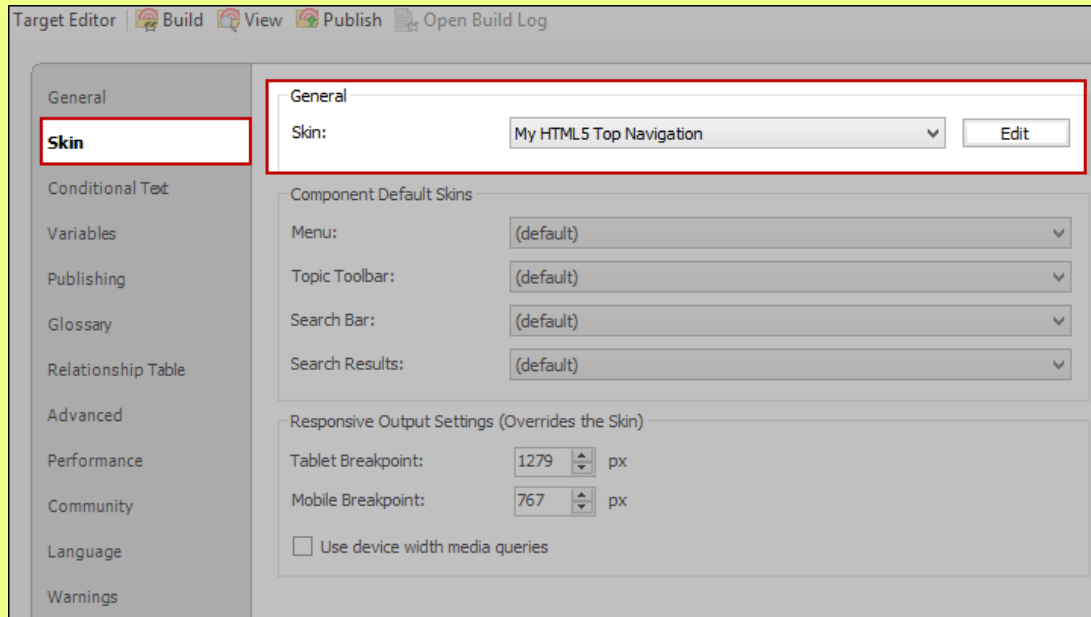


Next, you repeat the steps to add a new skin (**Project>New>Add Skin**), but this time you select **HTML5 Component - Menu**. This is the special skin component that will display a context-sensitive menu, depending on the topic that is open.

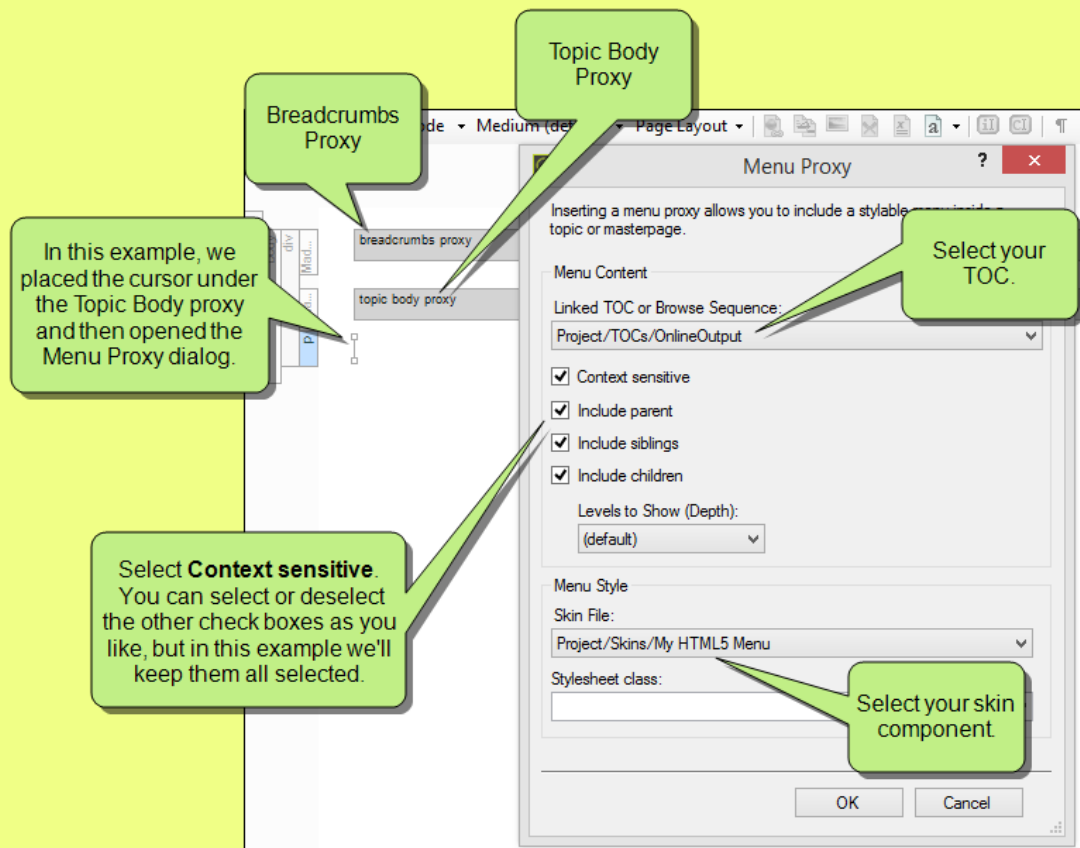


In the Skin Editor that opens as a result, you can modify the look of the menu and its individual items in many ways. However, for the purpose of this example, let's say you just keep the plain default look.

Now you need to tell Flare to use your new HTML5 Top Navigation skin. So you open your HTML5 target and select the **Skin** tab. In the **Skin** field you select your new Top Navigation skin.

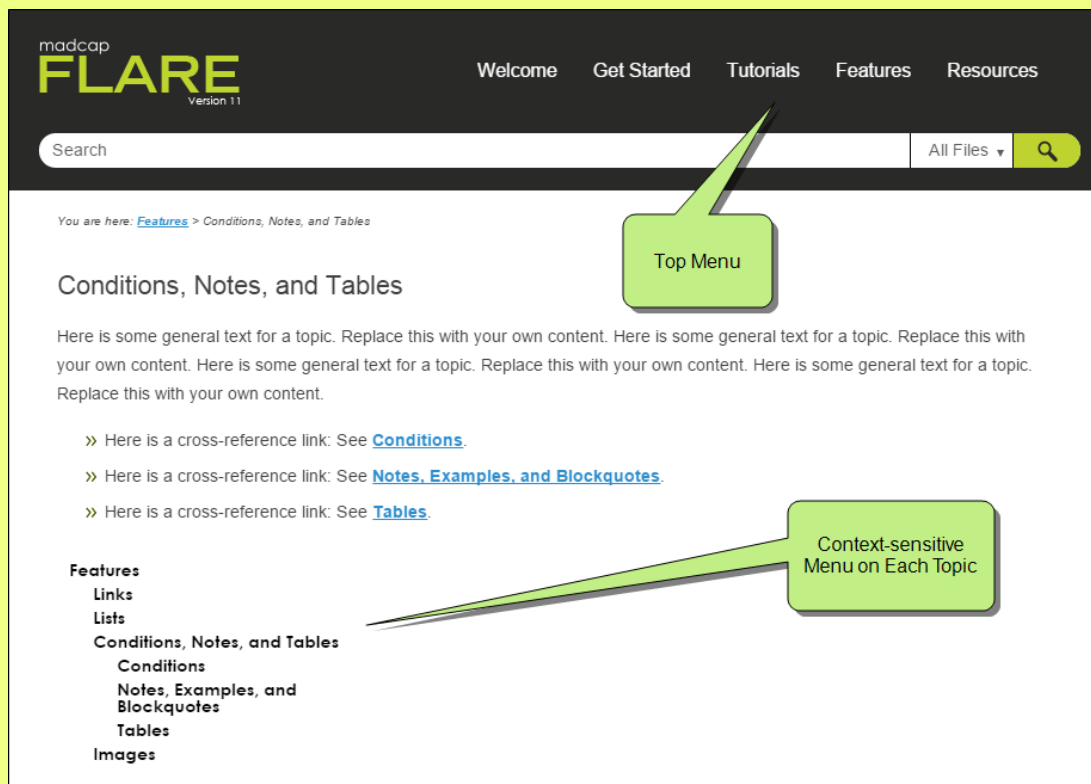


For this example, we'll assume that you've already been using a master page for your online output, perhaps to include breadcrumbs above all topics. You want your new context-sensitive menu to appear on all topics, so you open the master page. You navigate to an empty paragraph above or below the **Topic Body** proxy and select **Insert>Proxy>Insert Menu Proxy**. In the dialog that opens, you select the TOC you want to base the menu on. You also keep all of the check boxes selected, as well as the default depth level. Finally, you select the new Menu skin component that you added to your project.

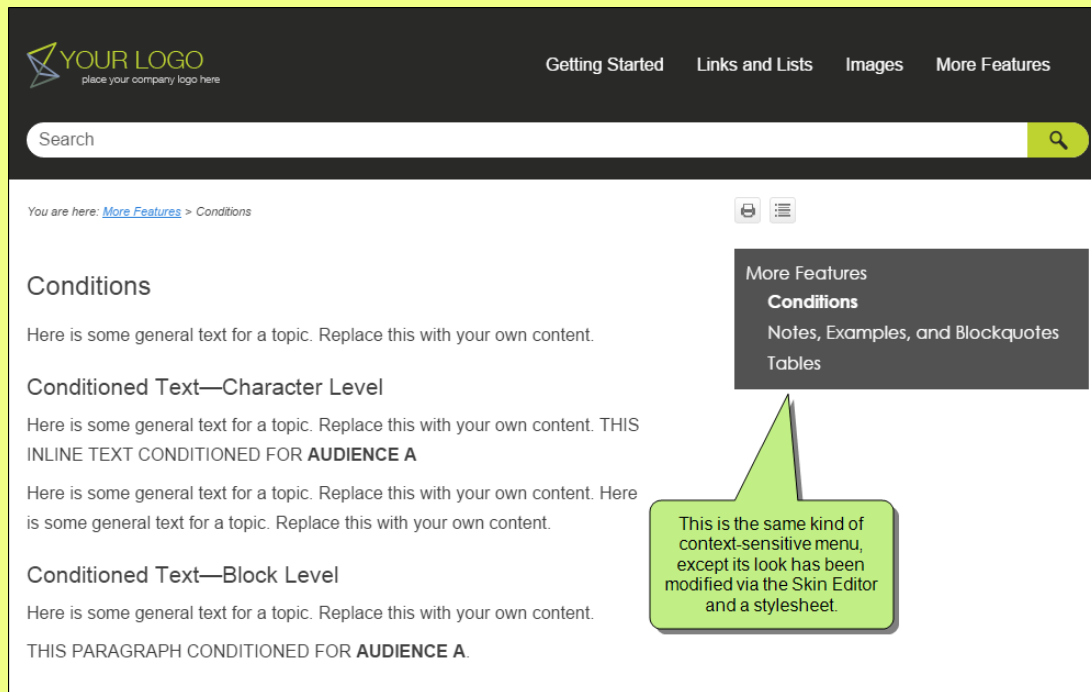


Because you're working with an existing project, we are going to assume you've already completed the other necessary steps for tying everything together. This includes associating your stylesheet with the project, target, or appropriate files; associating your master page with the target; and setting your intended Home page as the startup topic in the target. For more information about these tasks, see the online Help.

So now you just build the target. In the end, your output might look something like this:



You might be thinking, It looks good, but I really want the context-sensitive menu to appear on the side of topics, not below them. Like this:



That is certainly possible, but that type of configuration requires some extra work in your regular stylesheet and master page. You will need to create some styles and apply them to the area where you've inserted the Menu proxy in the master page. See "Stylesheets" on page 84.

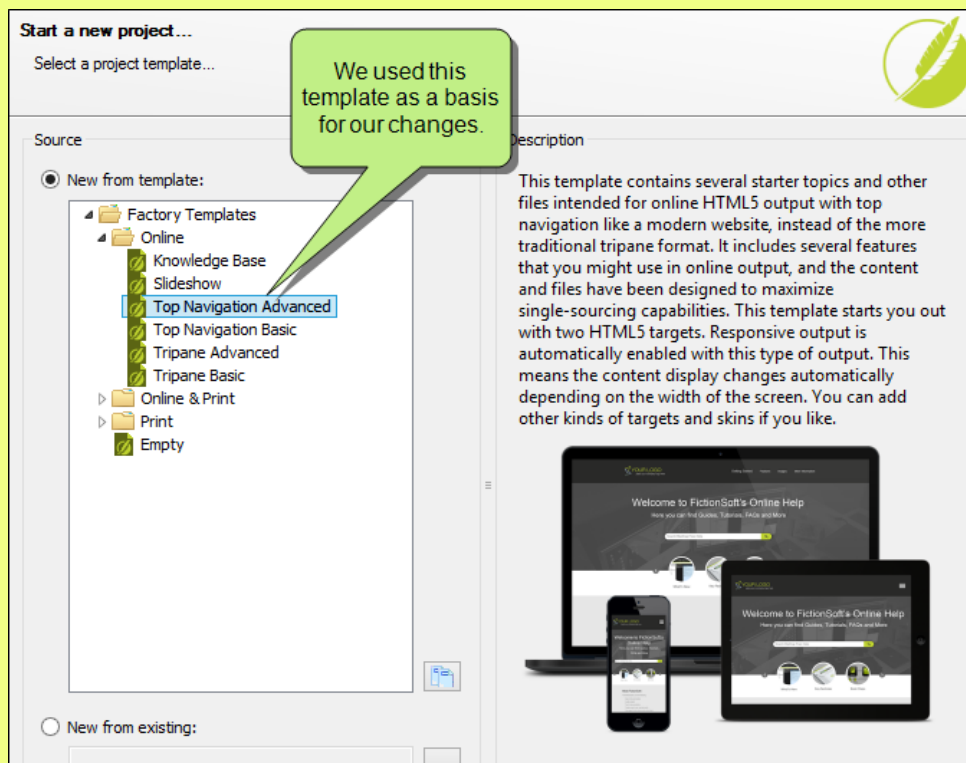
If this kind of advanced styling is beyond your ability, you can either find someone who is very knowledgeable in CSS to help you, or you can follow the other scenario where you borrow some files and styles from one of Flare's Top Navigation templates, where this look has already been achieved. See "Advanced Conversion to Top Navigation Output" on the next page.

Advanced Conversion to Top Navigation Output

Following are examples of how HTML5 Top Navigation was incorporated into an existing project by borrowing some files and styles from a Flare Top Navigation project template. Specifically, these examples show how we modified Flare's actual online Help project, converting it from Tripane to Top Navigation output.

STEP 1: CHOOSING AN APPROACH

Because Flare's Top Navigation project templates contain some elements that we wanted to use (e.g., the look and feel of the Home page and the side context-sensitive menus), we decided to borrow those pieces and make changes to them in our existing project. So we first selected **File>New Project** and created a small Flare project based on the **Top Navigation Advanced** template.



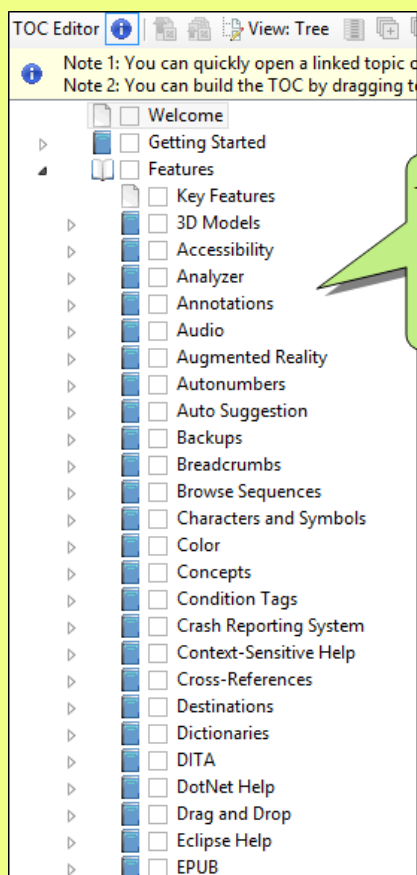
Once we had that small project, we had to make a decision. Do we bring all of the files from our existing project into the new small project, or do we bring the necessary files from the small project into our big one? Importing fewer files is always easier than importing many of them, so we decided to bring the few files from the small project into the big one.

The next decision was how to bring the files into the big project. One option was to simply open the project folders in Windows and manually copy the files from one project to the other. Another option was to use Flare's interface to import different types of files separately. The third option was to use Global Project Linking to create an import file in the big project and import only the files we wanted all at once. There isn't really a wrong answer, and regardless of the method, we would have to do a little bit of cleanup in the big project, so we decided to go with the third option—Global Project Linking—which seemed like the easiest, most streamlined way to do it.

STEP 2: PREPARING FOR TOP NAVIGATION INTEGRATION

With Flare's online Help project, we had been creating traditional Tripane output since the first version of the software. So we knew there would need to be some preparation work before switching to a different output model.

First up was the issue of limiting our menu items. Taking an initial look at our TOC, we knew something would have to change. Some books contained lots of other books and TOC entries.



This would be a lot of items in a menu format, most of which would disappear off the page when viewing them from a top menu.

For Tripane output, you can do something like that because there is more real estate to work with and a scroll bar lets end users find items that are not immediately in view. But we knew it would probably be an issue in Top Navigation output. So we had to decide what to do about it.

We could have turned off the top menu by increasing the responsive output tablet width on the Skin tab in the Target Editor. This would force the output to use the side flyout menu only, even if the output was being viewed in a regular browser. But we wanted to keep the top menu, so we decided against that.

In the end, we decided to restructure much of the TOC, placing items into different categories to keep books shorter. This was especially the case for the first couple of levels of the TOC, because we decided to use a default depth of 2 for the top menu (the root menu at the top and one submenu under it). Any levels deeper than that would show up in a side context-sensitive menu, where the number of items is less important than it is in the top menu. We also removed some TOC items that we felt did not need to be in a menu; instead, they would be accessible through search and links from higher profile topics.

This turned out to take more time than any of the other steps. But it was a good exercise because some TOC books had grown a great deal over the years, and it was time for some reorganization anyway. Once this was done, we were ready to proceed with importing files from the template project.

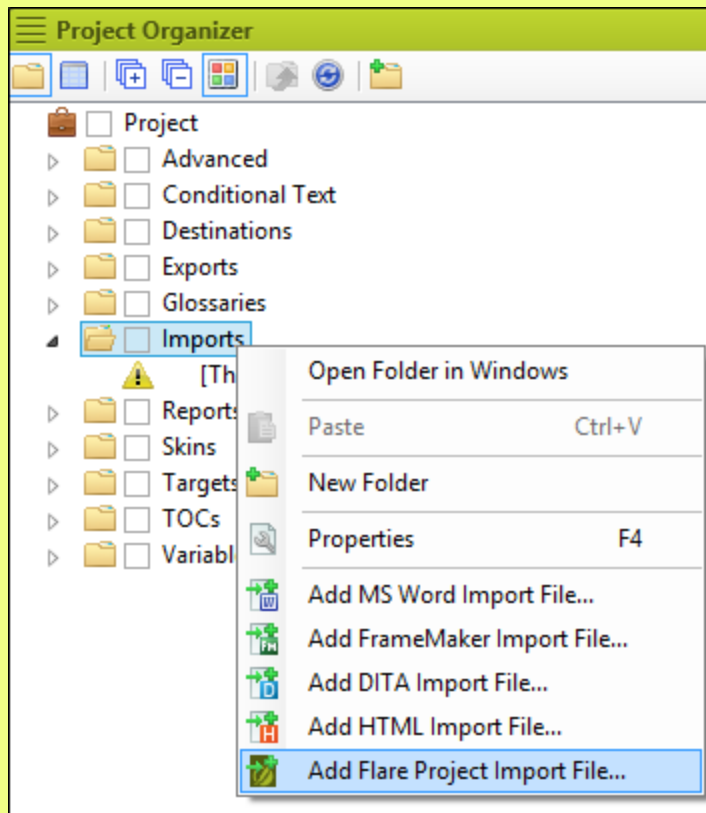
STEP 3: IMPORTING FILES VIA GLOBAL PROJECT LINKING

Next, we had to decide which of the files from the small Top Navigation project we wanted to import. Obviously we wanted the Top Navigation skin and its related files that give the Top Navigation project template its look and feel. We also knew we wanted to emulate the Home page in our project, including some images. But we didn't need most of the topics and many of the images, as well as most of the project files. So after taking a closer look at the files in the small project, we decided to import the following (your list of files might be different):

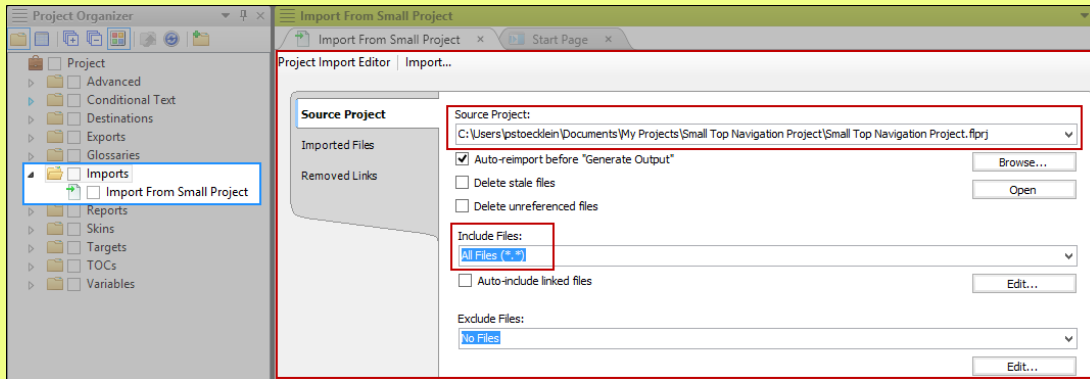
- » **Home.htm** This is the only topic file we imported. It is the unique topic used for the Home page in the template. We would replace the startup topic in our existing project with this one.
- » **BackgroundImage.png** This is the large "hero" background image seen on the Home page. We had our own background image in mind, but we thought it was best to bring this one over because there were references to it elsewhere in the template project. Once we had imported the image, we would replace it with our own image, which we would give the same file name. That way, we wouldn't have to "re-hook" it where other files referenced it.
- » **BasicSteps.png** This is one of the circular images used in the slideshow on the Home page. We wanted to use the same image in the same kind of slideshow, so this was an easy decision.
- » **FacebookIcon.png** This is the small Facebook image seen in the footer of the Home page.
- » **KeyFeatures.png** This is another slideshow image we wanted to use.
- » **TwitterIcon.png** This is the small Twitter image seen in the footer of the Home page.
- » **WhatsNew.png** This is the third slideshow image we wanted to use.
- » **HomePage.flmsp** This is the master page that is associated with the Home topic. It contains a unique configuration and references to styles we needed, so we definitely wanted this file, even though we already had a master page (which we would no longer use).

- » **OtherTopics.flmsp** This is a second master page in the template project that is used for all of the topics except the Home page. Again, it has a unique configuration, especially with its inclusion of a Topic Toolbar proxy and the context-sensitive Menu proxy, which we wanted to add to our output.
- » **StylesForHomePage.css** This is the stylesheet created specifically for use by the Home page. This is where a lot of the hard work exists to get the Home page to look the way it does, so it was one of the most important files we wanted to import. However, we decided not to import the other stylesheet in the template project because we already had our own stylesheet for the rest of our topics. However, there was one style in the template's main stylesheet that we wanted to use, because it affected the look of the context-sensitive menu. But because it was just one style, we just decided to copy it over manually into the main stylesheet in our big Flare project.
- » **HomeSearchBar.flskn** This is a special skin component that the template project uses for the central search bar on the Home page. It's already designed the way we want, so we brought it into our project.
- » **HTML5 - TopNavigation.flskn** This is the main HTML5 Top Navigation skin that gives the output the top menu and search bar, as well as the side flyout menu when the output is displayed on smaller devices.
- » **SideMenu.flskn** This is a special skin component used for the context-sensitive menu that appears next to each topic when it is opened. It was already modified with the look we wanted, so we imported this too.
- » **TopicToolBar.flskn** This is a special skin component used for the topic toolbar that appears above each topic. It was already modified with the look and alignment we wanted, so we chose this file as well.

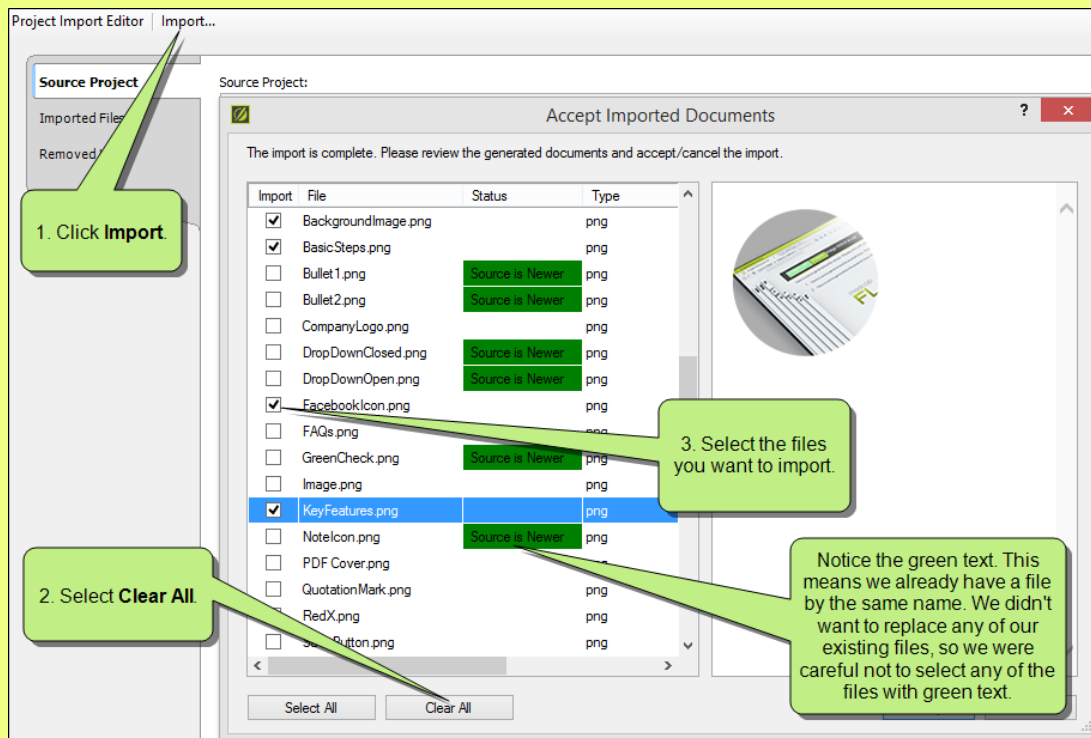
To bring the desired files into our project, we first created a project import file. We did this by opening the Project Organizer, right-clicking the **Imports** folder, and selecting **Add Flare Project Import File**.



After naming the import file, the Project Import Editor opened. In this editor, we pointed to our small Top Navigation project that we created from the template, and we told Flare to import all of the files, even though we really didn't want all of them. We'll explain in a moment why we did this.

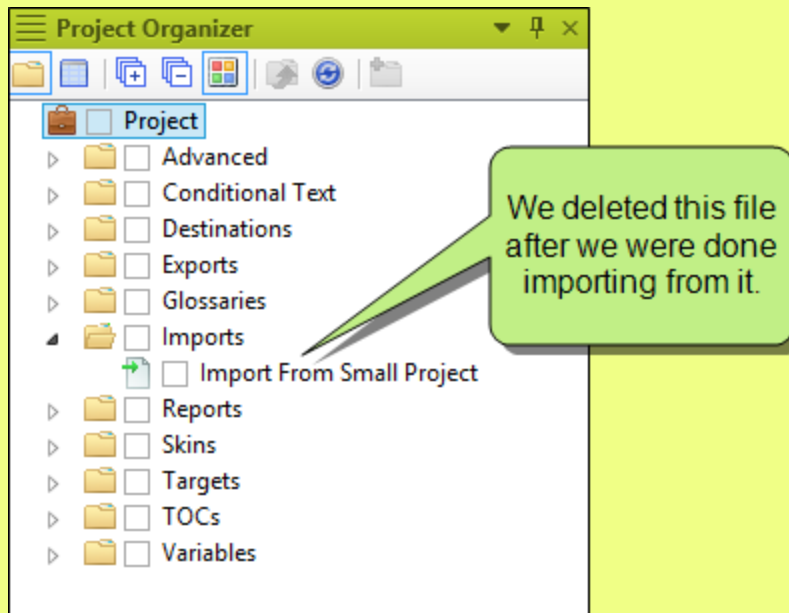


We saved the import file. Next, in the local toolbar we clicked **Import**. This opened the Accept Imported Documents dialog, which listed and automatically selected all of the files in the small project. However, we didn't want to import all of the files, but rather only a few of them. So we clicked **Clear All** to remove all of the check marks. Then we clicked in the check boxes for the files that we wanted to bring into our big project. Also, we wanted to avoid importing files that were already in our project with the same name, so we were careful not to select any of the files with green text next to them.



After clicking **Accept**, all of the selected files were imported into our big project. The imported files were placed in the same locations where they existed in the small project, ensuring that any links to and from each other were not broken. That would save us some work.

Finally, after all of the files were imported, we closed the Project Import Editor, opened the Project Organizer, and deleted the import file.



We did this for a few reasons. First, we wouldn't need it anymore. Second, Global Project Linking creates a link from the files in the source project to those in the child project, so anytime we tried to edit one of these files in our big project, Flare would give us a warning. And third, there is an option in the Project Import Editor that, if not deselected, would automatically import all of the files from the source project (not just the ones we want) if we begin to generate output. But we did not plan to use the import file the way you might normally use it for Global Project Linking, and we were all done with it. So by deleting the file, it removed any links between the small source project and our big one. Therefore, we wouldn't see any warnings when we tried to edit a file, and we wouldn't accidentally import files we didn't want.

One final note. After importing the files, we also placed condition tags on all of them. This isn't something that you must necessarily do with your own project, but our workflow is to always have a condition tag on every file in our project, so we made sure to put the correct condition on each file we imported.

STEP 4: REPLACING CONTENT AND LINKS

Now that the borrowed files were in our big project, we could make some adjustments to them. This meant replacing some content and links.

First, we opened the Home.htm topic, where most of our work would be done. Initially the file didn't look much like the output from the template project, but that was okay, because we knew it needed to receive some information from us, including which stylesheet it should use. In this topic, we did the following:

- » We replaced the few variables in it with our own, and we also entered different text in some places. Due to the way the Home page topic is designed—with elements such as multiple columns of information—we tried to keep our content to about the same amount and sizes as that used in the template. If you do that, your transition will go much more smoothly, and you will have to do less tweaking in the stylesheet. However, even in our case, we weren't able to do this precisely. Under our "Welcome to Flare" section in the Home page, we added a bit more text than is found in the same area in the template. This resulted in buttons overlapping with text when the output was viewed on a mobile device. Therefore, we adjusted a couple of styles in the Home page stylesheet to make it look right. For details on what we did, see step 8.
- » There were also some missing links to images for PDF guides. We just right-clicked on each "Missing file" bar, selected Image Properties, and selected the images we wanted to use. (We also resized our images to the same size as those in the template project so they would look good.)
- » We replaced the cross-references, as well as the image links on the slideshow and PDF guides so they would point to our own files.
- » We removed the fourth slide of the slideshow, because we weren't going to use it. Then we added two new slides with different images.
- » And finally, we wanted to include videos, but we needed them to have different YouTube links, so we changed those too.

Also, remember the big hero background image that we imported? We created our own hero image, named it the same thing ("BackgroundImage.png"), and in Windows we simply replaced the image we imported with our new one.

Next, we opened the HomePage.flmsp master page file. In this file, we did the following:

- » We replaced the missing logo file with our own.
- » We replaced the text and links to topics with different ones.
- » We replaced the broken variable with our own.


We also would have changed the links on the Twitter and Facebook images, but the template projects already point to our Documentation Team pages, so we didn't have to do anything with them.

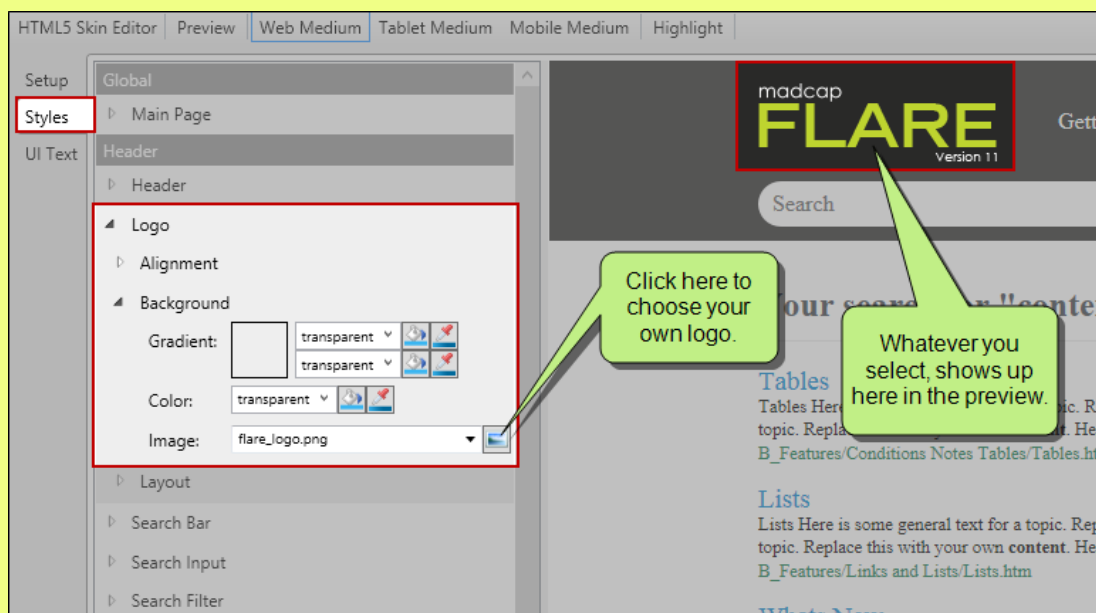
We didn't need to replace content or links in any other files, so we were ready to move on to the next step.

STEP 5: INCORPORATING THE TOP NAVIGATION SKIN

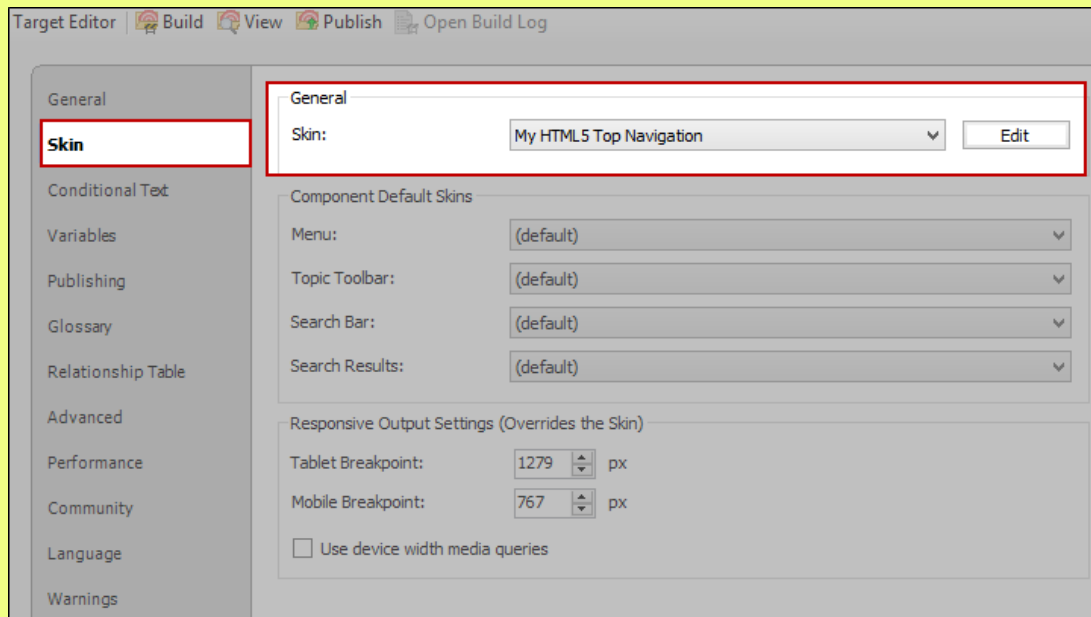
We imported the Top Navigation skin from the template project, so we didn't need to add one. Also, we decided to keep the look of the skin, with a few small exceptions.

First, we opened the Top Navigation skin in the Skin Editor, and we replaced the generic logo with our own.

To do this, we selected the **Styles** tab and expanded **Logo>Background**. Then we clicked  and selected our logo image.



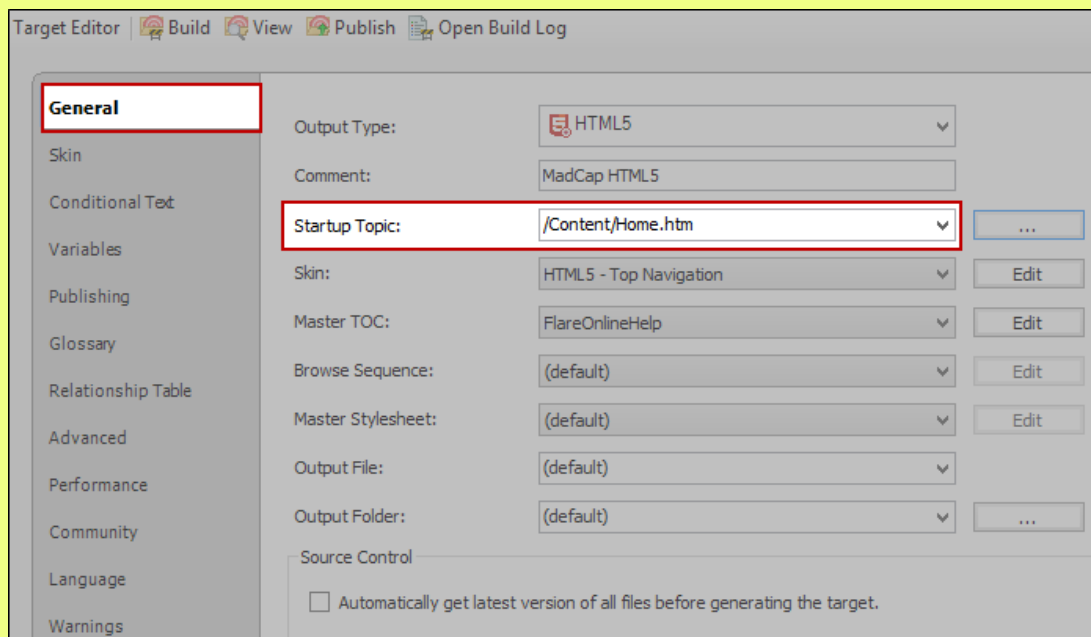
We also told Flare to use our new HTML5 Top Navigation skin. We opened our HTML5 target and selected the **Skin** tab. In the **Skin** field we selected the Top Navigation skin that we imported.



STEP 6: INCORPORATING TOPICS

The only topic we imported was the Home.htm topic. We already made some text, link, and image changes to it in a previous step. And we didn't add that topic to our TOC, because the best practice is not to include the Home topic in the top menu, but rather to link it from the logo. To make sure this happens, we just needed to tell Flare to use it as the startup topic.

So on the **General** tab of the Target Editor, we selected it in the **Startup Topic** field.

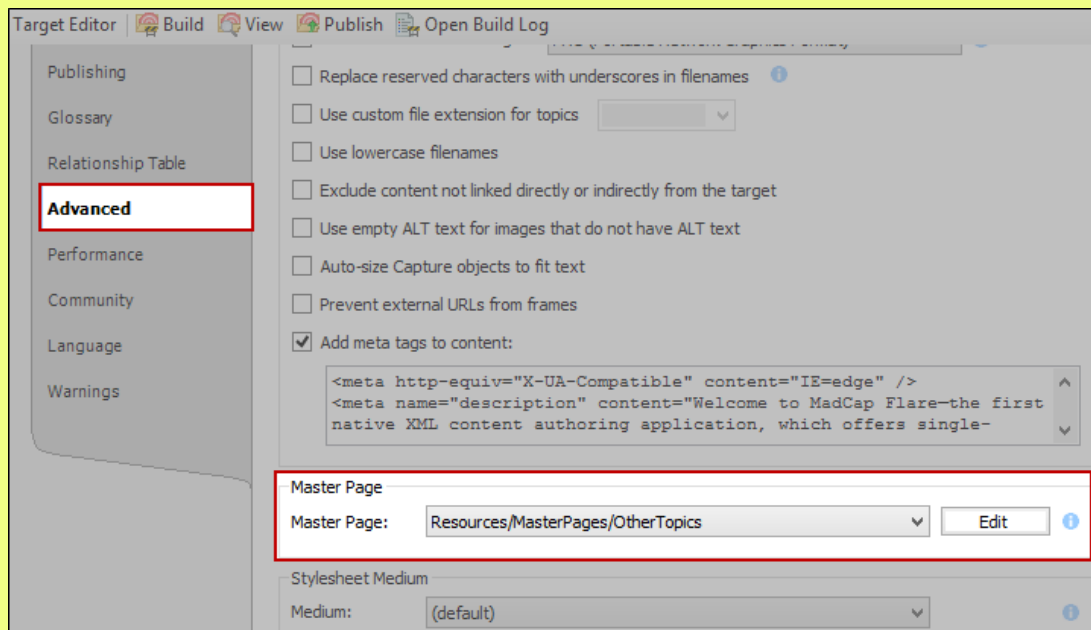


The default setting in the Skin Editor is to link the startup topic to the logo, so we didn't need to make any changes in the skin.

STEP 7: INCORPORATING MASTER PAGES

We imported two master pages—one to be used for the Home page and the other to be used with the rest of the topics. The good news is that we didn't need to do anything else to the master page for the Home topic, other than the few changes we made in a previous step. The Home topic is already tied to its master page via a style. The only thing we needed to do was associate the other master page with the rest of the topics.

