

Stylesheets

As is industry standard, you set the visual style of your site using CSS. On top of the benefits of CSS, Kibo themes provide a clear directory organization to enhance the separation of site content from site presentation, while also leveraging a LESS pre-processor to give you an easier and more powerful way of applying CSS rules to your site.

Directory Structure

The `stylesheets` directory contains all the folders and LESS files that implement the CSS for your theme. The subdirectories and notable stylesheets are organized as follows, which helps you locate style rules easily and then build on them using a maintainable pattern:

folder ./	
<code>back-office.less</code>	Stylesheet for back-office templates, such as order details or packaging slips.
<code>email.less</code>	The stylesheet (which also includes imported stylesheets) for email templates. In order to support older email clients, this stylesheet is outputted inline with email bodies using the <code>inline_style</code> tag, so it is a best practice to keep this stylesheet small, if possible.
<code>storefront.less</code>	This stylesheet defines and imports all the stylesheets used in page templates for your theme. Base templates, like <code>page.hypr</code> , need to include this stylesheet using a <code><link></code> tag within the <code><head></code> tags of the template; if omitted, the theme does not load any stylesheets automatically.
folder base	Contains the default style rules for your theme, including mixins and variables.
<code>global.less</code>	Rules for common elements such as <code>h1</code> , <code>body</code> , <code>p</code> , and so on, that you want shared across all pages.
<code>mixins.less</code>	Utilities for the LESS language, such as vendor prefix helpers and transition/animation helpers.
<code>normalize.less</code>	Industry-standard normalize stylesheet, similar to a reset stylesheet but less invasive. This stylesheet replaces non-standard or legacy browser stylesheets with sensible defaults.
<code>variables.less</code>	The recommended stylesheet in which to define LESS variables, which can then be used in all other stylesheets.
folder layouts	Contains style rules that divide the page into sections.
folder modules	Contains style rules that are reusable throughout the theme.
folder pages	Contains style rules that apply to a particular page type.

folder state	Contains style rules that describe how elements display when in a particular state, such as enabled or disabled, required or optional, loading or fading, etc.
folder widgets	Contains style rules associated with widgets.

The Benefits of LESS

All the stylesheets in your theme are LESS files, designated by the `.less` extension. The system processes the LESS files and turns them into CSS for your site to use. You might be wondering, why use LESS instead of just using CSS? In short, LESS adds powerful features to the CSS language that allow you to apply complex styling rules more quickly and efficiently than you would otherwise be able to using just straight CSS. Some of the benefits of LESS include:

Variables

Allow you to assign a value to a variable name of your choice and then use that variable in a style rule.

```
@nice-red: #df3018;
#header {
  color: @nice-red;
}
```

Mixins

Allow you to include the style rules from one rule set in another rule set. For example, the button in the following example inherits its background color from another rule set.

```
.nice-background {
  background-color: @nice-red;
}
#cool-button {
  padding: 10px;
  .nice-background;
}
```

Nesting

Allows you to nest rules instead of cascading them, resulting in more clear and efficient code. For example, you can write:

```
#header {
  color: @nice-red;
  p {
    font-size: 14px;
  }
  .center {
    margin: auto;
    width: 60%;
    border: 3px solid #73AD21;
    padding: 10px;
  }
}
```

Instead of the more wordy:

```
#header {
  color: @nice-red;
}

#header p {
  font-size: 14px;
}

#header .center {
  margin: auto;
  width: 60%;
  border: 3px solid #73AD21;
  padding: 10px;
}
```

Operations

Allow you to perform mathematical operations on numbers, colors, or variables.

```
@new-color: @nice-red / 2;
@large-font: @medium-font + 12px;
@see-through: @normal-opacity * .75;
```

...and many more

In addition to the features listed above, LESS supports many other features such as:

- operations
- escaping
- functions
- namespaces
- accessors
- scope
- comments
- importing

- loops

For a full list of features, refer to the [LESS website](#).

If you are not familiar with LESS, don't worry; you can code a LESS file exactly like a CSS file. But if you want to take advantage of the LESS pre-processor and make your code more powerful, refer to the [LESS website](#) or to the helpful [Wikipedia summary](#).

Create a New Stylesheet

If you want to create a new stylesheet rather than add to an existing one, complete the following steps:

1. Decide which folder the stylesheet belongs in.
2. Create the stylesheet and add it to the folder you decided on.
3. Import the stylesheet in `storefront.less` so that your theme knows to load it: within `storefront.less`, located in the `stylesheets` directory, find an appropriate section for your new stylesheet and import it using the following syntax:

```
@import "/stylesheets/folderPath/stylesheetName.less";
```