

**MADCAP FLARE 2024**

# Using Media Queries

Copyright © 2024 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  
9171 Towne Center Drive, Suite 335  
San Diego, California 92122  
858-320-0387  
[www.madcapsoftware.com](http://www.madcapsoftware.com)

**THIS PDF WAS CREATED USING MADCAP FLARE.**

# CONTENTS

---

## **CHAPTER 1**

Welcome .....	5
---------------	---

## **CHAPTER 2**

Creating a Project From a Template .....	6
How to Create a New Project From a Template .....	7

## **CHAPTER 3**

Understanding Media Queries .....	8
How to Select Media Queries in the Stylesheet Editor .....	9
How to Edit Tablet and Mobile Breakpoints .....	11

## **CHAPTER 4**

Editing Styles for Media Queries .....	13
How to Edit a Media Query Style .....	14
How to View Screen Sizes With a Web Browser .....	16

## **CHAPTER 5**

Viewing Media Queries in Editors .....	19
How to View Media Queries in the XML Editor .....	20
How to View Media Queries in the Internal Text Editor .....	21

## **CHAPTER 6**

Creating Custom Media Queries .....	23
How to Create a New Media Query .....	24
How to View Media Query Precedence .....	27

## **APPENDIX**

PDFs .....	29
Tutorials .....	29
Cheat Sheets .....	30
User Guides .....	31

## CHAPTER 1

# Welcome


*Approximate Time: 15 minutes*

Welcome to the Flare Styles Tutorial—Using Media Queries.

- **Tutorial Goal** Learning how to use media queries in a Flare project.
- **Key Concepts** Getting familiar with the tablet and mobile media queries that come with Flare, creating a custom media query, and seeing how media queries are tied to HTML5 responsive output.


When specific media conditions are met, media query style settings display in the output.


The image shows a screenshot of the Flare Styles Editor interface. On the left, a panel titled "Responsive Output Settings (Overrides the Skin)" contains two dropdown menus: "Tablet Breakpoint:" set to "1200 px" and "Mobile Breakpoint:" set to "700 px". Below these is a checkbox labeled "Use device width media queries" which is currently unchecked. A red arrow points from this panel towards the main editor area. The main editor area is split into two side-by-side panels. The left panel is titled "Medium: tablet" and shows a style rule for "h1" with a "Background" property set to "aqua". Below this, a preview shows the text "This is MadCap" on a light blue background. The right panel is titled "Medium: mobile" and shows the same "h1" style rule but with the "Background" property set to "yellow". Below this, a preview shows the text "This is MadCap" on a yellow background.

 **NOTE** Since media queries are tied to responsive output, it is recommended to work through the Responsive Output Tutorial after completing this tutorial.

# Creating a Project From a Template

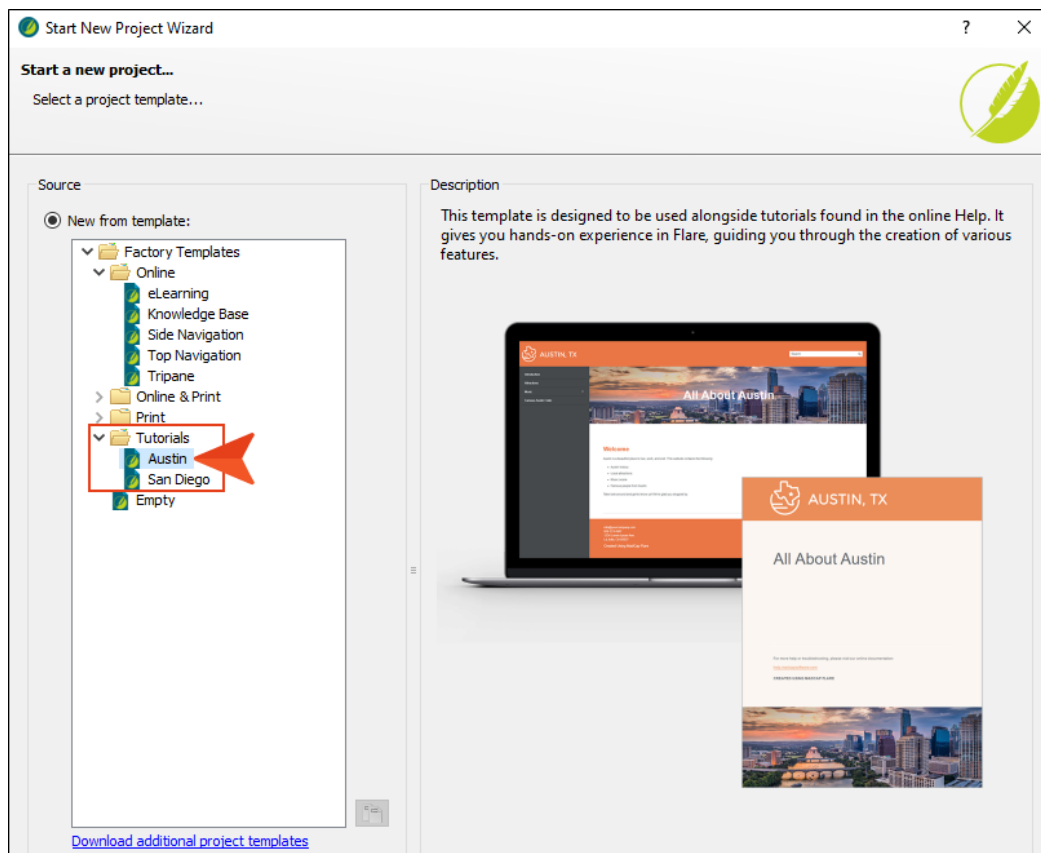
To start, let's create a new project based on Flare's Austin template.

 **NOTE** This is a standalone tutorial. However, if you have completed the previous tutorials, you can skip this topic since a project has already been created.

 **NOTE** If you plan to work with the project from the previous styles tutorials, the images you see as you go through this tutorial will be slightly different. If you want the images to match, then it is recommended to create a new project for this tutorial.

# I How to Create a New Project From a Template

1. Select **File > New Project**. The Start New Project Wizard opens.
2. In the **Project name** field, give your project any name you want.
3. (Optional) In the **Project folder** field, you can enter a location where you want to save your project. By default, projects are saved in your Documents > My Projects folder.
4. Click **Next**.
5. Under **Factory Templates**, expand the **Tutorials** folder. Select the **Austin** template.



6. Click **Next**.
7. For branding, keep the default settings. (This is where you could customize your project's look with branding, such as colors and logo.) Click **Next**.
8. Click **Finish**. The project is created and loaded into Flare.