So we opened our HTML5 target and selected the **Advanced** tab. From the **Master Page** field we selected the file that we imported.



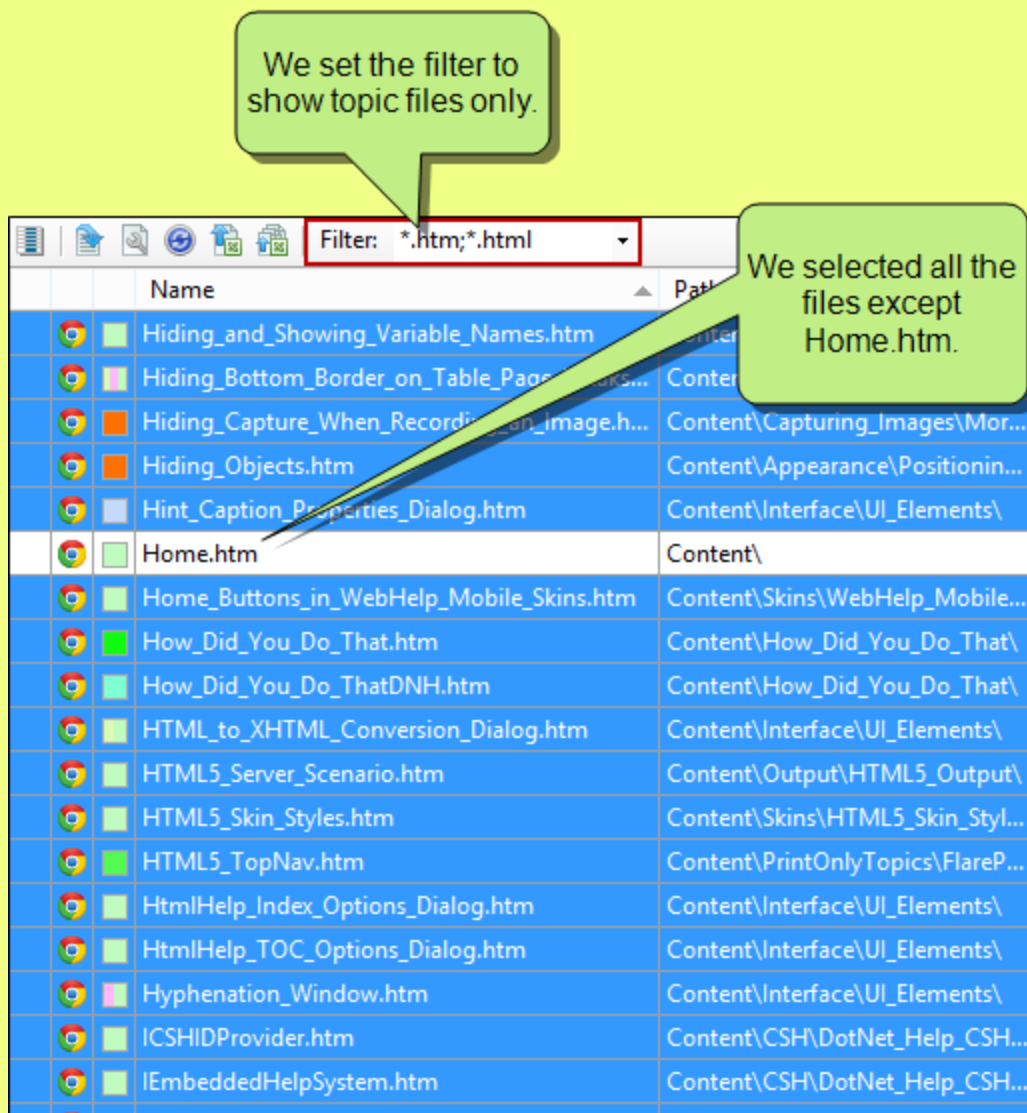
STEP 8: INCORPORATING STYLESHEETS

Remember when we initially opened the imported Home topic and it didn't look right? By incorporating the stylesheets we fixed that.

In our large project, we had one stylesheet set at the project level so that all content files would use it automatically. But then we imported another stylesheet from the template project that is designed to work solely with the Home topic. This meant we had to change the way we work with styles, associating stylesheets to individual files rather than to the entire project.

First, we selected **Project>Project Properties**, and in the dialog we selected the **Defaults** tab. We set the **Master Stylesheet** field to **(default)**.

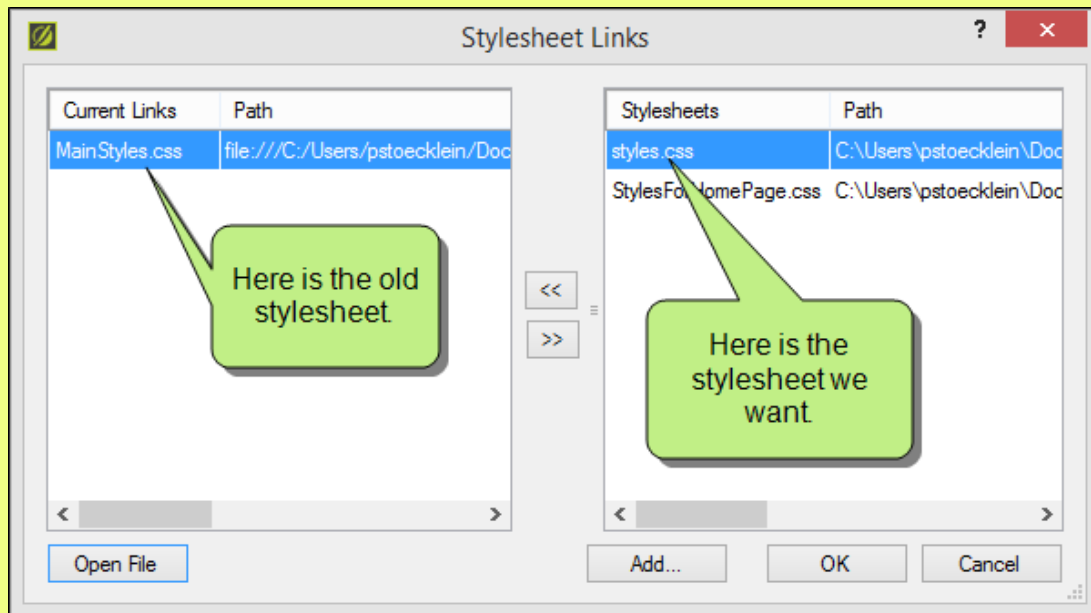
Next, we opened the File List window pane (**View>File List**). We set the filter to show all topic files, and in the grid we selected all of the files except for the Home.htm file.



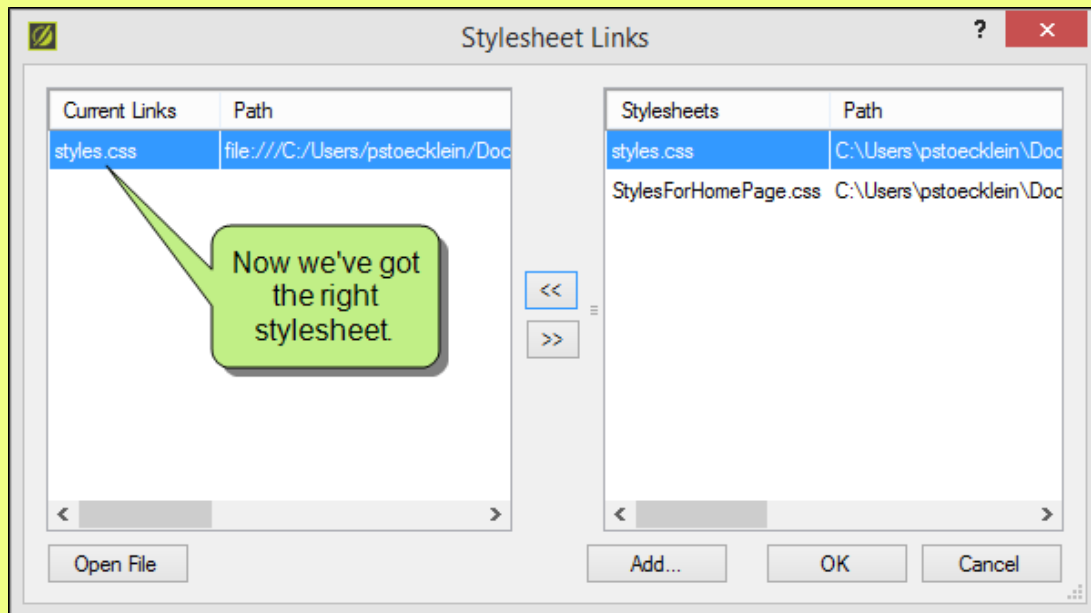
With those files highlighted, we selected **Home>Properties** (or **F4**). In the Properties dialog, we selected the **Topic Properties** tab and in the **Stylesheet** field we chose **styles** (the name of our existing stylesheet that we planned to use for all of our topics except Home.htm). After clicking **OK**, Flare applied that stylesheet to all of the selected topics.

The only other file that needed to be associated with a stylesheet was the "OtherTopics" master page. So we opened that file. When we did this, we saw a message telling us that the stylesheet linked to the master page was missing. That's because it was associated with a stylesheet in the small template project that we didn't import into our big project.

With the master page opened, we selected the **Home** ribbon and clicked **Stylesheet Links**. In the dialog, we saw the old stylesheet still linked to the master page.



We used the arrow buttons to remove the old stylesheet and add the one we wanted.

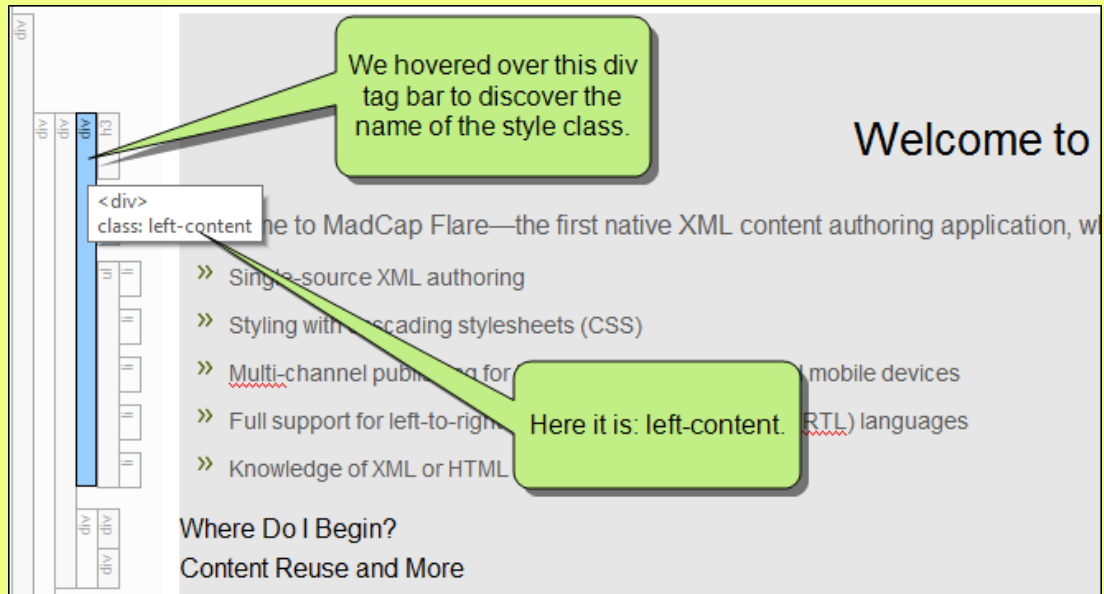


For the most part, we didn't touch the stylesheet that we used for the Home page topic (StylesForHomePage.css). However, previously in step 4 we mentioned that content shifted in the output so that buttons overlapped with text when viewed on a mobile device.



On larger screens, this didn't happen. So we needed to adjust the stylesheet so that we would have more space between the last bullet item and the first button below it, but only when the output was displayed on small screens like smart phones.

First, we opened the Home page, hovered over the **div** structure bar containing the bulleted list, and discovered it was using a style class called "left-content."



We opened the StylesForHomePage.css stylesheet in the Internal Text Editor. We scrolled to the bottom and found a media query called "screen and (max-width: 40em)." This media query contains style settings only for small screens, such as smart phones. Within this medium, we added the following:

```
.left-content
{
    margin-bottom: 80px;
}

/* Small Layout */
@media screen and (max-width: 40em)
{
    MadCap|searchBarProxy.homeSearchBar
    {
        width: 100%;
    }

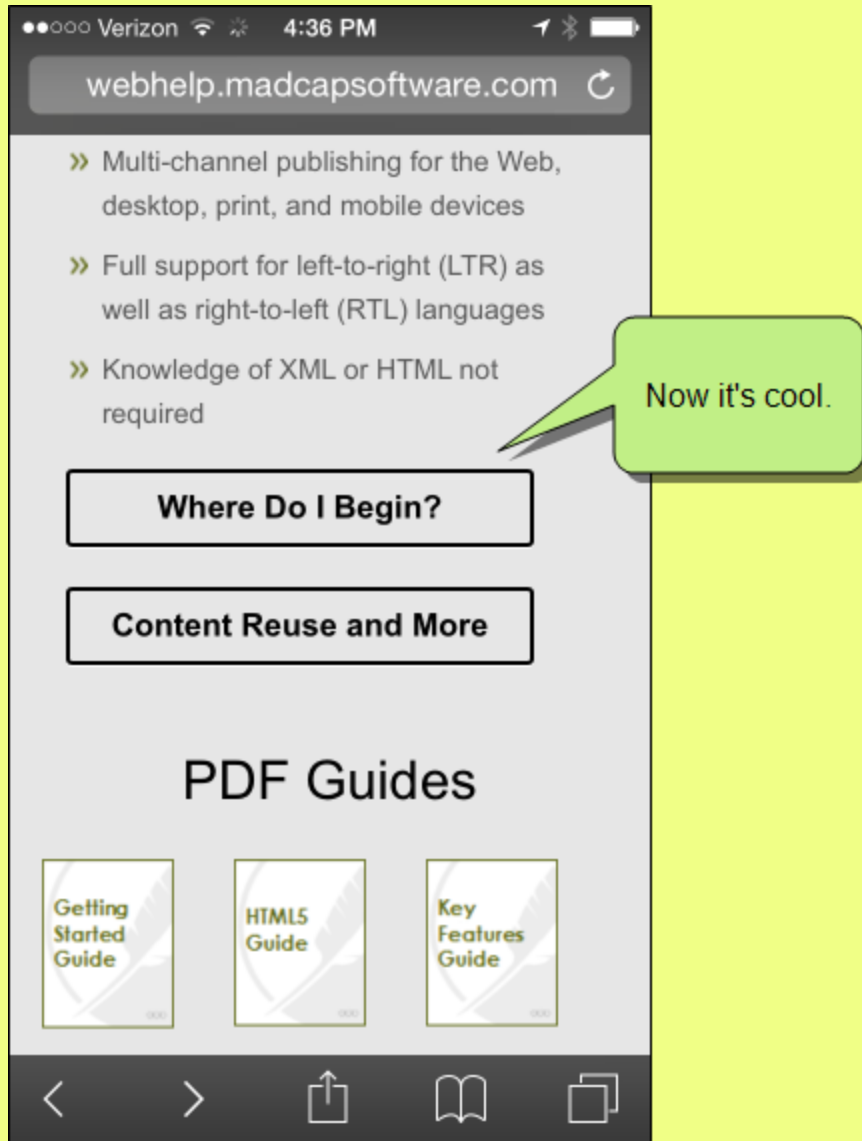
    div.topichero h1
    {
        font-size: 2em;
    }

    div.topichero h3
    {
        font-size: 1.2em;
    }

    #content-section
    {
        padding-top: 30px;
        padding-bottom: 30px;
    }

    .left-content
    {
        margin-bottom: 80px;
    }
}
```

After generating the output again, the spacing was much better.



(Another method we could have used to track down this problem is the "Inspect Element" feature in the browser output. This lets you tweak the code and see the modified results in the output at the same time. For more information about this technique, see "Modifying Foundation Grids in the Output" on page 50.)

There are a couple of other things that we did related to styles.

First, we thought that many users might want to print some topics, but we didn't want the navigation menus and search bar to be included in the printout. So we opened our primary stylesheet that is used for all topics except the Home page (in our case, our primary stylesheet is named "styles.css"). We opened it in the Internal Text Editor by right-clicking on the CSS file in the Content Explorer and selecting **Open with>Internal Text Editor**. We scrolled down to the @print medium, and we added the following:

```
nav.tab-bar, div.sideContent
{
    display: none;
}
```

Second, although we didn't want to use the "MainStyles.css" stylesheet from the small template project, there was one style that we wanted to use from it. It's a style that positions the context-sensitive menu in topics just the way we want it, allowing content to flow under it. In the small template project, we opened that CSS file in the Internal Text Editor. We scrolled all the way to the bottom of the file and copied this style and its definitions:

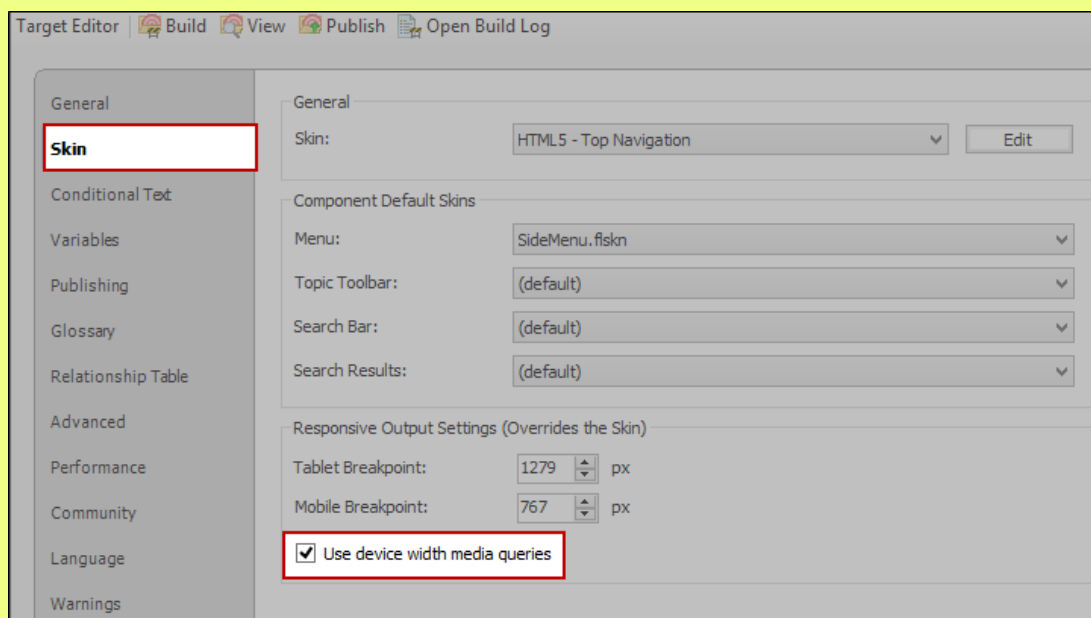
```
div.sideContent
{
    float: right;
    margin-left: 10px;
    margin-bottom: 10px;
    width: 33.33%;
}
```

Then in our big Flare project, we opened our primary stylesheet (styles.css) in the Internal Text Editor and we pasted this style and its definitions into it.

STEP 9: SETTING RESPONSIVE OUTPUT

HTML5 Top Navigation output is automatically set to use responsive output. But we wanted to set one last option to make the responsive output based on the width of specific devices, rather than based on the width in the browser. That way, the top menu would always be shown when the output is viewed in a browser, regardless of how small the output display gets. But the side flyout menu would be used when the output is viewed on a tablet or mobile phone.

So we opened the target, and on the **Skin** tab we selected **Use device width media queries**.



And that's it. The next time we generated our target, it displayed in the new HTML5 Top Navigation format.

Skin Styles

There are many ways to style your HTML5 skin, whether it be a Tripane skin, a Top Navigation skin, or one of the skin components. In addition, if you enable the skin to produce responsive output—meaning that the look of the output can automatically adjust depending on the size of the end user's device—there are some additional features for making the output look the way you want in those circumstances. See "Responsive Web Design" on page 19.

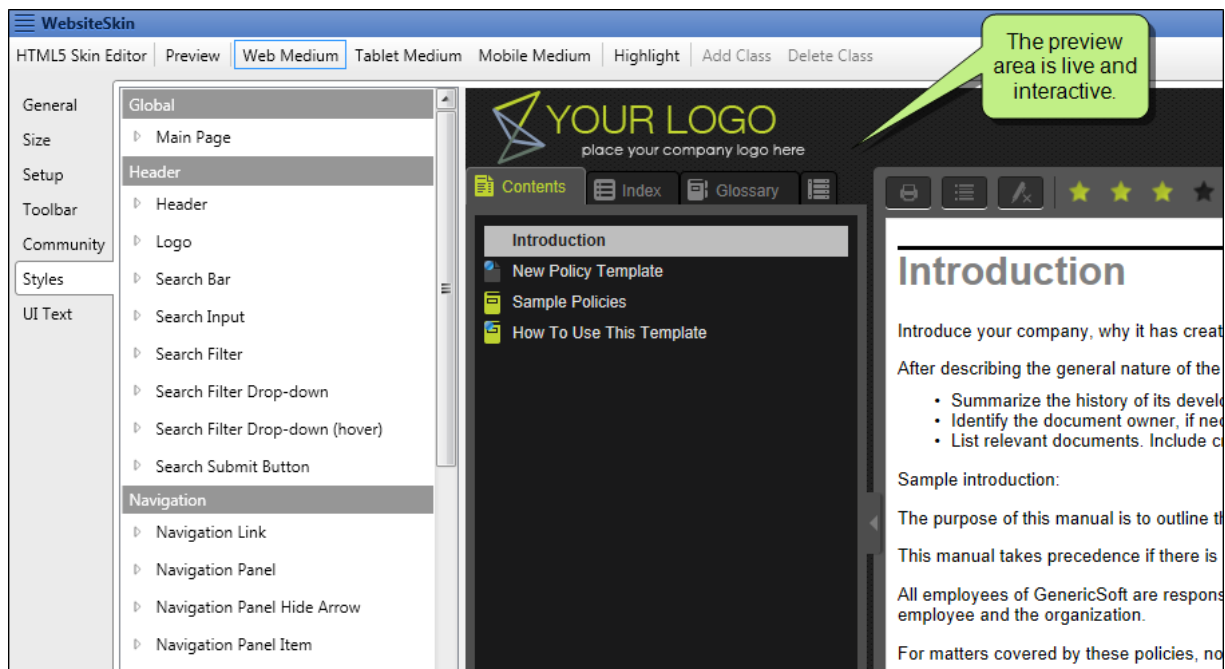
This chapter discusses the following:

Interactive Live Preview	164
Style Sections and Root/Child Properties	167
Default Values in Gray	177
Medium-specific Styles	178
Tasks for HTML5 Skins—Tripane, Top Navigation, and Components ...	180
Main Page in HTML5 Skins	204
Headers in HTML5 Skins	208
Menus in HTML5 Skins	217
Navigation in HTML5 Skins	222
Topics in HTML5 Skins	237
Search Results in HTML5 Skins	246
Feedback in HTML5 Skins	262
Topic Toolbars in HTML5 Skin Components	270

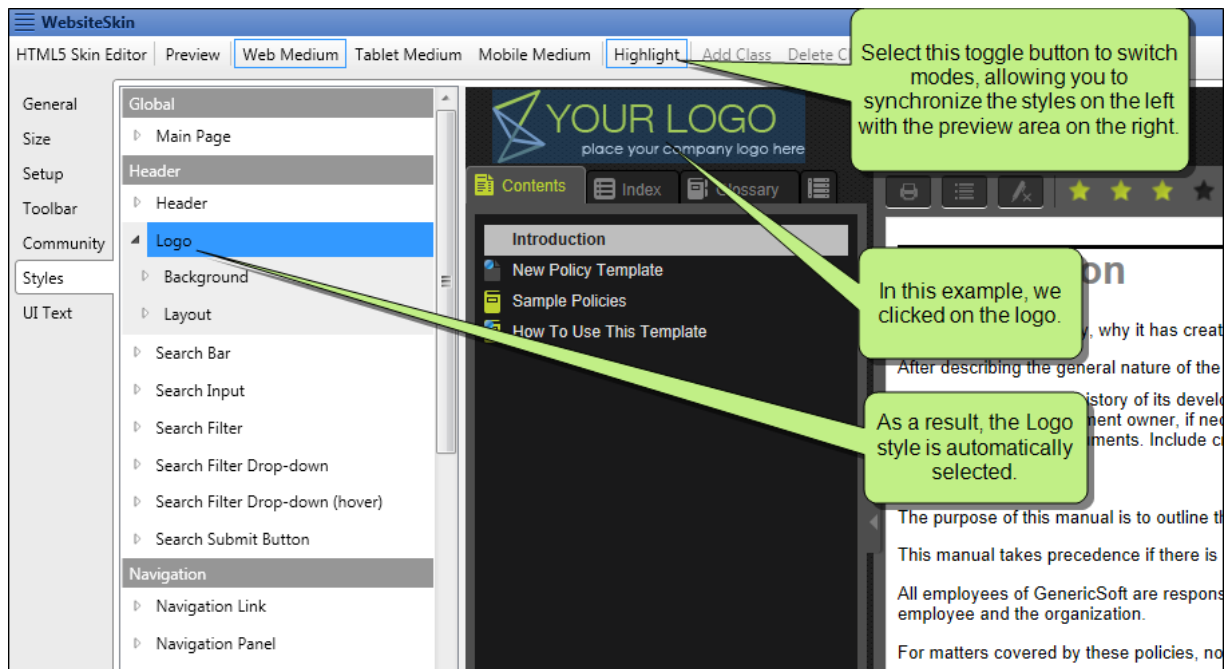


Interactive Live Preview

The preview area in the Styles tab of the HTML5 Skin Editor is more than just a static image. You can click in the preview to move around and open different areas.

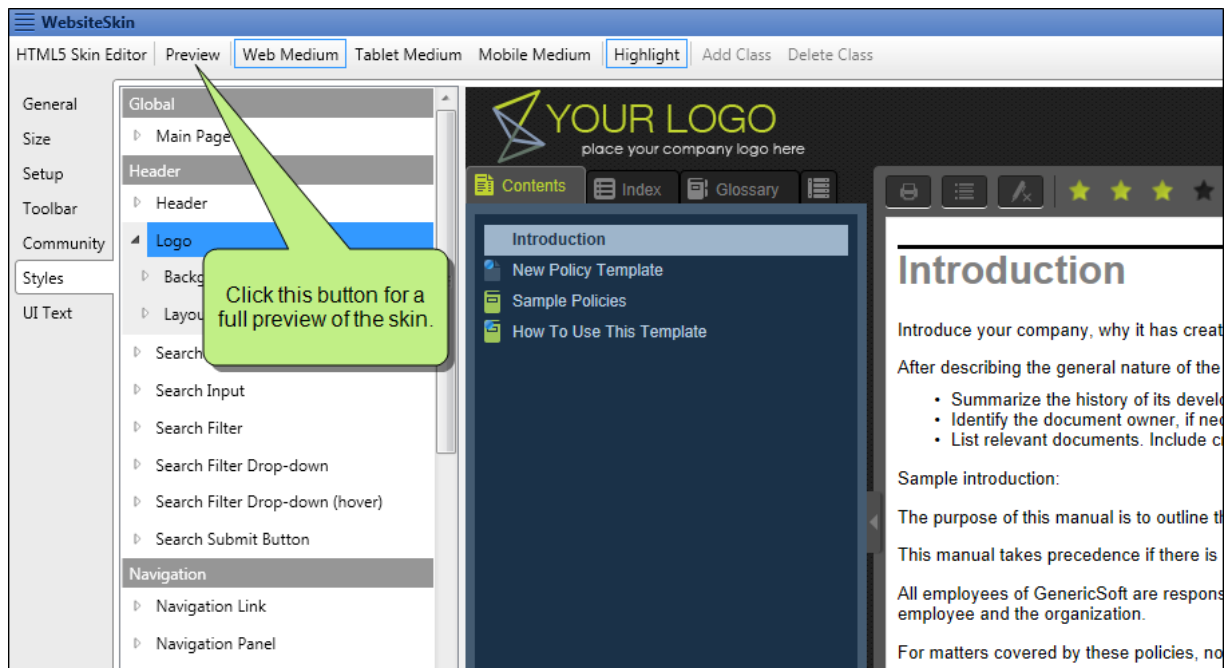


If you click **Highlight** in the local toolbar, the mode changes. As a result, different areas of the preview area become highlighted as you move the mouse over them. And when you click on an area, the corresponding style on the left is also selected. This makes it easier to know what styles you need to change in order to affect that part of the skin. The reverse is also true; as you click on styles, the corresponding area in the preview comes into focus if it is visible in the preview area.



After you make style changes, the preview changes accordingly.

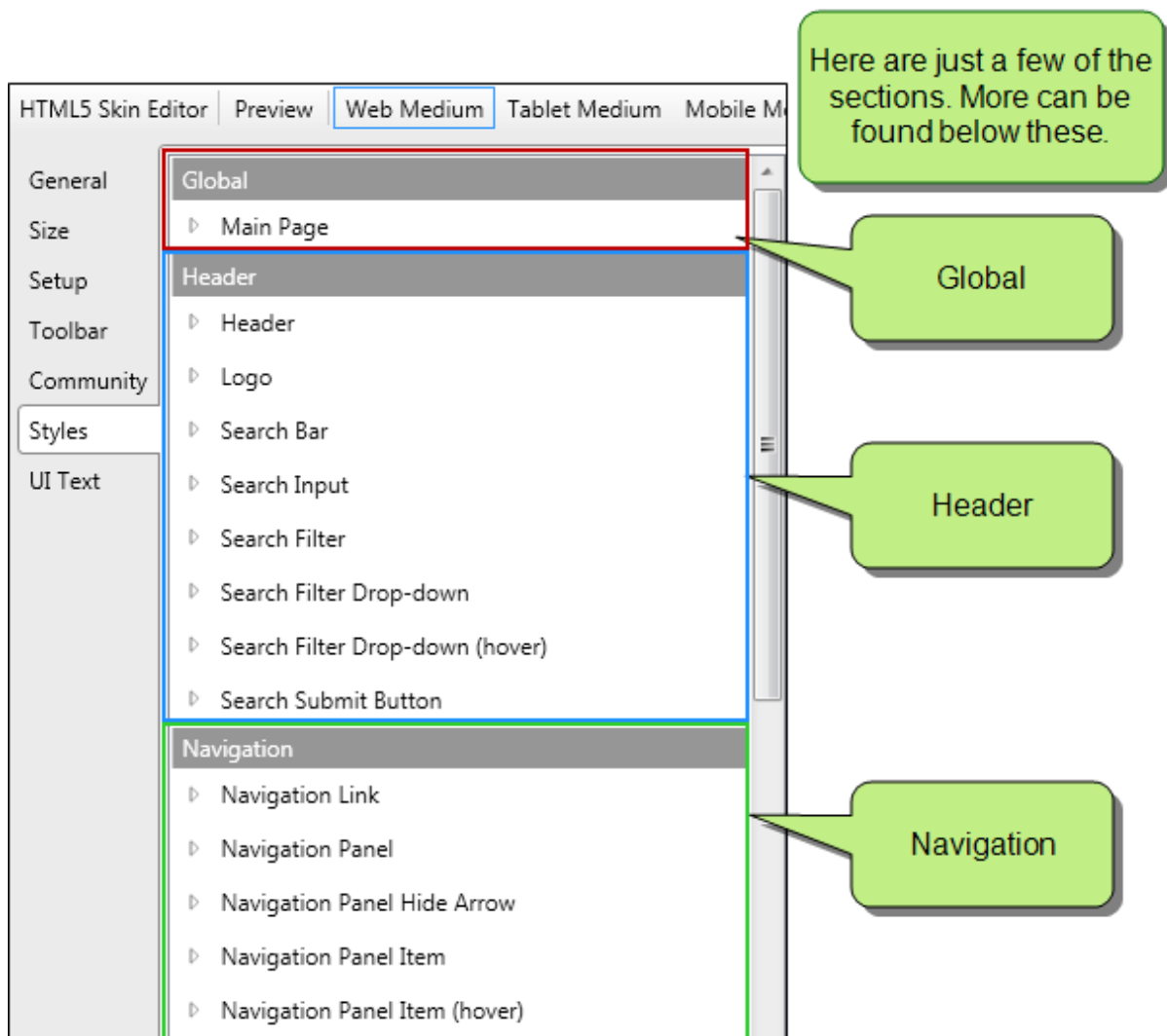
You can also click the full preview option in the local toolbar of the Skin Editor.



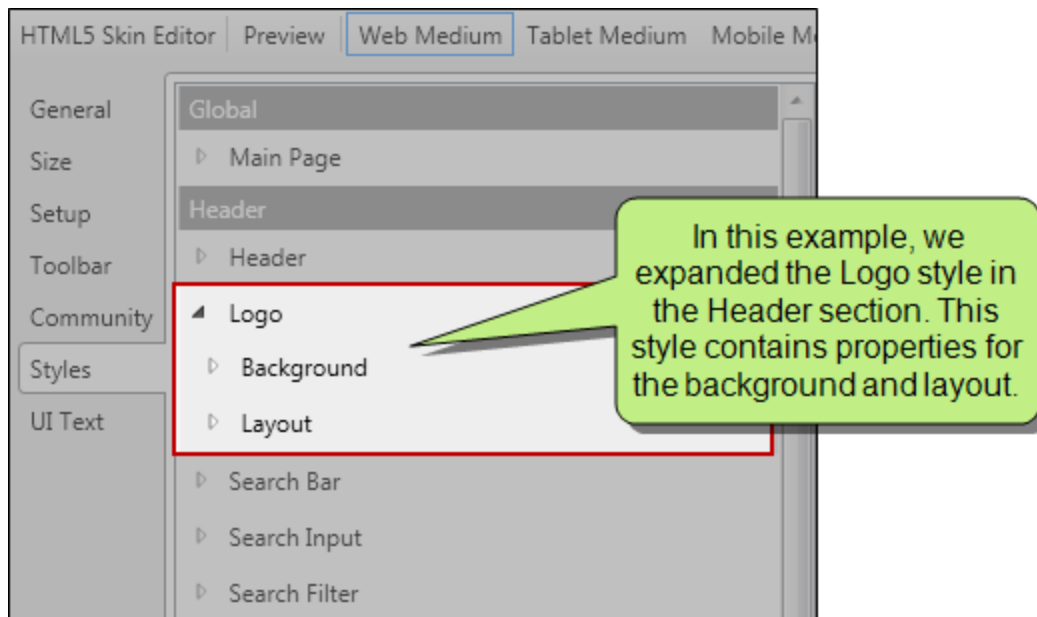
This full preview respects the responsive output setting in the skin. If you have responsive output enabled, the preview is also responsive. If you have the responsive output disabled, the preview is static.

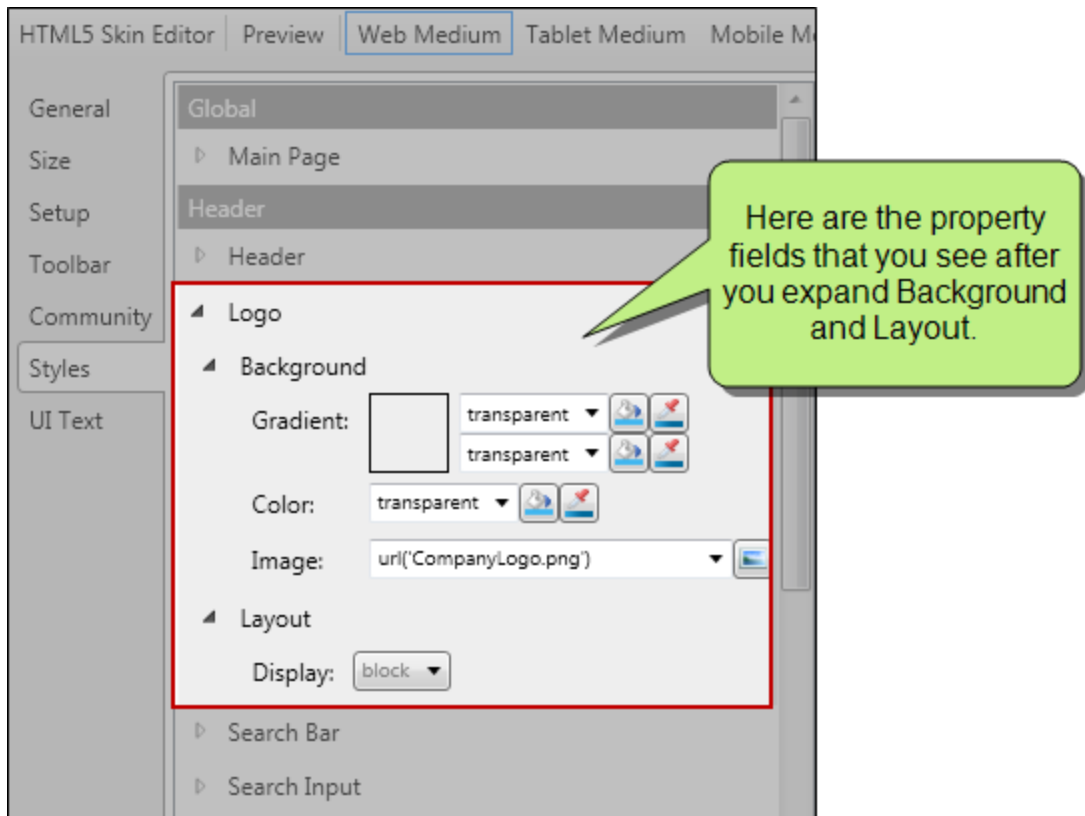
Style Sections and Root/Child Properties

When you open the Skin Editor to make changes to skin styles, you will see the styles organized in different sections.



Within these sections are the different styles you can adjust. You can expand any of these styles to see the properties within it that can be changed.



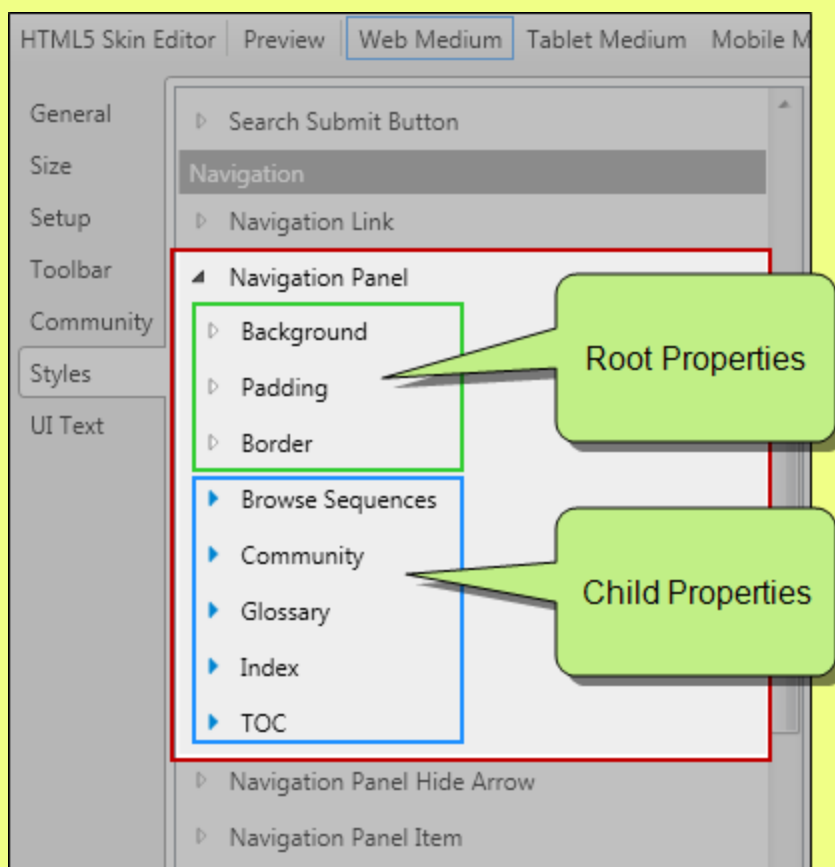


You may see two kinds of properties under a style—root and children.

Root properties are at the first level. If you make changes to a root style, all of the children styles below inherit those changes. So if you want all of the children to have the same setting, it's best to make your modifications to the root properties. The easiest way to tell the difference between root and children styles is by looking at the expand/collapse icons next to them; root styles have a white triangle icon, whereas children styles have a blue triangle icon.

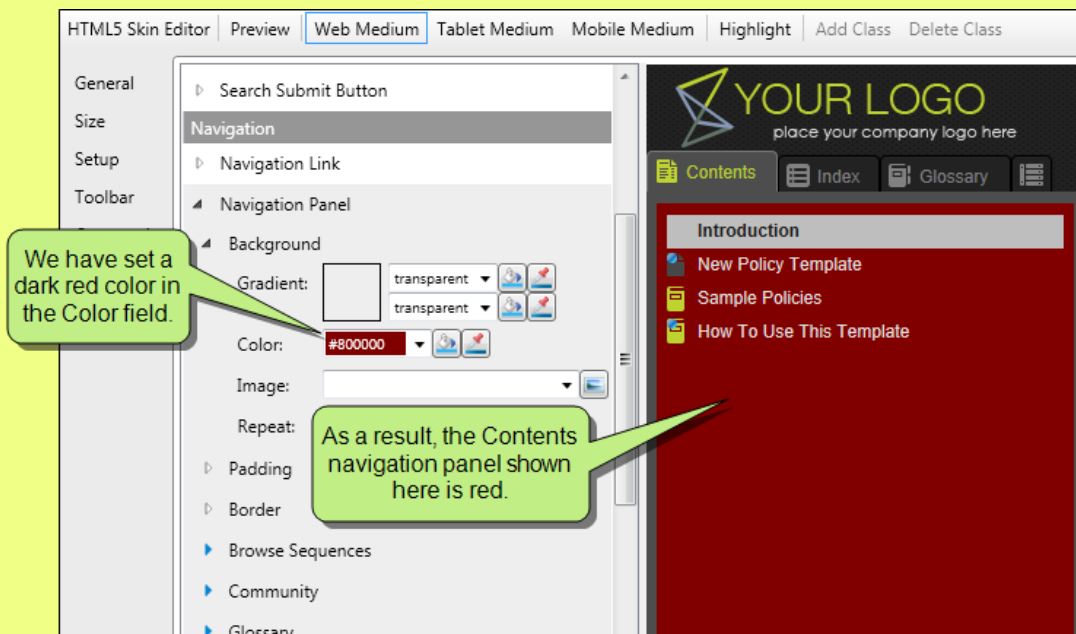
EXAMPLE

Let's say you want to change the background color of all of your navigation panels from black to dark red in the Web medium. First, in the **Navigation** section, you expand the **Navigation Panel** style. Under this style, you see three root property nodes—Background, Padding, and Border. Under these root nodes are five child nodes that represent each of the five kinds of panels you can include in your output—Browse Sequences, Community, Glossary, Index, TOC. (In the Tablet and Mobile mediums you would see Browse Sequences, Glossary, Index, Search Filters, and TOC.)

