Making Custom Modifications to PDF Templates

Clarify

Table of Contents

PDF Templates

How PDF templates are designed

PDF templates in Clarify are made up of HTML and CSS files. If you are have experience with HTML and CSS then you can customize a template however you see fit.

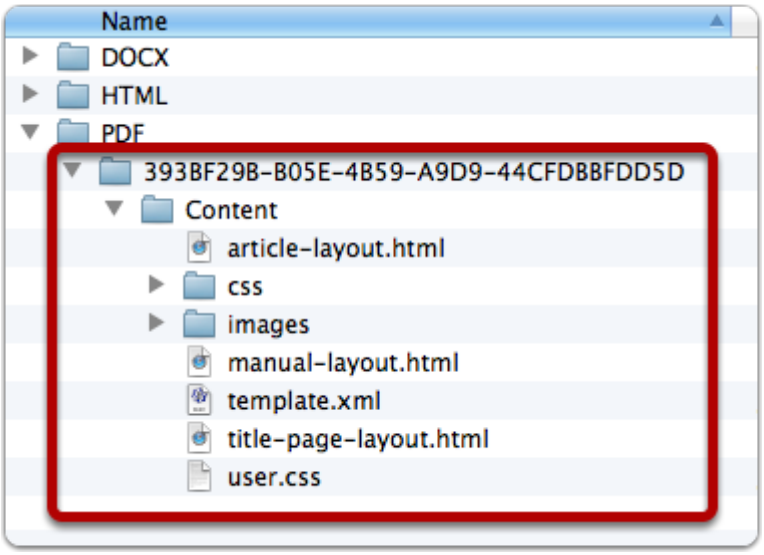
Note that the PDF templates share the same format as ScreenSteps PDF templates. There are some files in the templates that you won't use in Clarify. For example, the manual-layout.html file is only applicable when you export a manual from ScreenSteps.

Files in a custom template

These are the files that the Clarify PDF exporter has:

- **article-layout.html**: This file is processed when exporting an article as PDF.
- **manual-layout.html**: This file is processed when exporting a manual as PDF. [not applicable to Clarify]
- **template.xml**: This file stores properties of the template that the user specifies in the custom template interface in Clarify.
- **title-page-layout.html**: If this file is present than it will be used to generate the title page when exporting a manual as PDF. If it is not present then the selected title page in the ScreenSteps PDF template interface will be used. [not applicable to Clarify]
- **use.css**: This file contains any custom CSS that the user entered in the Clarify PDF template interface. This CSS has the highest precedence.

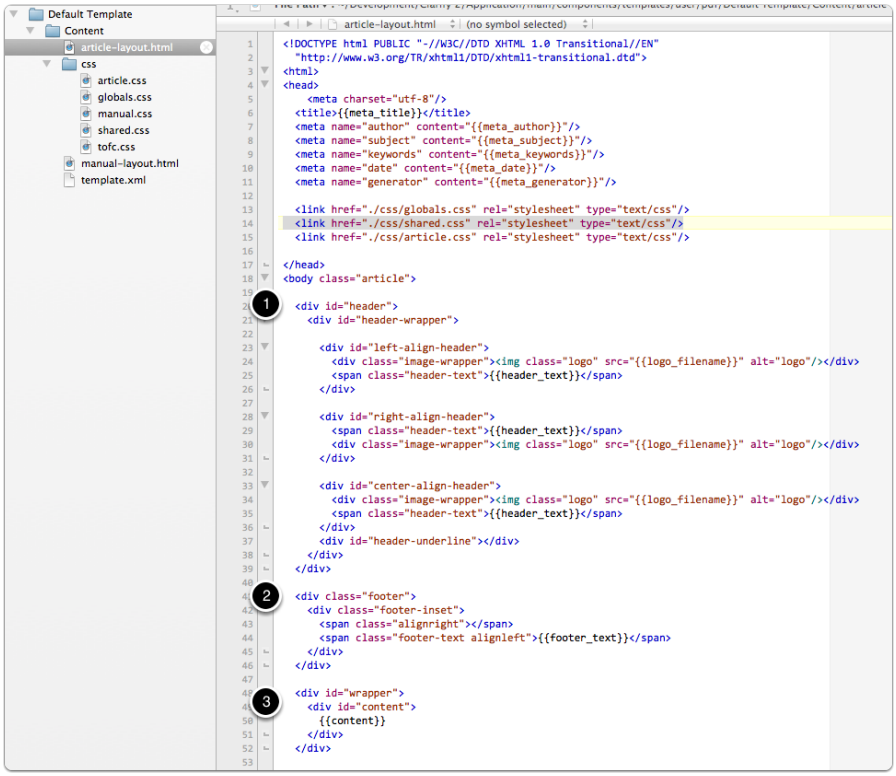
Any other files or folders in the **./Content** folder are not processed by the Clarify exporter. They are just resource files used by the HTML files. You can add any files you would like to use in your template in this folder. This folder will be the root folder for any references you make in your HTML.