# Understanding Media Queries

A media query is an alternative group of settings in a stylesheet. These settings are automatically used under certain conditions, such as when a screen of a certain size is displaying the output. Media queries are able to do this because they are configured with specific criteria (e.g., maximum width of the screen, orientation, resolution). When the criteria are met, the style settings in the media query are used to display the output. *You do not tell a Flare target to use a media query; it just happens automatically.*

You can place any valid CSS style settings in a media query, just as you can in a medium. For example, if you want paragraph text to suddenly turn blue when a topic is viewed on an iPhone, you can edit the mobile media query, telling it to use a blue font for all paragraph styles.

**This chapter discusses the following:**


How to Select Media Queries in the Stylesheet Editor .....	9
How to Edit Tablet and Mobile Breakpoints .....	11



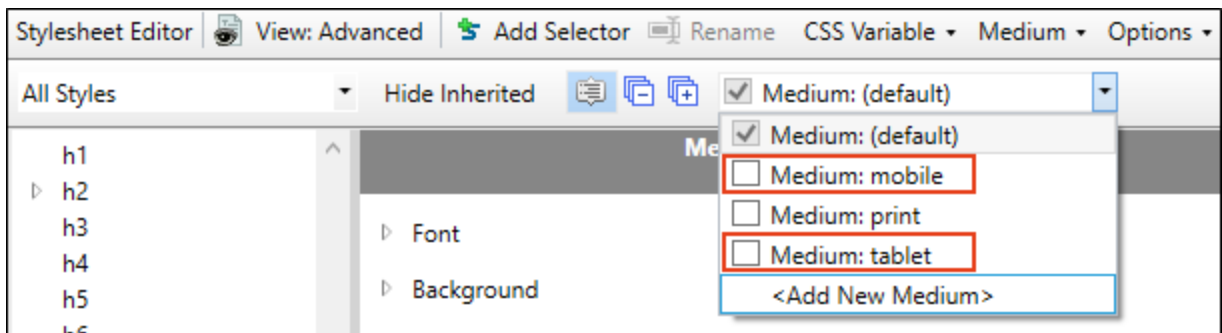
# I How to Select Media Queries in the Stylesheet Editor

One of the most common ways to use a media query is to identify a width setting for a device with a different screen size. Let's take a look at the media queries that come with Flare.

1. From the Content Explorer, expand **Resources > Stylesheets**, and double-click the **Styles.css** file.
2. From the local toolbar of the Stylesheet Editor, make sure the first button is labeled **View: Advanced**.

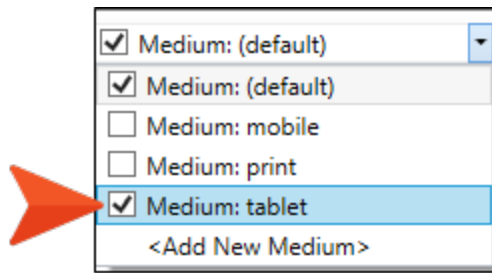
 **NOTE** You can select a media query in the editor's Simplified view. In that case, whichever media query you choose becomes the editable one.

3. In the left pane tree structure, select **p**.
4. From the local toolbar, click the **Medium** drop-down.

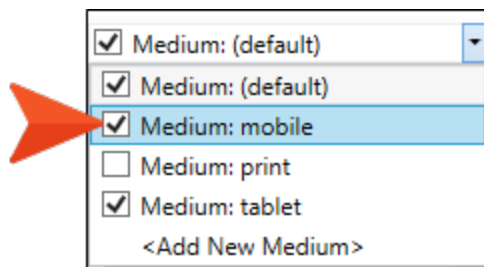


A couple of media queries (tablet and mobile) are already provided in Flare. You can create additional media queries from the Stylesheet Editor if you want, but most authors will be able to do everything they need with just the tablet and mobile media queries. The tablet media query is designed to be used on medium-sized screens, such as iPads. The mobile media query is designed to be used on smaller screens, such as smart phones. (The print item in the drop-down is actually a medium, not a media query. The default is a medium but can also be considered a media query in its relationship with the tablet and mobile items.)

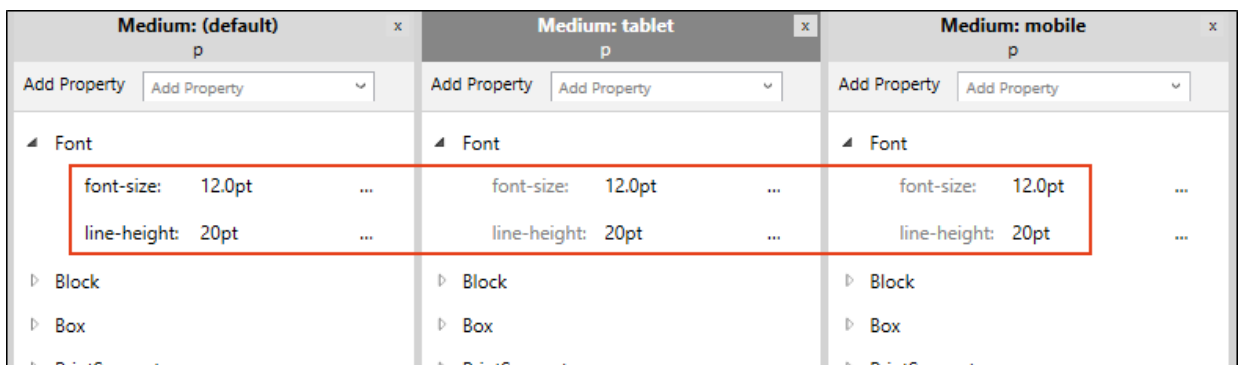
- Click the check box next to the **Medium: tablet** item. The media query opens in a new window pane next to the default medium in the editor.



- Select the drop-down again, and click the check box next to the **Medium: mobile** item. It opens in the editor.