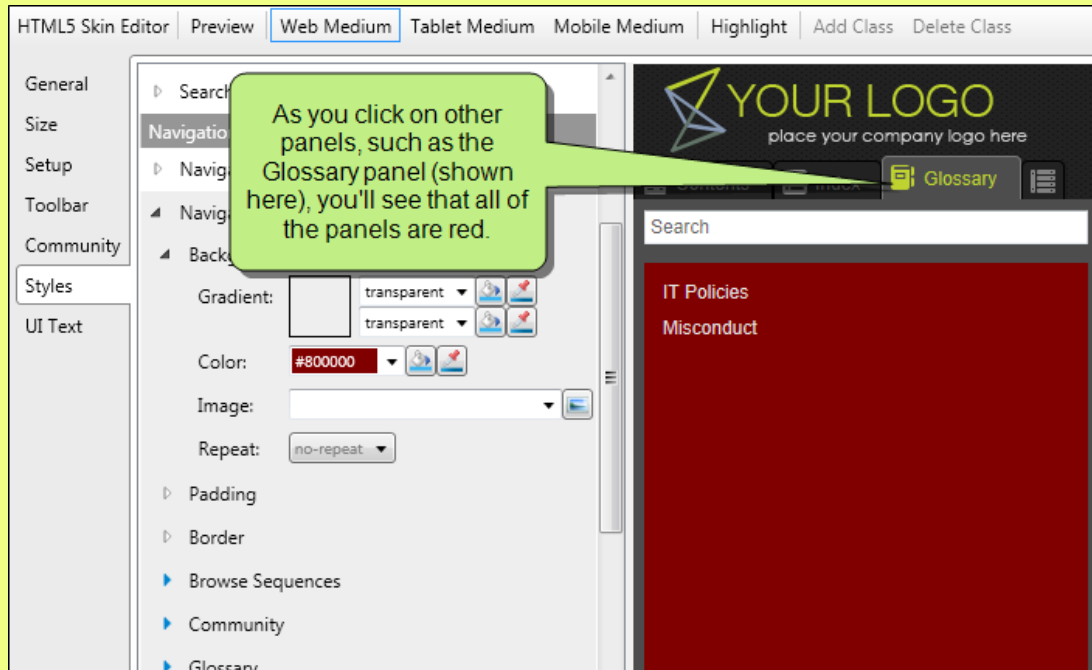


Because you want each of those five panels to have the same color, you expand the root **Back-ground** node. There are a few different fields that can control the background color. There are Gradient fields that let you add an effect where the progresses from one to another in the panel. There is a Color field, which lets you add a color without a gradient. And then there is an Image field, which lets you select an image to display in the background. The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

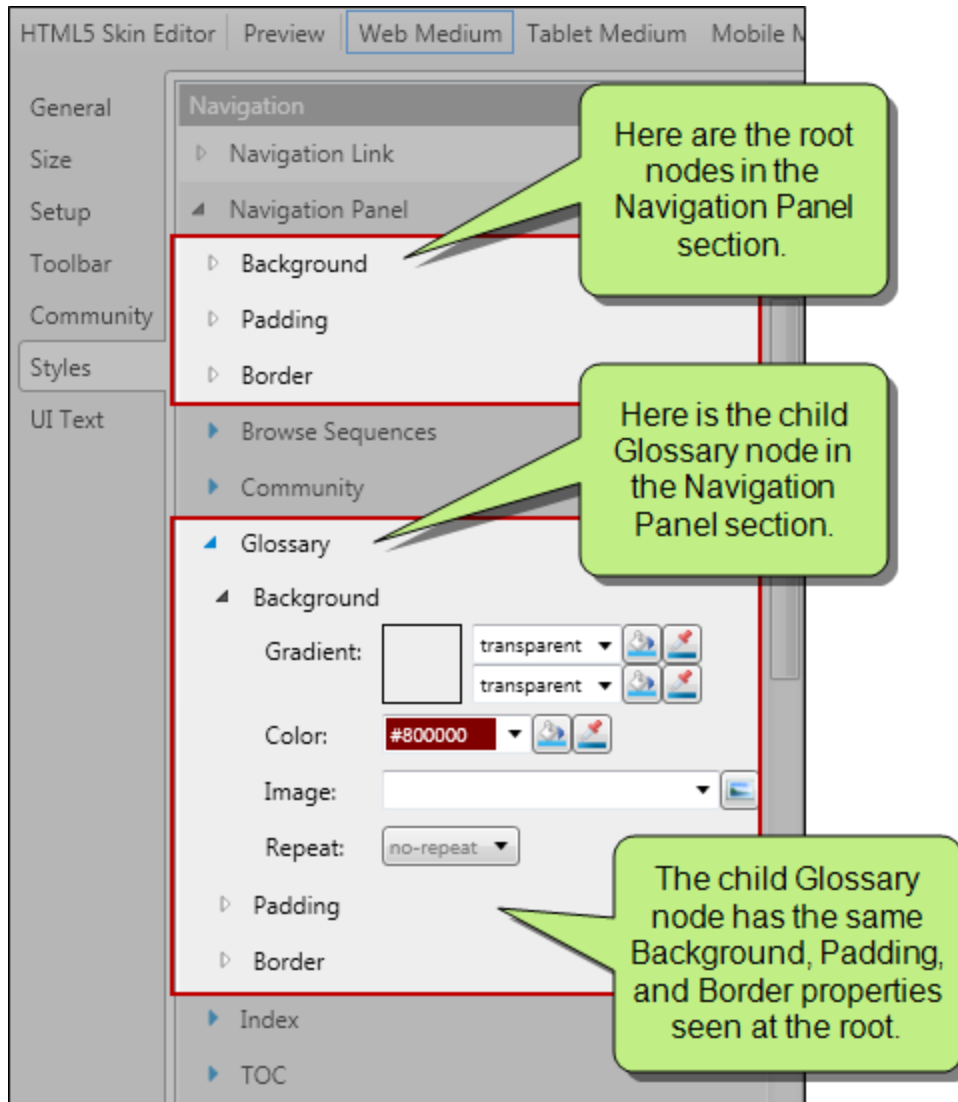
Let's say you use the Color field to choose a dark red color.



Regardless of the panel you select, it is red.



If you expand a child node, you will see the same properties that you see above in the root node.



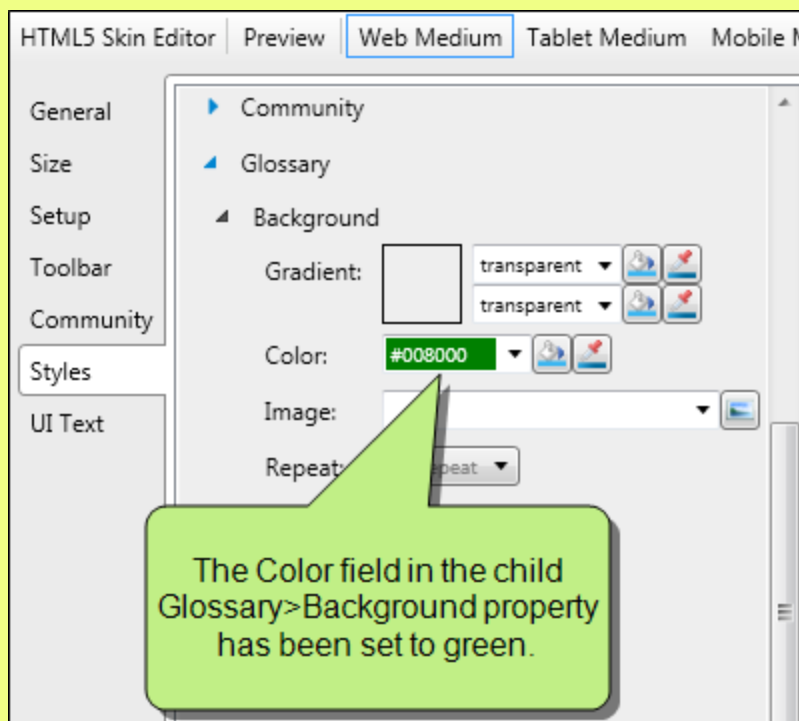
If you make changes to child properties, those settings override anything inherited from the root, and they pertain only to that child.

EXAMPLE

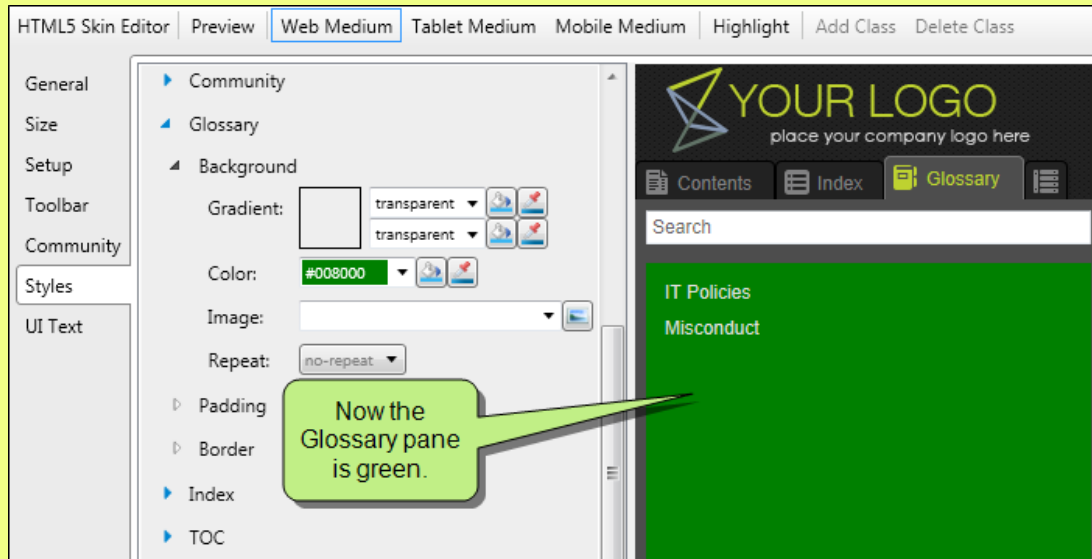
Let's say you have set the root Background property to show the navigation panel in dark red, as described in the previous example.

If you want the Glossary panel to show in green instead, you can expand the **Glossary** node, and then expand the **Background** property within it.

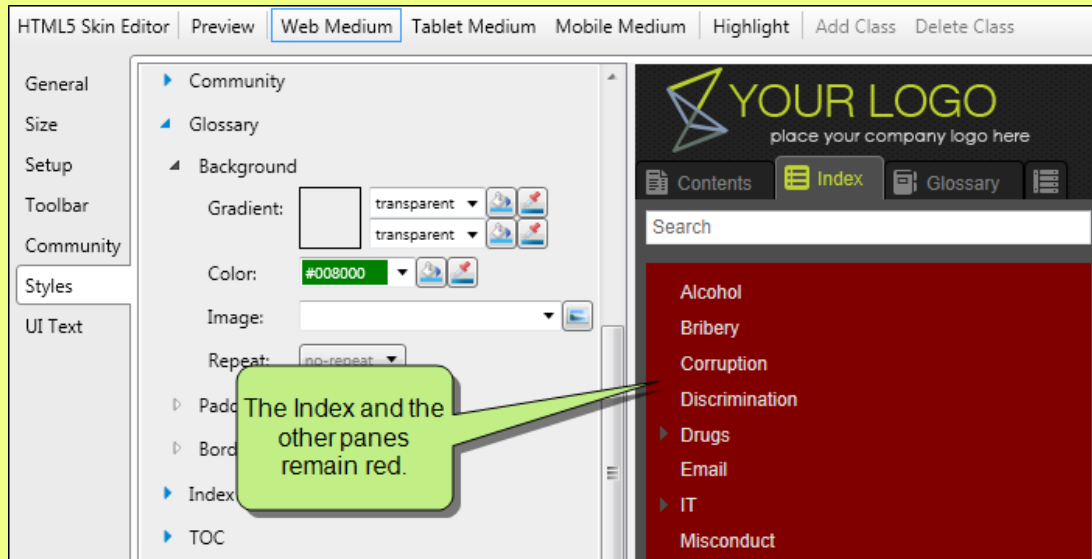
In the **Color** field, you select a green color.



As a result, the Glossary pane is green.

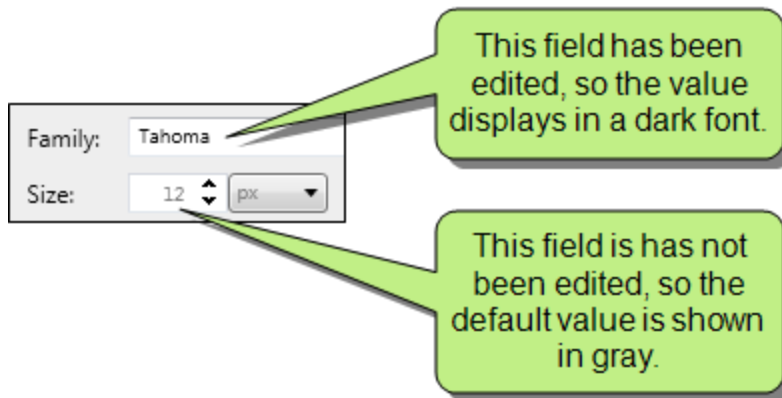


But all of the other panes remain red.



Default Values in Gray

When making changes to skin styles in HTML5 skins, you may notice that some fields display text or numbers in gray. This is the default value for that particular field. If you enter a different value, the font displays in a darker font.

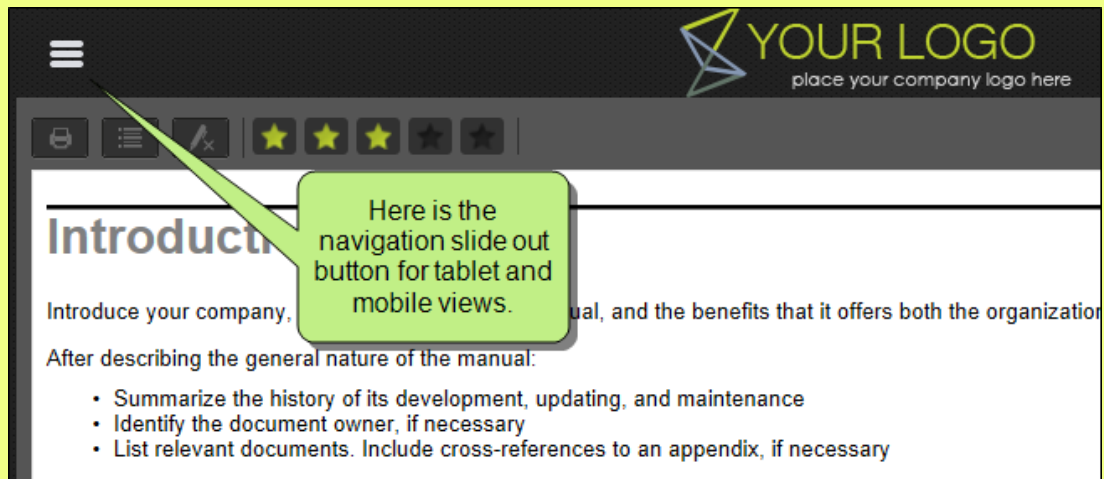


Medium-specific Styles

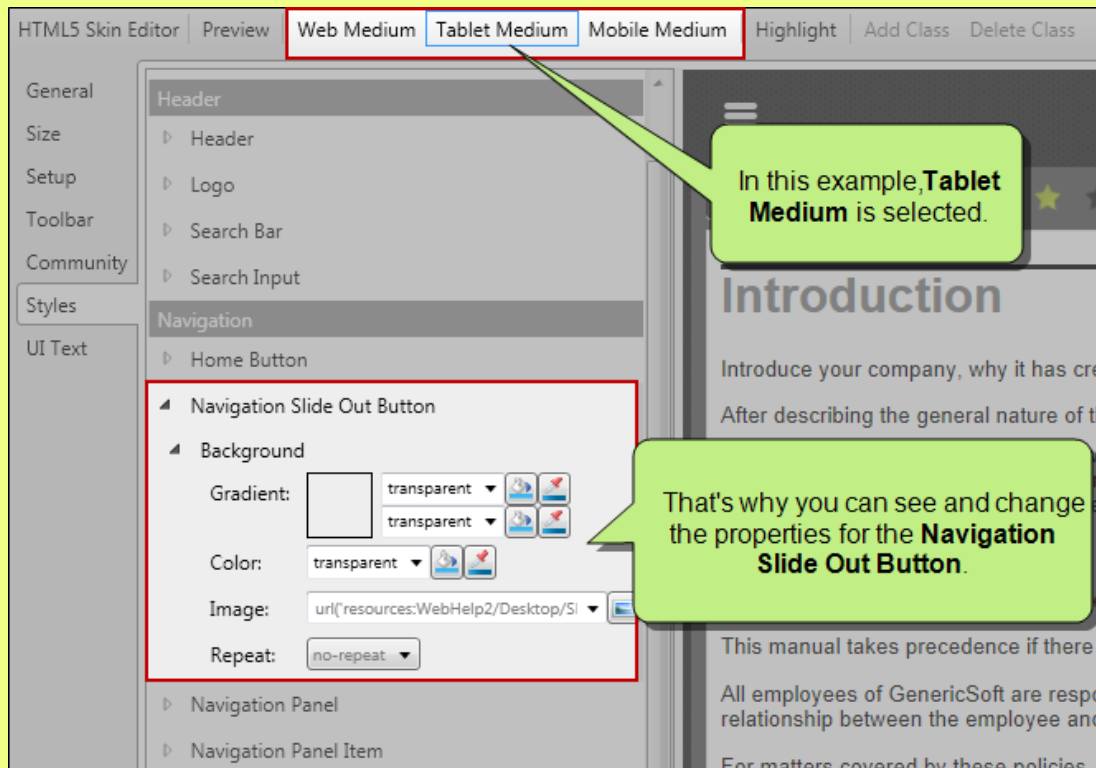
Responsive output allows you to place settings on styles using any of the three mediums—Web, Tablet, and Mobile. However, not all styles and properties are necessarily pertinent to all mediums. That's why you only see some styles and properties when you have a particular medium selected and not another.

EXAMPLE

Let's say you want to change the slide out button that you see when the output is shown in a tablet or mobile phone size.



When you select either the **Tablet** or **Mobile Medium**, you can see and adjust the **Navigation Slide Out Button**.



But when you select the Web medium, this style is not available because it isn't used in larger web browser views of the output.

Tasks for HTML5 Skins—Tripane, Top Navigation, and Components

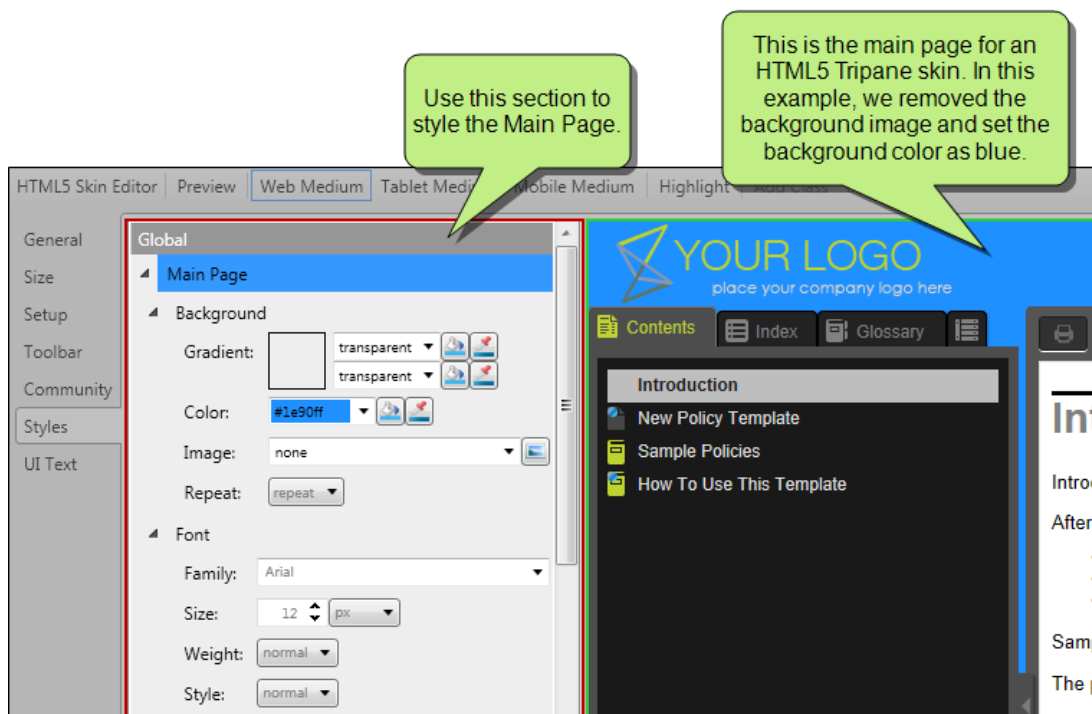
You might perform the following style tasks if you are working in an HTML5 skin or a skin component.

Global—Tripane and Top Navigation Skins

This section contains the Main Page style for Tripane and Top Navigation skins. See "Main Page in HTML5 Skins" on page 204.

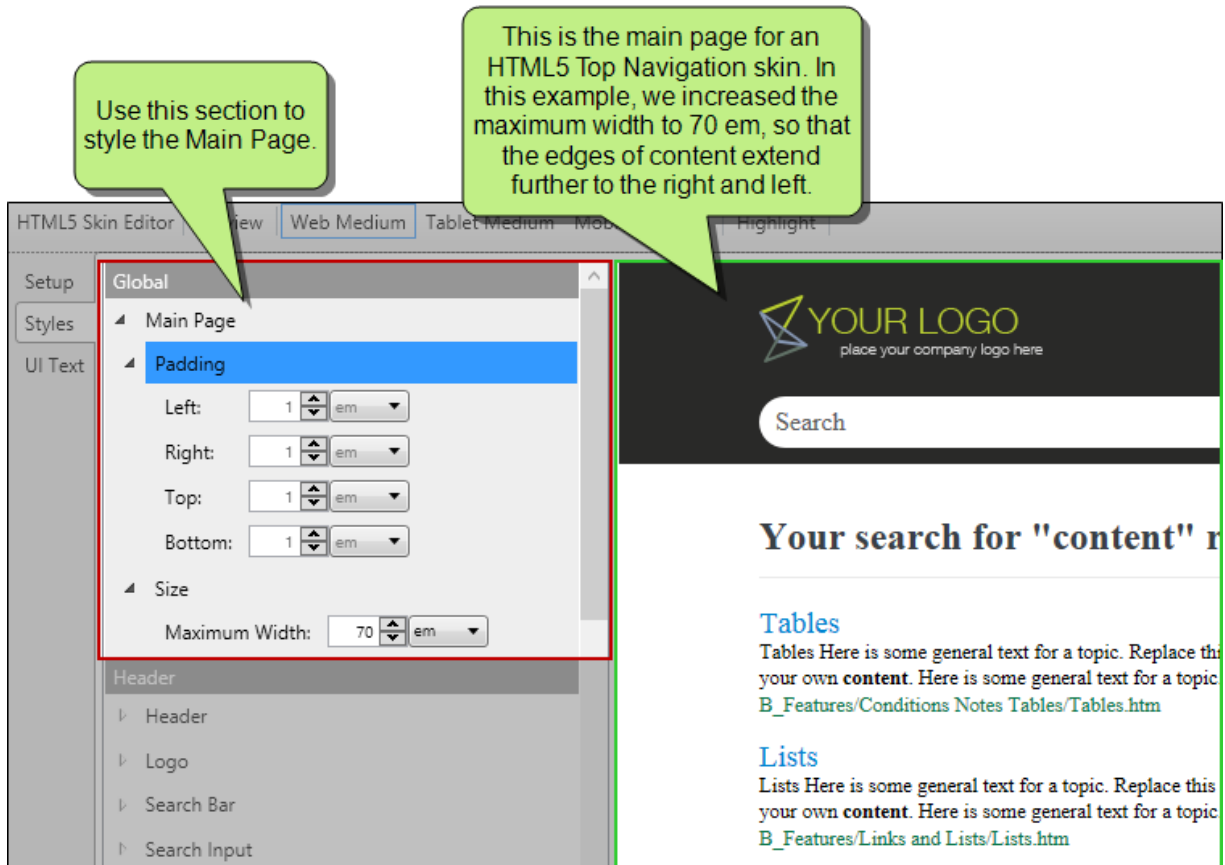
TRIPANE SKINS

For Tripane skins, you can control the font and background color for the page.



TOP NAVIGATION SKINS

For Top Navigation skins, you can control the padding and maximum width for the page.

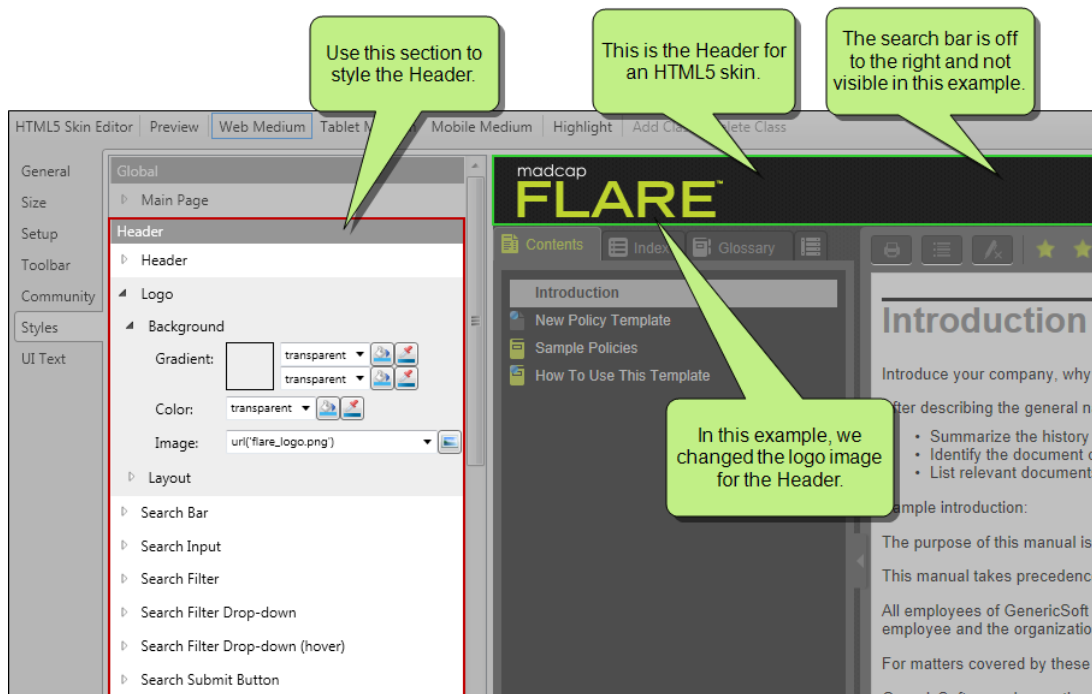


Header—Tripane and Top Navigation Skins

This is the area at the top of the screen for Tripane and Top Navigation skins. See "Headers in HTML5 Skins" on page 208.

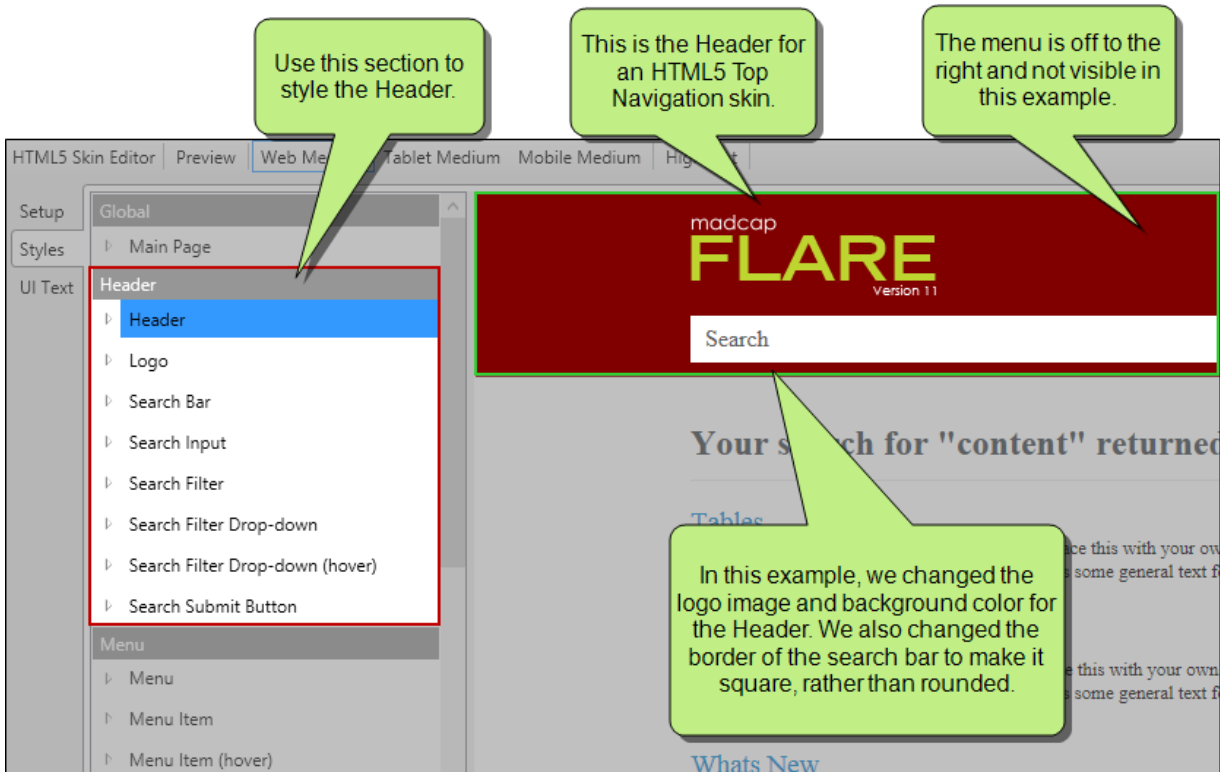
TRIPANE SKINS

For Tripane skins, this area displays a logo and the search bar.



TOP NAVIGATION SKINS

For Top Navigation skins, this area displays a logo, the search bar, and the top menu.

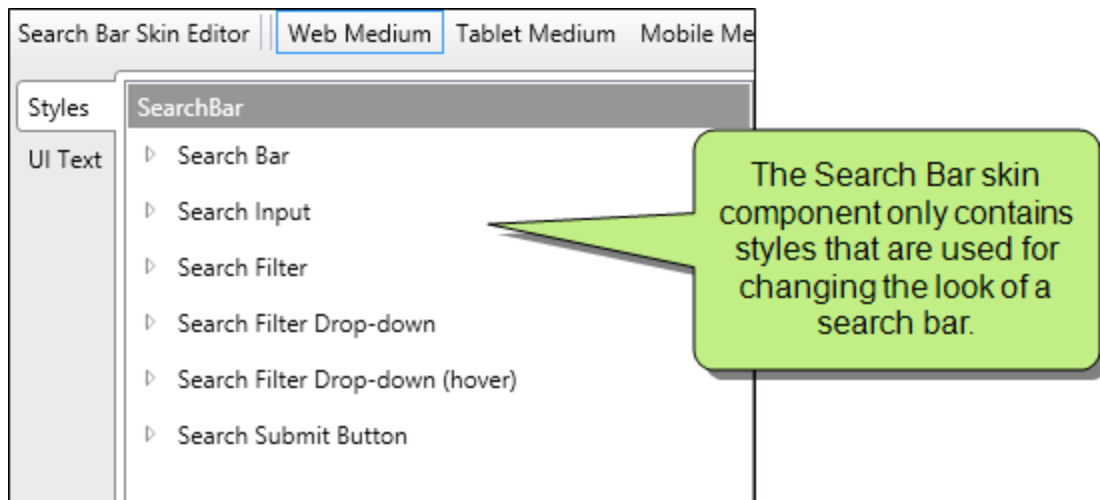


Note: If you want to style items for search results, you can use the Search section in the Skin Editor. Alternatively, you can use the Search Results skin component, which works with a Search Results proxy.

Search Bar Skin Components

One of the tasks that the Header area in Tripane and Top Navigation skins lets you perform is modifying the look of the search bar at the top of pages. Alternatively, you can use a Search Bar proxy to add a custom search bar elsewhere, and you can use a Search Bar skin component to control the look for it.

The options in the skin component are the same as the search bar options in the Top Navigation and Tripane skins.



The Search Results skin component is not available if you are generating Tripane output. It is only available when you are producing Top Navigation output, or HTML5 output without a skin selected.

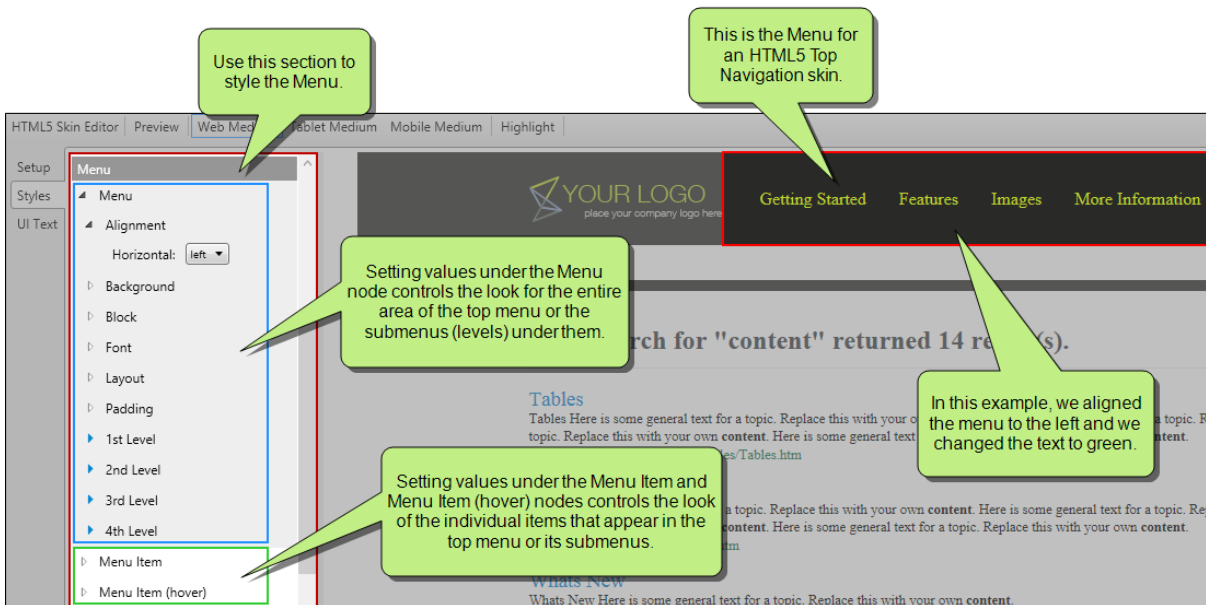
Menu—Top Navigation Skins and Menu Skin Components

You can control the look of the menu that is included with a Top Navigation skin, as well as additional menus that you add through the Menu proxy. See "Menus in HTML5 Skins" on page 217.

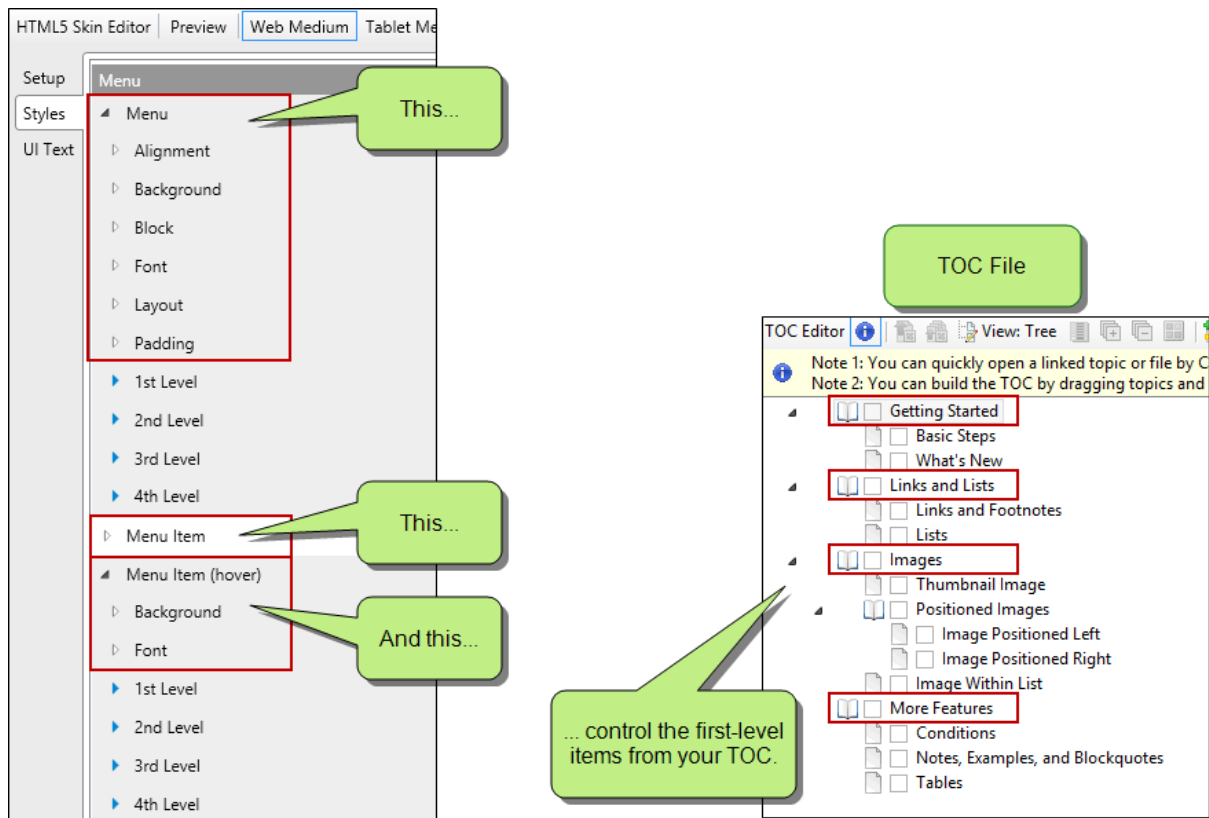
TOP NAVIGATION SKINS

For Top Navigation skins, this section is used to control the look of the menu that is displayed at the top of topics.

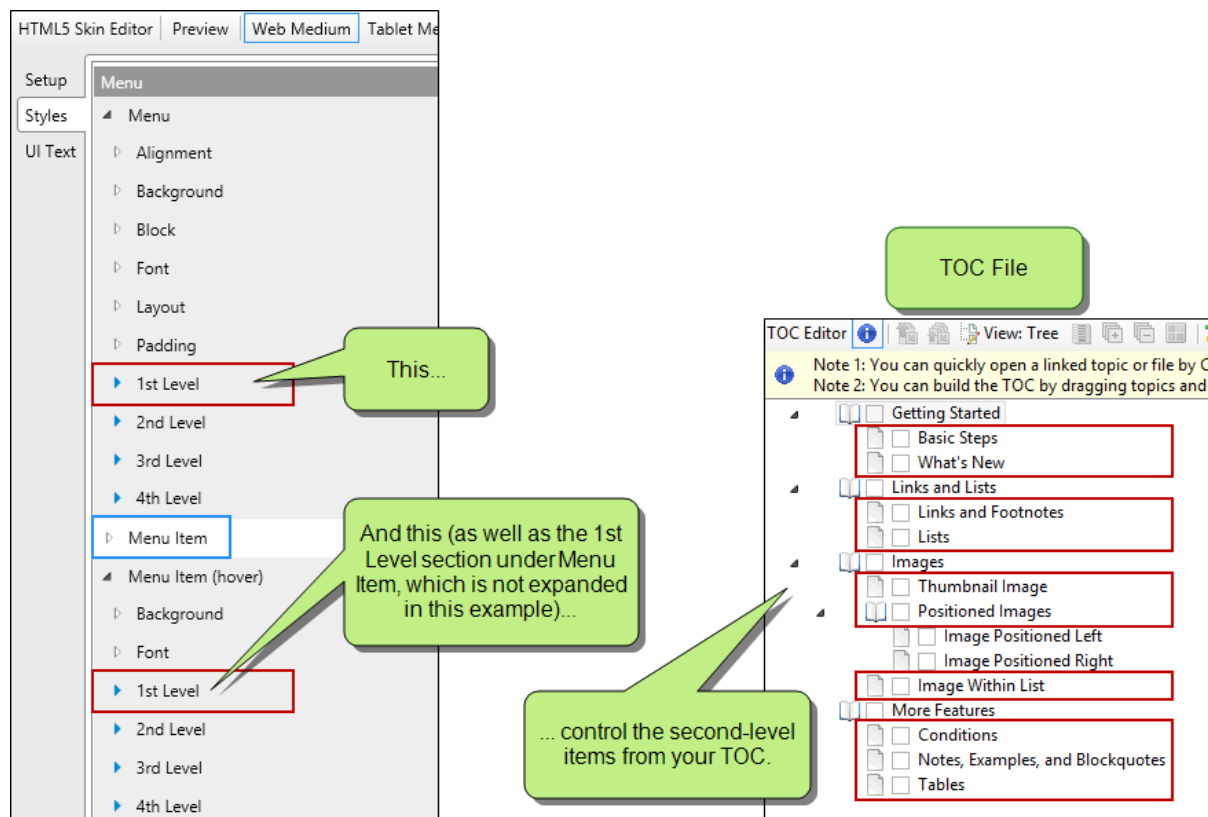
In the Menu section, you can specify settings for the entire menu and submenu areas. In the Menu Item and Menu Item (hover) sections, you can specify settings for the individual items within menus and submenus.