The HTML elements

Here is what the default article.html file looks like. As you can see it is a standard HTML file with links to CSS files. Some things to note about some of the divs and the behavior assigned to them using CSS:

1. The **header** div is taken out of the normal document flow and placed in the header of each page.
2. The **footer** div is taken out of the normal document flow and placed in the footer of each page.
3. The **content** div is where all document content is placed.



Variables

Any user configurable property in the PDF template editor will be represented by a variable in the PDF template fies.

In addition, document properties are inserted into the template files using variables.

User configured variables

These variables are defined by the user in the PDF template editor in Clarify. You will find most of these used in the CSS files.

- {{page_size}}: Can be assigned to the CSS *size* property.
- {{page_margins}}: Can be assigned to the CSS *margin* property.
- {{page_margin_top}}: Given in pts.
- {{page_margin_right}}: Given in pts.
- {{page_margin_bottom}}: Given in pts.
- {{page_margin_left}}: Given in pts.
- {{meta_author}}: The author.
- {{footer_text}}: The text to appear in the footer.
- {{text_color}}: Default color for document text.
- {{document_font}}
- {{header_font}}
- {{footer_font}}
- {{article_title_font}}
- {{article_description_font}}
- {{step_title_font}}
- {{substep_title_font}}
- {{step_instructions_font}}
- {{article_title_size}}
- {{article_title_color}}
- {{article_description_size}}
- {{article_description_color}}
- {{step_title_size}}
- {{step_title_color}}
- {{substep_title_size}}
- {{substep_title_color}}
- {{step_instructions_size}}
- {{step_instructions_color}}
- {{header_text}}
- {{header_text_color}}
- {{header_text_size}}
- {{header_background_color}}
- {{header_divider_color}}
- {{footer_text}}
- {{footer_text_size}}
- {{footer_text_color}}
- {{footer_divider_color}}

- {{page_numbering}}: The string used for page numbering in the footer.
- {{logo_filename}}
- {{logo_alignment}}
- {{content_flow}}: How the user wants content flowed on the page. keep-together, no-breaks, one-per-page.

Content variables

- {{meta_title}}: The title of the manual or article that goes in the <head>.
- {{meta_subject}}: The title of the manual or article.
- {{meta_keywords}}: Always empty.
- {{meta_date}}: The current date.
- {{meta_generator}}: This is always "ScreenSteps".
- {{title}}: The title of the manual or article that can appear within the <body>.
- {{content}}: Replaced with the manual or article content.
- {{t_of_c}}: Replaced with the table of contents for the document.
- {{small_image}}: Maximum height for images that are classed as *small*.
- {{medium_image}}: Maximum height for images that are classed as *medium*.
- {{large_image}}: Maximum height for images that are classed as *large*.
- {{xlarge_image}}: Maximum height for images that are classed as *xlarge*.
- {{max_logo_height}}: The maximum height for the logo in points.
- {{template_folder}}: The path to the template folder. Can be used in CSS where relative image paths won't resolve properly.
- {{working_page_height}}: The size of the PDF page where content can appear.

Example HTML files with content

This example shows the HTML that Clarify generates for documents You can use the example to see the classes and ids assigned to different elements. Note that the images used are not included.

clarify-pdf-article.zip

This example shows the normal output that Clarify generates when exporting a PDF.

Download clarify-pdf-article.zip

clarify-pdf-article-tofc.zip

This example shows the output that Clarify generates when the {{t_of_c}} variable is included in a template.

Download clarify-pdf-article-tofc.zip

How PDF templates are processed

Prince

When a document is exported from Clarify the template being used is processed and all content is inserted into the template. The root folder for any relative links in the HTML file will be the **./Content** folder in the template. This is where you can put any supporting images or CSS files.

After the HTML, CSS, and image files have been generated they are passed off to an application named **Prince**. Prince processes the HTML and creates the resulting PDF document. For full documentation on Prince and what it supports please visit their site where you will find a User Guide and Forums.