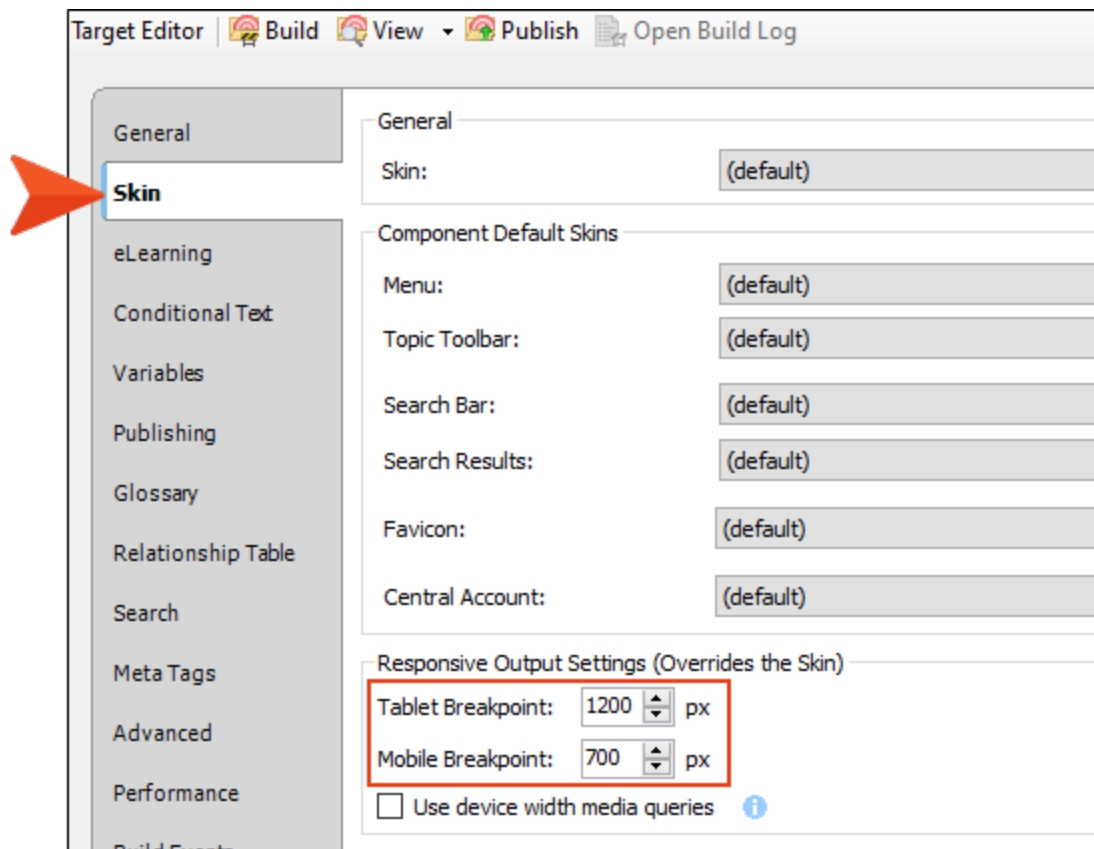
- Expand different property groups in the media query window panes. Notice how the styles are initially inherited from the default medium.




# I How to Edit Tablet and Mobile Breakpoints


The tablet and mobile media queries are tied to the responsive output settings on the Skin tab of the Target Editor for HTML5 targets. The breakpoints provided in the target determine the point at which your media queries will become active in displaying the different style settings.

1. From the Project Organizer, expand **Targets**, and double-click **All-About-Austin-HTML5**.
2. In the Target Editor, select the **Skin** tab.
3. The default tablet breakpoint is 1279px, and the default mobile breakpoint is 767px. Let's round those numbers down respectively, type 1200 pixels (tablet), and type 700 pixels (mobile).



4. Click  to save your work.

 **NOTE** Notice the option for **Use device width media queries**. 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.

 **NOTE** HTML5 Side and Top Navigation skins are always enabled for responsive output. If you prefer using Tripane HTML5 output, you can set breakpoints in the skin.

# Editing Styles for Media Queries

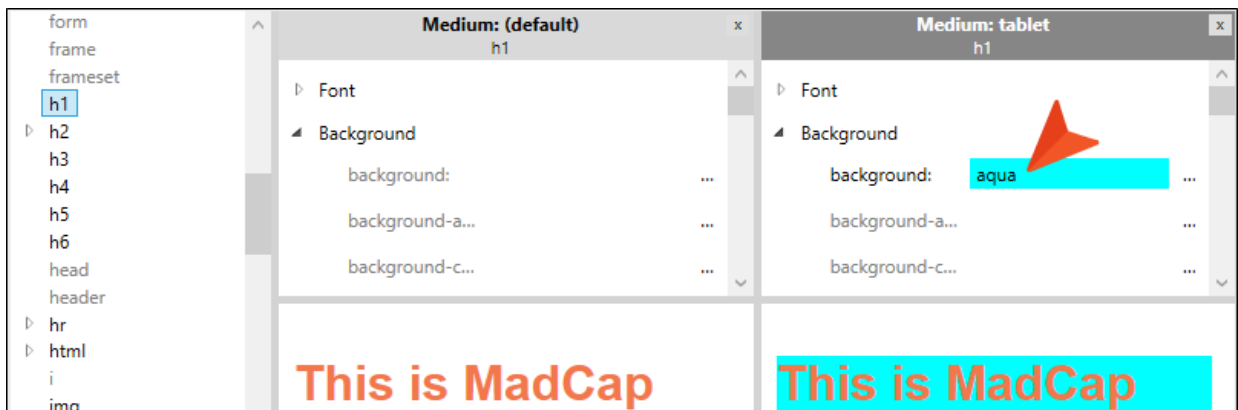
Let's view the output to see the results of how media queries work. But first, let's edit the h1 styles for each media query by giving them a vibrant background, so the breakpoints are easy to identify in the web browser.

This chapter discusses the following:

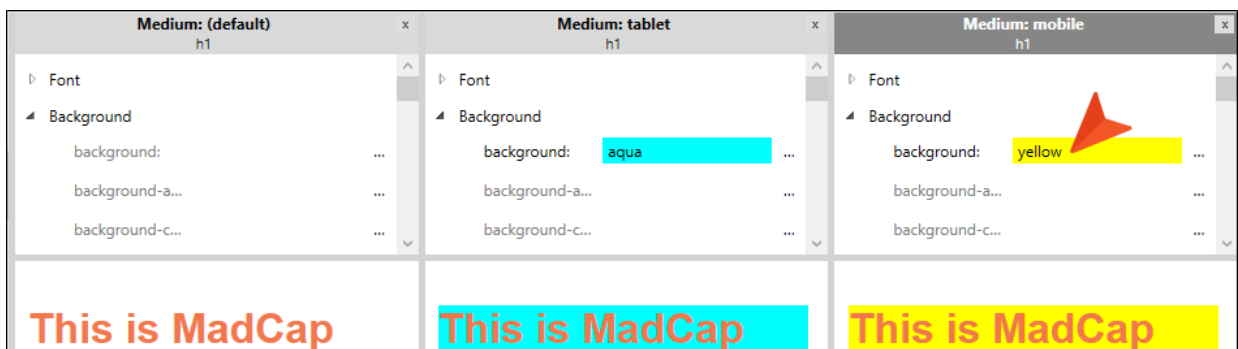
How to Edit a Media Query Style .....	14
How to View Screen Sizes With a Web Browser .....	16

# I How to Edit a Media Query Style


1. Go back to the `Styles.css` file.
2. In the left pane tree structure, select the `h1` style.
3. In the local toolbar, select the **Show** drop-down, and choose **Show: All Properties**.
4. In the Medium: tablet window pane, expand the **Background** property group. (The media query you give focus to becomes editable.)
5. Next to the top **Background** property, type `aqua`, and press **ENTER**. Notice the property is no longer inheriting its style from the default medium.