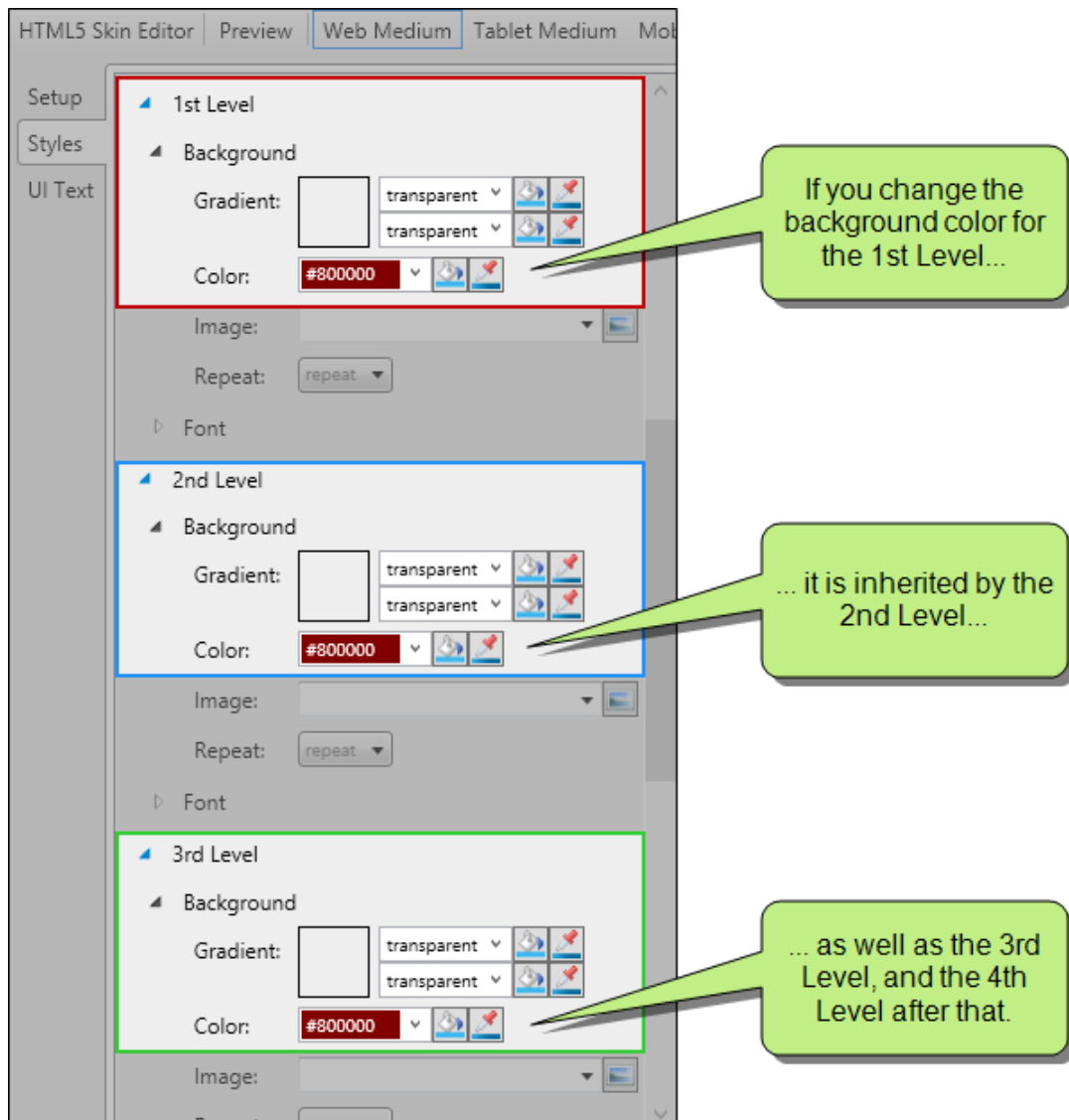
If you place settings on the properties under the first node—Menu, Menu Item, or Menu Item (hover)—they are applied to the root (top) level of the menu, which is the same as the first-level items in your TOC file.



If you place settings on the properties under the second node—1st Level—they are applied to the first sub-menu under the top menu, which is the second-level items in your TOC file. Therefore, the 2nd Level is the same as the third-level items in your TOC file, the 3rd Level is the same as the fourth-level items in your TOC file, and the 4th Level is the same as the fifth-level items in your TOC file.

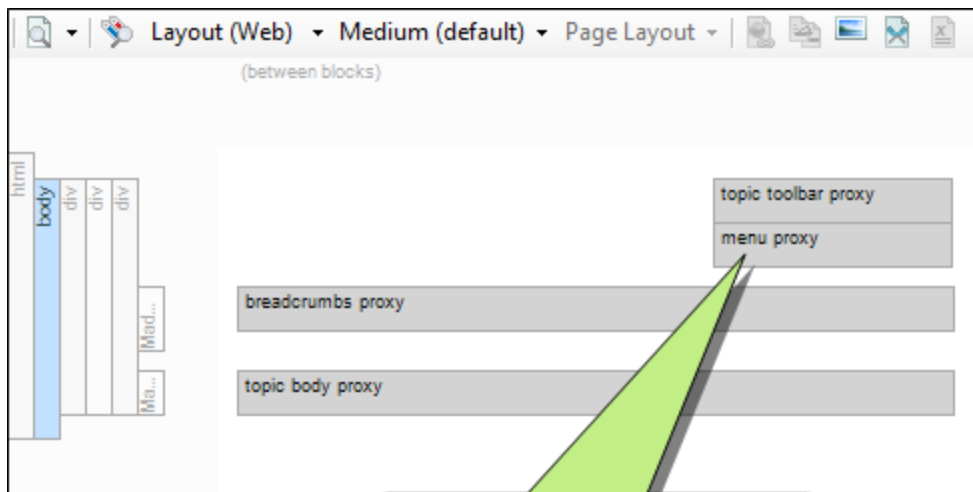


If you place settings on the properties under the second node—1st Level—the 2nd, 3rd, and 4th Levels inherit them. If you set values under the 2nd Level, the 3rd and 4th Levels inherit them, and so on.

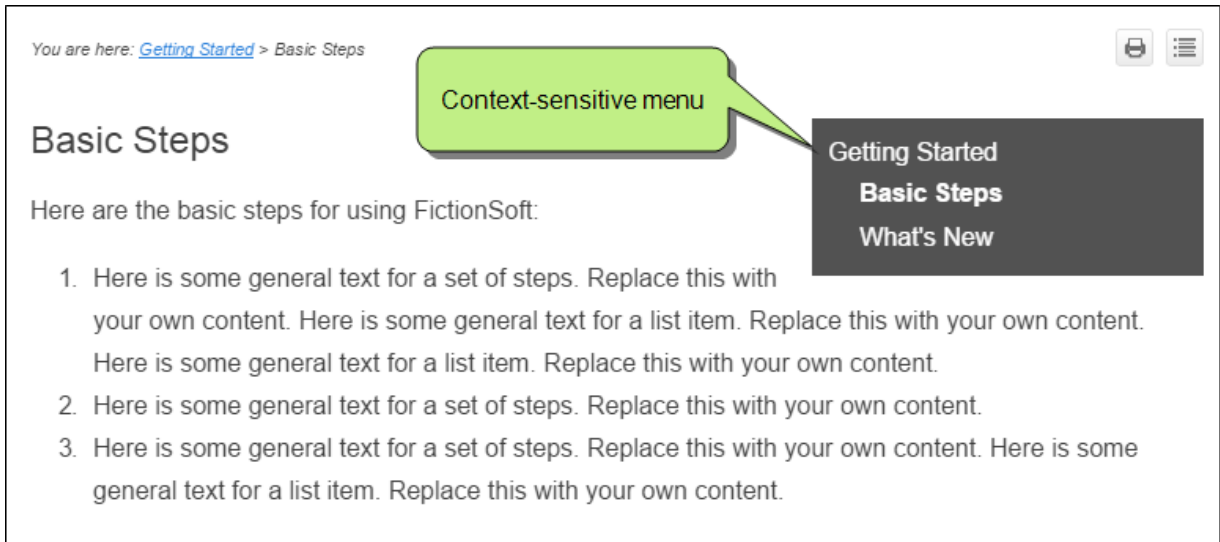


MENU SKIN COMPONENTS

A Menu skin component can work in tandem with a Menu proxy that you insert into a topic or master page.



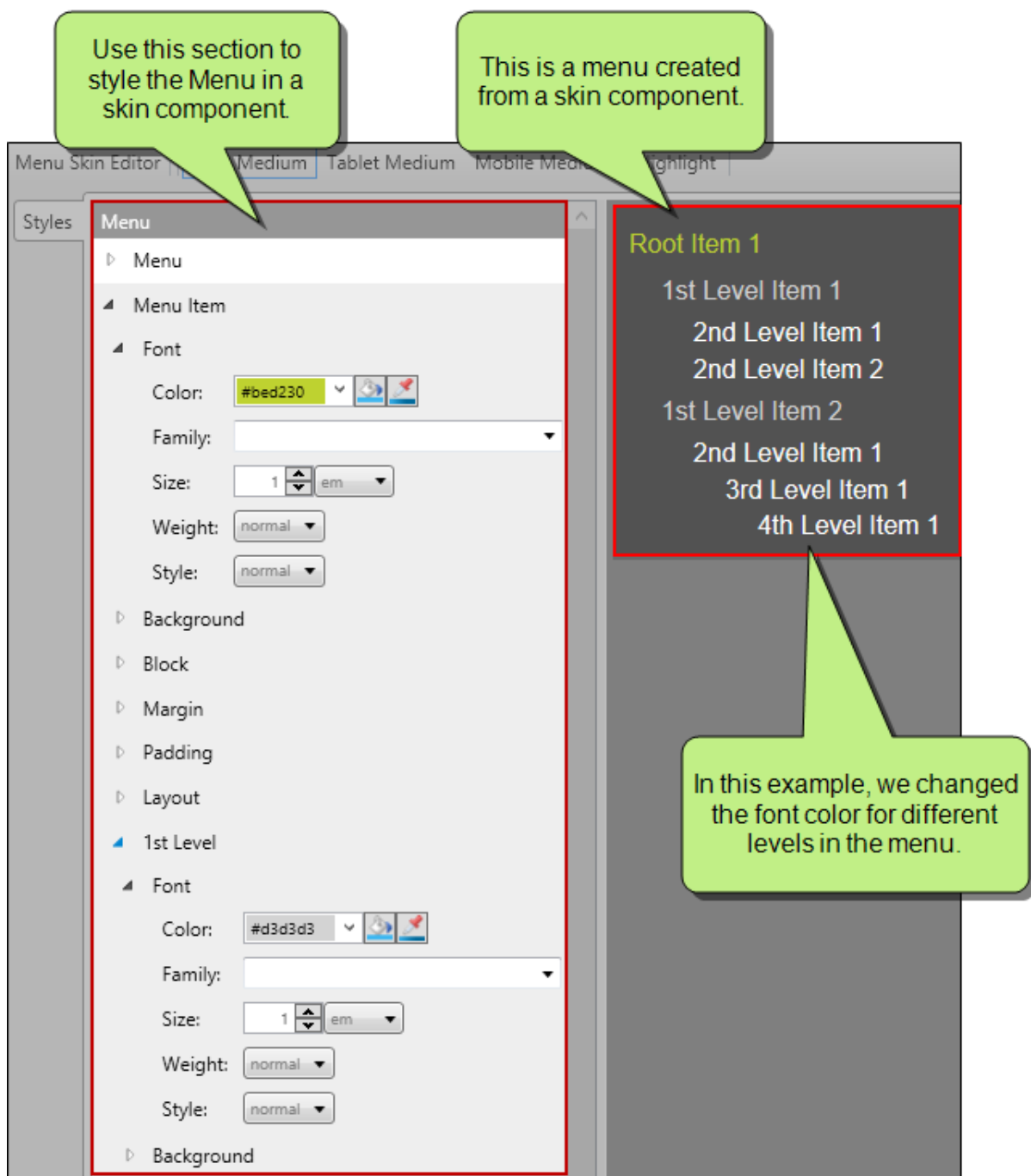
This Menu proxy is set to be context-sensitive. It has been styled to display on the right side of topics with content wrapping under it.



The styles for a Menu skin component are quite similar to those for the full Top Navigation skin. However, there are some differences and some options that are unique to each.

The primary thing to remember about a Menu component is that it can be context-sensitive, meaning it refers only to the topic that is open and, depending on your settings, the parent, sibling, and child items from the TOC as well. On the other hand, the menu styles in the full Top Navigation skin are always concerned with your TOC file structure as a whole.

That being said, the menu item style levels in a Menu component can mean something different than they do for the full Top Navigation skin.



When you set values directly under the Menu Item or Menu Item (hover) node, you are controlling the look for the menu items that are at the top level of the menu. If you insert a context-sensitive Menu proxy that includes the parent, the root Menu Item style refers to the parent topic.

In this example, we set the font color to green directly under the Menu Item node.

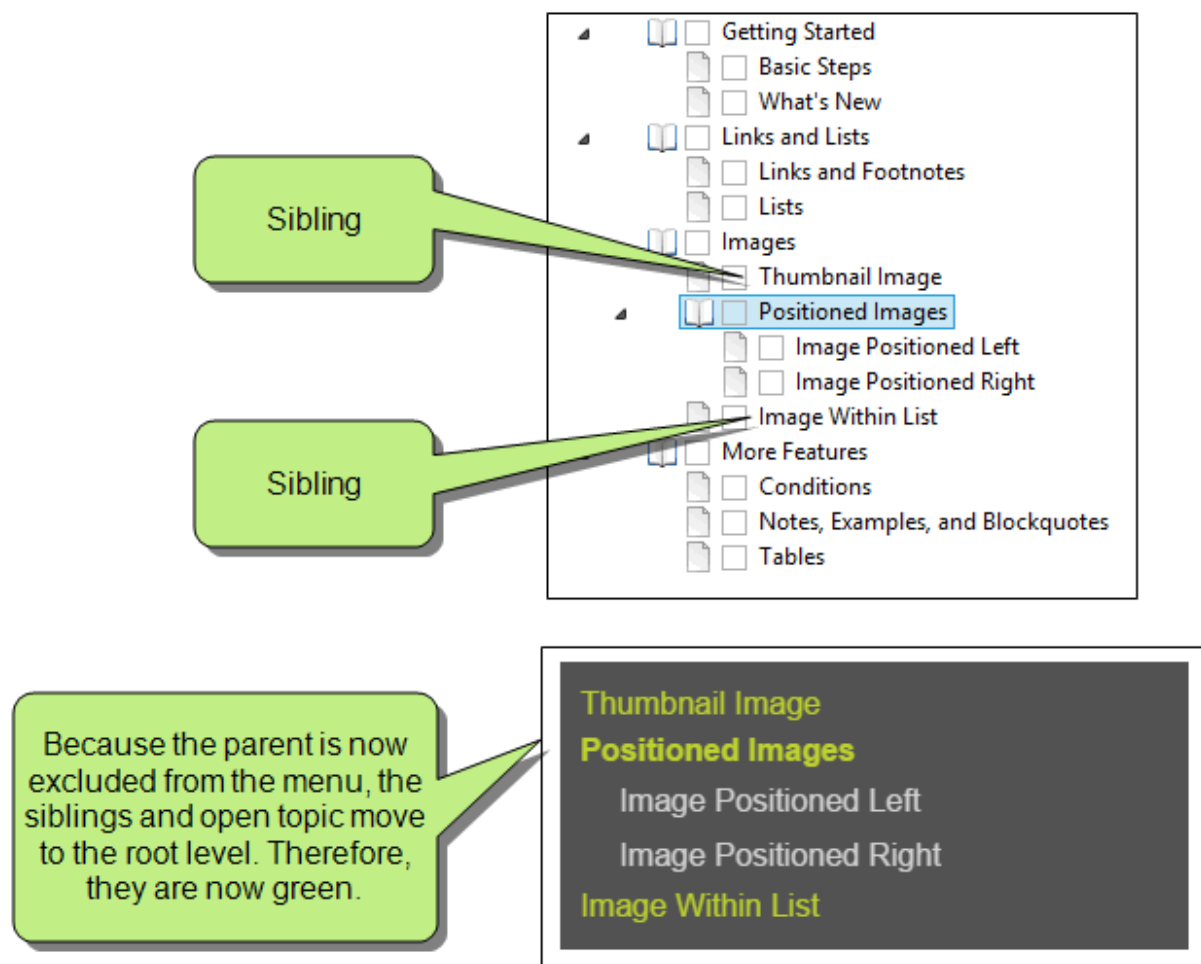
This is our TOC structure. Let's say "Positioned Images" is the topic that is open in the output.

Parent topic.

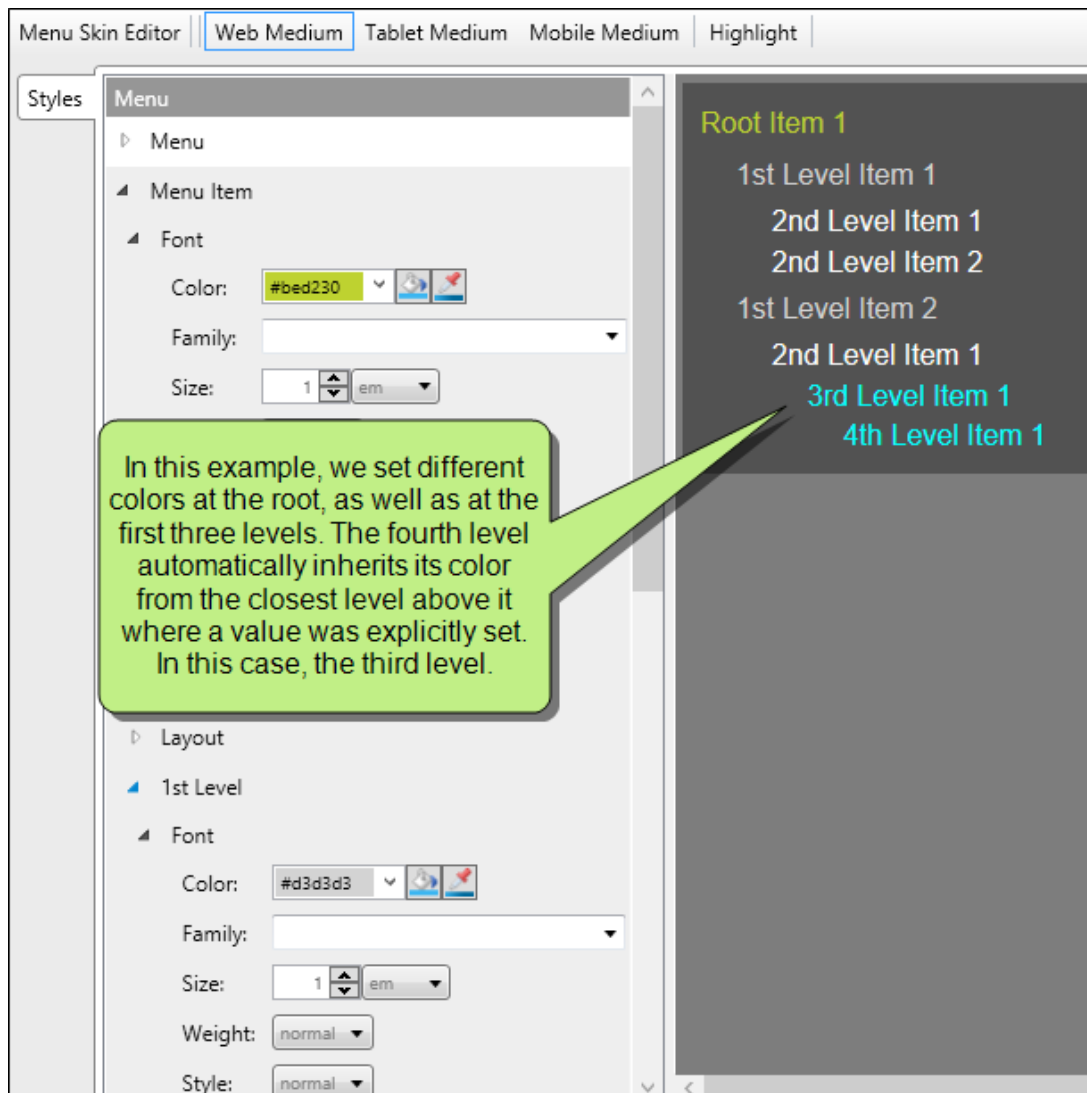
The parent of "Positioned Images" is "Images." Therefore, in our context-sensitive menu, it is displayed in a green font.

Images
Thumbnail Image
Positioned Images
Image Positioned Left
Image Positioned Right
Image Within List

If you don't include the parent, but you do include the siblings, the root Menu Item style refers to those sibling topics in the TOC.



The 1st Level style refers to the first level under the root, followed by 2nd Level, 3rd Level, and 4th Level. The same kind of inheritance used for menu items in the full Top Navigation skin is used for a Menu component.



There is also a style called "Selected" for Menu components. This lets you apply a unique look to the menu item that refers to the topic that is currently open.

The image shows a web editor interface. On the left, a style configuration panel for the 'Selected' style is highlighted with a red border. It includes settings for Font Color (set to #ffffff), Font Size (set to 1.2em), Weight (bold), and Style (normal). A green callout bubble points to the color and size settings, stating: 'In this example, we set a white font on the Selected style. We also increased its size.'

Below the configuration panel, a preview of a web page titled 'Positioned Images' is shown. The page content includes a breadcrumb 'You are here: [Images](#) > Positioned Images', a heading 'Positioned Images', and a paragraph 'Here is some general text for a topic. Replace this with your own content.' followed by two cross-reference links: '» Here is a cross-reference link: See [Image Positioned Left](#).' and '» Here is a cross-reference link: See [Image Positioned Right](#).' A green callout bubble points to the page content, stating: 'The menu item referring to the open topic is now white and larger than the other links.'

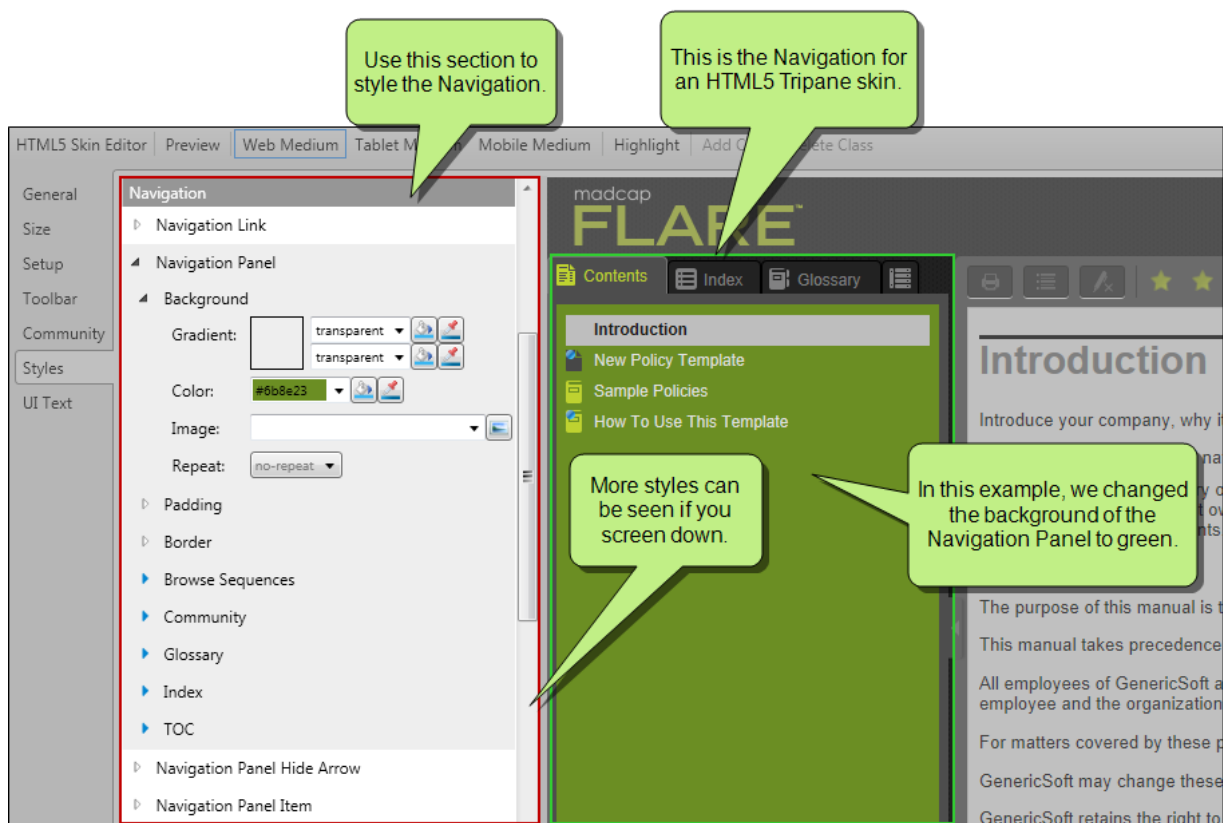
On the right side of the preview, a dark gray sidebar contains a 'Thumbnail Image' and a list of links: 'Positioned Images', 'Image Positioned Left', 'Image Positioned Right', and 'Image Within List'.

Navigation—Tripane and Top Navigation Skins

For Tripane and Top Navigation skins, this section is used to control the look of navigation elements, which allow users to open different parts of your output. See "Navigation in HTML5 Skins" on page 222.

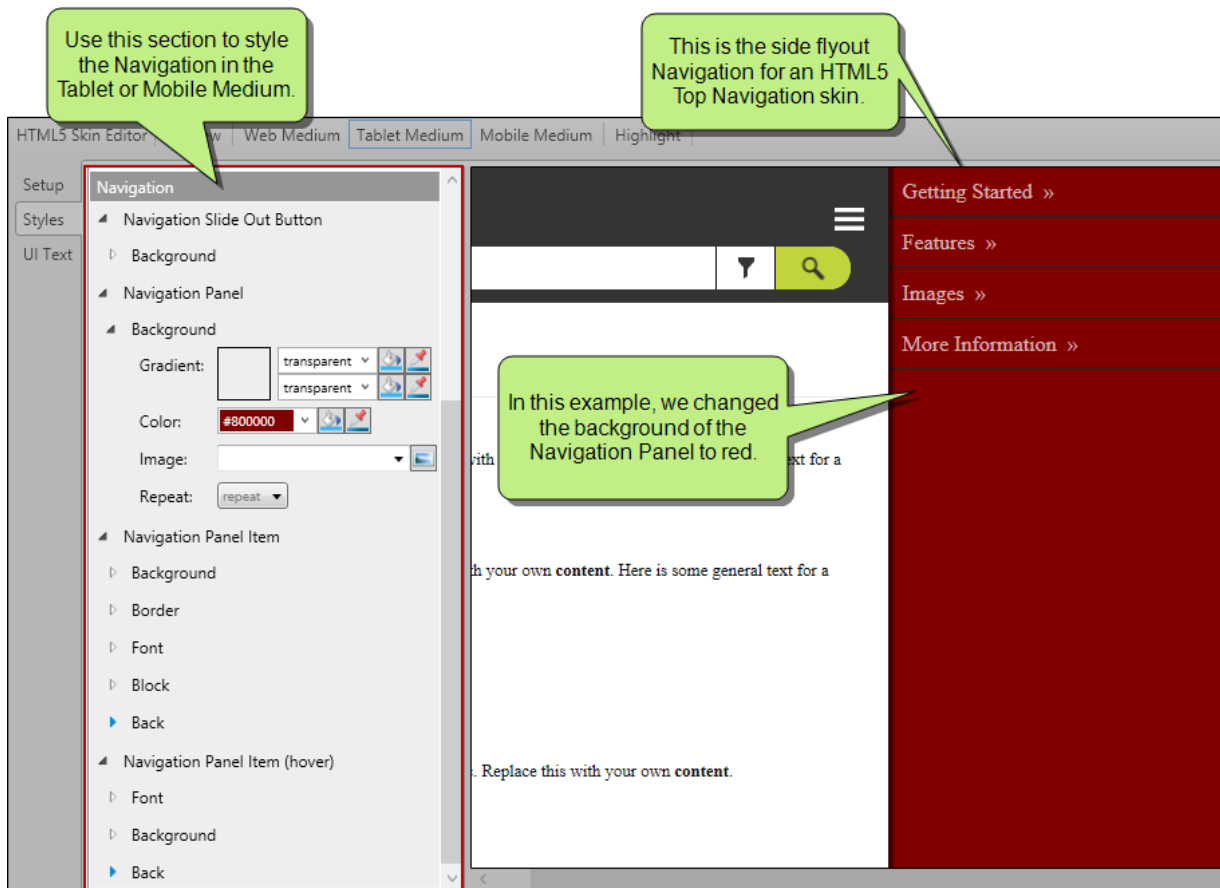
TRIPANE SKINS

For Tripane skins, this section is available in all three mediums, although there are different options available for the Tablet and Mobile mediums.



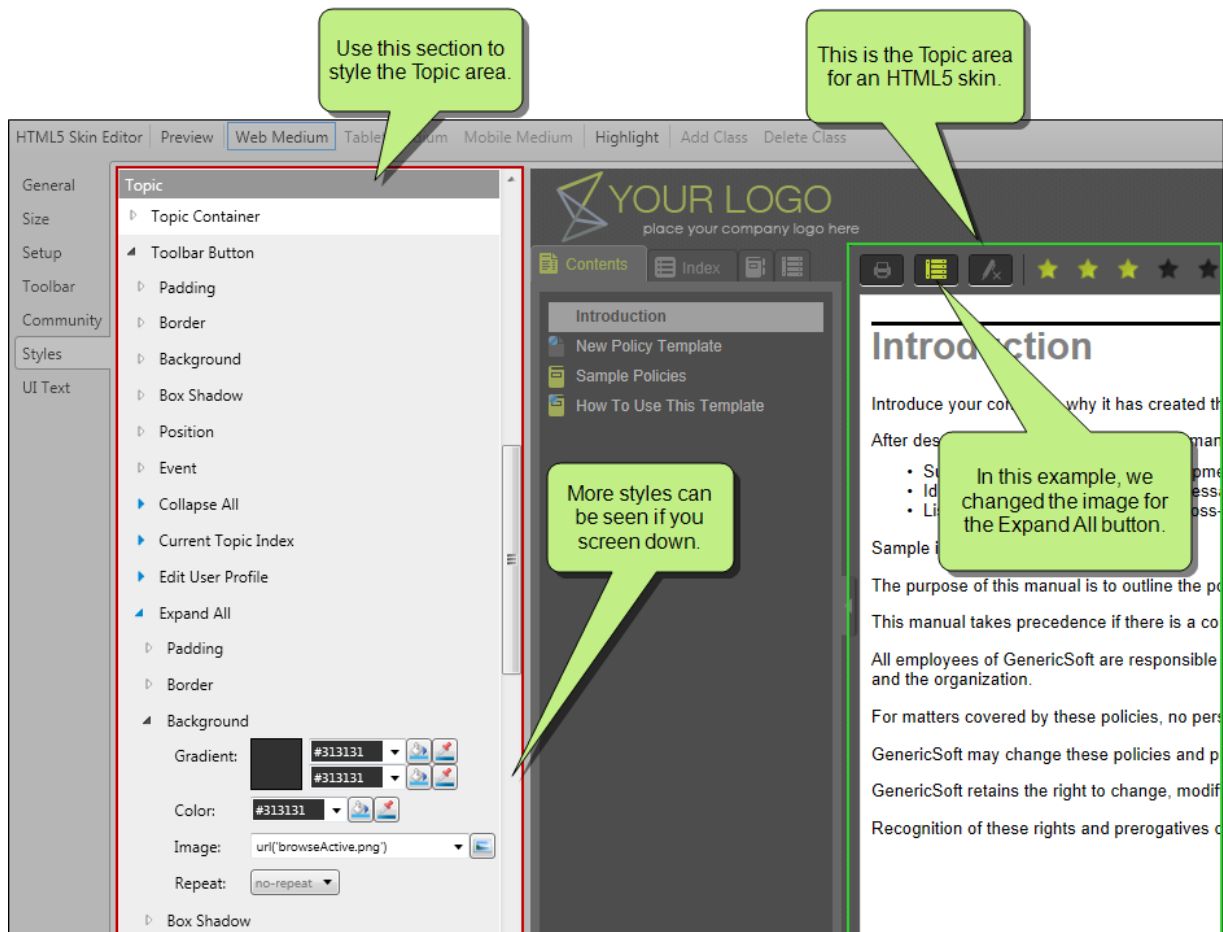
TOP NAVIGATION SKINS

For Top Navigation skins, this section is used to control the look of the side flyout navigation pane when being displayed on a tablet or mobile device. Therefore, it is not shown if you are working in Web Medium view in the Skin Editor, but rather only in the Tablet and Mobile mediums.



Topic—Tripane Skins

For Tripane skins, this section is used to control the look of the topic area of the output. This includes the container holding the topic content. It also includes the toolbar just above the topic content. See "Topics in HTML5 Skins" on page 237.

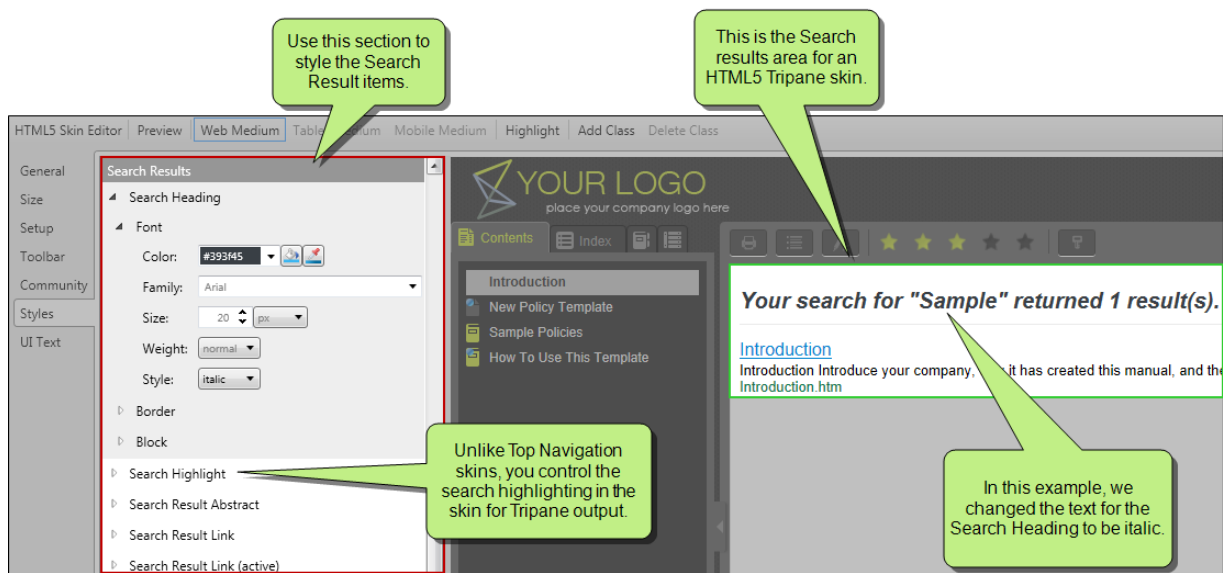


Search Results—Tripane Skins, Top Navigation Skins, and Search Results Components

You can use a Tripane or Top Navigation skin to change the appearance of search result items. You can also do this with a smaller Search Results skin component. See "Search Results in HTML5 Skins" on page 246.

TRIPANE SKINS

For Tripane skins, you can control the look of search result elements such as the heading, search highlighting, abstract (i.e., descriptive text of the file), links, glossary terms, and pagination.



TOP NAVIGATION SKINS

The search results options available for Tripane skins are similar for Top Navigation skins. The biggest difference is that search highlight settings are controlled in the skin only for Tripane output. For Top Navigation output, the search highlighting is controlled in your regular stylesheet by modifying special classes under the span style.

The screenshot displays the HTML5 Skin Editor interface. On the left, the 'Setup' panel is open to the 'Search Results' section. The 'Search Heading' is selected, and the 'Font' settings are visible: Color is set to #800000, Family is empty, Size is 1.5, Weight is normal, and Style is normal. The 'Search Results' list on the left includes: Search Results, Search Result Abstract, Search Result Link, Search Result Link (active), Search Result Link (focus), and Search Result Link (hover). On the right, the preview shows a search results page with a dark header, a search heading in red, and search results for 'content'. The search heading text is 'Your search for "content" returned 14 result(s)'. The search results list includes 'Tables' and 'Lists'. The 'Tables' section has a description and a link to 'B_Features/Conditions Notes Tables/Tables.htm'. The 'Lists' section has a description and a link to 'B_Features/Images/ImagePositionedLeft.htm'. The preview also shows a company logo and a search bar.

Use this section to style the Search Result items.

Many of the same options are available for Top Navigation skins that are available for Tripane skins. However, there are some minor differences in the styles and properties that are available.

This is the Search results area for an HTML5 Top Navigation skin.

Notice that Top Navigation skins do not include an option for search highlighting. That's because search highlighting is controlled in a regular stylesheet for Top Navigation output.

In this example, we changed the text for the Search Heading to be red.

SEARCH RESULTS SKIN COMPONENTS

Alternatively, you can use the Search Results skin component and its related proxy to design a custom container to display search results.

The options in the skin component are the same as those in the Top Navigation skin. Chances are good that you will never need to use a Search Results skin component, but it is available in case you want the flexibility of having another page showing search results.

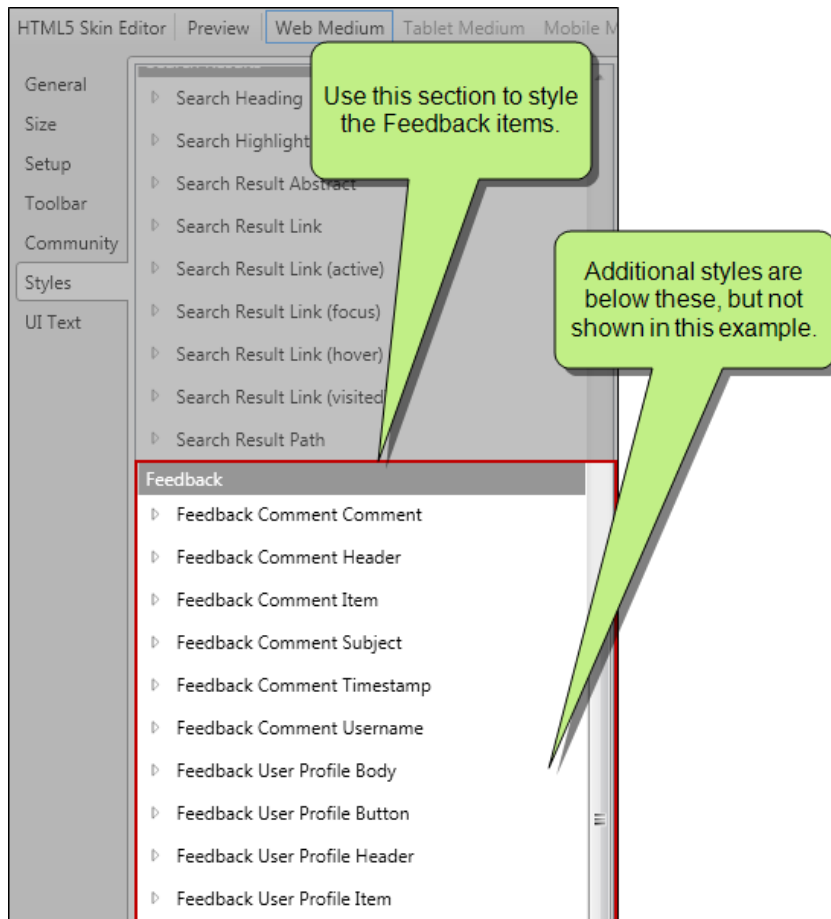
The Search Results skin component is not available if you are generating Tripane output. It is only available when you are producing Top Navigation output, or HTML5 output without a skin selected.



Note: If you want to style the search bar and its elements, you can use the Header section in the Skin Editor, or you can use a Search Bar skin component.

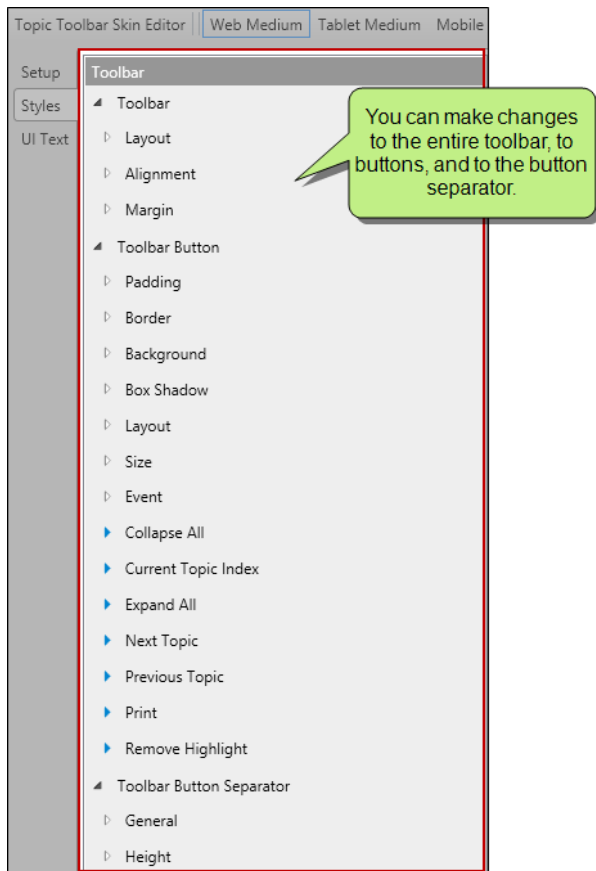
Feedback—Tripane Skins

For Tripane skins, these are styles for elements that are displayed if you integrate your output with MadCap Feedback. See "Feedback in HTML5 Skins" on page 262.



