

Creating a New Theme in Form Builder 5

Download the PDF of this article.

In this Article

Related Articles

Overview

Enterprise admins have the ability to create themes in the new Form Builder 5. Additionally, you have control over which style elements will be exposed when your users want to customize the themes you have created.

This process involves using SCSS to expose the specific elements you want to be customizable.

Details on this process are available below.

Requirements

Enterprise

Compliance Cloud

For information on upgrading, please contact our Sales Department at sales@formassembly.com.

Getting Started

To begin, click the **Theme** option in the form builder menu and choose to **Create New Theme**.

From there, you can add SCSS variables to the Custom CSS section of the theme editor in order to choose what variables will be exposed to those using and customizing the theme.

Additionally details on the types of SCSS variables you can use are discussed below.

Variable Syntax

1. You must use valid SCSS variables. These must start with "\$", and will be followed by valid CSS identifier. [More information is available here](#).
2. You must have a valid "type" prefix. The type is defined as the string between the \$ character and the first dash "-".
 - Examples:
 - Variable of type "text": \$text-abcdef
 - Variable of type "number" \$number-ghijklm
3. You must have valid "label". The label is defined as the string following the first dash ("-"). Subsequent dashes indicate spaces.
 - Examples:
 - Label "Input Font": \$text-input-font
 - Label "Input Font Size": \$number-input-font-size

Variable Definition Syntax:

1. You must use a valid SCSS variable definition. It will start with a

variable name (as defined above) followed by a colon (":"), a variable value, and terminated with a semi-colon (";").

- Examples:

- `$text-input-font: helvetica, sans-serif;`
- `$number-input-font-size: 12px;`

2. Valid definitions will show up immediately as a Theme Option. If a definition does not appear, its syntax is invalid or not supported.

Valid Type Prefixes

Type	Usage	UI Control
text	Any text	A text input
length	A numeric value followed by a size unit (e.g. "px, em")	A number input and a drop-down menu with valid unit types
color	A color, 6 digit hex format (#FFFFFF)	A color picker
number	Any number	A number input
range	A number between 0 and 1	A slider
image	A background image	File picker for image upload
font	A Google Font name	A dropdown menu with select Google Fonts

Type	Usage	UI Control
------	-------	------------

Supported Length Unit Types

Support length unit types are: em, rem, px, %, ex, ch

[Please see this article for more information.](#)

Other units are not widely supported, and are not recommended in a web page settings.

Theme Source Code

You must use valid CSS code, augmented with the SCSS variable syntax. Other SCSS features are not supported. Syntax errors are currently reported when saving and will prevent the theme from being saved.

You must avoid any attempt to alter the layout of the form. The layout is controlled with the form builder (via wforms-layout.css) and overriding layout in a Theme would break this Form Builder functionality.

Notable CSS Classes

Class Name	Usage Notes
------------	-------------

Class Name	Usage Notes
.wFormContainer	Top level container for the entire form, header and footer included. Container's width is determined by the "Form Width" setting in the Form Builder. Managing the width, min-width or max-width in the Theme is not recommended.
.wFormHeader	Area above the form. Empty. Available for logo. New in Form Builder 5.0.0
.wForm	Container for the entire form, from the form title to the submit button.
.wFormFooter	Area below the form. Contains Assistance Link, optional "Powered By FormAssembly" logo, abuse report link. New in Form Builder 5.0.0
.oneField	<p>Top level container for a form field (including all checkbox/radio buttons of a multiple-choice field, field label, field hint, and any field-level error message).</p> <p>Container's width is determined by the "Form Width" setting in the Form Builder. Managing the width, min-width or max-width in the Theme is not recommended.</p>
.inputWrapper	Container for the field's input, and optional field hint and delimiters ("text around field" setting in the Form Builder)
.preField	Label preceding the input. The width of the label can be controlled in the Form Builder. A default width can be set in the theme.
.postField	Label following the input. Applies to checkbox/radio

Class Name	buttons only. Usage Notes
.oneChoice	Container for a radio or checkbox option. Includes the input and its label (.postField)
.matrixLayout	Table container for radio/checkbox matrix layout option
.headerRow	For matrix layout only. Table's header row
.headerCol	For matrix layout only. Table's first column, with field labels.
.wfTabNav	Container for optional page navigation.
.wfTabCurrentPage	Current page in the page navigation
.wfTab	A page in the page navigation
.wFormTitle	Form Title (H3 element)

Public Theme

Enterprise customers can make themes public for their users. After you have created the theme, you can navigate to: Admin Dash>Forms>Form Themes

Look for the theme name. There will be a dropdown arrow beside the Preview button and an option to make it public.