6. In the Medium: mobile window pane, locate the same **Background** property.
7. Type `yellow`, and press **ENTER**. Notice the property is no longer inheriting its style from the default medium or from the tablet media query.




8. Click  to save your work.

 **NOTE** When working with media queries, the following is applicable regarding inheritance. When a screen is larger than a tablet, then styles associated with the default medium are used. Tablet-sized screens inherit from the default medium, but if there are any conflicts between files, then the styles from the tablet media query will win. Mobile-sized screens inherit from the default medium and the tablet media query. If there are any conflicts, then the mobile media query will win.

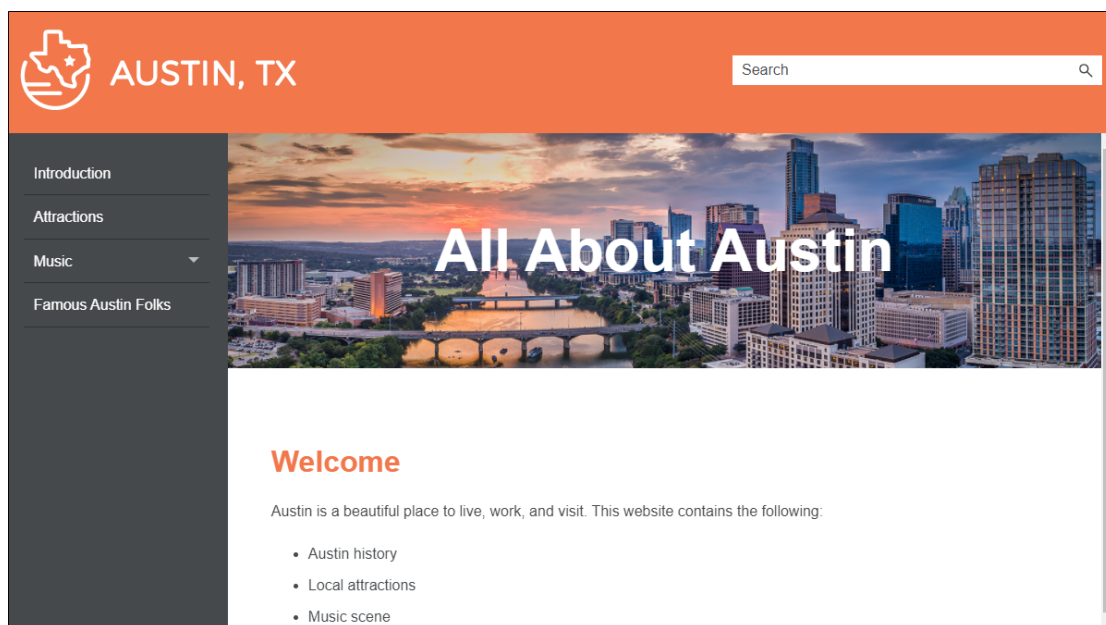
Even though you may not see the inheritance reflected in the stylesheet when editing mediums and media queries, it is happening.

# I How to View Screen Sizes With a Web Browser

Let's build the HTML5 output to view the responsive output as it displays for the default web medium, and for the tablet and mobile media query breakpoint screen settings.

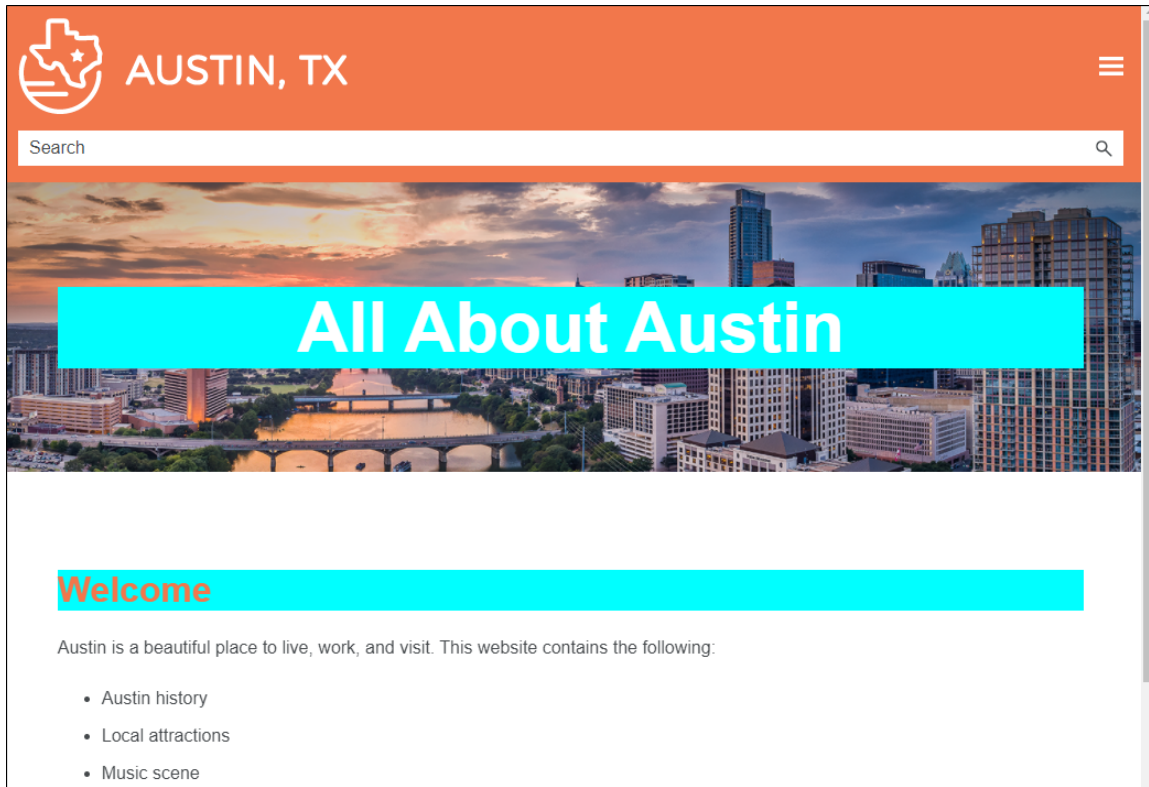
 **NOTE** You don't need to have a tablet or a mobile device on hand to test responsive output. It can all be simulated using the generated output in your web browser.