Topic Toolbar Skin Components

In Topic Toolbar skin components, you can use skin styles used to control the look of a toolbar you add via a proxy. See "Topic Toolbars in HTML5 Skin Components" on page 270.

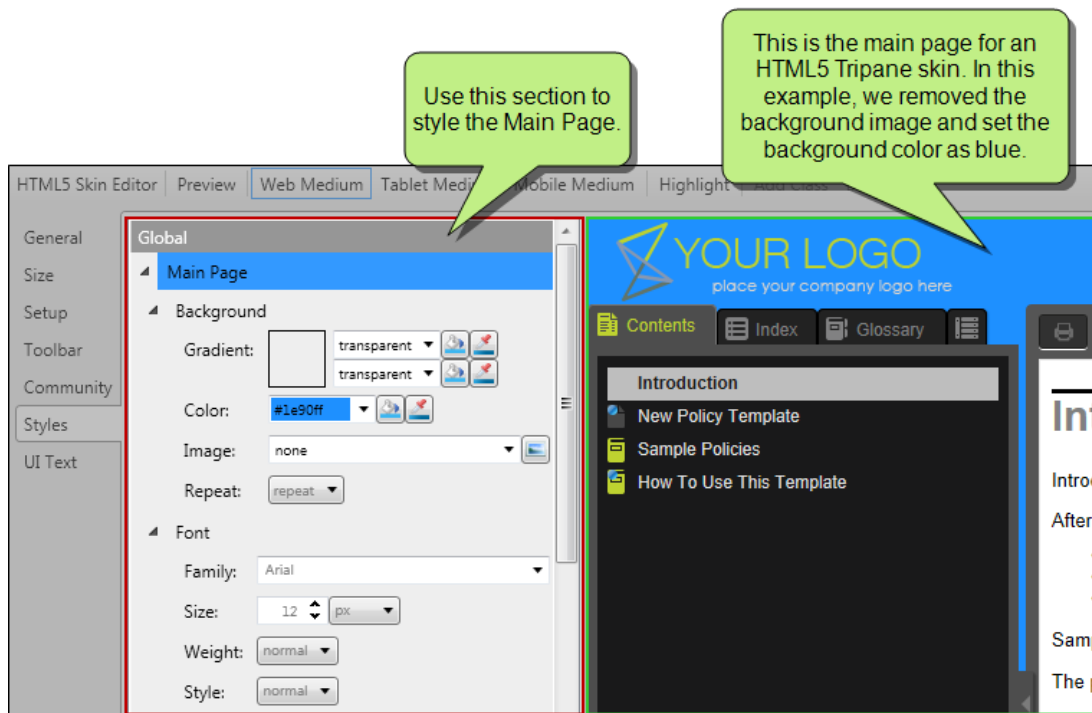


Note: If you want to change labels for some of these styles, or if you want to display the output user interface in a particular language, you can use the UI Text tab in the Skin Editor.

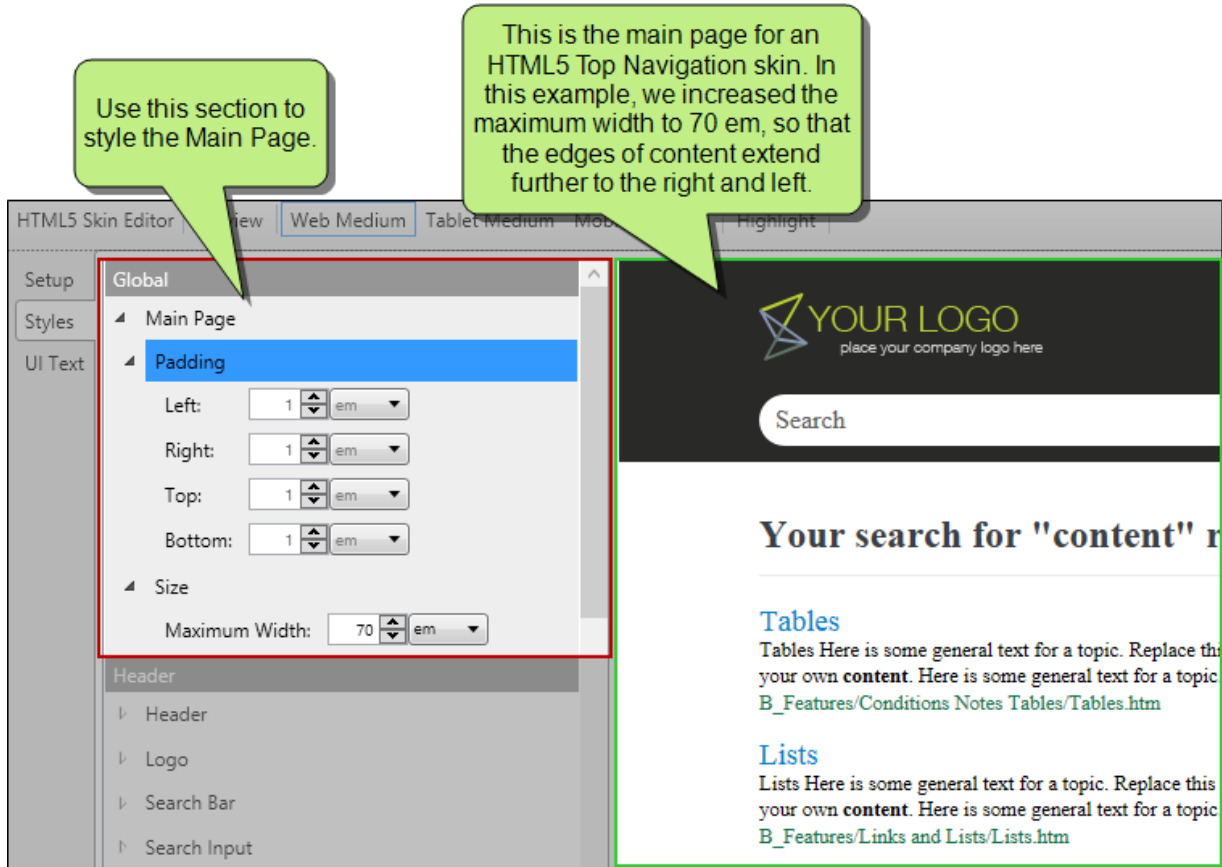
Main Page in HTML5 Skins

This section contains the Main Page style for Tripane and Top Navigation skins.

For Tripane skins, you can control the font and background color for the page.



For Top Navigation skins, you can control the padding and maximum width for the page.



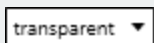


HOW TO SPECIFY STYLE SETTINGS FOR THE MAIN PAGE IN HTML5


1. Open an HTML5 Tripane or Top Navigation skin.
2. (Optional) If you are using responsive output, make sure you select the appropriate medium—Web, Tablet, or Mobile—in the local toolbar before making changes to styles. If the skin has not been enabled for responsive output, you can make changes only for the Web medium. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.
3. Select the **Styles** tab.
4. On the left side of the editor, in the **Global** section, expand the **Main Page** style.
5. Expand any of the nodes and complete the necessary fields.

BACKGROUND (TRIPANE SKINS ONLY)

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.



Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

FONT (TRIPANE SKINS ONLY)

You can change any of the following to affect the font:

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

PADDING (TOP NAVIGATION SKINS ONLY)

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

SIZE (TOP NAVIGATION SKINS ONLY)

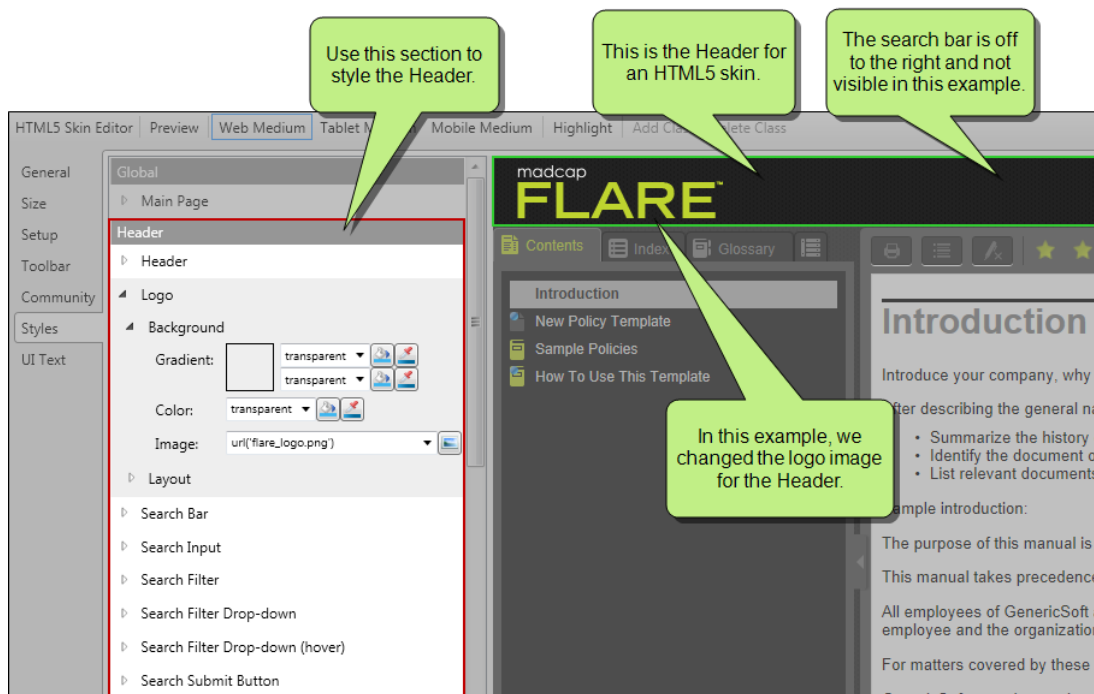
You can change the maximum width of the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

6. Click  to save your work.

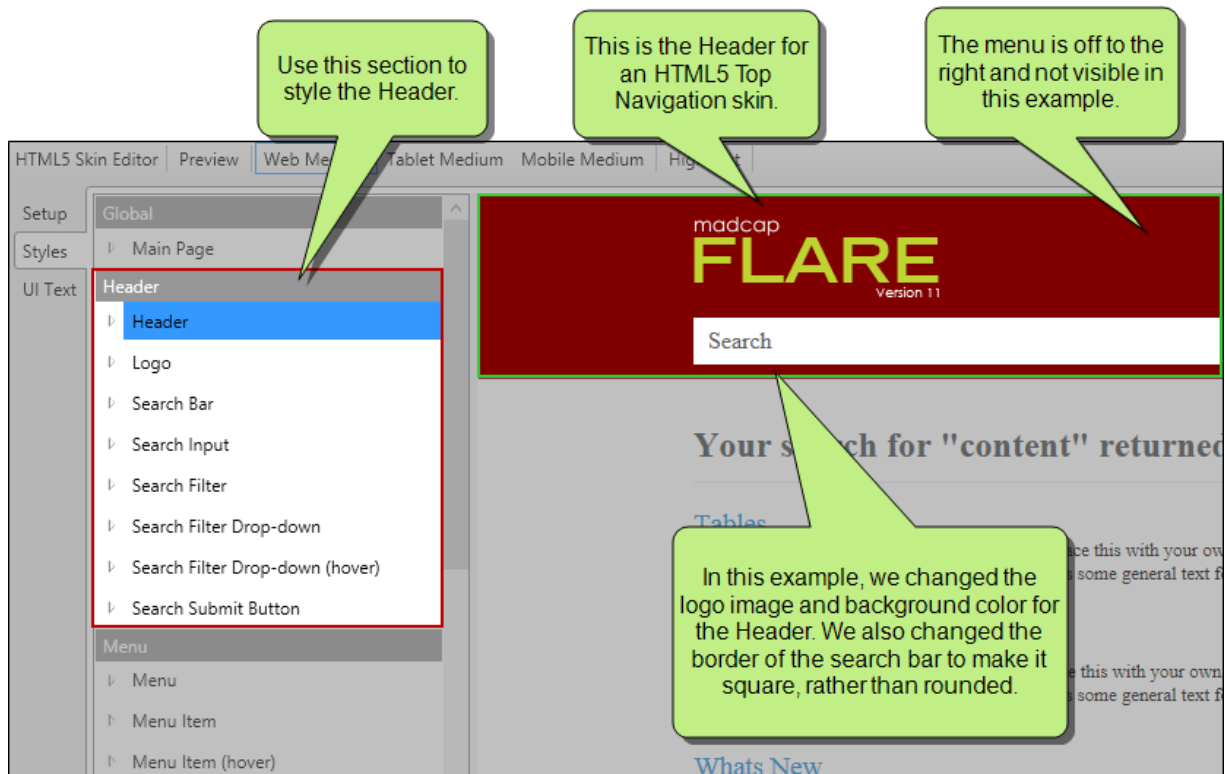
Headers in HTML5 Skins

This is the area at the top of the screen for Tripane and Top Navigation skins.

For Tripane skins, this area displays a logo and the search bar.



For Top Navigation skins, this area displays a logo, the search bar, and the top menu.



HOW TO SPECIFY STYLE SETTINGS FOR HEADERS IN HTML5

1. Open an HTML5 skin.
2. (Optional) If you are using responsive output, make sure you select the appropriate medium—Web, Tablet, or Mobile—in the local toolbar before making changes to styles. If the skin has not been enabled for responsive output, you can make changes only for the Web medium. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.
3. Select the **Styles** tab.
4. On the left side of the editor, in the **Header** section, expand any of the following styles:

Style	Description
Header	This controls the entire header area.
Logo	This controls the logo in the header.
Search Bar	This controls the search bar in the header.
Search Input	This controls the text used when users perform searches.
Search Filter	This controls the look of the search filter area, if you have included one.
Search Filter Drop-down	This controls the drop-down portion of the search filter.
Search Filter Drop-down (hover)	This lets you change how the search filter drop-down looks when users hover over it.
Search Submit Button	This controls the look of the button used to initiate searches.

5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Also, you might see nodes for a Tripane skin that you do not see for a Top Navigation skin, and vice versa. Following is a general list in alphabetical order:

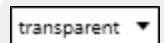


ALIGNMENT


You can click in the **Horizontal** field and choose whether to position the element to the **left**, **middle**, or **right**. You can also click in the **Vertical** field and choose whether to position the element on the **top**, **middle**, or **bottom**.

BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.


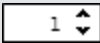

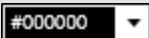




Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.

	You can select a type of line for the border (e.g., solid, dashed, dotted).
	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.




- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field

type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

BOX SHADOW

You can change any of the following to affect the box shadow on the element:

- » **Horizontal Shadow Length** This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Vertical Shadow Length** This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Blur Length** This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Color** This lets you select a color for the shadow. You can use any of the following:




	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Shadow Insert** This changes the shadow from an outer shadow (outset) to an inner shadow. Click in the field and select an option. If you select **false** the shadow will be outset. If you select **true** the shadow will be inset.

FONT

You can change any of the following to affect the font:

- » **Color** This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

HEIGHT

You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

LAYOUT

You can click in the **Display** field and choose whether to show the element (block) or not (none).

MARGIN

You can change the margin for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

SIZE

For some properties, you can change the height or width of the element. For other properties, you can change the maximum width of the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

WIDTH

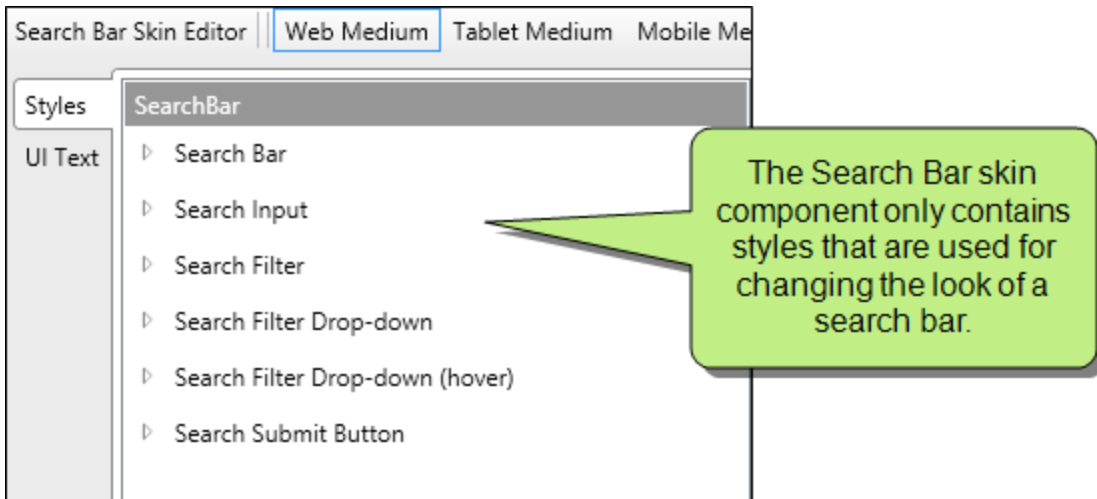
You enter a width for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

6. Click  to save your work.

SEARCH BAR SKIN COMPONENTS

One of the tasks that the Header area in Tripane and Top Navigation skins lets you perform is modifying the look of the search bar at the top of pages. Alternatively, you can use a Search Bar proxy to add a custom search bar elsewhere, and you can use a Search Bar skin component to control the look for it.

The options in the skin component are the same as the search bar options in the Top Navigation and Tripane skins.



The Search Results skin component is not available if you are generating Tripane output. It is only available when you are producing Top Navigation output, or HTML5 output without a skin selected.



Note: For HTML5 logos, the following image types are supported: PNG, BMP, JPEG, TIF, TIFF, and GIF.



Note: If you want to style items for search results, you can use the Search section in the Skin Editor. See "Search Results in HTML5 Skins" on page 246.

Menus in HTML5 Skins

Using skin styles is one way to control the look of the menu that is included with a Top Navigation skin, as well as additional menus that you add through the Menu proxy.

HOW TO SPECIFY STYLE SETTINGS FOR MENUS IN HTML5

1. Open an HTML5 Top Navigation skin or a Menu skin component.
2. If you opened a skin component, make sure you select the appropriate medium—Web, Tablet, or Mobile—in the local toolbar before making changes to styles. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.

If you opened a full Top Navigation skin, you must have Web Medium selected, because menu elements are not shown in Tablet or Mobile views. For those mediums, you would set navigation elements instead. See "Navigation in HTML5 Skins" on page 222.

3. Select the **Styles** tab.
4. On the left side of the editor, in the **Menus** section, expand any of the following styles:

Style	Description
Menu	This controls the look of the entire menu and submenu areas.
Menu Item	This controls the look of individual items within a menu.
Menu Item (hover)	This controls the look of individual items within a menu when a user hovers over them.

5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Also, you might see nodes in the full skin that you do not see in a skin component, and vice versa. Following is a general list in alphabetical order:




ALIGNMENT


You can click in the **Horizontal** field and choose whether to position to the **right** or **left** side of the display.

BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.



Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BLOCK

This is the container (or "block") holding content in an element. You can change any of the following to affect the block:

- » **Line Height** This is how tall the container is that holds the content. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Text Align** This changes the alignment of the content. Click in the field and select an option (center, justify, left, right).
- » **Text Indent** This is how far the text is moved inward. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Word Wrap** You can click in the **Word Wrap** field and choose whether to wrap text for the element (normal) or not (nowrap).

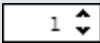
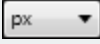
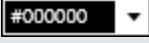


BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.

solid ▼

You can select a type of line for the border (e.g., solid, dashed, dotted).




	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

FONT

You can change any of the following to affect the font:

- » **Color** This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

LAYOUT

You can click in the **Display** field and choose whether to show the element (block) or not (none).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

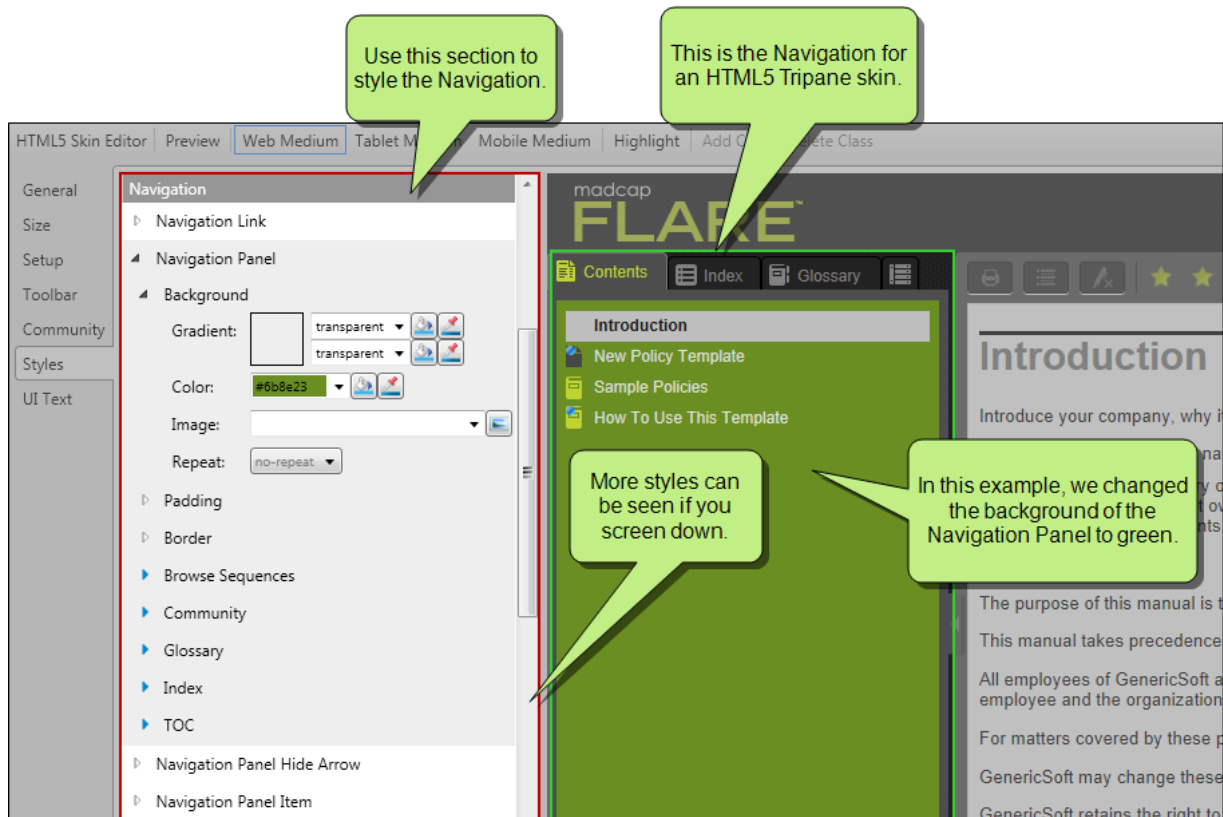
SIZE

You can change the height or width of the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

6. Click  to save your work.

Navigation in HTML5 Skins




This section is used to control the look of navigation elements in the skin, which allow users to open different parts of your output.




HOW TO SPECIFY STYLE SETTINGS FOR NAVIGATION ELEMENTS IN HTML5

1. Open an HTML5 skin.
2. (Optional) If you are using responsive output, make sure you select the appropriate medium—Web, Tablet, or Mobile—in the local toolbar before making changes to styles. If the skin has not been enabled for responsive output, you can make changes only for the Web medium. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.
3. Select the **Styles** tab.
4. On the left side of the editor, in the **Navigation** section, expand any of the following styles:

Style	Description
Home Button	This controls the button that users see in the slide out in Tablet and Mobile views.
Navigation Link	<p>These are links that users may see at the top or bottom of topics if they open those topics as standalone files. In order to add these links to the output, you must enable them on the Setup tab of the Skin Editor. See and "Showing Navigation Links in Standalone Topics" on page 281.</p> <p>This style has properties both at the root and on children nodes. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p>
Navigation Slide Out Button	This controls the button that people use in Tablet and Mobile views to access the slide out navigation.

Style	Description
Navigation Panel	<p>This controls the panel area that contains the following elements: Browse Sequence, Community, Glossary, Index, Search Filters, and TOC.</p> <p>This style has properties both at the root and on children nodes. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p> <div>  Note: The TOC node under the Navigation Panel style lets you change icons in <i>general</i> for all books/topics in a TOC. If instead you want to change the book or topic icons only for <i>specific</i> entries in your TOC, use the TOC Entry style instead, creating classes for it. </div> <div>  Note: The Community element is available only in the Web medium. </div> <div>  Note: The Search Filters element is available only in the Tablet and Mobile mediums. </div>
Navigation Panel Hide Arrow	This controls the look of the arrow button that is shown on the divider between the navigation elements and content in Web view. Users can click this button to show or hide the navigation elements.
Navigation Panel Item	This controls the look of text in navigation panel elements.

Style	Description
Navigation Panel Item (hover)	This controls the look of text in navigation panel elements when a user hovers over it.
Navigation Panel Item (selected)	This controls the look of text in navigation panel elements when it is selected.
Navigation Panel Search Bar	This controls the look of the search field that users see in some navigation elements (e.g., Index, Glossary).
Navigation Tab	<p>This controls the look of the tab area at the top of each navigation element. For example, you might want to change the height or width, or you might want to change the tab layout to specify whether the tab should show an image then text, text then an image, text only, or an image only. If you want to change other characteristics, such as the color or the icon images themselves, you need to use either the Navigation Tab (active) or Navigation Tab (inactive) style.</p> <p>This style has properties both at the root and on children nodes. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p>
Navigation Tab (active)	<p>This controls the look of the tab area at the top of each navigation element when the tab is selected.</p> <p>This style has properties both at the root and on children nodes. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p> <p>To change the color of the navigation tab, including the strip around the edge of the panel, you can modify the background of the Navigation Tab (active) style.</p>

Style	Description
Navigation Tab (inactive)	<p>This controls the look of the tab area at the top of each navigation element when the tab is not selected.</p> <p>This style has properties both at the root and on children nodes. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p>
Toc Entry	<p>These are the images that users see for <i>specific</i> books or topics when they open your table of contents (TOC) in the output.</p> <p>Keep these important points in mind when you use the TOC Entry style.</p> <ul style="list-style-type: none"> » Specifying icons at the root of the TOC Entry style itself will do nothing. Instead, you must create <i>classes</i> of the TOC Entry style and select the icons for those classes. Steps for this are given below. » The purpose of this style is to choose book or topic icons only for specific entries in your TOC, not for all of them. For example, you might want most of your TOC books to be blue, but there is one that you want to be red. To make that red book, you would create a class on the TOC Entry style and select a red book image for it. If instead you want to change the icons in <i>general</i> for all books/topics in a TOC, expand the Navigation Panel style, and set the icons on the TOC node. <p>HOW TO CREATE A STYLE CLASS FOR TOC ENTRIES IN HTML5</p> <ol style="list-style-type: none"> 1. Select the Toc Entry style. 2. In the local toolbar, click Add Class. 3. Give the class a name (e.g., RedBook), and click OK. A new node with that name is added under the Toc Entry style. 4. Use the fields under the new node to choose images. 5. Click  to save your work.

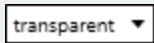


Style	Description
Index Highlight	This is the highlighting that is seen when a user begins typing in the Index search field.
Glossary Highlight	This is the highlighting that is seen when a user begins typing in the Glossary search field.
Glossary Item Definition	This controls the look of definition text in the Glossary pane. If you want to change the look of the glossary terms, you need to use the Navigation Panel Item style.


5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Following is a general list in alphabetical order:

BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.



Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BLOCK


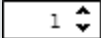



This is the container (or "block") holding content in an element. You can change any of the following to affect the block:

- » **Line Height** This is how tall the container is that holds the content. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Text Align** This changes the alignment of the content. Click in the field and select an option (center, justify, left, right).
- » **Word Wrap** You can click in the **Word Wrap** field and choose whether to wrap text for the element (normal) or not (nowrap).

BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.

	You can select a type of line for the border (e.g., solid, dashed, dotted).
	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.



This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

BOTTOM

You can change any of the following to affect the font for the navigation link when it appears at the bottom of a topic:

- » **Color** This lets you select a color for the font. You can use any of the following:



You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.



This opens the Color Picker dialog, which lets you choose a color in many ways.






This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

FONT

You can change any of the following to affect the font:




- » **Color** This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

GENERAL

You can change the color for the element.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

GLOSSARY

You can expand the **Layout** node, click in the **Display** field, and choose whether to show the search bar in the Glossary pane. If you want to show it, select **block**. If you do not want to show it, select **none**.

HEIGHT

You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

ICON

You can click  and select a different image to be used for the element.

ICONS

You can click  and select a different image for any of the following TOC entry icons:

- » **Closed Book Icon** You can select a different icon image for the book items in the TOC when they are closed.
- » **Open Book Icon** You can select a different icon image for the book items in the TOC when they are open (or selected).
- » **Topic Icon** You can select a different icon image for the topic items in the TOC.
- » **Mark As New Icon** You can select a different icon image for the topic items in the TOC when they are marked as new.



Important: If you are setting these under the Toc Entry style (instead of the Navigation Panel style), you must first create a class of the Toc Entry style and then choose the images for that class. *Selecting images in the root fields under Toc Entry will do nothing.* To create a class, select **Toc Entry** and in the local toolbar click **Add Class**. You will also need to open your TOC, right-click on an entry, select **Properties**, and on the **General** tab choose the **Style Class** that you created.

INDEX

You can expand the **Layout** node, click in the **Display** field, and choose whether to show the search bar in the Index pane. If you want to show it, select **block**. If you do not want to show it, select **none**.

LAYOUT

You can click in the **Display** field and choose whether to show the element (block) or not (none).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

POSITION

You can change the position of the element, meaning its width and height:

- » **Width** You enter a width for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Height** You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

TAB LAYOUT

You can change what appears in the navigation tabs and their order. Click the **Layout** field and select one of the layout options (TextOnly, ImageOnly, TextThenImage, ImageThenText).

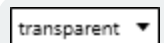




Note: One of the benefits of having image icons on the navigation tabs is that it lets users drag the pane to reduce its width more than they could otherwise. The text disappears, and tooltips on the image icons let you know what a particular tab contains.

TEXT SHADOW

You can change any of the following to affect the text shadow on the element:




- » **Shadow Visibility** You can click in this field and choose to make the shadow either hidden or visible.
- » **Horizontal Shadow Length** This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Vertical Shadow Length** This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Blur Length** This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Color** This lets you select a color for the shadow. You can use any of the following:


	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

TOP

You can change any of the following to affect the font for the navigation link when it appears at the top of a topic:

- » **Color** This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
 - » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
 - » **Weight** You can click in this field and make the font normal or bold.
 - » **Style** You can change click in this field and make the font normal or italic.
6. Click  to save your work.



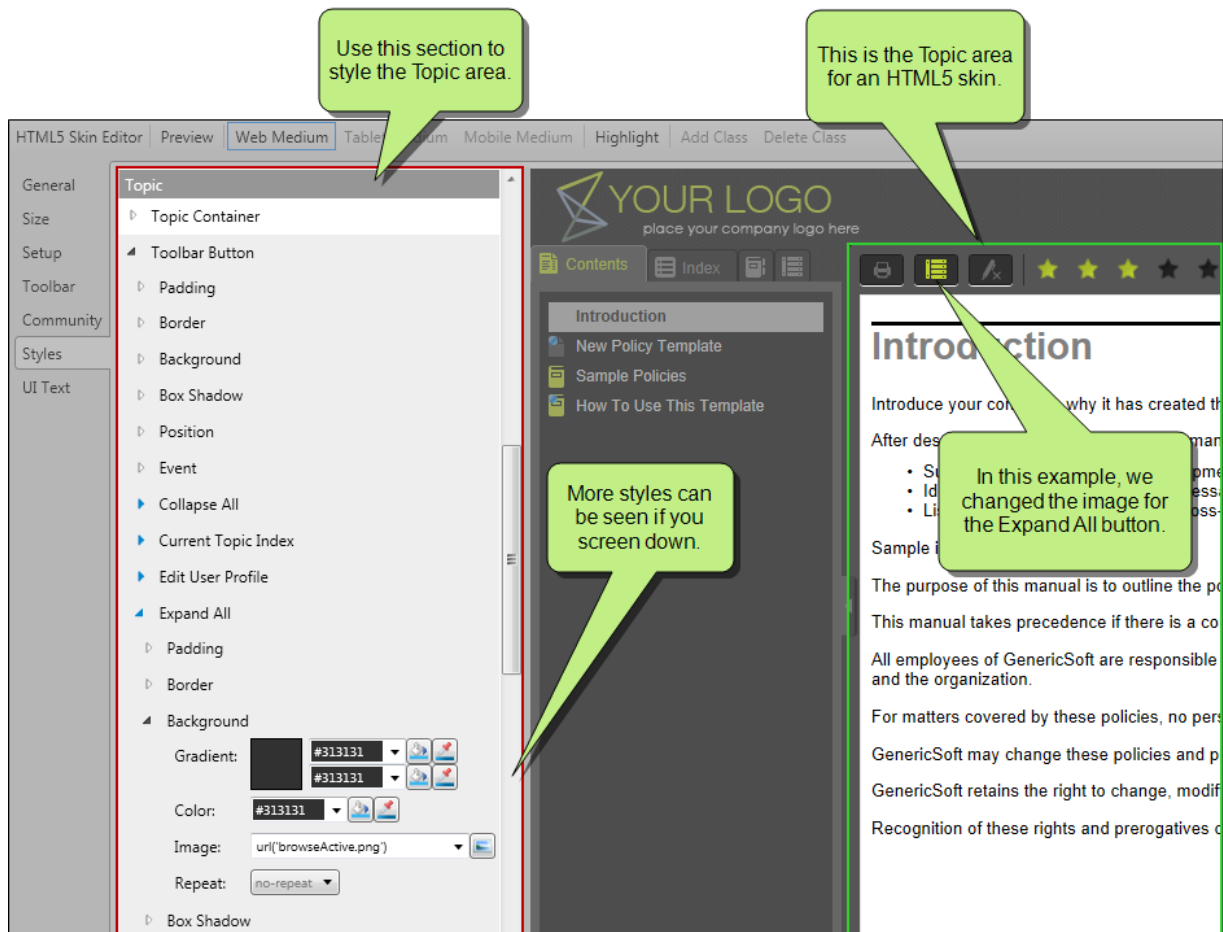
Note: In order to see navigation items in the output, you must make sure they are added to the skin. For more information see the online Help.



Note: You can also use the UI Text tab in the Skin Editor to change labels for these styles, including default values.

Topics in HTML5 Skins

This section is used to control the look of the topic area of the output. This includes the container holding the topic content. It also includes the toolbar just above the topic content.



HOW TO SPECIFY STYLE SETTINGS FOR THE TOPIC AREA IN HTML5

1. Open an HTML5 skin.
2. (Optional) If you are using responsive output, make sure you select the appropriate medium—Web, Tablet, or Mobile—in the local toolbar before making changes to styles. If the skin has not been enabled for responsive output, you can make changes only for the Web medium. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.
3. Select the **Styles** tab.
4. On the left side of the editor, in the **Topic** section, expand any of the following styles:

Style	Description
Topic Container	This is the container holding topics in the output.

Style	Description
Toolbar Button	<p>These are the buttons and other elements that are part of toolbars that you can add to output.</p> <p>The Toolbar Button style has properties both at the root and on children nodes. If you want the setting(s) to be applied to all of the items (except the separator), select the Toolbar Button style itself. Otherwise, expand any of the child styles to apply settings to them individually. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p> <p>Following are the child styles that control individual elements in the toolbar:</p> <ul style="list-style-type: none"> » Collapse All Lets users collapse all elements such as togglers, drop-down effects, and expanding text effects in a topic (if they are expanded). This toggles with the Expand All button. » Current Topic Index Lets users see the display for the current topic in the sequence. This is typically used in conjunction with the "Next Topic" and "Previous Topic" buttons when the topic in question exists in a browse sequence. Flare finds the current topic in a browse sequence and display its position in relation to the rest of the topics in that sequence. (The root node is used to determine the count; in other words, if there are "child" topics in the sequence structure under the current topic, they will be included in the count.) » Edit User Profile Lets users edit their Pulse or Feedback registration profile settings. This toggles with the Login button. The Edit User Profile button is shown if the user is already logged in. » Expand All Lets users expand all elements such as togglers, drop-down effects, and expanding text effects in a topic (if they are not yet expanded). This toggles with the Collapse All button. » Login Lets users log in to Pulse or Feedback, if you have integrated it with the output. This toggles with the Edit User Profile button. The Login button is shown if the user is not yet logged in.

Style	Description
	<ul style="list-style-type: none"> » Next Topic Lets users open the next topic in the sequence. First, Flare attempts to find the current topic in a browse sequence and navigate to the next topic from there. If you do not have a browse sequence, Flare looks at the position of the current topic in your table of contents (TOC) and opens the next topic after it. » Previous Topic Lets users open the previous topic in the sequence. First, Flare attempts to find the current topic in a browse sequence and navigate to the previous topic from there. If you do not have a browse sequence, Flare looks at the position of the current topic in your table of contents (TOC) and opens the previous topic before it. » Print Lets users open the Print dialog so that they can send the open topic to the printer. » Remove Highlight After a user performs a quick search in a topic, the search text found in the topic is highlighted. This button lets users turn the highlights off. » Topic Ratings (full) Displays ratings for a topic (if you have incorporated your output with MadCap Pulse or Feedback). For example, if a topic has been rated three out of five stars, "Topic Ratings (full)" lets you control how the three stars look. » Topic Ratings (empty) This shows an empty rating (by default a non-colored star). For example, if a topic has been rated three out of five stars, "Topic Ratings (empty)" lets you control how the two stars look.
Toolbar Button Separator	This controls any separators that you may have added to the toolbar. A separator is the divider between the toolbar buttons and the navigation pane.



Note: In order to see these items in the output, you must include a WebHelp toolbar or topic toolbar in your project. For more information see the online Help.



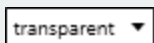


Note: You can add a custom toolbar button class by clicking **Toolbar Button** and in the local toolbar selecting **Add Class**.


5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Following is a general list in alphabetical order:

BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.


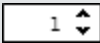
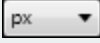
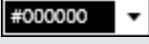




Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.

	You can select a type of line for the border (e.g., solid, dashed, dotted).
	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.




- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field

type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

BOX SHADOW

You can change any of the following to affect the box shadow on the element:

- » **Horizontal Shadow Length** This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Vertical Shadow Length** This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Blur Length** This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Color** This lets you select a color for the shadow. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Shadow Insert** This changes the shadow from an outer shadow (outset) to an inner shadow. Click in the field and select an option. If you select **false** the shadow will be outset. If you select **true** the shadow will be inset.

EVENT

You can provide a command for the item when it is clicked by the user.

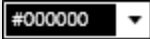


EXAMPLE

Let's say you want to provide a command for an item (e.g., so that clicking the button opens a website). The following command opens the MadCap Software website when a user clicks the button:

```
window.open("http://www.madcapsoftware.com");
```

GENERAL

You can change the color for the element.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

HEIGHT

You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

LAYOUT

You can click in the **Display** field and choose whether to show the element (block) or not (none).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

POSITION

You can change the position of the element, meaning its width and height:

- » **Width** You enter a width for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Height** You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

6. Click  to save your work.



Note: You can also use the UI Text tab in the Skin Editor to change labels for some of these styles.

Search Results in HTML5 Skins

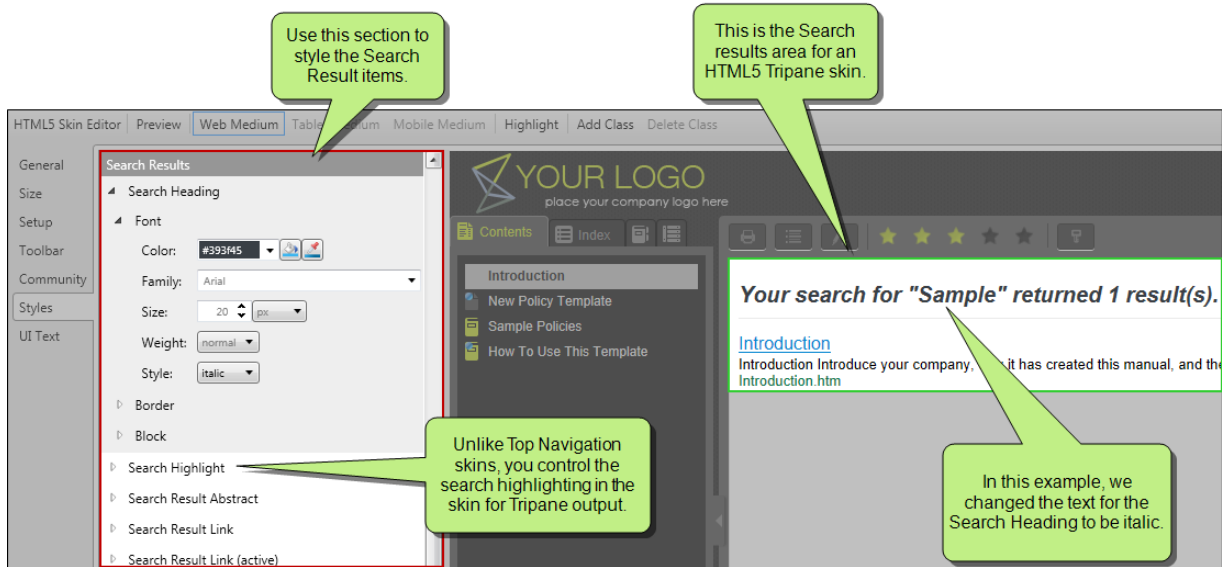
A Tripane or Top Navigation skin can be used to change the appearance of search result items. You can also do this with a smaller Search Results skin component.

The screenshot displays the HTML5 Skin Editor interface. On the left, a sidebar contains a 'Search Results' section with various styling options. A red box highlights this section, with a callout stating: 'Use this section to style the Search Result items.' The 'Search Results' section includes a 'Search Heading' subsection with font settings (Color: #393f45, Family: Arial, Size: 20, Weight: normal, Style: italic) and a list of expandable options: Search Highlight, Search Result Abstract, Search Result Link, and Search Result Link (active). A callout points to the 'Search Highlight' option, stating: 'Unlike Top Navigation skins, you control the search highlighting in the skin for Tripane output.'