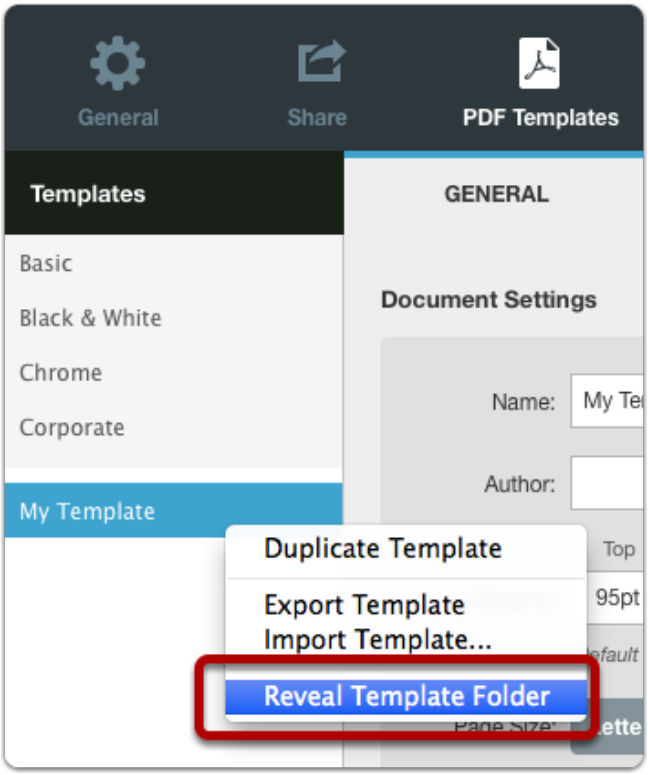
http://www.princexml.com

After the HTML files have been processed and the PDF file generated, the files are deleted from the computer.

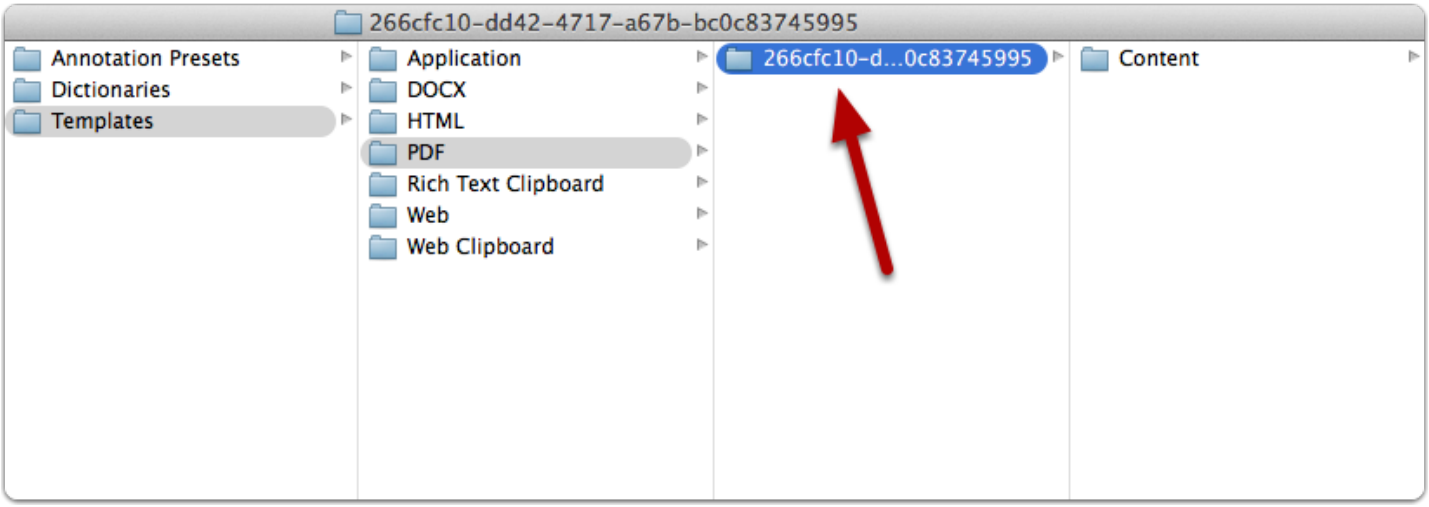
Modifying a PDF template by hand

Locating custom PDF templates

To modify a PDF template by hand beginning by creating a custom template and then locating the template folder. To locate the template folder right-click on the template and select **Reveal Template Folder** from the contextual menu.



The template folder will be revealed in the Finder (Mac) or Windows Explorer (Windows). You can then begin editing the files.



Example PDF Templates

Adding a table of contents to the front of the document

A PDF template can be modified to display a table of contents at the beginning of the document.

Download Table of Contents template

table-of-contents

In order to add the table of contents a ** ** with an id of **table-of-contents** was added to the **article-layout.html** file. It displays the PDF template variable **{{t_of_c}}**.

Custom CSS

The following CSS is then added to the Custom CSS tab. The classes referenced in the CSS are output by the {{t_of_c}} template variable.

```
#table-of-contents {
  /* Name the page with this div so we can target header/footer and other props with
@page */
  page: table-of-contents;

  /* So we don't get too close to the header/footer */
  padding-bottom: 8pt;
}
```

```
#table-of-contents .table-of-contents-title {
  /* The text for the <h1> tag displaying the t of c title */
  content: 'My Table of Contents';
}

#table-of-contents a {
  text-decoration: none;
  color: rgb(0,0,0);
}

#table-of-contents .step-tofc {
}

#table-of-contents .sub-step-tofc {
  margin-left: 50px;
}

#table-of-contents a::after {
  content: leader(".") target-counter(attr(href), page);
}
```