1. Select the **Project** ribbon. Then from the **Build Primary** drop-down, select **Build All-About-Austin-HTML5**. The Builds window pane opens at the bottom of the Flare interface, and the target begins to generate. When the build finishes successfully, the Build Progress cell turns dark green.
2. Double-click the row to open the HTML5 output, and view it in a regular browser with the window maximized. You will see the settings for the default web medium.



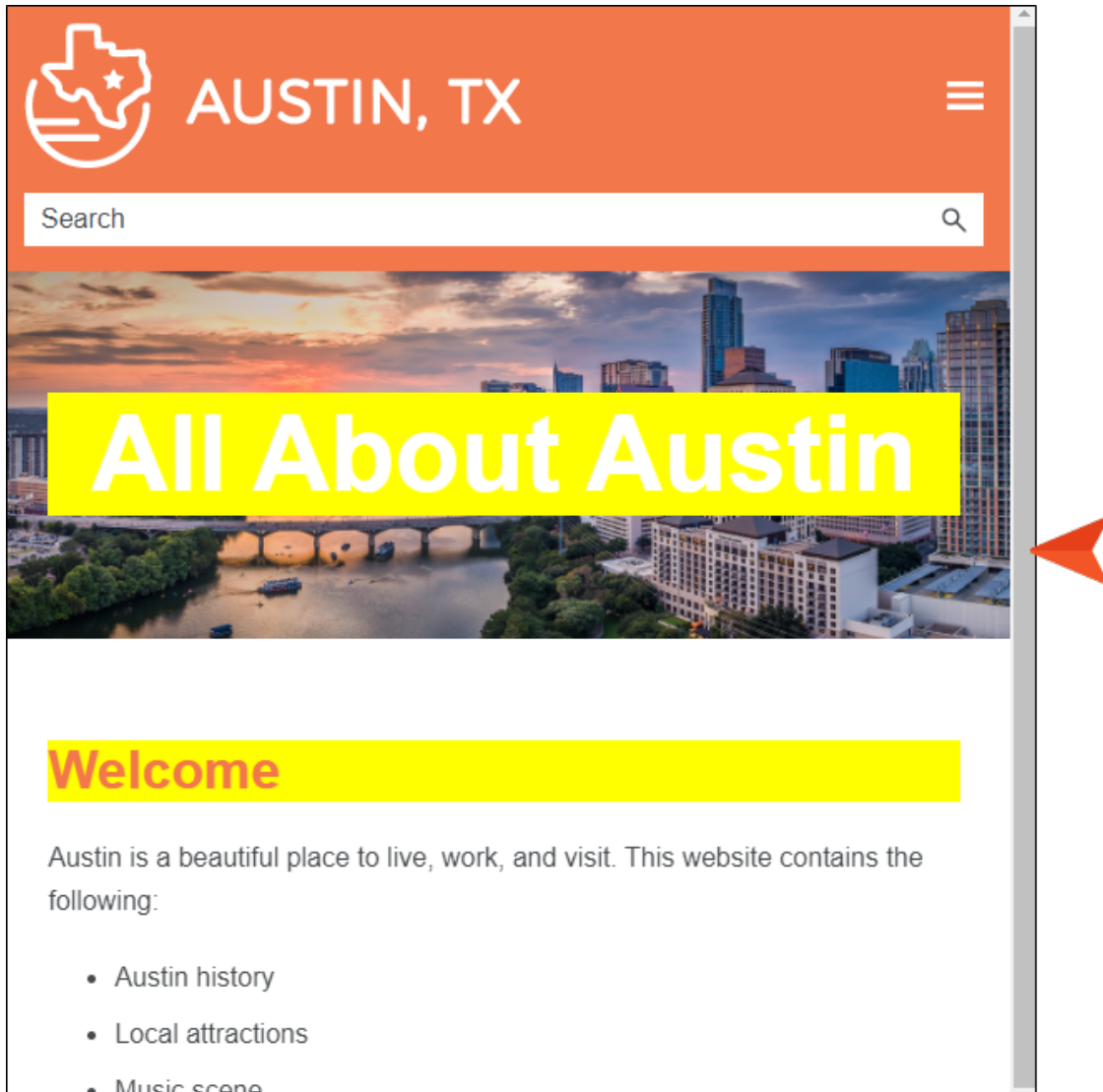



3. Click and drag the browser window to reduce it. Once the width of the browser window reaches 1200 pixels of width, the display changes to show the tablet media query (with a bright aqua background for the h1 style element).



 **NOTE** If you are working the styles tutorials in a sequential order without creating a new project each time, keep in mind that the background color and text do not respond as nicely as they do when working this tutorial individually.

- Click and drag the browser window to reduce it further. Once the width of the browser window reaches 700 pixels, the display changes to show the mobile media query (with a yellow background for the h1 style element).



 **NOTE** As you reduce the size of the browser window, look for other differences as well. Noticeably, the table of contents disappears from the left side (and appears as a flyout menu in the upper-right corner), and the repositioning of the search bar.

- Close the HTML5 output.

# Viewing Media Queries in Editors

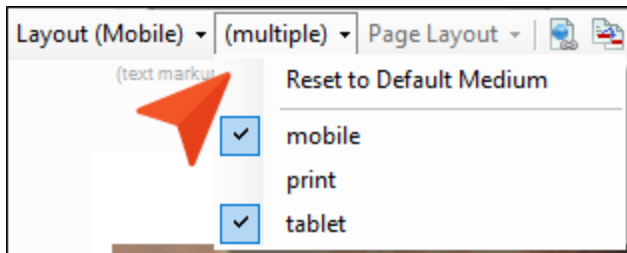
Let's take a closer look at media queries in Flare's XML Editor and Internal Text Editor.