The main preview area shows a web skin with a 'YOUR LOGO' header and a 'Contents' sidebar. The search results area is highlighted with a green box, with a callout stating: 'This is the Search results area for an HTML5 Tripane skin.' The search results display the heading 'Your search for "Sample" returned 1 result(s).', a link 'Introduction', and a snippet 'Introduction Introduce your company, it has created this manual, and the Introduction.htm'. A callout points to the search heading, stating: 'In this example, we changed the text for the Search Heading to be italic.'

TRIPANE SKINS

For Tripane skins, you can control the look of search result elements such as the heading, search highlighting, abstract (i.e., descriptive text of the file), links, glossary terms, and pagination.



TOP NAVIGATION SKINS

The search results options available for Tripane skins are similar for Top Navigation skins. The biggest difference is that search highlight settings are controlled in the skin only for Tripane output. For Top Navigation output, the search highlighting is controlled in your regular stylesheet by modifying special classes under the span style.

The screenshot displays the HTML5 Skin Editor interface. On the left, the 'Setup' panel is open to the 'Search Results' section. The 'Search Heading' is selected, and the 'Font' settings are visible: Color is set to #800000, Family is empty, Size is 1.5, Weight is normal, and Style is normal. The 'Search Results' list on the left includes: Search Results, Search Result Abstract, Search Result Link, Search Result Link (active), Search Result Link (focus), and Search Result Link (hover). On the right, the preview shows a search results page with a dark header containing a logo and the text 'company logo here'. Below the header, the search results are displayed. The search heading is 'Your search for "content" returned 14 result(s).'. Below this, there are sections for 'Tables' and 'Lists', each with a heading and a paragraph of placeholder text. The 'Tables' section includes a link to 'B_Features/Conditions Notes Tables/Tables.htm'. The 'Lists' section includes a link to 'B_Features/Images/ImagePositionedLeft.htm'. Several callout boxes provide additional information: 'Use this section to style the Search Result items.' points to the 'Search Results' list in the Setup panel. 'Many of the same options are available for Top Navigation skins that are available for Tripane skins. However, there are some minor differences in the styles and properties that are available.' points to the 'Font' settings. 'This is the Search results area for an HTML5 Top Navigation skin.' points to the preview area. 'Notice that Top Navigation skins do not include an option for search highlighting. That's because search highlighting is controlled in a regular stylesheet for Top Navigation output.' points to the 'Search Result Link' item in the Setup panel. 'In this example, we changed the text for the Search Heading to be red.' points to the search heading in the preview.

Use this section to style the Search Result items.

Many of the same options are available for Top Navigation skins that are available for Tripane skins. However, there are some minor differences in the styles and properties that are available.

This is the Search results area for an HTML5 Top Navigation skin.

Notice that Top Navigation skins do not include an option for search highlighting. That's because search highlighting is controlled in a regular stylesheet for Top Navigation output.

In this example, we changed the text for the Search Heading to be red.

SEARCH RESULTS SKIN COMPONENTS

Alternatively, you can use the Search Results skin component and its related proxy to design a custom container to display search results.

The options in the skin component are the same as those in the Top Navigation skin. Chances are good that you will never need to use a Search Results skin component, but it is available in case you want the flexibility of having another page showing search results.

The Search Results skin component is not available if you are generating Tripane output. It is only available when you are producing Top Navigation output, or HTML5 output without a skin selected.

HOW TO SPECIFY STYLE SETTINGS FOR SEARCH RESULTS IN HTML5

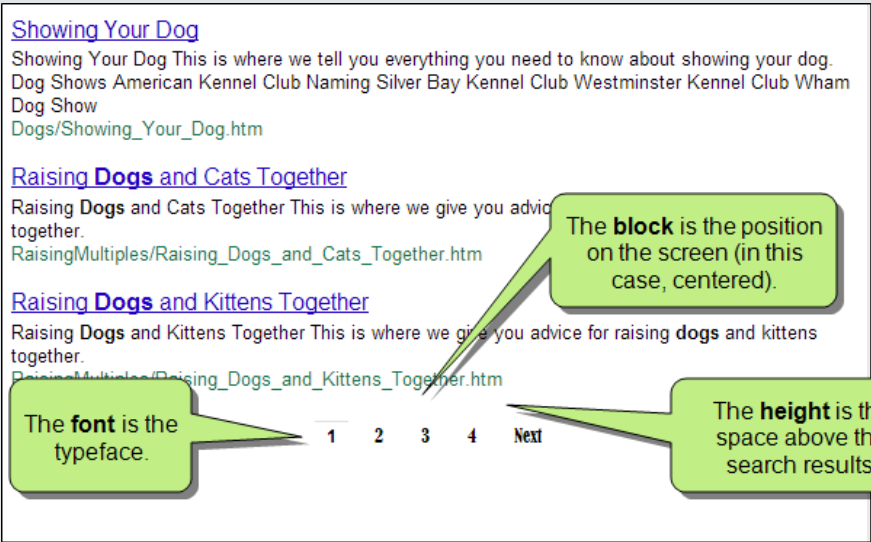
1. Open an HTML5 skin.
2. (Optional) If you are using responsive output, make sure you select the appropriate medium—Web, Tablet, or Mobile—in the local toolbar before making changes to styles. If the skin has not been enabled for responsive output, you can make changes only for the Web medium. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.
3. Select the **Styles** tab.
4. On the left side of the editor, in the **Search Results** section, expand any of the following styles:

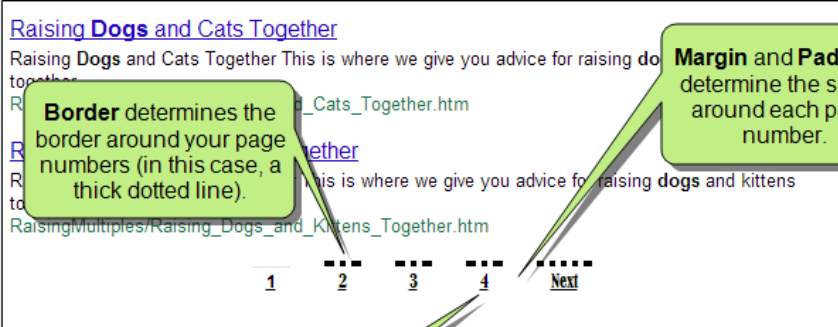
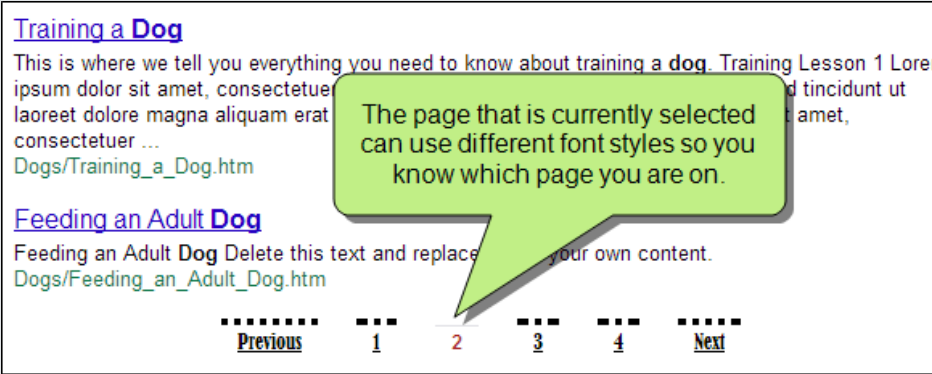
Style	Description
Search Heading	This is the heading at the beginning of the search results, indicating how many search results were found.

Style	Description
Search Highlight	<p>When users perform searches in your online output, the keywords that are found may be highlighted in the topics. The background for each term found in a topic can be highlighted in a different color. In Flare you can use styles to change not only the color background, but other settings as well (e.g., font style, text decoration).</p> <p>The Search Highlight style has properties both at the root and on children nodes. If you want the setting(s) to be applied to all of the items, complete the fields under the Search Highlight style itself. Otherwise, expand any of the child styles (Search Highlight 01 to 10) to apply settings to them individually. Each search highlight style is numbered to coordinate with the order search terms are entered. For example, the style named "Search Highlight 01" will be used for matches to the first search term entered and "Search Highlight 10" would coordinate with the tenth entered term. You can customize up to 10 highlight colors. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p> <div> <p>EXAMPLE</p> <p>Let's say that, for "Search Highlight 01" in this skin, you change the background color to blue. For "Search Highlight 02," you change the background color to orange. And for "Search Highlight 03," you change the background color to yellow.</p> <p>If a user enters "topic information help find" in the search field, the term "topic" will be displayed with a blue background everywhere it occurs in a topic. The term "information" will be displayed with an orange background, and the term "help" will be displayed with a yellow background. The term "find" will be displayed with the default background color specified by Flare (since you did not change it).</p> <p>If another user enters "help find topic information" in the search field, the</p> </div>

Style	Description
	<p>term "help" will be displayed with a blue background. The term "find" will be displayed with an orange background, and the term "topic" will be displayed with a yellow background. The term "information" will be displayed with the default background color specified by Flare (since you did not change it).</p>
Search Result Abstract	This is the opening paragraph text for a search result, giving you context.
Search Result Link	This is the link at the top of the search result.
Search Result Link (active)	This is the search result link when it is in an active state.
Search Result Link (focus)	This is the search result link when it has focus.
Search Result Link (hover)	This is the search result link when a user hovers over it.

Style	Description
Search Result Link (visited)	This is the search result link after it has been selected.
Search Result Path	This shows the path to the file containing the search result.
Search Glossary Result	This controls the look of the area holding glossary terms and definitions if they are included at the beginning of search results.
Search Glossary Term	This controls the look of glossary terms (not definitions) if they are included at the beginning of search results.
Search Glossary Term Link	This controls the look of a glossary term (not definition) if it is included at the beginning of search results, and if it is linked to another topic. In other words, when you create a glossary term, you have a choice of simply typing the definition or linking it to a topic; this style controls the latter situation.
Search Glossary Term Link (active)	This controls the look of a glossary term (not definition) if it is included at the beginning of search results, and if it is linked to another topic. This controls the look of the link when it is in an active state.
Search Glossary Term Link (focus)	This controls the look of a glossary term (not definition) if it is included at the beginning of search results, and if it is linked to another topic. This controls the look of the link when it has focus.

Style	Description
Search Glossary Term Link (hover)	This controls the look of a glossary term (not definition) if it is included at the beginning of search results, and if it is linked to another topic. This controls the look of the link when a user hovers over it.
Search Glossary Term Link (visited)	This controls the look of a glossary term (not definition) if it is included at the beginning of search results, and if it is linked to another topic. This controls the look of the link after it has been selected.
Search Glossary Definition	This controls the look of glossary definitions (not terms) if they are included at the beginning of search results.
Search Pagination	<p>This controls the block, font, and height (i.e., distance from the last search result) of search pagination elements.</p> 

Style	Description
Search Pagination Item	<p>This look of text in search pagination elements.</p>  <p>Border determines the border around your page numbers (in this case, a thick dotted line).</p> <p>Margin and Padding determine the space around each page number.</p> <p>Decoration determines text styles (in this case, underline).</p>
Search Pagination Item (selected)	<p>This controls the look of text of a search pagination element when it is selected.</p>  <p>The page that is currently selected can use different font styles so you know which page you are on.</p>




Style	Description
Search Pagination Item (hover)	<p>This controls the look of text of a search pagination element when you hover over it.</p> <div> <p>Raising Dogs and Cats Together</p> <p>Raising Dogs and Cats Together This is where we give you advice for raising dogs and cats together.</p> <p>RaisingMultiples/Raising_Dogs_and</p> <p>Raising Dogs and Kittens Together</p> <p>Raising Dogs and Kittens Together This is where we give you advice for raising dogs and kittens together.</p> <p>RaisingMultiples/Raising_Dogs_and_Kittens_Together</p> <p>1 2 3 4 Next</p> </div> <p>You can use different styles when you hover over a page number.</p>


5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Also, you might see nodes for a Tripane skin that you do not see for a Top Navigation skin, and vice versa. Following is a general list in alphabetical order:

BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.



Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BLOCK


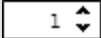



This is the container (or "block") holding content in an element. You can change any of the following to affect the block:

- » **Line Height** This is how tall the container is that holds the content. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Text Align** This changes the alignment of the content. Click in the field and select an option (center, justify, left, right).
- » **Word Wrap** You can click in the **Word Wrap** field and choose whether to wrap text for the element (normal) or not (nowrap).

BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.

	You can select a type of line for the border (e.g., solid, dashed, dotted).
	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.



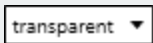
This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

BOX SHADOW

You can change any of the following to affect the box shadow on the element:

- » **Horizontal Shadow Length** This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Vertical Shadow Length** This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Blur Length** This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Color** This lets you select a color for the shadow. You can use any of the following:



You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.



This opens the Color Picker dialog, which lets you choose a color in many ways.



This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Shadow Insert** This changes the shadow from an outer shadow (outset) to an inner shadow. Click in the field and select an option. If you select **false** the shadow will be outset. If you select **true** the shadow will be inset.

FONT

You can change any of the following to affect the font:

- » **Color** This lets you select a color for the font. You can use any of the following:



You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.



This opens the Color Picker dialog, which lets you choose a color in many ways.



This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

HEIGHT

You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

LAYOUT

You can click in the **Display** field and choose whether to show the element (block) or not (none).

MARGIN

You can change the margin for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

TEXT DECORATION

You can click in the **Decoration** field and select a text decoration (e.g., underline, line-through, overline, underline).

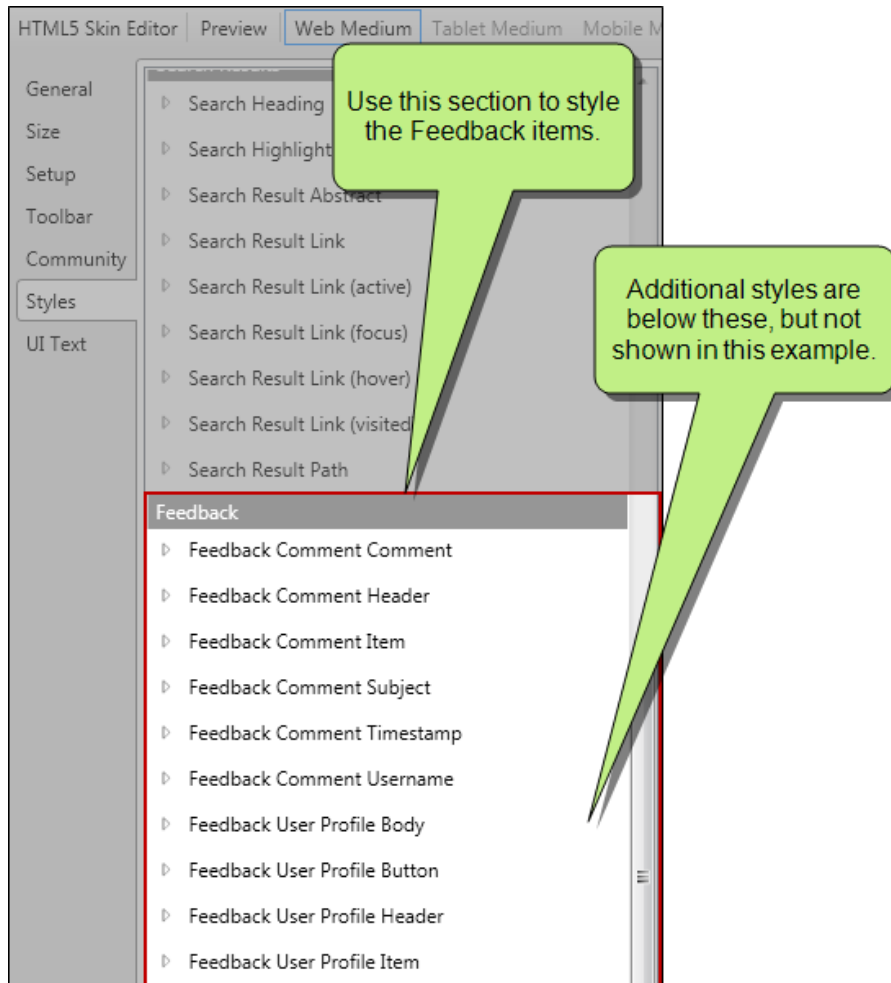
6. Click  to save your work.



Note: If you want to style the search bar and its elements, you can use the Header section in the Skin Editor. See "Headers in HTML5 Skins" on page 208.

Feedback in HTML5 Skins

If you integrate your output with MadCap Feedback, you can use styles in a skin to control different elements.



HOW TO SPECIFY STYLE SETTINGS FOR FEEDBACK IN HTML5

1. Open an HTML5 skin.
2. If you are using responsive output, make sure you select **Web Medium** in the local toolbar. Feedback integration is not supported in the Tablet and Mobile views, although topic rating stars will display in the topic toolbar. For more information about these mediums and responsive output, see "Responsive Web Design" on page 19.
3. Select the **Styles** tab.
4. On the left side of the editor, in the **Feedback** section, expand any of the following styles:

Style	Description
Feedback Comment Comment	This lets you control the look of topic comment text.
Feedback Comment Header	This lets you control the look of the header text that appears above comments in a topic.
Feedback Comment Item	This lets you control the look of the container displaying the contents, including header and the comments.
Feedback Comment Subject	This lets you control the look of the subject text that appears just above a comment.
Feedback Comment Timestamp	This lets you control the look of the date and time that appears in a comment.
Feedback Comment Username	This lets you control the look of the user's name that appears in a comment.

Style	Description
Feedback User Profile Body	This lets you control the look of the interface used for creating a profile.
Feedback User Profile Button	This lets you control the look of the buttons in the interface that is used for creating a profile.
Feedback User Profile Header	This lets you control the look of the header text in the interface that is used for creating a profile.
Feedback User Profile Item	<p>The Feedback User Profile dialog always includes a "Username" and "E-mail Address" field. In addition to those fields, you can add many others by using the Community tab in the Skin Editor.</p> <p>Once you've added the fields, you can edit the Feedback User Profile Item style in the skin to specify which fields are required for users to complete.</p> <p>The Feedback User Profile Item style has properties both at the root and on children nodes. In addition to the many standard fields that can be included (e.g., Address1, City, Department, Occupation), you can add up to 10 custom fields (e.g., Custom1, Custom2, Custom3), which can be used for any kind of field not already available.</p> <p>If you want the setting(s) to be applied to all of the items, complete the General>Required field under the Feedback User Profile Item style itself. Otherwise, expand any of the child styles (Address1, Address2, Birthdate, etc.) to apply settings to them individually. For more information about root and children properties, see "Style Sections and Root/Child Properties" on page 167.</p>
Feedback User Profile Item Group	This lets you control the look of the container holding fields in the interface when creating a profile.

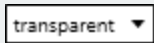


Style	Description
Feedback User Profile Item Input	This lets you control the look of the text that users type in fields when creating a profile.
Feedback User Profile Item Label	This lets you control the look of the field label text in the interface that is used for creating a profile.


5. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Following is a general list in alphabetical order:

BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.









Note: The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.




	You can select a type of line for the border (e.g., solid, dashed, dotted).
	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

FONT

You can change any of the following to affect the font:

- » **Color** This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

GENERAL

You can click in the **Required** field and choose whether to make the element required (true) or not (false).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

6. Click  to save your work.



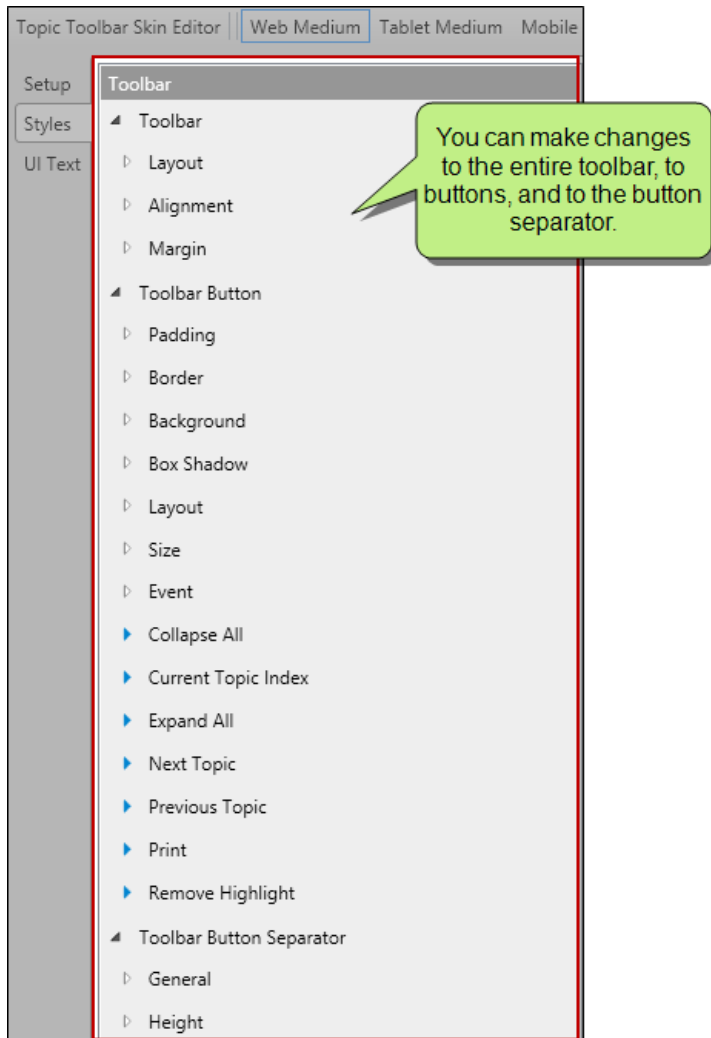
Note: In order to see these items in the output, you must enable Feedback with your output. For more information see the online Help.



Note: You can also use the UI Text tab in the Skin Editor to change labels for some of these styles.

Topic Toolbars in HTML5 Skin Components

In Topic Toolbar skin components, you can use skin styles to control the look of a toolbar you add via a proxy.



For more about web toolbars versus topic toolbars, see the online Help.

HOW TO SPECIFY STYLE SETTINGS FOR A TOPIC TOOLBAR IN HTML5

1. Open an HTML5 Topic Toolbar skin component.
2. Select the **Styles** tab. (You can use the **Setup** tab to select the buttons to be included in the toolbar.)
3. On the left side of the editor, in the **Toolbar** section, expand any of the following styles:

Style	Description
Toolbar	This controls the look of the entire toolbar, such as its alignment and margins.
Toolbar Button	This controls the look of individual buttons in the toolbar. If you set the properties directly under this style, they will be used for all buttons. If you want specific values for a particular button, expand the style for that button and then set the properties under it.
Toolbar Button Separator	This controls the look of the separator between buttons.

4. Expand any of the nodes and complete the necessary fields. The properties and fields that you see are different for each node. Following is a general list in alphabetical order:

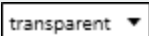


ALIGNMENT


You can click in the **Horizontal** field and choose whether to position to the **right** or **left** side of the display.


BACKGROUND

You can change any of the following to affect the background:

- » **Gradient** This lets you change the background color using a progression effect. Select the beginning gradient color in the first field, and then select a second gradient color in the second field. You can use any of the following to enter or select a color.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.


- » **Color** This lets you select a single color.
- » **Image** This lets you select an image for the background. Click  and use the dialog to select an image file.
- » **Repeat** Use this field to tell Flare whether the image should repeat or not.

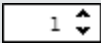




 **Note:** The Image field has the highest precedence, which means that if you enter settings in all of the Background fields, the image will win. Gradient has the next highest precedence. If you want to use the Color field, you need to make sure the Gradient fields are set to transparent and that there is no image selected.

BORDER

You can change any of the following to affect the border on the element:

- » **Border Edge Fields (Top, Right, Bottom, Left)** You can add a border line around any of the edges of the element. You can use any of the following. The first three fields are used to control the type and size of the border line. The last three fields are alternative ways to choose a color for the border.

	You can select a type of line for the border (e.g., solid, dashed, dotted).
---	---

	Type a number in the field or use the arrows to increase or decrease it.
	Click in this field and select a unit of measurement (e.g., pixels, points, centimeters).
	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.




- » **Border Radius Fields (Top Left, Top Right, Bottom Right, Bottom Left)** If you want the element to have rounded corners, you can enter values in any of these fields. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters). The higher the number, the more of a curve the corner will have.

BOX SHADOW

You can change any of the following to affect the box shadow on the element:

- » **Horizontal Shadow Length** This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Vertical Shadow Length** This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Blur Length** This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

» **Shadow Color** This lets you select a color for the shadow. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

» **Shadow Insert** This changes the shadow from an outer shadow (outset) to an inner shadow. Click in the field and select an option. If you select **false** the shadow will be outset. If you select **true** the shadow will be inset.

EVENT




You can enter a special command that is triggered when an end user clicks a button. For example, the following command opens the MadCap Software website when a user clicks the button:

```
window.open("http://www.madcapsoftware.com");
```

FONT

You can change any of the following to affect the font:



- » **Color** This lets you select a color for the font. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

- » **Family** You can select or type a specific font family (e.g., Arial, Tahoma, Verdana).
- » **Size** You can change the size of the font. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Weight** You can click in this field and make the font normal or bold.
- » **Style** You can change click in this field and make the font normal or italic.

GENERAL

You can change the color for the element.

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color.
	This opens the Color Picker dialog, which lets you choose a color in many ways.



This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

HEIGHT

You enter a height for the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

LAYOUT

You can click in the **Display** field and choose whether to show the element (block) or not (none).

MARGIN

You can change the margin for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

PADDING

You can change the padding for any of the sides on the element (Left, Right, Top, Bottom). In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

SIZE

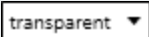


You can change the height or width of the element. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).

TEXT SHADOW

You can change any of the following to affect the text shadow on the element:

- » **Shadow Visibility** You can click in this field and choose to make the shadow either hidden or visible.

- » **Horizontal Shadow Length** This is the position of the horizontal shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Vertical Shadow Length** This is the position of the vertical shadow. Negative values are allowed. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Blur Length** This is the blur distance. In the first field type a number or use the arrows. Then click in the second field and select the unit of measurement (e.g., pixels, points, centimeters).
- » **Shadow Color** This lets you select a color for the shadow. You can use any of the following:

	You can type a hexadecimal number (e.g., #000000) directly in this field. Alternatively, you can click the down arrow and choose a color or make the background transparent.
	This opens the Color Picker dialog, which lets you choose a color in many ways.
	This temporarily adds a small bar above the cursor. As you move the cursor over any area of your computer screen, the color changes in the bar to reflect the color that is directly behind the tip of the cursor. When you click, that color is automatically loaded into the Color Picker dialog.

5. Click  to save your work.

Other Skin Tasks

There are various skin-related tasks unique to HTML5 output that you can perform.

This chapter discusses the following:


Changing Text Strings in HTML5 Skins	280
Showing Navigation Links in Standalone Topics	281
Specifying Browser Settings	282
Specifying Navigation Pane Settings for HTML5 and WebHelp Outputs ..	283
Specifying Web Toolbar Settings—HTML5 Output	287



Changing Text Strings in HTML5 Skins

Unlike other output types, separate language skins are not used for HTML5 targets. Instead, a UI Text tab is located in the Skin Editor, which lets you perform all of the localization tasks that you would normally accomplish in a language skin. This tab not only lets you enter text in other languages, but it lets you change the substance of text strings altogether, whether they are in your primary language or another.

HOW TO CHANGE TEXT STRINGS IN AN HTML5 SKIN OR COMPONENT


1. Open an HTML5 skin or a skin component.
2. Select the **UI Text** tab.
3. From the **Language** drop-down, select the language for the skin.
4. In the grid below, type text in the **Value** field for each relevant row.
5. Click  to save your work.

Showing Navigation Links in Standalone Topics

You can add a navigation link to the top or bottom of topics in HTML5, WebHelp, WebHelp AIR, or WebHelp Plus outputs. This navigation link will not display unless the output topic is opened as a standalone (outside of the main navigation framework of the output). By clicking the link, a user can view the standalone topic in the main navigation framework.

You can also modify the skin styles for these navigation links to change the way they look or the words used in the links. To do this in a Standard skin, open the Skin Editor, select the **Styles** tab, expand the **Control** group, and set the values for the **Navigation Link Bottom** and **Navigation Link Top** properties. To do this in an HTML5 skin, open the Skin Editor, select the **Styles** tab, expand the **Navigation Link** group, and set the values for the **Bottom** and **Top** properties. See "Skin Styles" on page 163.

HOW TO SHOW NAVIGATION LINKS IN WEBHELP STANDALONE TOPICS

1. Open a Standard or HTML5 skin.
2. If you opened a Standard skin, select the **WebHelp Setup** tab. If you opened an HTML5 skin, select the **Setup** tab.
3. In the **Topic Settings** section, click **Show navigation link at top of topic** or **Show navigation link at bottom of topic**.
4. Click  to save your work.



Note: This is supported in HTML5 Tripane skins, but not in HTML5 Top Navigation skins.


Specifying Browser Settings

You can specify which features will be used in the output window when a browser is involved.

HOW TO SPECIFY BROWSER SETTINGS FOR WEBHELP OUTPUT

1. Open a Standard or HTML5 skin.
2. Select the **WebHelp Setup** tab. If you opened an HTML5 skin, select the **Setup** tab.
3. Select the options that you want to include in the browser output window.
 - » **Use Browser Default Settings** Click this check box if you want the output window to use the settings specified in the user's Internet browser. If you do not select this check box, you can select specific browser features below to include in the output window.

If you do not select "Use Browser Default Settings" above, click the check boxes next to each browser feature that you want to include in the output window.

- » **ToolBar** Displays the browser toolbar at the top of the output window.
 - » **Menu** Displays the browser menu at the top of the output window.
 - » **LocationBar** Displays the browser location bar (or address bar) at the top of the output window.
 - » **StatusBar** Displays the browser status bar at the bottom of the output window.
 - » **Resizable** Allows users to resize the output window.
4. Click  to save your work.



Important: If you specify browser settings for WebHelp, WebHelp Plus or HTML5, you *must* provide your users with a proper link to open your online Help. Otherwise the browser features mentioned below will not work properly.



Note: This is supported in HTML5 Tripane skins, but not in HTML5 Top Navigation skins.

Specifying Navigation Pane Settings for HTML5 and WebHelp Outputs

You can specify navigation settings for HTML5, WebHelp, WebHelp AIR, or WebHelp Plus output. For most of these outputs, the navigation pane is used to hold the TOC, Index, Search, Glossary, Browse Sequences, and Favorites in an accordion-type structure. For HTML5 Top Navigation output, the navigation pane settings let you choose the pane position, depth level of menu items, and a URL for a logo.

HOW TO SPECIFY NAVIGATION PANE SETTINGS FOR HTML5 AND WEBHELP OUTPUTS

1. Open a Standard or HTML5 skin.
2. Do one of the following, depending on the type of output:
 - » **HTML5** Select the **Setup** tab.
 - » **Others** Select the **WebHelp Setup** tab.
3. Select the appropriate options.

PANE POSITION

Select a location in the output window for the navigation pane. A small preview to the right changes as you make your selection. For HTML5 Top Navigation output, this refers to the flyout pane that appears when responsive output is activated and the screen is too small to show the top menu.

TOP MENU LEVELS TO SHOW (DEPTH)

Specify how many levels of your TOC items are included in the top menu navigation. The default is 3. It is recommended that you avoid including too many depth levels in the top navigation. (This option is available only for HTML5 Top Navigation skins—see "Top Navigation Output" on page 63.)

LOGO URL

Select a topic to be linked to the logo in the output. By default, the logo is linked to your Home (i.e., startup) topic. However, you can select a different topic or even enter the URL to your company's website instead (remember to include `http://` at the beginning of the path if you do this). (This option is available only for HTML5 Top Navigation skins—see "Top Navigation Output" on page 63.)

PANE SIZE

Enter the width of the navigation pane for the output window (in pixels). You can type a number or use the up and down arrows. (This option is not available in HTML5 Top Navigation skins.)

VISIBLE ACCORDION ITEMS

Specify how many navigation items (e.g., TOC, Index, Search) you want to be included at full size in the navigation pane. (This option is not available in HTML5 skins.)

E X A M P L E

If you include six tabs in the output window and set this field to "4," the first four navigation items will be shown with full-sized links. The remaining two items will still be accessible in the navigation pane, but their links will not be full-sized, but rather smaller icons.

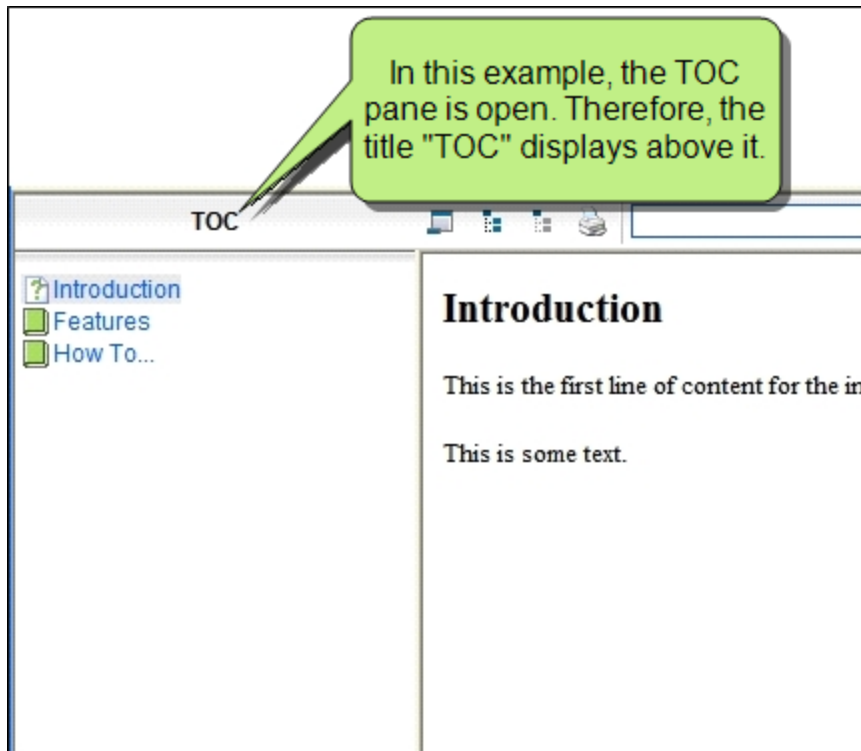
HIDE NAVIGATION PANE ON STARTUP

Click this check box if you do not want the navigation pane to be immediately displayed when the Help is accessed in this skin. You might use this option, for example, if you are creating a skin to be used for context-sensitive Help (CSH) topics (as opposed to your main Help system with full navigation). With CSH topics, the navigation pane is not usually necessary because the user wants to see information only for a very specific area. (This option is not available in HTML5 Top Navigation skins.)

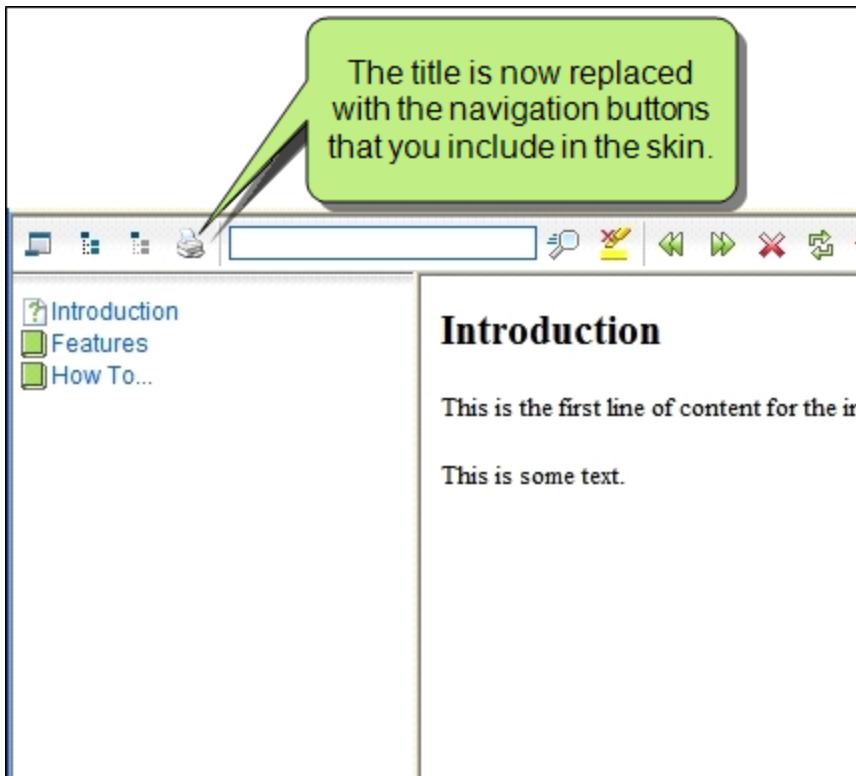
EXCLUDE ACCORDION TITLE

Removes the accordion title from the WebHelp output. (This option is not available in HTML5 skins.)

If you generate output using WebHelp, WebHelp AIR, or WebHelp Plus, the title of the active accordion bar displays by default in the output, like this:



You have the option of excluding this accordion title from the output. This shifts the navigation buttons for the output to the left, like this:




4. Click  to save your work.

Specifying Web Toolbar Settings—HTML5 Output

You can specify web toolbar settings for some online outputs. This includes determining which buttons are displayed in the toolbar. You can also add custom JavaScript for the toolbar.

For more about web toolbars versus topic toolbars, see the online Help.

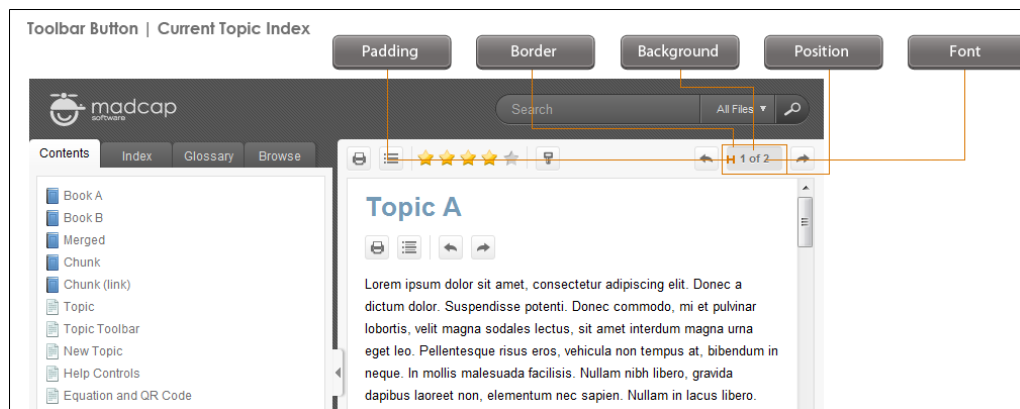
HOW TO SPECIFY CUSTOM TOOLBAR SETTINGS FOR HTML5 OUTPUT

1. Open an HTML5 skin.
2. Select the **Toolbar** tab.
3. Select options to include specific features (buttons) in the toolbar. Items that will be included in the toolbar are displayed in the Selected section on the right. Items that will not be included in the toolbar are displayed in the Available section on the left. You can use the right arrow button  to move items from the Available section to the Selected section.

Following are descriptions of the various items.

CURRENT TOPIC INDEX

Lets users see the display for the current topic in the sequence. This is typically used in conjunction with the "Next Topic" and "Previous Topic" buttons when the topic in question exists in a browse sequence. Flare finds the current topic in a browse sequence and display its position in relation to the rest of the topics in that sequence. (The root node is used to determine the count; in other words, if there are "child" topics in the sequence structure under the current topic, they will be included in the count.)



EDIT USER PROFILE

Lets users edit their Pulse or Feedback registration profile settings.



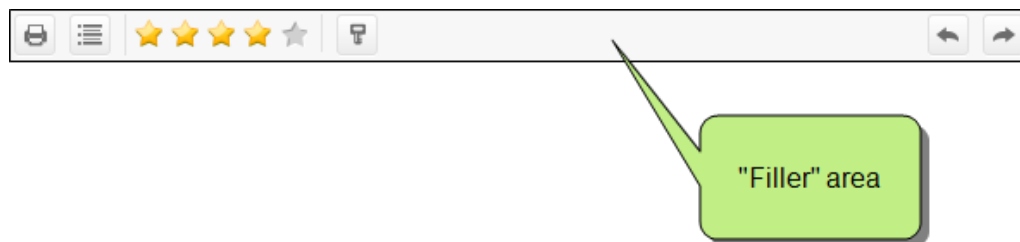
EXPAND ALL

Lets users expand all elements such as togglers, drop-down effects, and expanding text effects in a topic (if they are not yet expanded).



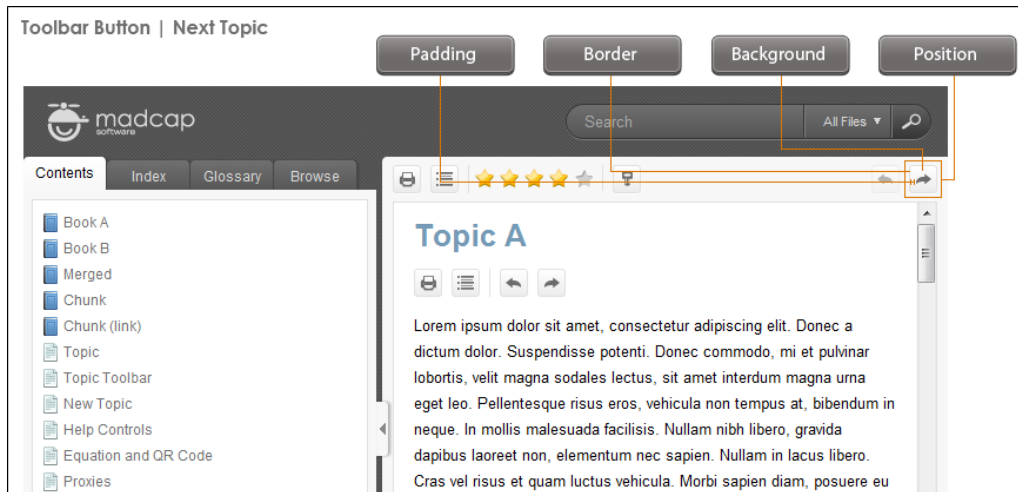
FILLER

Lets users add a filler item, which is used to specify where the empty space should go in the toolbar.