This chapter discusses the following:

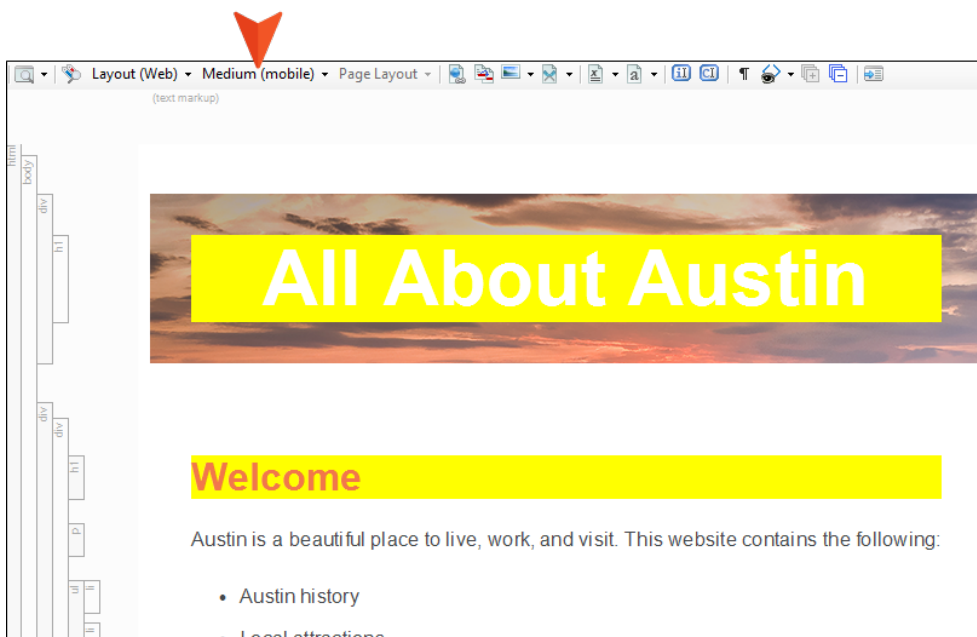
How to View Media Queries in the XML Editor .....	20
How to View Media Queries in the Internal Text Editor .....	21

# I How to View Media Queries in the XML Editor

1. From the Content Explorer, double-click the **Home.htm** topic.
2. In the local toolbar, select the **Medium** drop-down. Notice that **(multiple)** displays to indicate multiple styles are applicable (i.e., tablet and mobile).



You can work in a topic with your preferred media query styles set. This enables you to see what content will look like before building the output.

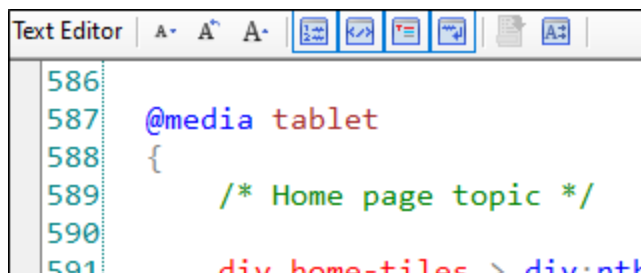


3. Cycle through the mediums to see what they look like in the editor. From the **Medium** drop-down, select **Reset to Default Medium**, and then select **Tablet**, and then select **Mobile** (to end on).

# I How to View Media Queries in the Internal Text Editor


Let's see how a media query looks in CSS code in the Text Editor.

1. From the Content Explorer, expand **Resources > Stylesheets**, and right-click the **Styles.css** file.
2. From the context menu, select **Open with**, and then select **Internal Text Editor**. All the styles and properties from the Stylesheet Editor are displayed to the right in the Text Editor.
3. In the Text Editor, all the default styles are in the upper region of the editor. As you scroll down to roughly line 587, you'll see a line that starts with `@media`. This means the style is either a medium or a media query. The styles underneath the media entry are indented to make them easier to read and distinguish from the default medium styles.



The screenshot shows a window titled "Text Editor" with a toolbar at the top. The code is displayed in a monospaced font with syntax highlighting. Line 586 is empty. Line 587 starts with `@media tablet`. Line 588 starts with a curly brace `{`. Line 589 is indented and contains `/* Home page topic */`. Line 590 is empty. Line 591 is partially visible and shows `div.home-tiler > div:nth`.

```
586
587 @media tablet
588 {
589     /* Home page topic */
590
591     div.home-tiler > div:nth
```

 **NOTE** If you are working the styles tutorials in a sequential order, the line number outlined in the step and image may not match what you see in the editor. Scroll down in the Text Editor until you see the `@media tablet` item.

- Continue to scroll down to roughly lines 650 and 710. You will see the h1 background colors for the tablet and mobile media queries respectively.

```
649  
650     h1  
651     {  
652         background: aqua;  
653     }  
654 }
```

```
707  
708     h1  
709     {  
710         background: yellow;  
711     }  
712 }
```

- Close the file.

# Creating Custom Media Queries

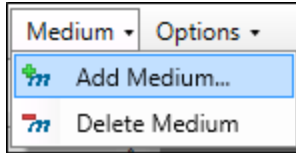
As mentioned before, most authors will be able to do everything they need with just the tablet and mobile media queries. However, there are situations where content might look better if you create a custom media query. For example, a custom query might come in handy if you want content to display a certain way in between the tablet and mobile breakpoints, or if you want content to be narrower than the mobile width.

**This chapter discusses the following:**

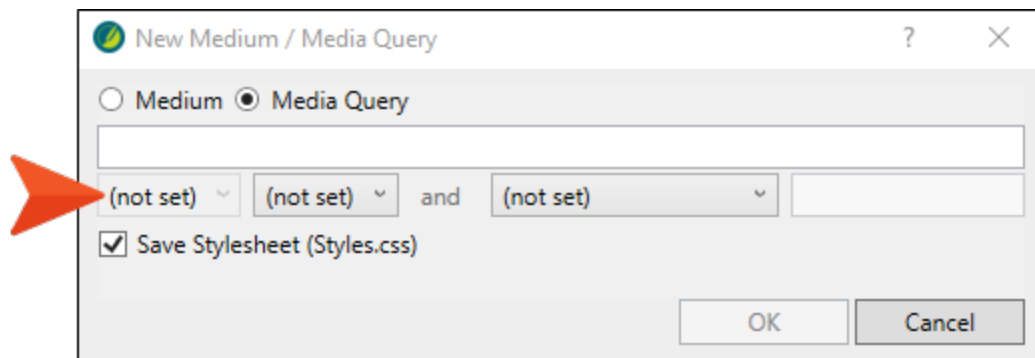
How to Create a New Media Query .....	24
How to View Media Query Precedence .....	27


# I How to Create a New Media Query

1. Open the `Styles.css` file.
2. In the local toolbar of the Stylesheet Editor, click the **Medium** drop-down, and select **Add Medium**. The New Medium / Media Query dialog opens.