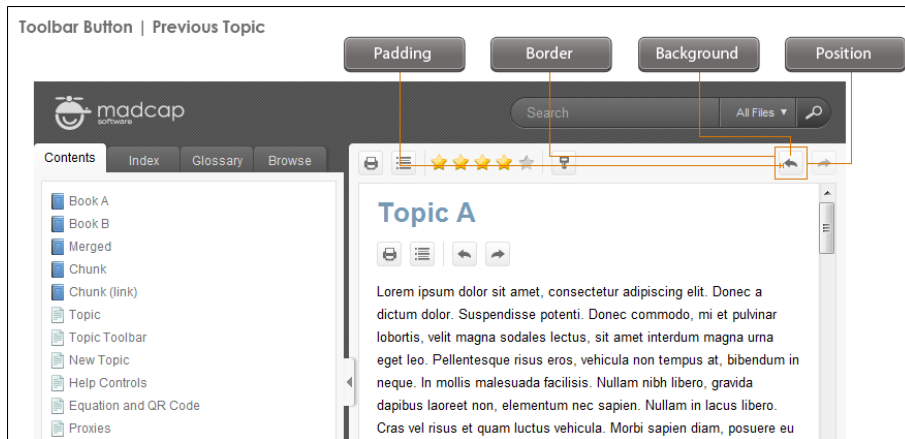
NEXT TOPIC

Lets users open the next topic in the sequence. First, Flare attempts to find the current topic in a browse sequence and navigate to the next topic from there. If you do not have a browse sequence, Flare looks at the position of the current topic in your table of contents (TOC) and opens the next topic after it.



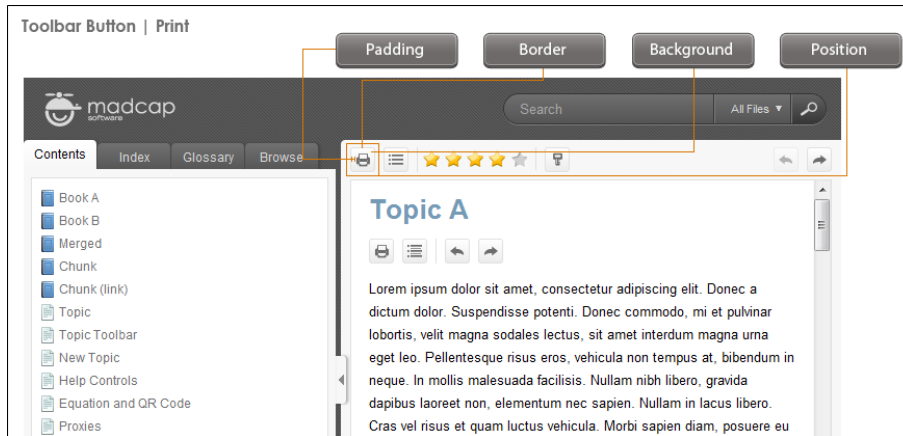
PREVIOUS TOPIC

Lets users open the previous topic in the sequence. First, Flare attempts to find the current topic in a browse sequence and navigate to the previous topic from there. If you do not have a browse sequence, Flare looks at the position of the current topic in your table of contents (TOC) and opens the previous topic before it.



PRINT

Lets users open the Print dialog so that they can send the open topic to the printer.



REMOVE HIGHLIGHT








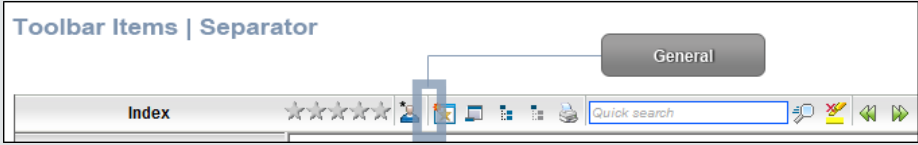

After a user performs a quick search in a topic, the search text found in the topic is highlighted. This button lets users turn the highlights off.

TOPIC RATINGS

Lets users submit ratings for a topic (if you have incorporated your output with MadCap Pulse or Feedback).



4. (Optional) You can use the buttons at the bottom of the tab to perform additional tasks.

	<p>Opens the New Toolbar Button dialog, which lets you add a custom button to the toolbar. After creating the custom button, select it and click  to move it to the Selected section.</p> <p>If you want to provide a command for the custom button (e.g., so that clicking the button opens a website), you can select the Styles tab in the Skin Editor, select one of the styles listed above, expand the Event property group, and enter a command in the Click field. For example, the following command opens the MadCap Software website when a user clicks the button:</p> <pre>window.open("http://www.madcapsoftware.com");</pre> <p>Alternatively, you can use the JavaScript area on the Toolbar tab to enter a command for the button. See the next step.</p>
	<p>Removes the selected custom button from the Available section.</p>
	<p>Moves the selected item up in the order in the Selected section. In the output, buttons are displayed left to right according to the order in which they appear on this tab, with the item at the top being the first button shown on the left in the output.</p>
	<p>Moves the selected item down in the order in the Selected section.</p>
	<p>Removes the selected item(s) from the Selected section.</p>
	<p>Adds a separator to the Selected section. A separator is the divider between the toolbar buttons and the navigation pane.</p> <div data-bbox="410 1318 1323 1461">  </div>
	<p>Resets the toolbar to the default settings.</p>

5. (Optional) If you want to include custom JavaScript in the toolbar, click **Edit** in the area to the right. This opens the Toolbar JavaScript dialog, which you can use to enter or load custom JavaScript for the toolbar.

For example, the following command opens the MadCap Software website when a user clicks the button:

```
function MyFunction()  
{  
    window.open("http://www.madcapsoftware.com");  
}
```

6. Click  to save your work.



Note: This is supported in HTML5 Tripane skins, but not in HTML5 Top Navigation skins.

CSH Calls

You can work with your developer (or you can function as the developer yourself) to connect an HTML5 system to a software application or to open specific parts of your online documentation from a simple web link. The application or web link(s) can be connected to the basic Help output, to specific topics in the output (context-sensitive Help), or both.

- » **Basic Help** The developer can connect the application or web link to your basic HTML5 output, rather than to a specific topic. The Help will open in the browser window, displaying the startup topic that you designate and the navigation elements that you include.
- » **Context-sensitive Help (CSH)** The developer can use CSH to connect the application or web link(s) to specific topics in the HTML5 output (as long as you have created CSH in your Flare project and share the header file information with the developer).

This chapter discusses the following:

What You Need to Do	298
CSH Calls for HTML5 Output—Developers	299



What You Need to Do

1. Work with your developer to determine how you want to connect the online output to the application or web link(s).
2. Create and build your HTML5 target in Flare.
3. Provide the developer with the output files, as well as the CSH header file.
4. Provide the developer with the information in the following topic: "CSH Calls for HTML5 Output—Developers" on the next page.

CSH Calls for HTML5 Output—Developers

Information for Developers

Use the following information if you are producing HTML5 and want to incorporate context-sensitive Help (CSH) into the software application.

There are two methods you can use.

- » **Method 1—JavaScript** Using this method requires calling a JavaScript function that Flare provides.
- » **Method 2—URL** Using this method, you can create a hyperlink to launch the Help system.

Which Method is Best for You?

Each method has its unique benefits. Generally speaking, the JavaScript method lets you have more control, whereas the URL method is a bit more quick and simple.

One reason to choose the JavaScript method is to better control the window size and location. With the URL method, the browser window automatically starts to open at the same size and location as the previous time that browser window was opened. But if you have specified a different size and location for your output window, the window will visibly move and resize accordingly. The JavaScript method prevents this type of behavior by opening the window directly to the size and location you specified. You would set the window size and location in the skin. Then in the JavaScript call you would specify the appropriate skin.

Another benefit to using the JavaScript method is that it is required in order for the Browser Settings option to take effect. This option can be found on the Setup tab of the Skin Editor.

HOW TO USE JAVASCRIPT TO OPEN HTML5 OUTPUT

1. **Author** Add a header file to your project.
2. **Author** Add an alias file to your project.
3. **Author** In the Alias Editor, create and assign an identifier (ID) for the topic that you want to link to from the CSH call.

For example, if you have a topic called "Welcome.htm," you might create a new ID and also name it "Welcome." Then you might assign a value of 1000 to it. You would make sure that the topic Welcome.htm is assigned to the ID.

Finally, an optional step in the Alias Editor is to assign a skin to that ID. However, you can bypass this step and specify a skin later when you create the actual JavaScript, or you can choose not to select a skin at all.

4. **Author** Build your Help system using an HTML5 target and publish the output files to the final destination.
5. **Developer** Add a reference to the JavaScript file (which is created automatically when the author builds the output). This .js file should be named "csh.js." The reference to the JavaScript file should use the following format: `<script type="text/javascript" src="[path of file]/csh.js"></script>`.



Note: Make sure you use forward slashes (/) in the src path to the file, even if the file is referenced locally.

6. **Developer** Create a trigger and add the command to let users open the Help system. Here is a format that you can use to add a button.

```
<input type="button" value="Open Help System" onclick-  
k="MadCap.OpenHelp(ID, skin name, search string, first pick  
search string value );" />
```

You can change the input type and the value if necessary. The most important parts that you will adjust are the elements within parentheses ID, skin name, search string, first pick search string value).

- » **ID** This is the CSH ID that the author created in Flare (see Step 3 above). This can be either the ID name or value. The topic and skin associated with the ID will be used. If no skin is associated with the ID in Flare, the skin name that you provide in this command will be used. Alternatively, the ID may contain a topic path. In this case, the specified topic will be loaded with the skin that is specified in this command. The topic path must be relative to the Content folder of the Flare project. You also have the option of entering "null" instead of an ID to use the Help system's default starting topic.

- » **Skin Name** This is the name of the skin to use when opening the Help system. If a skin has been assigned to the ID in Flare (see Step 3 above) *and* you enter a skin name in this command, the skin name in the command will take precedence. You also have the option of entering "null" instead of a skin name if you want to use the Help system's default skin or to use the skin that is associated with the CSH ID in Flare.
- » **Search String** This is an optional element that automatically performs a search for a specific string.
- » **First Pick Search String Value** This element can be used in conjunction with the search string. If you use the first pick option, you can include a true or false value. If the value is true, the first topic found with the specified search string will be opened automatically. If the value is false, the search results will simply be displayed; the first topic will not be opened automatically.

In the following example, the topic and skin associated with "Welcome" will be used. No search string information is included.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp('Welcome', null, null, null );" />
```

In the following example, the topic associated with "Welcome" will be used. "BlueSkin" will override the skin associated with "Welcome." No search string information is included.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp('Welcome', 'BlueSkin', null, null );" />
```

In the following example, the topic and skin associated with the ID value 1000 will be used. No search string information is included.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp(1000, null, null, null );" />
```

In the following example, the topic associated with the ID value 1000 will be used. "BlueSkin" will override the skin associated with ID value 1000. No search string information is included.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp(1000, 'BlueSkin', null, null );" />
```

In the following example, "Company/Employees.htm" will be used with the default skin. No search string information is included.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp('Company/Employees.htm', null, null, null );" />
```

In the following example, both the default topic and skin will be used. A search will automatically be performed for the words "quarterly report," but the first topic found in the search *will not* be opened automatically.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp(null, null, 'quarterly report', false );" />
```

In the following example, the default topic will be used with "BlueSkin." A search will automatically be performed for the words "quarterly report," and the first topic found in the search *will* be opened automatically.

```
<input type="button" value="Open Help System" onclick="MadCap.OpenHelp(null, 'BlueSkin', 'quarterly report', true );" />
```

HOW TO USE A URL TO OPEN HTML5 OUTPUT

1. **Author** (Optional) Add a header file to your project.
2. **Author** (Optional) Add an alias file to your project.
3. **Author** (Optional) In the Alias Editor, create and assign IDs for the topics to which you want to provide links. If you do not want to create a header file, alias file, and IDs for topics, you can instead use the file names for the topics to which you want to link.

An optional step in the Alias Editor is to assign a skin to that ID.

4. **Author** Build your Help system using an HTML5 target and publish the output files to the final destination.
5. **Developer** Create a trigger and add a link to let users open a specific area of the Help system.

There is a certain amount of flexibility in terms of how you create the link and what you can include in it. Here is the basic structure of the link.

```
[main entry file].htm#cs hid=[ID number, ID name, or topic path/-  
name]&searchQuery=[search string]&firstPick=true
```

After the hash tag (#), you can specify any combination of the parameters (cs hid, searchQuery, firstPick), separated by ampersands (&). The order of the parameters does not matter.

- » **Main entry file** Provide the path to the main entry file for your output. The file name is determined by whatever you enter into the **Output File** field in the **General** tab of the Target Editor. If you do not provide a name in this field, the name "Default" will be used.
- » **cs hid** This is the CSH ID that you created in Flare (see Step 3 above). This can be either the identifier name or value. The topic and skin that is associated with the ID will be used (unless you override it in this link by specifying a different skin name). Alternatively, you can enter the path and name of the specific topic to which you want to link. If you use this element, you do not need to create an ID as described above. The topic path must be relative to the Content folder of the Flare project.
- » **searchQuery** This is an optional element that automatically performs a search for a specific string.

- » **firstPick** This element can be used in conjunction with the search string. If you include the first pick option, the first topic found with the specified search string *will* be opened automatically. If you do not include this element, the search results will simply be displayed; the first topic *will not* be opened automatically.

EXAMPLES

```
<a href-  
f="http://my.mycompany.com/Default.htm#cs hid=1000&searchQuery=World  
Cup Standings&firstPick=true">Click here to open</a>  
  
<a href-  
f="http://my.mycompany.com/Default.htm#cs hid=Soccer&searchQuery=World  
Cup Standings&firstPick=true">Click here to open</a>  
  
<a href-  
="h-  
ttp://my.mycompany.com/Default.htm#cs hid=Soccer.htm&searchQuery=World  
Cup Standings&firstPick=true">Click here to open</a>
```

In these examples, the following were used.

- » Default.htm = main entry file name
- » 1000 = CSH ID value
- » Soccer = CSH ID name
- » Soccer.htm = topic in the project, at the root level of the Content Explorer
- » World Cup Standings = search term

More About HTML5

Because HTML5 is a particularly powerful output type, there are several additional tasks to consider when using this format.

This chapter discusses the following:

Enabling HTML5 Server-based Output	306
Installing IIS and ASP.NET	307
Setting Up an HTML5 Target	318
Configuring IIS on Production Server	320
Starting Microsoft Indexing Service	322
Starting Microsoft Windows Search	323
Enabling HTML5 Server-based Search	324
Testing HTML5 Server-based Search	325
Viewing HTML5 Server-based Output	326
Merging Output at Runtime Using HTML5	331
Including Non-XHTML Files in HTML5 Search	341
HTML5 Server-based Output Scenario	345



Enabling HTML5 Server-based Output

You can create HTML5 output in its regular state, or you can select an option in the Publishing tab of the Target Editor to enable server-based functionality. This allows you to accomplish the same results as WebHelp Plus output—server-side search and searching of non-XHTML content.

If you want to take advantage of the advanced server-side features of HTML5 (i.e., automatic runtime project merging, server-side search, searching of non-XHTML files), you must enable HTML5 server-based output. This includes performing the following tasks: (1) installing Microsoft Internet Information Services (IIS) and ASP.NET, (2) setting up the HTML5 target and generating/publishing, (3) configuring IIS on the production server, (4) starting Microsoft Indexing Service or Microsoft Windows Search (depending on the operating system), and (5) enabling HTML5 search.

It is not necessary to install these in order to simply build HTML5 server-based output. However, if you want to be able to view and display HTML5 server-based output, the following steps are necessary. You must perform these steps on the machine that will be hosting the published HTML5 server-based output. If you want to view the output on your local machine to test the results, then you also need to perform these steps on your local computer, except for the procedure "Configuring IIS on Production Server."

For a complete scenario that illustrates these and other HTML5 server-based output steps in a real-life situation, see "HTML5 Server-based Output Scenario" on page 345.

Installing IIS and ASP.NET

The following steps show you how to install IIS and ASP.NET, depending on the operating system.

WINDOWS 8

The following steps show you how to install IIS and ASP.NET 4.5 for Windows 8.

INSTALLING IIS

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Select **Programs and Features**.
3. Click **Turn Windows features on or off**.
4. Select **Internet Information Services**.
5. Expand **Internet Information Services>Web Management Tools>IIS 6 Management Compatibility**.
6. Click **IIS Metabase and IIS 6 configuration compatibility**.

INSTALLING ASP.NET 4.5

7. Under **Internet Information Services** expand **World Wide Web Services**.
8. Expand **Application Development Features**.
9. Click the check box next to **ASP.NET 4.5** to add a check mark.
10. Click **OK**.

WINDOWS 7

The following steps show you how to install IIS and ASP.NET for Windows 7.

INSTALLING IIS

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Select **Programs and Features**.
3. Click **Turn Windows features on or off**.
4. Select **Internet Information Services**.
5. Expand **Internet Information Services>Web Management Tools>IIS 6 Management Compatibility**.
6. Click **IIS 6 Metabase and IIS 6 configuration compatibility**.

INSTALLING ASP.NET

7. Under **Internet Information Services** expand **World Wide Web Services**.
8. Expand **Application Development Features**.
9. Click the check box next to **ASP.NET** to add a check mark.
10. Click **OK**.
11. Click **Start**.
12. In the search field enter `run`.
13. Click **Run**. The Run dialog opens.
14. In the **Open** field, copy and paste one of the following into the field, depending on whether you have a 32-bit or 64-bit system:

32-BIT

```
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis -i
```

64-BIT

```
C:\WINDOWS\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis -i
```

15. Click **OK**.

A window opens, displaying the installation progress of ASP.NET. The window will close automatically when the installation finishes.

WINDOWS SERVER 2012

The following steps show you how to install IIS and ASP.NET 4.5 for Windows Server 2012.

INSTALLING IIS

1. Open the Server Manager.
2. Click **Add roles and features**. The Add Roles and Features Wizard opens.
3. Click **Next** two times until you get to the Server Roles page.
4. Click the check box next to **Web Server (IIS)**.
5. In the dialog that opens click **Add Features**.

INSTALLING ASP.NET 4.5

6. In the Add Roles Wizard click **Next**.
7. Expand **.NET Framework 4.5 Features** and click **ASP.NET 4.5**.
8. Click **Next**.
9. At the Web Server Role (IIS) tab, click **Next** again.
10. Expand **Management Tools>IIS 6 Management Compatibility** and click **IIS 6 Metabase Compatibility**.
11. On the Confirm Installation Selections page of the wizard, click **Install**.
12. After the installation is completed, the Installation Results page opens. Click **Close**.

WINDOWS SERVER 2008 R2

The following steps show you how to install IIS and ASP.NET for Windows Server 2008 R2.

INSTALLING IIS

1. From the Start menu open the Server Manager dialog.
2. Click **Add roles**. The Add Roles Wizard opens.
3. Click **Next**.
4. Click the check box next to **Web Server (IIS)**.
5. In the dialog that opens click **Add Required Features**.
6. In the Add Roles Wizard click **Next**.
7. Click **Next** again.
8. On the "Select Role Services" page of the wizard, expand **Management Tools>IIS 6 Management Compatibility** and click **IIS 6 Metabase Compatibility**.
9. On the Confirm Installation Selections page of the wizard, click **Install**.
10. After the installation is completed, click **Finish**.

INSTALLING ASP.NET

1. In the Server Manager dialog, expand **Roles**.
2. Select **Web Server IIS**.
3. In the **Role Services** section, click **Add Role Services**.
4. Click the check box next to **ASP.NET** to add a check mark.
5. Click **Next**.
6. Click **Install**.
7. Click **Close**.
8. In the search field enter `run`.
9. Click **Run**. The Run dialog opens.

10. In the **Open** field, copy and paste one of the following into the field, depending on whether you have a 32-bit or 64-bit system:

32-BIT

```
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis -i
```

64-BIT

```
C:\WINDOWS\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis -i
```

11. Click **OK**.

A window opens, displaying the installation progress of ASP.NET. The window will close automatically when the installation finishes.

WINDOWS SERVER 2008

The following steps show you how to install IIS and ASP.NET for Windows Server 2008.

INSTALLING IIS

1. From the Start menu open the Server Manager dialog.
2. Click **Add roles**. The Add Roles Wizard opens.
3. Click **Next**.
4. Click the check box next to **Web Server (IIS)**.
5. In the dialog that opens click **Add Required Features**.
6. In the Add Roles Wizard click **Next**.
7. Click **Next** again.
8. On the "Select Role Services" page of the wizard, expand **Management Tools>IIS 6 Management Compatibility** and click **IIS 6 Metabase Compatibility**.
9. On the Confirm Installation Selections page of the wizard, click **Install**.
10. After the installation is completed, click **Finish**.

INSTALLING ASP.NET

1. In the Server Manager dialog, expand **Roles**.
2. Select **Web Server IIS**.
3. In the **Role Services** section, click **Add Role Services**.
4. Click the check box next to **ASP.NET** to add a check mark.
5. Click **Next**.
6. Click **Install**.
7. Click **Close**.

WINDOWS SERVER 2003

The following steps show you how to install IIS and ASP.NET for Windows Server 2003.

INSTALLING IIS VIA THE MANAGE YOUR SERVER DIALOG

1. From the Start menu open the Manage Your Server dialog.
2. Click **Add or remove a role**. The Configure Your Server Wizard opens.
3. Click **Next**.
4. Select **Application server (IIS, ASP.NET)** and click **Next**.
5. Click **Enable ASP.NET** and click **Next**. Your selections are summarized.
6. Click **Next**. The Windows Components Wizard opens, displaying the status of the installation.
7. After the installation is completed, click **Finish**.

INSTALLING IIS VIA THE CONTROL PANEL

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Double-click **Add or Remove Programs**. The Add or Remove Programs window opens.
3. Click **Add/Remove Windows Components**. The Windows Components Wizard opens.
4. In the list of components select **Application Server** and then click **Details**. The Application Server Dialog opens.
5. Click the check box next to **Internet Information Services (IIS)** to add a check mark.
6. In the Windows Component Wizard, "Application Server" should now be selected. Click **Next**.
7. Click **Finish**.

INSTALLING ASP.NET

1. Click **Start>Run**. The Run dialog opens.
2. In the **Open** field, copy and paste one of the following into the field, depending on whether you have a 32-bit or 64-bit system:

32-BIT

```
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis -i
```

64-BIT

```
C:\WINDOWS\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis -i
```

3. Click **OK**.

A window opens, displaying the installation progress of ASP.NET. The window will close automatically when the installation finishes.



Note: ASP.NET pages by default are prohibited in IIS. Therefore, you must set these pages to be allowed if you are running Windows Server 2003. To do this: (1) In Internet Information Services (IIS) Manager, select **Web Service Extensions** from the left pane; (2) in the right pane, make sure that **ASP.NET v4.0.30319** is set to **Allowed**. If it is not, select it and click the **Allow** button.

WINDOWS XP

The following steps show you how to install IIS and ASP.NET for Windows XP.

INSTALLING IIS

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Double-click **Add or Remove Programs**. The Add or Remove Programs window opens.
3. Click **Add/Remove Windows Components**. The Windows Components Wizard opens.
4. In the list of components select **Application Server** and then click **Details**.



Note: The options in step 4 do not apply to the process for installing IIS on Windows XP Professional. If you are installing IIS on Windows XP Professional, skip step 4 and move on to step 5.

5. Click the check box next to **Internet Information Services (IIS)** to add a check mark.
6. In the Windows Component Wizard, "Application Server" should now be selected. Click **Next**.
7. Click **Finish**.

INSTALLING ASP.NET

1. Click **Start>Run**. The Run dialog opens.
2. In the **Open** field, copy and paste one of the following into the field, depending on whether you have a 32-bit or 64-bit system:

32-BIT

```
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis -i
```

64-BIT

```
C:\WINDOWS\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis -i
```

3. Click **OK**. A window opens, displaying the installation progress of ASP.NET. The window will close automatically when the installation finishes.

WINDOWS VISTA

The following steps show you how to install IIS and ASP.NET for Windows Vista.

INSTALLING IIS

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Double-click **Programs and Features**.
3. Click **Turn Windows features on or off**.
4. Select **Internet Information Services**.
5. Expand **Internet Information Services>Web Management Tools>IIS 6 Management Compatibility**
6. Click **IIS 6 Metabase and IIS 6 configuration compatibility**.

INSTALLING ASP.NET

7. Under **Internet Information Services** expand **World Wide Web Services**.
8. Expand **Application Development Features**.
9. Click the check box next to **ASP.NET** to add a check mark.
10. Click **OK**.
11. Click **Start>Run**. The Run dialog opens.
12. In the **Open** field, copy and paste one of the following into the field, depending on whether you have a 32-bit or 64-bit system:

32-BIT

```
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis -i
```

64-BIT

```
C:\WINDOWS\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis -i
```

13. Click **OK**. A window opens, displaying the installation progress of ASP.NET. The window will close automatically when the installation finishes.

Setting Up an HTML5 Target

The following steps show you how to set up your HTML5 target for server-based output.


HOW TO SET UP THE HTML5 TARGET

1. In Flare, open the HTML5 target. The Target Editor opens.
2. Select the **Publishing** tab.
3. In the **Server-based Output** section, click the check box.
4. In the **Indexing Service Catalog Name** field, type the catalog that you are using for the output. In most cases, this will be `Web`, which is the default value. However, if you or someone in your company (e.g., network administrator) creates a custom catalog, you need to enter that name in the field.

What is a catalog? Microsoft Indexing Service stores all of its index information in catalogs. A catalog comprises index information and stored properties for a particular group of file system directories. When Indexing Service is installed with Windows XP, it automatically builds a catalog, called the System catalog, listing contents of all permanently attached disk drives. The System catalog contains an index for all documents except certain system and temporary files. If Internet Information Services (IIS) is installed, the Indexing Service also creates a web catalog, which contains an index of IIS, the default virtual server of the World Wide Web.



Note: This step is necessary only for Windows XP and Windows Server 2003.

5. Click  to save your work.
6. Generate the target.
7. Publish the output to any location on the web server. If you want to use the default location that was created after you installed IIS, you can publish the output to `C:\inetpub\wwwroot`. If you do this, you can use the "Default Web Site" folder that you will encounter when performing the next set of steps (i.e., configuring IIS on the production server).

If you are working on your local machine for testing purposes, you do not need to publish the output. You can simply view your WebHelp Plus output. When you view HTML5 output on your local computer, you need to create a special folder called "MCPreview" within your "`C:\inetpub\wwwroot`" folder. Place a copy of your HTML5 output files in it. This enables you to test the advanced features

of HTML5 on your local machine. (For Windows Vista users testing locally, the physical location is C:\MadCap\[Name of Project]\[Name of Main Entry File].htm.)

Configuring IIS on Production Server

The following steps show you how to configure IIS, depending on the operating system. These steps are necessary only for the server where you will be publishing the final output. It is not necessary to perform these steps on your local computer for testing HTML5 server-based output.

WINDOWS SERVER 2003 AND WINDOWS XP

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Double-click **Administrative Tools**. The Administrative Tools dialog opens.
3. Double-click **Internet Information Services**. The Internet Information Services dialog opens.
4. Find your output folder and right-click the **Service** subfolder. Then select **Properties**. The Service Properties dialog opens.
5. Select the **Virtual Directory** tab.
6. Next to the **Application name** field, click the **Create** button.



Note: If this button has already been selected, it will display as "Remove" instead. In this case, you do not need to click the button.

7. In the **Execute Permissions** drop-down, make sure **Scripts only** is selected. It should already be selected by default.
8. Click **OK**.

WINDOWS SERVER 2008 AND 2008 R2, WINDOWS 2012, WINDOWS 7, WINDOWS 8, AND WINDOWS VISTA

1. Open the Control Panel.
2. Select **System and Security**.
3. Select **Administrative Tools**. The various operating systems have different ways to find this option.
4. Open the Internet Information Services (IIS) Manager.
5. Expand the node with the computer name.
6. Expand the **Sites** folder.

7. Expand the website folder corresponding to the one that you published.
8. Find your output folder and right-click the **Service** subfolder. Then select **Convert to Application**.
9. Make sure the application pool is using the .NET 4 framework.
 - a. In the dialog click **Select**.
 - b. From the drop-down select an application pool that is using the .NET 4 framework.
 - c. Click **OK**.
10. Click **OK**.

Starting Microsoft Indexing Service

The following steps show you how to start the Microsoft Indexing Service on machines that are running Windows XP and Windows Server 2003. This procedure is not necessary for the other operating systems.

WINDOWS SERVER 2003 AND WINDOWS XP—HOW TO START MICROSOFT INDEXING SERVICE

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Double-click **Administrative Tools**. The Administrative Tools dialog opens.
3. Double-click **Computer Management**. The Computer Management dialog opens.
4. Expand **Services and Applications**.
5. Right-click **Indexing Service**.
6. In the context menu, click **Start**. The Indexing Service is now started.

Starting Microsoft Windows Search

The following steps show you how to start the Microsoft Windows Search on machines that are running Windows Server 2008. This procedure is not necessary for the other operating systems.

WINDOWS SERVER 2008—HOW TO START MICROSOFT WINDOWS SEARCH

1. From the Start menu open the Server Manager dialog.
2. Click **Add roles**. The Add Roles Wizard opens.
3. Click **Next**.
4. Click the check box next to **Files Services**.
5. Click **Next** twice.
6. Click the check box next to **Windows Search Service**.
7. Click **Next** twice.
8. Click **Install**.
9. Click **Close**.

Enabling HTML5 Server-based Search

The following steps show you how to enable HTML5 server-based search. These steps must be done for each catalog that you are using on the server (or on your local machine, if you are viewing HTML5 server-based output on your computer for testing purposes).

HOW TO ENABLE HTML5 SERVER-BASED SEARCH

1. Navigate to the output folder for the HTML5 target on the server or on your local machine.
If you are enabling HTML5 server-based output on your local machine, you can quickly find the output folder by doing the following.
 - a. Open the Project Organizer.
 - b. Double-click the **Targets** folder.
 - c. Right-click on the target and select **Open Output Folder**.
2. In the output folder, double-click the **Service** folder.
3. Double-click the folder labeled **Console.ConfigureSearch**.
4. Double-click **ConfigureSearch.exe**. A window appears very briefly and then disappears.
5. Create a folder called "AutoMergeCache" at the root of the site. Then you must set security preferences on the AutoMergeCache folder so the application can create and update files.
 - a. Right-click on the root folder and choose **Explore** to open Windows Explorer.
 - b. Right-click on **AutoMergeCache** and choose **Properties**.
 - c. On the **Security** tab click the **Edit** button.
 - d. Click **Add**.
 - i. Type `Everyone` and click **Check Names**, making sure it gets underlined.
 - ii. Click **OK**.
 - e. Make sure **Everyone** is highlighted and check the option for **Full Control**.
 - f. Click **Apply**.
 - g. Click **OK** to exit the Permission dialog.
 - h. Click **OK** to exit the Properties dialog.

Testing HTML5 Server-based Search

The following steps are optional for testing HTML5 server-based search (for Microsoft Windows Search).

HOW TO TEST SEARCH

1. In IIS right-click on the directory where your HTML5 project is published to and choose **Explore**.
2. In the upper-right corner of the window you will see a Search input box. Perform a search for a term that is commonly found in your project.
3. If you see results, your project is ready to go live.

Viewing HTML5 Server-based Output

You can view HTML5 server-based output on your local machine for testing purposes, and you can view the output on the server where the files are published (where users access them). Before viewing the output, however, you must first follow all of the steps for enabling HTML5 server-based output. See "Enabling HTML5 Server-based Output" on page 306.

Basics of Viewing HTML5 Server-Based Output

When you view HTML5 output on your local computer, you need to create a special folder called "MCPreview" within your "C:\inetpub\wwwroot" folder. Place a copy of your HTML5 output files in it. This enables you to test the advanced features of HTML5 on your local machine. (For Windows Vista users testing locally, the physical location is C:\MadCap\[Name of Project]\[Name of Main Entry File].htm.)

When you test the output on your local computer, the output initially needs to be viewed by using the Flare interface (in order to generate the MCPreview folder). After that, you can view the output by using the Flare interface or by opening the file copies from the MCPreview folder. It is recommended that you use the MCPreview folder for subsequent viewings of WebHelp Plus output, rather than the Flare interface.

For a complete scenario that illustrates these and other HTML5 server-based output steps in a real-life situation, see "HTML5 Server-based Output Scenario" on page 345.

If you want to test HTML5 server-based output on your local computer, the advanced search features of HTML5 are not operable.

Testing Output Locally—Windows Vista and Windows 7

Windows Vista and Windows 7 include a feature called User Access Control (UAC). By default, this feature is enabled in Windows Vista and Windows 7, which means that all users, including administrators, run programs in an under-privileged state. It is not possible to view for testing on your local computer if Windows Vista or Windows 7 is running in this under-privileged state. If this occurs, you have two options: (1) disable UAC on the local machine, or (2) run Flare with elevated permission without globally disabling UAC.

OPTION 1—DISABLE UAC ON THE LOCAL MACHINE

1. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
2. Select **User Accounts** (select it twice if you are in Category View).
3. (Windows Vista) Click **Turn User Account Control on or off**. Click the check box to remove the check mark. Click **OK**.

(Windows 7) Click **Change User Account Control Settings** and then move the slider to **Never notify** and click **OK**.

4. Restart the computer.

UAC is now disabled. However, in order to preview HTML5 server-based output, you must be part of the Administrator's group, because Standard users cannot preview the output. Check with your system administrator to determine if you are in this group.

OPTION 2—RUN FLARE WITH ELEVATED PERMISSION WITHOUT GLOBALLY DISABLING UAC

1. Make sure you are part of the Administrator's group. Check with your system administrator to determine if you are in this group.
2. In Windows, navigate to the Flare executable file (C:\Program Files\MadCap Software\MadCap Flare V11\Flare.app\Flare.exe).
3. Set yourself as an administrator for Flare (either temporarily or permanently).

TO RUN ONLY A SINGLE INSTANCE OF FLARE AS AN ADMINISTRATOR

- a. Right-click on the executable file.
- b. Select **Run as administrator**.

TO ALWAYS RUN FLARE AS AN ADMINISTRATOR

- a. Right-click on the executable file.
- b. Select **Properties**.
- c. Select the **Compatibility** tab.
- d. In the **Privilege Level** area, click in the check box labeled **Run this program as an administrator**.

Procedures for Viewing Output

Use the following procedures to view HTML5 server-based output on the production server (where you publish your final output files) and on your local computer (for testing purposes).

HOW TO VIEW OUTPUT ON THE PRODUCTION SERVER

1. Open a browser window on any computer.
2. In the address field, type the URL to the main entry file of your output files (where you published them).

Unless you change the name of the output file (on the General tab of the Target Editor), the name of the main output file is Default.htm.

E X A M P L E

`http://www.mycompanysite/help/myHTML5target/Default.htm`

3. Press **Enter**.

HOW TO VIEW OUTPUT ON THE LOCAL MACHINE FOR TESTING (INITIAL VIEWING)

After generating the output, do one of the following using the Flare interface.

- » In the message that displays immediately at the end of the compilation process, click **Yes**.
- OR
- » Open the Target Editor and in the local toolbar click **View**.

HOW TO VIEW OUTPUT ON THE LOCAL COMPUTER (SUBSEQUENT VIEWINGS)

You can follow the same steps above for the initial viewing (using the Flare interface), or you can do the following to quickly access the files from the MCPreview folder.

1. Open a browser window.
2. In the address field, type the following:

```
http://localhost/MCPreview/[Name of Main Entry File].htm
```

For example: `http://localhost/MCPreview/Default.htm`



Note: The above address pertains to Windows XP and Windows Server 2003 only. If you are testing on a computer running Windows Vista, use the following instead: `http://localhost:70/[Name of Project]/[Name of Main Entry File].htm`. The portion `":70"` refers to the port number. For example, if testing on Vista, you might enter `http://localhost:70/MyProject/Default.htm`.

3. Press **Enter**.



Note: If you are testing HTML5 server-based output on your local machine, you may need to wait a few minutes after viewing the output for the Indexing Service to fully scan your files. Otherwise, you may not immediately see the effects of the scan (e.g., searches of non-XHTML files, incorporation of merged output files) in the output. If you avoid performing other tasks during this period, the scanning of the files will be completed more quickly.



Note: Flare's HTML5 Top Navigation skin does not support project merging.

Merging Output at Runtime Using HTML5

This is an easy way to merge the output from multiple HTML5 server-based Flare targets into one Help system. These targets can be originated from the same Flare project or from different Flare projects. You simply place the output files in the correct location on the server (i.e., within your master project's AutoMerge folder). Flare then automatically merges the output from all of the targets when users access the Help. From the end user's perspective, the results are seamless, appearing as one large Help system. All of the TOCs, browse sequences, indexes, glossaries, and search capabilities for the projects are merged.

Tasks Associated with Merging Output Using HTML5

Following are basic tasks involved with merging output using HTML5.

HOW TO MERGE OUTPUT AT RUNTIME USING HTML5

1. **Enable HTML5 Server-based Output** If you want to take advantage of the advanced server-side features of HTML5 (i.e., automatic runtime project merging, server-side search, searching of non-XHTML files), you must enable HTML5 server-based output. This includes performing the following tasks: (1) installing Microsoft Internet Information Services (IIS) and ASP.NET, (2) setting up the HTML5 target and generating/publishing, (3) configuring IIS on the production server, (4) starting Microsoft Indexing Service or Microsoft Windows Search (depending on the operating system), and (5) enabling HTML5 search. See "Enabling HTML5 Server-based Output" on page 306.
2. **Determine Master Output** Decide which of your project outputs will serve as the "master." This is the main output that users will open. All other outputs will be accessed from that output, although it will appear as one large Help system to end users.
3. **(Optional) Specify TOC and/or Browse Sequence Locations for Automerger** By default, the TOCs and browse sequences from the secondary outputs will be appended at the end of the master output's TOC and browse sequence. However, if you want them to be appended somewhere within the master output's TOC or browse sequence, you can specify the exact location. See "Specifying the Automerger Location in a TOC for HTML5" on page 334 and "Specifying the Automerger Location in a Browse Sequence for HTML5" on page 336.

4. **(Optional) Specify Order of Merged Outputs** When you automerger HTML5 outputs, the secondary outputs are merged to the master project's table of contents (TOC) and/or browse sequence in alphabetical order. However, you can override this default configuration and merge the secondary outputs in any order that you like. See "Specifying the Order of Automerger HTML5 Outputs" on page 338.
5. **Generate Outputs** Build the output for each of the targets to be included in the automerger. You must use HTML5 for all of the targets.
6. **Publish Output to Server** Publish the output for the master project to the web server running Microsoft IIS. The output files for the secondary targets need to be published to the "AutoMerge" subfolder located in the master target's main output folder (e.g., <http://www.xyzincorporated.com/help systems/MyTargetName/AutoMerge>).

When readers open the online output, they will see all of the HTML5 outputs merged into one Help system. They can easily navigate and use the index and search across all of the output files.



Note: If you are testing HTML5 server-based output on your local machine, you need view the output at least one time. When you view HTML5 output on your local computer, you need to create a special folder called "MCPreview" within your "C:\inetpub\wwwroot" folder. Place a copy of your HTML5 output files in it. This enables you to test the advanced features of HTML5 on your local machine. (For Windows Vista users testing locally, the physical location is C:\MadCap\[Name of Project]\[Name of Main Entry File].htm.) When testing the automerger feature on your computer, you need to place the secondary ("child") outputs in the AutoMerge subfolder at this location (as opposed to the output folder that was generated where your Flare project is located).



Note: If you are testing HTML5 server-based output on your local machine, you may need to wait a few minutes after viewing the output for the Indexing Service to fully scan your files. Otherwise, you may not immediately see the effects of the scan (e.g., searches of non-XHTML files, incorporation of merged output files) in the output. If you avoid performing other tasks during this period, the scanning of the files will be completed more quickly.



Note: If you want to test HTML5 server-based output on your local computer, the advanced search features of HTML5 are not operable.





Note: Flare's HTML5 Top Navigation skin does not support project merging.


Specifying the Automerger Location in a TOC for HTML5

You can determine where other Flare project outputs are merged relative to your "master" project's TOC if you are generating HTML5 server-based output and you are publishing the files to a web server running Microsoft IIS.

By default, the other HTML5 server-based outputs will be merged at the end of your master project's TOC. However, you can use the following steps to select one of the available options to override this placement.

HOW TO SPECIFY THE AUTOMERGE LOCATION IN A TOC

1. Open the TOC in the Flare project that will serve as the parent project.
2. Do one of the following:
 - » If you want to merge the other outputs in relation to one of the existing entries in the TOC (e.g., before it, after it), select that entry (whether it is an individual item or a book).OR
 - » If you want to merge the other outputs at the location of an entry that is not linked to any other file, and you want to provide a label to indicate the location of the merge, create a new TOC item. To do this:
 - a. Place your cursor in the TOC where you want to add the new item.
 - b. Click .
 - c. Press **F2**.
 - d. Replace the default text with new text.
 - e. Press **Enter**.
 - f. If necessary, use the arrow buttons in the local toolbar to position the new entry in the TOC.
3. In the local toolbar of the TOC Editor, click . The Properties dialog opens.
4. Select the **Advanced** tab.
5. In the **Server-based Automerger field**, select one of the following:
 - » **Before** The automerger will occur immediately before the selected TOC entry.
 - » **After** The automerger will occur immediately after the selected TOC entry.



- » **First Child** The automerge will occur at the first location directly after the selected TOC book (i.e., before any other entries within the book). If you use this option on a simple TOC entry instead of a book, the entry will automatically become a book once the outputs are automerged.
 - » **Last Child** The automerge will occur at the last location after the selected TOC book (after the last entry within the book). If you use this option on a simple TOC entry instead of a book, the entry will automatically become a book once the outputs are automerged.
 - » **Replace** The automerge will occur at the location of the TOC entry where you have specified this option. It would replace any links that might otherwise be applied to that entry. You might use this option, for example, if you want to create a new entry in the master TOC so that you can add a label at the point where the automerge occurs.
6. In the Properties dialog, click **OK**.
 7. Click  to save your work.