3. Select **Media Query**. Extra fields are shown in the dialog.

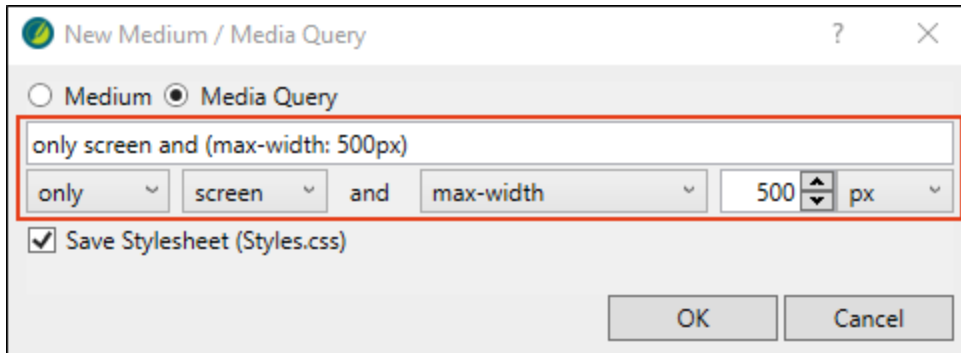


 **NOTE** Media queries are written as logical expressions that check for specified media conditions. You can enter your media query directly in the text field (as long as you use valid media query syntax). Alternatively, you can use the fields below it to choose criteria for the media query; the text field is populated accordingly. For more information about creating custom media queries and the options available in the New Medium / Media Query dialog, see:

<http://www.w3.org/TR/css3-mediaqueries/>

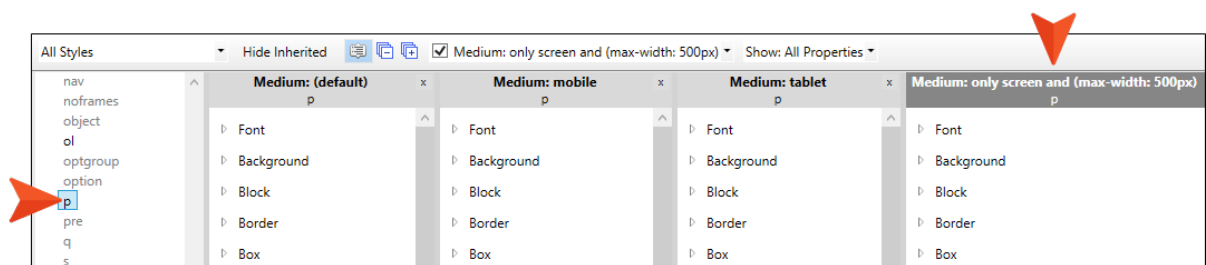


- Set the fields in the dialog as follows, and then click **OK**. (The first drop-down is disabled, so just start at the second drop-down, and go back to the other when it is enabled.)



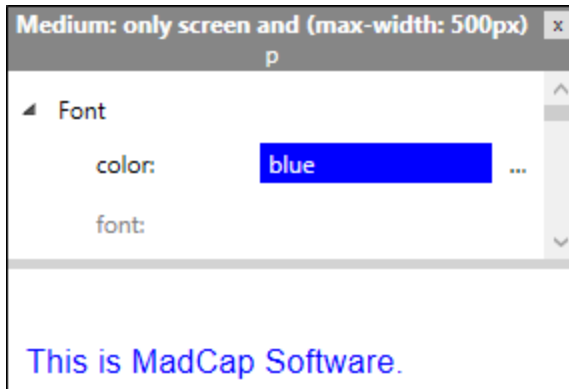
By setting the maximum width, when you reduce the width of the browser to 500px, it takes on a different look with the specific media query style setting for that device size.


- The custom media query, named Medium: only screen and (max-width:500px), displays automatically next to the other mediums in the editor. In the left pane tree structure, select the **p** style.





- In the local toolbar, from the **Show** drop-down, choose **Show: All Properties**.
- In the custom media query window pane, expand the **Font** group.

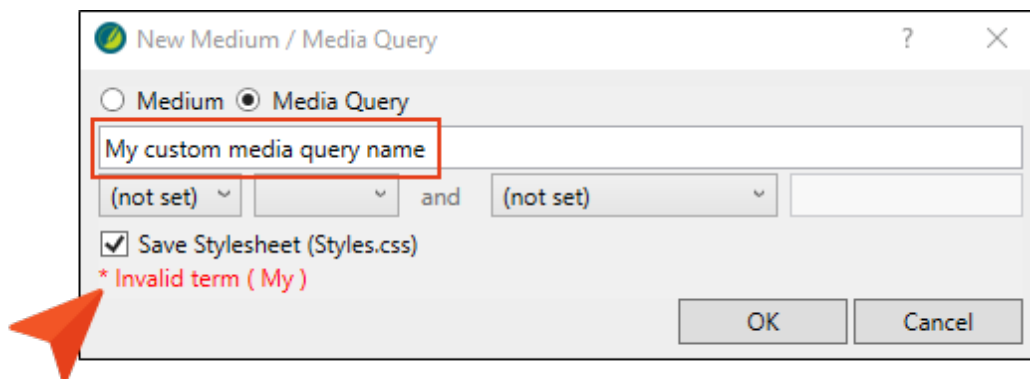
- Next to the **Color** property, type `blue`, and press **ENTER**. The blue font color is not intended to make our project pretty, we just want the change to stand out after the build is generated.



 **NOTE** If you are working the styles tutorials in a sequential order, the paragraph element font color was set to gray. Just update the value to blue.

- Click off the **Color** property.
- Click  to save your work.

 **NOTE** If you want to rename your media query, you can only rename it by making modifications to its settings in the dialog. It must retain valid syntax. Flare will let you know if the syntax is invalid.

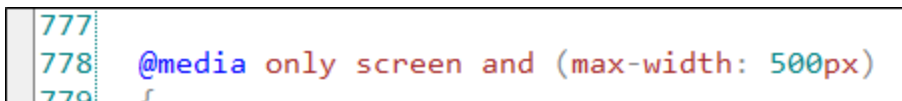


# I How to View Media Query Precedence

The order that media queries display in the Text Editor affects how they are used.

What if there is a conflict between your selected medium and a media query, or a conflict between media queries themselves? Media queries will always have precedence over your selected medium. As far as multiple media queries are concerned, the end result depends on the order of the media queries in your stylesheet (when viewing it in the Internal Text Editor). The general rule is that priority is given to whichever media query is listed last (i.e., the one that was added most recently) in the stylesheet. Then the next one above has the next highest priority, and so on. You can always open your stylesheet in the Internal Text Editor and change the order of the media queries.