Specifying the Automerger Location in a Browse Sequence for HTML5

You can determine where other Flare project outputs are merged relative to your "master" project's browse sequence if you are generating HTML5 server-based output and you are publishing the files to a web server running Microsoft IIS.

By default, the other HTML5 server-based outputs will be merged at the end of your master project's browse sequence. However, you can use the following steps to select one of the available options to override this placement.

HOW TO SPECIFY THE AUTOMERGE LOCATION IN A BROWSE SEQUENCE

1. Open the browse sequence in the Flare project that will serve as the "master" project.
2. Do one of the following:
 - » If you want to merge the other outputs in relation to one of the existing entries in the browse sequence (e.g., before it, after it), select that entry (whether it is an individual item or a book).OR
 - » If you want to merge the other outputs at the location of an entry that is not linked to any other file, and you want to provide a label to indicate the location of the merge, create a new browse sequence item. To do this:
 - a. Place your cursor in the browse sequence where you want to add the new item.
 - b. Click .
 - c. Press **F2**.
 - d. Replace the default text with new text.
 - e. Press **Enter**.
 - f. If necessary, use the arrow buttons in the local toolbar to position the new entry in the browse sequence.
3. In the local toolbar of the Browse Sequence Editor, click . The Properties dialog opens.
4. Select the **Advanced** tab.
5. In the **Server-based Automerger field**, select one of the following:
 - » **Before** The automerger will occur immediately before the selected browse sequence entry.
 - » **After** The automerger will occur immediately after the selected browse sequence entry.

- » **First Child** The automerge will occur at the first location directly after the selected browse sequence book (i.e., before any other entries within the book). If you use this option on a simple browse sequence entry instead of a book, the entry will automatically become a book once the outputs are automerged.
 - » **Last Child** The automerge will occur at the last location after the selected browse sequence book (after the last entry within the book). If you use this option on a simple browse sequence entry instead of a book, the entry will automatically become a book once the outputs are automerged.
 - » **Replace** The automerge will occur at the location of the browse sequence entry where you have specified this option. It would replace any links that might otherwise be applied to that entry. You might use this option, for example, if you want to create a new entry in the master browse sequence so that you can add a label at the point where the automerge occurs.
6. In the Properties dialog, click **OK**.
 7. Click  to save your work.

Specifying the Order of Automerged HTML5 Outputs

When you automerge HTML5 outputs, the secondary outputs are merged to the master project's table of contents (TOC) and/or browse sequence in alphabetical order. However, you can override this default configuration and merge the secondary outputs in any order that you like. See "Merging Output at Runtime Using HTML5" on page 331.

EXAMPLE

Let's say you have a master project with a target named "Main" and three smaller projects with targets named "First Child," "Second Child," and "Third Child," respectively. If you place the three smaller project outputs in the AutoMerge subfolder where "Main" is published, their TOCs will be appended to the master project's TOC in the following order:

1. First Child
2. Second Child
3. Third Child

But what if you want "Third Child" to be appended first? You can use the steps in this topic to position it above the other targets. Therefore, the result would be:

1. Third Child
2. First Child
3. Second Child

Now let's say that you add two more targets (called "Final Child" and "Another Child") to the AutoMerge folder. If you do not adjust your custom order, Flare will automerge these outputs in alphabetical order at the end of your custom order. Therefore, the final result would be:

1. Third Child
2. First Child
3. Second Child

4. Another Child

5. Final Child

HOW TO SPECIFY THE ORDER OF AUTOMERGED OUTPUTS

1. Create an XML file and name it SortOrder.xml. You can do this by opening an editor such as Notepad. When you save the file, type the xml file extension at the end of the name.
2. Enter the following code into the blank file.

```
<?xml version="1.0" encoding="utf-8"?>

<SortOrder>

<Item>ProjectA</Item>

<Item>ProjectB</Item>

<Item>ProjectC</Item>

</SortOrder>
```

3. Replace "ProjectA," "ProjectB," and "ProjectC" with the names of your own child targets. You can add or remove line items as necessary.

If you were to use the example at the top of this topic, you would enter the following.

```
<?xml version="1.0" encoding="utf-8"?>

<SortOrder>

<Item>Third Child</Item>

<Item>First Child</Item>

<Item>Second Child</Item>

</SortOrder>
```

4. Save the file.
5. Copy and paste the SortOrder.xml file into the AutoMerge directory of your published HTML5 output. This file sits beside the child output folders that you are using for the automerge.

Including Non-XHTML Files in HTML5 Search

When end users perform a search in your online output, you can ensure that non-XHTML files (e.g. PDF, DOC, XLS) are included in that search. The files do not even need to be linked to any of the content in your Flare project. This feature is available if you publish HTML5 server-based output to a web server running Microsoft Internet Information Services (IIS).

For a complete scenario that illustrates these and other HTML5 server-based output steps in a real-life situation, see "HTML5 Server-based Output Scenario" on page 345.

HOW TO INCLUDE NON-XHTML FILES IN SEARCH

1. **Enable HTML5 Server-based Output** If you want to take advantage of the advanced server-side features of HTML5 (i.e., automatic runtime project merging, server-side search, searching of non-XHTML files), you must enable HTML5 server-based output. This includes performing the following tasks: (1) installing Microsoft Internet Information Services (IIS) and ASP.NET, (2) setting up the HTML5 target and generating/publishing, (3) configuring IIS on the production server, (4) starting Microsoft Indexing Service or Microsoft Windows Search (depending on the operating system), and (5) enabling HTML5 search. See "Enabling HTML5 Server-based Output" on page 306.
2. **Generate HTML5 Server-based Output** In Flare, generate a target using the WebHelp Plus output format.
3. **Publish HTML5 Output to Server** Publish the HTML5 server-based output to your Microsoft IIS Web server.
4. **(Optional) Add IFilter** File types that are automatically supported for HTML5 server-based search include: Microsoft Office files, HTML files, and TXT files. For other non-supported file types, such as PDF documents, you can download and install a simple IFilter in order to include that file type in the search. To find and download the IFilter for the file type that you want to include, go to a website that provides these (such as <http://www.ifilter.org>) and follow the instructions for downloading IFilters. You must install the appropriate IFilters on the server where the output files will be published, and if you are testing HTML5 server-based output on your computer, you need to install the IFilter(s) locally as well.
5. **Copy and Paste Non-XHTML Files to Appropriate Folder** Copy all of the non-XHTML files that you want to be included in the search. Then paste them in the appropriate subfolder of the HTML5 server-based output that you published. By default, HTML5 server-based will find any files located in the project output folder, as well as the "AutoSearch" subfolder. Therefore, you can paste your extra non-XHTML files in the AutoSearch subfolder.

EXAMPLE

<http://www.xyzincorporated.com/help systems/MyTargetName/AutoSearch>.

However, if there are any other folders where you want either XHTML or non-XHTML files to be stored (instead of, or in addition to, AutoSearch), you can perform some extra steps to allow this.

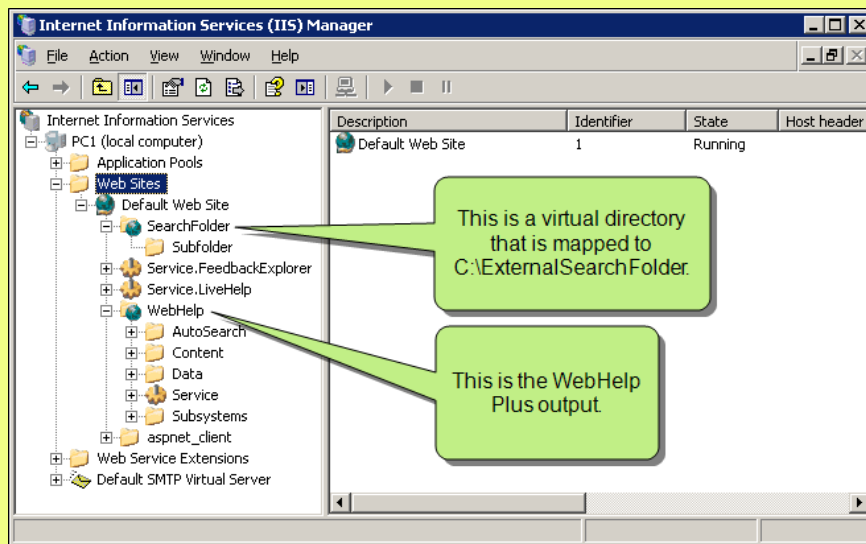
Let's say you also want searches in the Help system to provide results from files in the folder C:\ExternalSearchFolder. Here are the steps to allow this:

1. This folder must be accessible by IIS. Therefore, create a virtual directory in IIS for it.

Following are steps for doing this.

- a. Open the Control Panel. On many computers, you can do this by clicking **Start>Control Panel**.
- b. Double-click **Administrative Tools**. The Administrative Tools dialog opens.
- c. Double-click **Internet Information Services**. The Internet Information Services (IIS) Manager dialog opens.
- d. Expand the **Web Sites** folder.
- e. Right-click on the **Default Web Site** folder and select **New>Virtual Directory**.
- f. Click **Next**.
- g. In the **Alias** field, enter a name for the subfolder (e.g., SearchFolder).
- h. Click **Next**.
- i. Click the **Browse** button, then find and select the folder holding the content that you want to publish (e.g., C:\ExternalSearchFolder).

- j. Click **Next**.
- k. Make sure a check mark is next to **Read**.
- l. Click **Next**.
- m. Click **Finish**.



2. Create an XML file called "SearchFolders.xml" and place it in the "AutoSearch" folder in the HTML5 server-based output directory. The contents of this file will specify any virtual directories that HTML5 should search. Following is what you would type in the SearchFolders.xml file. This example specifies just one folder, but you can add references to as many virtual directories as you need.

```
<?xml version="1.0" encoding="utf-8"?>

<SearchFolders>

    <Url>/SearchFolder</Url>
```

</SearchFolders>

When users perform a search in your output, those non-XHTML files will also be scanned and become accessible to the users.



Note: If you are testing WebHelp Plus output on your local machine, you need to view the output at least one time. When you view HTML5 output on your local computer, you need to create a special folder called "MCPreview" within your "C:\inetpub\wwwroot" folder. Place a copy of your HTML5 output files in it. This enables you to test the advanced features of HTML5 on your local machine. (For Windows Vista users testing locally, the physical location is C:\MadCap\[Name of Project]\[Name of Main Entry File].htm.) When testing this search feature on your computer, you need to place the non-XHTML files in the AutoSearch subfolder at this location (as opposed to the output folder that was generated where your Flare project is located).



Note: If you are testing HTML5 server-based output on your local machine, you may need to wait a few minutes after viewing the output for the Indexing Service to fully scan your files. Otherwise, you may not immediately see the effects of the scan (e.g., searches of non-XHTML files, incorporation of merged output files) in the output. If you avoid performing other tasks during this period, the scanning of the files will be completed more quickly.



Note: If you want to test HTML5 server-based output on your local computer, the advanced search features of HTML5 are not operable.

HTML5 Server-based Output Scenario

The following scenario provides a real-life example of how someone might set up a local computer and server for producing HTML5 server-based output. For specific steps of the tasks represented in this scenario, see "Enabling HTML5 Server-based Output" on page 306 and "Viewing HTML5 Server-based Output" on page 326.

SCENARIO

Here is the situation: An author is running Windows XP on her computer. She has a Flare project with a target named "Jackpot" and plans to publish the output to the company website (xyzincorporated.com), using the HTML5 server-based format. In addition, the author has 20 Microsoft Word documents and 15 PDF files that she wants to be included in user searches. Finally, she has four other Flare projects and wants to merge them with the output from the Jackpot target, which will serve as the "parent" target. The author wants to publish the output to this directory on the server: <http://www.xyzincorporated.com/help> systems. But before she publishes the final output, she wants to test the output on her local computer. The following steps in this scenario require the completion of some tasks by the author and the completion of other tasks by the web administrator, who is responsible for the production server where the final output files will be placed.

1. **Author** On her local computer, the author completes the steps for installing Microsoft Internet Information Services (IIS) and ASP.NET.
See "Installing IIS and ASP.NET" on page 307.
2. **Author** In Flare, the author follows the steps for setting up the Jackpot target, which is based on the HTML5 server-based output type. This includes entering "Web" as the catalog name on the Publishing tab (we're assuming the intention is to use the default value, as opposed to a custom catalog).
See "Setting Up an HTML5 Target" on page 318.
3. **Author** The author generates the Jackpot target in Flare. The author then creates a folder called "MCPreview" on her computer at `C:\inetpub\wwwroot` (e.g., `C:\inetpub\wwwroot`) and places a copy of the Jackpot output files in that folder.

4. **Author** The author follows the steps for starting Microsoft Indexing Service.

See "Starting Microsoft Indexing Service" on page 322.

5. **Author** In order to enable the advanced HTML5 server-based search features on her computer, the author opens the output folder and double-clicks the file `ConfigureSearch.exe`, which is located in the output's `Service\Console.ConfigureSearch` subfolder. In the case of this author, she finds her executable file here:

```
C:\Documents and Settings\AuthorName\Documents\My Projects\JackpotProject\Output\authname\ Jack-  
pot\Service\Console.ConfigureSearch\ConfigureSearch.exe
```

See "Enabling HTML5 Server-based Search" on page 324.

6. **Author** The author generates her four "child" Flare targets (those that will be merged with the parent Jackpot target). In Flare, she selects **Project>Open Output Folder** and copies each output folder related to a child target. She then pastes these output folders in the following location:

```
C:\inetpub\wwwroot\MCPreview\AutoMerge
```

The author is careful not to view the output of any of the child targets from Flare immediately after she builds them. If she were to do this, the output files for the parent Jackpot project would be overwritten in the MCPreview folder.

You can also choose specific places in the table of contents or browse sequence where child projects are merged. See "Specifying the Order of Automerged HTML5 Outputs" on page 338, "Specifying the Automerge Location in a TOC for HTML5" on page 334, and "Specifying the Automerge Location in a Browse Sequence for HTML5" on page 336. In this scenario, let's assume the author simply uses the default merge order.

7. **Author** Now the author wants to add the Word and PDF files to the output, so that they can be included in searches. However, an initial step needs to be completed first. Certain files (Microsoft Office, text, and HTML) do not need any attention because they can be included in the search just the way they are. PDF documents, however, require an IFilter. Therefore, the author downloads an IFilter from Adobe and installs it on her computer.

See "Including Non-XHTML Files in HTML5 Search" on page 341.

8. **Author** The author copies and pastes the Word and PDF documents to:

C:\inetpub\wwwroot\MCPreview\AutoSearch

9. **Author** The author views the HTML5 server-based output on her computer to test the merged output and the search for Word and PDF content. However, this time she does not initiate the view from within Flare. This time, she opens a browser window and types the following path into the address bar:

<http://localhost/MCPreview/Default.htm>

She does this because if she had initiated the view from within Flare, the output files in the MCPreview folder would have been replaced with new files, which would have overwritten the merged child output, Word documents, and PDF files that were added manually.

See "Viewing HTML5 Server-based Output" on page 326.

10. **Author** Let's say the author is now finished testing her output and is ready to publish the final output to the server. Therefore, in Flare she creates a publishing destination for the parent Jackpot target that points to <http://www.xyzincorporated.com/help systems/Jackpot>. She generates the output for the parent Jackpot target again and publishes the Jackpot HTML5 server-based output to that location.

11. **Author/Web Administrator** The author generates the four child Flare target outputs again. In Flare, she selects **Project>Open Output Folder** and copies each output folder related to a child target. The author can provide these folders to the web administrator or (if the author has access) directly paste them in the following location:

<http://www.xyzincorporated.com/help systems/Jackpot/AutoMerge>

For example, if the author has child targets named "Child1," "Child2," "Child3," and "Child4," those output folders would be added to the website as follows:

<http://www.xyzincorporated.com/help systems/Jackpot/AutoMerge/Child1>

<http://www.xyzincorporated.com/help systems/Jackpot/AutoMerge/Child2>

<http://www.xyzincorporated.com/help systems/Jackpot/AutoMerge/Child3>

<http://www.xyzincorporated.com/help systems/Jackpot/AutoMerge/Child4>

Alternatively, the author (or web administrator) could manually create subfolders on the web-site that are named after each target folder; these subfolders would be contained within the AutoMerge folder. The author can then create publishing destinations that point to the appropriate subfolders and automatically publish each child target accordingly.

12. **Web Administrator** On the server where the output files will be published, the company's web administrator completes the steps for installing Microsoft Internet Information Services (IIS) and ASP.NET.

See "Installing IIS and ASP.NET" on page 307.

13. **Web Administrator** On the server where the output files will be published, the company's web administrator completes the steps for configuring IIS on the production server.

See "Configuring IIS on Production Server" on page 320.

14. **Web Administrator** On the server where the output files will be published, the company's web administrator completes the steps for starting Microsoft Indexing Service.

See "Starting Microsoft Indexing Service" on page 322.

15. **Web Administrator** The web administrator opens the output folder on the server to enable HTML5 search. He double-clicks the file ConfigureSearch.exe, which is located here:

<http://www.xyzincorporated.com/help systems/Jackpot/Service/Console.ConfigureSearch>

See "Enabling HTML5 Server-based Search" on page 324.

16. **Web Administrator** In order to include the PDF files in the search, the web administrator installs the appropriate IFilter on the server.

See "Including Non-XHTML Files in HTML5 Search" on page 341.

17. **Author/Web Administrator** The author (if she has access) or web administrator copies and pastes the 20 Word documents and 15 PDF files to:

<http://www.xyzincorporated.com/help systems/Jackpot/AutoSearch>

18. **Author** To view the published output, the author opens a browser window and types the following path into the address bar:

<http://www.xyzincorporated.com/help systems/Jackpot/Default.htm>

If the author had specified a custom file name for the Jackpot target, such as "JackpotOnline," the address instead would be:

<http://www.xyzincorporated.com/help systems/Jackpot/JackpotOnline.htm>



Note: If you are testing HTML5 server-based output on your local machine, you may need to wait a few minutes after viewing the output for the Indexing Service to fully scan your files. Otherwise, you may not immediately see the effects of the scan (e.g., searches of non-XHTML files, incorporation of merged output files) in the output. If you avoid performing other tasks during this period, the scanning of the files will be completed more quickly.

PDF Guides

The following PDF guides are available for download from the online Help:

Accessibility Guide

Analyzer Guide

Autonumbers Guide

Condition Tags Guide

Context-sensitive Help Guide

DotNet Help Guide

Eclipse Help Guide

Getting Started Guide

Global Project Linking Guide

HTML Help Guide

HTML5 Guide

Images Guide

Importing Guide

Index Guide

Key Features Guide

Language Support Guide

Movies Guide

Navigation Links Guide

Print-based Output Guide

Project Creation Guide

Pulse Guide

QR Codes Guide

Reports Guide

Reviews & Contributions Guide

Search Guide

SharePoint Guide

Shortcuts Guide

Skins Guide



Snippets Guide

Templates Guide

Source Control Guide: Git

Topics Guide

Source Control Guide: Perforce

Touring the Workspace Guide

Source Control Guide: Subversion

Transition From FrameMaker Guide

Source Control Guide: Team Foundation Server

Tutorials Guide: Product Foldout 3-Fold Template

Source Control Guide: Visual SourceSafe

Tutorials Guide: Top Navigation Adv Template

Styles Guide

Tutorials Guide: Tripane and PDF Adv Template

Tables Guide

Variables Guide

Tables of Contents Guide

WebHelp Outputs Guide

Targets Guide

What's New Guide

Output Type Comparison Tables

Following are tables that show the various output types available, with the distinguishing features of each.

This chapter discusses the following:

General	355
Accessibility	356
Context-sensitive Help	357
Feedback Statistics and Reporting Features	358
Generated Content	359
Glossaries	362
Images	363
Indexes	364
Language Support	365
Master Pages and Page Layouts	366
Merging Output	367
Miscellaneous Features	368
Multimedia	370
Navigation Links	374
Pulse (Socially Enabled Output)	375



Search	376
Skins	382
Tables of Contents and Mini-TOCs	385
Target Settings	386
Variables	388



Note: If an item does not have a check mark, that does not necessarily mean the feature is unavailable in the output. Rather, it means that the feature cannot be added in Flare. For example, search cannot be added in Flare for EPUB output, but some ebook readers may have search built in.









General

Following are general features supported in each output type:

	 Dot Net Help	 Eclp Help	 MS HTML Help	 HTML5	 Web Help	 Web Help Mobile	 Web Help Plus	 AIR
.NET Integration	✓	✗	✗	✗	✗	✗	✗	✗
Responsive Output	✗	✗	✗	✓	✗	✗	✗	✗
Single Output File Possible	✗	✓	✓	✗	✗	✗	✗	✓

Accessibility

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

								
Accessibility Supported	✓	✓	✓	✓	✓	✓	✓	✓
Compiler Warnings	✗	✓	✓	✓	✓	✓	✓	✓
Empty Alt Text for Images	✓	✓	✓	✓	✓	✓	✓	✓
Scrolling for Toolbars and Navigation Frames—Enable	✗	✗	✗	✗	✓	✗	✓	✗






Context-sensitive Help

Following are context-sensitive Help (CSH) features supported in each output type:

								
CSH Supported	✓	✓	✓	✓	✓	✓	✓	✗
Embedded CSH Supported	✓	✗	✗	✗	✗	✗	✗	✗

Feedback Statistics and Reporting Features

Following are Feedback statistics and reporting features supported in each output type:

								
Feedback Supported	✓	✗	✓	✓	✓	✗	✓	✗
Search Results	✓	✗	✗	✓	✓	✗	✓	✗









¹For HTML5, Feedback is supported only in Tripane output. It is not supported in Top Navigation or skinless outputs.

Generated Content

Following are generated content features supported in each output type:

								
Autonumbers ¹	✓	✓	✓	✓	✓	✓	✓	✓
Breadcrumbs	✓	✗	✓	✓	✓	✓	✓	✓
Browse Sequences	✓	✗	✓	✓	✓	✓	✓	✓
Concept Links	✓	✓	✓	✓	✓	✓	✓	✓
Keyword Links	✓	✓	✓	✓	✓	✓	✓	✓
Glossaries from Proxies	✗	✗	✗	✗	✗	✗	✗	✗
Indexes from Proxies	✗	✗	✗	✗	✗	✗	✗	✗
List of Concepts	✓	✓	✓	✓	✓	✓	✓	✓
List of Elements	✓	✓	✓	✓	✓	✓	✓	✓
List of Endnotes	✗	✗	✗	✗	✗	✗	✗	✗

								
Lists (Numbered and Bulleted)	✓	✓	✓	✓	✓	✓	✓	✓
Menus from Proxies	✗	✗	✗	✓	✗	✗	✗	✗
Mini-TOCs from Proxies	✓	✓	✓	✓	✓	✓	✓	✓
Page Numbers	✗	✗	✗	✗	✗	✗	✗	✗
Related Topics Links	✓	✓	✓	✓	✓	✓	✓	✓
Relationship Links	✓	✓	✓	✓	✓	✓	✓	✓
Scripts	✓	✓	✓	✓	✓	✓	✓	✓
Search Bars from Proxies	✗	✗	✗	✓ ²	✗	✗	✗	✗
Search Results from Proxies	✗	✗	✗	✓ ²	✗	✗	✗	✗
Shortcut Controls	✗	✗	✓	✗	✗	✗	✗	✗







								
TOCs from Proxies	✓	✓	✓	✓	✓	✓	✓	✓
Topic Toolbars from Proxies	✓	✗	✓	✓	✓	✓	✓	✓

¹It is possible to use autonumbering in online outputs, but only in a limited way. For example, if you have notes throughout your project, you can create an autonumber format on a paragraph style class that simply has the text "Note:" followed by a space. Then whenever you want to apply that style class to note content, it will automatically start with "Note:" so that you don't have to type it each time. However, in the traditional sense where autonumbering is used to increment numbers for elements such as chapters, figures, and tables, it is really only intended for print-based outputs.

²In HTML5, the proxies and related skin components for search bars and search results are supported only in Top Navigation output, as well as in targets not using a skin. They are not supported in Tripane output.

Glossaries

Following are glossary features supported in each output type:

								
Glossaries—Created via Glossary File and Proxy	✗	✓	✗	✗	✗	✗	✗	✗
Glossaries—Created via Glossary File and Skin Setting	✓	✗	✓	✓	✓	✓	✓	✓
Auto-generate	✗	✗	✗	✗	✗	✗	✗	✗

Images

Following are image features supported in each output type:

								
Absolute Positioning	✓	✓	✓	✓	✓	✓	✓	✓
Image Hyperlinks	✓	✓	✓	✓	✓	✓	✓	✓
Image Maps	✓	✓	✓	✓	✓	✓	✓	✓
Pre-Compile Resized	✓	✓	✓	✓	✓	✓	✓	✓
Thumbnail Images	✓	✓	✓	✓	✓	✓	✓	✓
Web-Safe—Convert from Non—Web-safe	✓	✓	✓	✓	✓	✓	✓	✓

Indexes

Following are index features supported in each output type:

								
Indexes—Created Using Keywords and Proxy	✗	✓	✗	✗	✗	✗	✗	✗
Indexes—Created Using Keywords and Skin Setting	✓	✗	✓	✓	✓	✓	✓	✓
Auto-generate	✗	✗	✗	✗	✗	✗	✗	✗
Binary Indexes	✗	✗	✓	✗	✗	✗	✗	✗
Bookmarks in Index Entries	✗	✗	✓	✗	✗	✗	✗	✗
Index Links	✓	✗	✓	✓	✓	✓	✓	✓
Search—Can Exclude Index Entries from Search	✓	✗	✗	✓	✓	✓	✓	✓

Language Support

Following are language support features supported in each output type:

								
Author and Generate Output in Left-to-Right (LTR) and Right-to-Left (RTL) Languages	✓	✓	✓	✓	✓	✓	✓	✓
Invert Image Callouts to RTL	✗	✓	✓	✓	✓	✓	✓	✓
Invert Page Layouts to RTL	✗	✗	✗	✗	✗	✗	✗	✗
Invert Styles to RTL	✗	✓	✓	✓	✓	✓	✓	✓
Output Interface—Display in LTR and RTL	✗	✓	✗	✓ ¹	✓ ²	✓ ²	✓ ²	✓ ²
Output Interface—Display English, French, Japanese, or German	✓ ³	✗	✗	✗	✗	✗	✗	✗
Language Skins	✓	✗	✓	✗	✓	✓	✓	✓
Separate UI Text tab in Skin Editor for Localization	✗	✗	✗	✓	✗	✗	✗	✗





¹This is controlled on the UI Text tab in a regular HTML5 skin.

²This is controlled in a language skin.

³This is possible with the MadCap Help Viewer, where the end user selects the language.


Master Pages and Page Layouts

Following are master page and page layout features supported in each output type:

								
Master Pages Supported	✓	✓	✓	✓	✓	✓	✓	✓
Page Layouts Supported	✗	✓	✗	✗	✗	✗	✗	✗

Merging Output






Following are merging output features supported in each output type:

								
Merge Output Supported	✓	✓	✓	✓ ^{1,2}	✓	✓	✓	✗
Merge Output at Runtime	✗	✗	✗	✓ ¹	✗	✗	✓	✗
¹ Flare's HTML5 Top Navigation skin does not support project merging. ² HTML5 outputs do not support pre-merging.								

Miscellaneous Features

Following are miscellaneous features supported in each output type:

								
Augmented Reality	✓	✓	✓	✓	✓	✓	✓	✓
Equations	✓	✓	✓	✓	✓	✓	✓	✓
Footnotes	✓ ¹	✓	✓ ¹	✓ ¹	✓ ¹	✓ ¹	✓ ¹	✓ ¹
Horizontal Rules	✓	✓	✓	✓	✓	✓	✓	✓
Object Positioning	✓	✓	✓	✓	✓	✓	✓	✓
PDF Stitching	✓	✓	✓	✓	✓	✓	✓	✓
QR Codes	✓	✓	✓	✓	✓	✓	✓	✓
Preview Topics in Output Type	✓	✓	✓	✓	✓	✓	✓	✓
Redacted Text	✗	✗	✗	✗	✗	✗	✗	✗
Slideshows	✓	✗	✓	✓	✓	✓	✓	✓
Snippets	✓	✓	✓	✓	✓	✓	✓	✓
Tables	✓	✓	✓	✓	✓	✓	✓	✓

								
Text Boxes	✓	✓	✓	✓	✓	✓	✓	✓
¹ Footnotes are converted to popups.								









Multimedia

Following are multimedia features supported in each output type:

								
3D Models (U3D Files)	✓	✓	✓ ²	✓	✓	✓	✓	✓
Audio and Video Files								
ASF	✓	✓	✓	✓	✓	✓	✓	✗
ASX	✓	✓	✓	✓	✓	✓	✓	✗
AU	✓	✓	✓	✓	✓	✓	✓	✗
AVI ³	✓ ⁴	✓	✓	✓	✓	✓	✓	✗
M4V ³	✓ ⁴	✓	✗	✓	✓	✓	✓	✗
MID	✓	✓	✓	✓	✓	✓	✓	✗
MIDI	✓	✓	✓	✓	✓	✓	✓	✗
MP3	✓	✓	✓	✓	✓	✓	✓	✗
MP4 ³	✓ ⁴	✓	✗	✓	✓	✓	✓	✗

								
MPA	✓	✓	✓	✓	✓	✓	✓	✗
MPE	✓	✓	✓	✓	✓	✓	✓	✗
MPEG	✓	✓	✓	✓	✓	✓	✓	✗
MPG	✓	✓	✓	✓	✓	✓	✓	✗
OGG ³	✓ ⁴	✗	✗	✓	✗	✗	✗	✗
OGV ³	✓ ⁴	✗	✗	✓	✗	✗	✗	✗
OPUS	✗	✗	✗	✓ ⁵	✓ ⁵	✗	✗	✗
SWF	✓	✓	✓	✓	✓	✓	✓	✓
WAV	✓	✓	✓	✓	✓	✓	✓	✗
WEBM ³	✓ ⁴	✗	✗	✓	✗	✗	✗	✗
WMA	✓	✓	✓	✓	✓	✓	✓	✗
WMV	✓	✓	✓	✓	✓	✓	✓	✗

								
MadCap Mimic Movie Links								
Flash (SWF) Format	✓	✓	✓	✓	✓	✓	✓	✓
HTML5 Format (Depending on Browser) ³	✓	✓	✓	✓	✓	✗	✓	✓
Mimic Movie Format	✓	✓	✓	✗	✗	✗	✗	✓
Silverlight Format	✓	✓	✓	✓	✓	✓	✓	✓
Vimeo Embedded Videos	✓ ⁶	✓	✓ ⁶	✓	✓	✓	✓	✓

								
YouTube Embedded Videos	✓ ⁶	✓	✓ ⁶	✓	✓	✓	✓	✓

¹Support for each element or file type depends on if a particular browser supports it. This is especially true with WebHelp mobile output and mobile browsers.

²For this output, the 3D Model opens in a separate window.

³To view HTML5 movie output, end users must have a newer browser that supports these types of videos.

⁴If you generate DotNet Help, embedded HTML5 movies are not supported out of the box. That's because DotNet Help uses Internet Explorer 7, which does not support HTML5 movies. However, you can make it work by adding the default meta tag to your target: `<meta http-equiv="X-UA-Compatible" content="IE=Edge" />`.

⁵This file format does not work in Internet Explorer.

⁶If you generate DotNet Help or Microsoft HTML Help, YouTube videos are rendered smaller than in other outputs and Vimeo Advanced settings are not supported. However, you can make it work by adding the default meta tag to your target: `<meta http-equiv="X-UA-Compatible" content="IE=Edge" />`.









Navigation Links

Following are navigation link features supported in each output type:

								
Cross-References Supported	✓	✓	✓	✓	✓	✓	✓	✓
Cross-References—Context-sensitive	✗	✗	✗	✗	✗	✗	✗	✗
Drop-down Text	✓	✓	✓	✓	✓	✓	✓	✓
Expanding Text	✓	✓	✓	✓	✓	✓	✓	✓
Text Hyperlinks	✓	✓	✓	✓	✓	✓	✓	✓
Text Popups	✓	✓	✓	✓	✓	✓	✓	✓
Togglers	✓	✓	✓	✓	✓	✓	✓	✓
Topic Popups	✓	✓	✓	✓	✓	✓	✓	✓

Pulse (Socially Enabled Output)

Following are Pulse features supported in each output type:









								
Pulse Integration	✗	✗	✗	✓	✓	✗	✓	✗

Search

Following are search features supported in each online output type.

General Support









The table below summarizes search support in each output type:









								
Search Supported Via Flare Integration	✓	✗	✓	✓	✓	✓	✓	✓

End User Search Features

The table below summarizes the key search features and capabilities for each output type. With HTML5 and WebHelp outputs, you can distribute either client- or server-based outputs. This is why the table below lists them twice—under client- and server-based output.

End user search features are supported by the Eclipse Help Viewer and search operations are provided by a plug-in that you develop using the Eclipse platform (for more information, see org.eclipse.help.ui.searchEngine). The Flare search engine does not handle Eclipse Help search operations.

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
								
Glossary Search—Find Matching Terms	✓ ¹	✓	✗	✓	✗	✗	✗	✓
Index Search—Find Matching Terms	✓	✓	✓	✓	✓	✗	✓	✓
Search Favorites—Save Queries to Favorites List	✓	✗	✓	✗	✓	✗	✓	✗
Search Query—Asterisk (*) Wildcard Character	✓ ²	✓	✓	✗	✗	✗	✗	✗
Search Query—Boolean Operators	✓ ²	✓ ³	✓	✓	✓	✓	✗	✗
Search Query—Full-text Search (not case sensitive)	✓	✓	✓	✓	✓	✓	✓	✓
Search Query—Enclose Terms in Quotes (" ")	✓	✓	✓	✓	✓	✓	✓	✓
Search Results—Narrow Search Scope Using Search Filter Sets	✓	✗	✓	✓	✓	✗	✓	✓

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
								
Search Results—Results Listed in Ranked Order and Show Rank Number	✓	✓ ⁴	✓	✓ ⁴	✓	✓ ⁴	✓	✓ ⁴
Search Results—Search Hits Highlighted in Topics	✓	✓	✓	✓	✓	✓	✓	✓
Search Results—Search Hits Bolded	✗	✗	✗	✓	✗	✗	✗	✗
¹ Use CTRL+F in the Glossary pane. ² SQL Server Compact is required. ³ Select the operator from the built-in list. ⁴ Results are listed in order. Rank number is not visible.								

Content Developer Search Features

This table summarizes the key search features and capabilities that are available to you, the content developer, for each output type. With HTML5 and WebHelp outputs, you can distribute either client- or server-based outputs. This is why the table below lists them twice—under client- and server-based output.

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
								
Enable Search Feature	✓ ¹	✓ ²	✓ ²	✓ ³	✓ ²	✓ ⁴	✓ ²	✓ ³
Content Optimization—Adjust Abstract Character Limit	✗	✗	✗	✓	✗	✗	✗	✗
Content Optimization—Enable or Disable Importance in Search Results	✗	✗	✗	✓	✗	✗	✗	✗
Content Optimization—Include or Exclude Glossary Terms in Search Results	✗	✗	✗	✓	✗	✗	✗	✓
Content Optimization—Set Number of Results Displayed Per Page	✗	✗	✗	✓	✗	✗	✗	✗
Search Highlighting—Set Highlight Color	✓	✓	✓	✓	✓	✓	✓	✓
Search Performance—Chunk Large Search Database Files	✓	✗	✓	✓	✓	✓	✓	✓

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
								
Search Performance—Enable Partial-word Searching and Set Minimum Word Size	✓	✗	✓	✓	✓	✓	✓	✓
Search Performance—Enable Stop Words in Search	✓	✓ ⁵	✓	✗	✓	✓	✓	✗
Search Performance—Exclude Index Entries from Search	✓	✓ ⁶	✓	✓	✓	✓	✓	✗
Search Performance—Exclude Non-words from Search	✓	✓ ⁷	✓	✓	✓	✓	✓	✓
Search Performance—Include SQL Search Database	✓ ⁸	✗	✗	✗	✗	✗	✗	✗
Search Performance—Pre-merge Search Database	✓	✗	✓	✓	✓	✓	✓	✓
Search Performance—Set N-Gram Size for Chinese, Japanese, and Korean Projects	✓	✗	✓	✓	✓	✓	✓	✓
Search Toolbar—Include and Customize a Quick Search Field and/or Select Search Button	✓	✗	✓	✗	✓	✗	✓	✗
Search Results—Customize Order of Search Filter Sets	✓	✗	✓	✓	✓	✗	✓	✓






	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
								
Styles—Generated Search Results Pages	✗	✗	✗	✓ ⁹	✗	✗	✗	✓ ⁹
¹ Search cannot be disabled. ² Standard skin. ³ HTML5 skin. ⁴ Mobile skin. ⁵ Stop words are always enabled. ⁶ Index entries are always excluded. ⁷ Non-words are always excluded. ⁸ SQL Server Compact is required. ⁹ These styles are supported only in HTML5 Top Navigation and skinless outputs. They are not supported in HTML5 Tripane output.								

Skins

Following are skin features supported in each output type:

								
Skin Type	Standard	✗	Standard	HTML5	Standard	Mobile	Standard	Standard
About Box	✗	✗	✓	✗	✓	✓	✓	✓
Accordion Titles—Exclude	✗	✗	✗	✗	✓	✗	✓	✓
Browser Settings	✗	✗	✗	✓ ¹	✓	✗	✓	✗
Caption for Output Window	✗	✗ ²	✓	✓ ¹	✓	✓	✓	✓
Elements (e.g., tabs, accordions)—Specify Default Element	✗	✗	✓	✓ ¹	✓	✗	✓	✓
Elements (e.g., tabs, accordions)—Specify Which to Include	✓	✗	✓	✓ ¹	✓	✓	✓	✓
Feedback Comments, Email Notifications, User Profile	✓	✗	✓	✓ ¹	✓	✗	✓	✗
Generate All	✓	✗	✓	✗	✓	✗	✓	✓
Language Skins	✓	✗	✓	✗	✓	✓	✓	✓

								
Menu Skin Component	✗	✗	✗	✓	✗	✗	✗	✗
Navigation Links in Standalone Topics	✗	✗	✗	✓ ¹	✓	✗	✓	✓
Navigation Pane Settings	✗	✗	✓	✓	✓	✗	✓	✓
Preview Skin for Output Type	✓	✗	✓	✓	✓	✓	✓	✓
Search Bar Skin Component	✗	✗	✗	✓ ³	✗	✗	✗	✗
Search Results Skin Component	✗	✗	✗	✓ ³	✗	✗	✗	✗
Separate UI Text Tab in Skin Editor for Localization	✗	✗	✗	✓	✗	✗	✗	✗
Styles	✓ ⁴	✗	✓ ⁴	✓	✓	✓	✓	✓
TOC Entry—Select Skin For	✗	✗	✓	✗	✗	✗	✗	✗
Topic Toolbar—Custom Settings	✓	✗	✓	✓ ⁵	✓	✓ ⁶	✓	✓
Web Toolbar—Add via Skin	✗	✗	✓	✗	✗	✗	✗	✗
Web Toolbar—Custom Settings	✗	✗	✓	✓ ¹	✓	✗	✓	✓

								
Window—Size/Position	✓	✗	✓	✓ ¹	✓	✗	✓	✓
Window—User-Defined Size/Position	✗	✗	✓	✗	✗	✗	✗	✗

¹This is available in the Tripane skin only. It is not available in the Top Navigation skin.

²This is handled through the org.eclipse.help.base plug-in.

³This is available in the Top Navigation skin only. It is not available in the Tripane skin.

⁴Only some styles (e.g., Feedback, toolbar) are supported.

⁵Topic toolbar settings can be controlled in a Topic Toolbar skin component, as well as in a Topic Toolbar proxy.

⁶Limited settings are available.

Tables of Contents and Mini-TOCs









Following are table of contents (TOC) and mini-TOC features supported in each output type:

								
TOCs—Created via Headings and Proxy	✗	✓	✗	✗	✗	✗	✗	✗
TOCs—Created via TOC File and Skin Setting	✓	✗	✓	✓	✓	✓	✓	✓
Auto-generate	✗	✗	✗	✗	✗	✗	✗	✗
Binary TOCs	✗	✗	✓	✗	✗	✗	✗	✗
Mark as New	✗	✓	✓	✓	✓	✓	✓	✓
Mini-TOCs	✓	✓	✓	✓	✓	✓	✓	✓
Skin—Select for TOC Entry	✗	✗	✓	✗	✗	✗	✗	✗
Synchronize With Topics	✓	✗	✓	✓	✓	✗	✓	✓

Target Settings


Following are target features supported in each output type:

								
Auto-generate Glossary, Index, TOC	✗	✗	✗	✗	✗	✗	✗	✗
Characters and Spaces—Replace With Underscores	✓	✓	✓	✓	✓	✓	✓	✓
CMYK	✗	✗	✗	✗	✗	✗	✗	✗
Content Folder—Omit from Output	✗	✓	✓	✓	✓	✗	✓	✓
Crop and Registration Marks—Include in Output	✗	✗	✗	✗	✗	✗	✗	✗
DOCTYPE Declaration	✓	✗	✓ ¹	✗	✓	✓ ²	✓	✓
Empty Pages—Inject Automatically	✗	✗	✗	✗	✗	✗	✗	✗
File Extensions—Custom	✓	✓	✗	✓	✓	✓	✓	✓
Mark of the Web	✗	✓	✗	✓	✓	✗	✓	✗
Meta Tags—Custom	✓	✓	✓	✓	✓	✓	✓	✓
Startup Topic	✓	✗	✓	✓	✓	✗	✓	✓

								
Stylesheet Medium	✓	✓	✓	✓	✓	✓	✓	✓
Warnings—Ignore	✓	✓	✓	✓	✓	✓	✓	✓
¹ We recommend disabling this option for non-English content. ² The standard mobile DOCTYPE is always used.								

Variables

Following are variable features supported in each output type:

								
Custom Variables	✓	✓	✓	✓	✓	✓	✓	✓
Heading Variables	✓	✓	✓	✓	✓	✓	✓	✓
Running HF Variables	✗	✗	✗	✗	✗	✗	✗	✗
System Variables	✓	✓	✓	✓	✓	✓	✓	✓