1. From the Content Explorer, expand **Resources > Stylesheets**, and right-click the **Styles.css** file.
2. From the context menu, select **Open with**, and then select **Internal Text Editor**.
3. In the Text Editor, scroll to the bottom of the file (roughly line 778). The most recently added media query, `@media only screen and (max-width:500px)`, has the highest precedence (it displays lowest in the stylesheet).

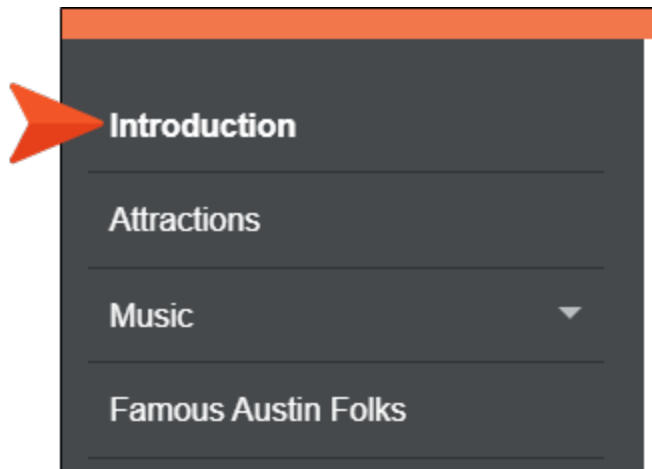


```
777:
778: @media only screen and (max-width: 500px)
779: /
```

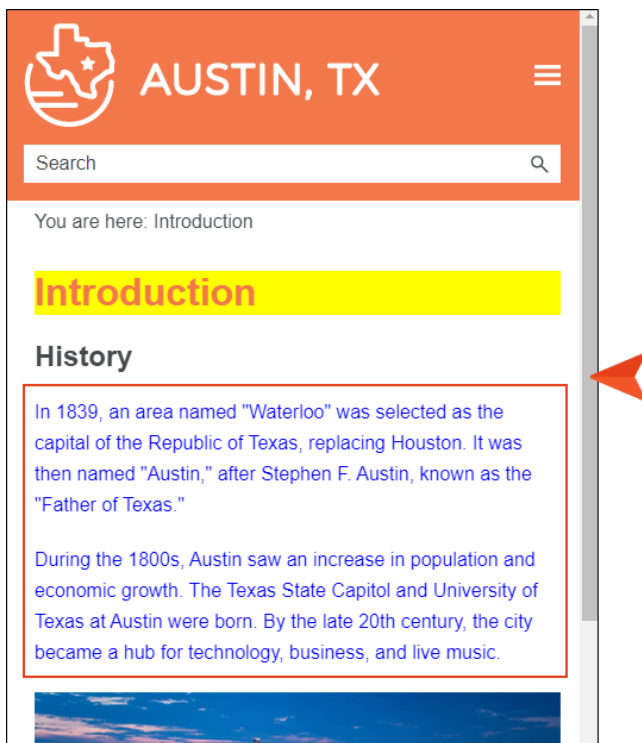
 **NOTE** For more information see the online Help.

4. Select the **Project** ribbon. Then from the **Build Primary** drop-down, select **Build All-About-Austin-HTML5**. The Builds window pane opens at the bottom of the Flare interface, and the target begins to generate. When the build finishes successfully, the Build Progress cell turns dark green.
5. Double-click the row to open the HTML5 output, and view it in a regular browser with the window maximized.

6. From the table of contents, select the **Introduction** topic.



7. Click and drag the browser window to reduce it. Once the width of the browser window reaches 500 pixels, the display changes to show the custom media query (with blue paragraph font).



8. Close the HTML5 output.

9. From the **Window** ribbon, select **Close All Documents**.

## APPENDIX

---

# PDFs

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

## I Tutorials

*Getting Started Tutorial*

*Autonumbers Tutorial*

*Back-to-Top Button Tutorial*

*Context-Sensitive Help Tutorial*

*Custom Toolbar Tutorial*

*eLearning Tutorial—Basic*

*eLearning Tutorial—Advanced*

*Image Tooltips Tutorial*

*Lists Tutorial*

*Meta Tags Tutorial*

*Micro Content Tutorial—Basic*

*Micro Content Tutorial—Advanced*

*Responsive Output Tutorial*

*Single-Sourcing Tutorial*

*Snippet Conditions Tutorial*

*Styles Tutorials*

*Tables Tutorial*

*Word Import Tutorial*

# | Cheat Sheets

*Context-Sensitive Help Cheat Sheet*

*Folders and Files Cheat Sheet*

*Learning & Development Cheat Sheet*

*Lists Cheat Sheet*

*Micro Content Cheat Sheet*

*Print-Based Output Cheat Sheet*

*Search Cheat Sheet*

*Shortcuts Cheat Sheet*

*Structure Bars Cheat Sheet*

*Styles Cheat Sheet*

# I User Guides

*Accessibility Guide*

*Analysis and Reports Guide*

*Architecture Guide*

*Autonumbers Guide*

*Branding Guide*

*Condition Tags Guide*

*Context-Sensitive Help Guide*

*Eclipse Help Guide*

*eLearning Guide*

*Getting Started Guide*

*Global Project Linking Guide*

*HTML5 Guide*

*Images Guide*

*Import Guide*

*Indexing Guide*

*Key Features Guide*

*Lists Guide*

*MadCap Central Integration  
Guide*

*Meta Tags Guide*

*Micro Content Guide*

*Navigation Links Guide*

*Plug-In API Guide*

*Print-Based Output Guide*

*Project Creation Guide*

*QR Codes Guide*

*Reviews & Contributions With  
Contributor Guide*

*Scripting Guide*

*Search Guide*

*SharePoint Guide*

*Skins Guide*

*Snippets Guide*

*Source Control Guide: Git*

*Source Control Guide:  
Perforce Helix Core*

*Source Control Guide:  
Subversion*

*Source Control Guide: Team  
Foundation Server*

*Styles Guide*

*Tables Guide*

*Tables of Contents Guide*

*Targets Guide*

*Template Pages Guide*

*Templates Guide*

*Topics Guide*

*Touring the Workspace Guide*

*Transition From FrameMaker  
Guide*

*Translation and Localization  
Guide*

*Variables Guide*

*Videos Guide*

*What's New Guide*