

Default

Christian Külker

2020-04-26

Contents

| | | |
|----------|--|----------|
| 1 | Site cfg.yaml Example | 1 |
| 1.1 | page_type | 1 |
| 1.2 | Example Values For 3 Themes | 2 |
| 1.3 | keyword_page_type | 2 |
| 1.4 | index_list_type | 2 |
| 1.5 | Siblings | 2 |
| 1.6 | Keywords | 2 |
| 2 | l10n_locale | 3 |
| 3 | pdf_create | 3 |
| 4 | css_local | 3 |
| 4.1 | Example Theme Support Template Snippet | 3 |
| 5 | js_local | 4 |

Some values need to have pre-defined values, so called default values. [Pankyll](#) explicitly supports for some values that this values are set to a default value for the whole web site.

1 Site cfg.yaml Example

```
default:
  page_type:      doc
  keyword_page_type: keyword
  index_list_type: list
  l10n_locale:    en_US
  pdf_create:     False
```

```
css_local:      False
js_local:       False
```

1.1 page_type

Every theme has different kind of web pages in its template layout directory. Usually the front matter defined with the key `page_type` the kind of template to be used. For example the `page_type: doc` uses the `themes/NAME/layouts/doc/index.html` as a template to render the page. (Replace `NAME` with name of the used [theme](#). There are cases in which almost all pages of a given web site are of one type. Rather forcing to add a `page_type` in every page front matter, this default value sets the page type for pages without a `page_type` in the front matter. The possible values are dependent on the theme. Every theme can define as many page types as the theme wants.

1.2 Example Values For 3 Themes

| Theme | Page Types |
|----------------|---------------------------|
| newspaper | title, keyword, doc, blog |
| rankle | title, keyword, doc, blog |
| simplicissimus | title, keyword, doc, blog |

1.3 keyword_page_type

Similar to the `page_type` one can set a default type for keyword pages. Usually the `page_type` is set for `index.md` files. While also for every keyword, like **tags** or **categories** also `index.md` files are created automatically, this files can have a different default type determined by `keyword_page_type`. This is because the page of a keyword usually need to have different kind of lists to show all keywords, that are not practically on `title` pages for example. It is possible to set the value to the all values of `page_type`, see above section for details.

1.4 index_list_type

On some pages you would like to have an automatically generated link list of sibling pages. This can be done in two ways. (1) creating a directory without and `index.md` page. In this case the `index.md` page will be created with the default list type specified by

`index_list_type:`. (2) You create an `index.md` file or any other file inside a directory containing sibling pages and define `list: True` or `cards: True` inside the front matter. Also in cases a list will be printed if this option matches the `index_list_type:` value. For now [Pankyll](#) supports 3 kind of listings, of which only two can be used as values: either `'list'` or `cards`. A theme can but do not need to support any list.

1.5 Siblings

- list - simple list for title, doc, and blog pages
- cards - a different kind of layout list for title, doc and blog pages

1.6 Keywords

- keyword - a special list for keywords

2 l10n_locale

As [Pankyll](#) support multilingual pages, it is assumed that a page has at least one language. The key `l10n_locale:` require the name of the locale, that is basically the top level directory of your content folder. If you would like to create a German page your content directory might look like this `content/de_DE/index.md`, than `de_DE` is the default locale and you should set `l10n_locale: de_DE`. If you would like to create an English page your content directory might look like this `content/en_US/index.md`, than `en_US` is the default locale and you should set `l10n_locale: en_US`.

3 pdf_create

[Pankyll](#) supports the creation of [PDF](#) version if the [Markdown](#) page via [Pandoc](#). Some pages would like to have a link pointing to download the [PDF](#) version of a page, some sites would not. Also when developing a new web site, the creation of [PDF](#) pages takes time. To speed up the development, the creation can be globally enabled or disabled for the site. As a remark, for now [Pankyll](#) do not create [PDF](#) documents for `index.md` pages. Just use a different page name for now. To enable [PDF](#) creation set the key to `True` like this: `pdf_create: True` and to disable it to: `pdf_create: False`. Keep in mind that [YAML](#) is case sensitive. `pdf_create: true` might not work.

4 css_local

Some themes (or your site) might offer the delivery of [CSS](#) files locally as in contrast to point to some arbitrary third party web servers. There are advantages and disadvantages for this. `css_local: True` might deliver some CSS locally from some themes or sites, while `css_local: False` do not. You have to look into your theme to see if it supports it.

4.1 Example Theme Support Template Snippet

```
{% if cfg.default.css_local %}
  <link rel="stylesheet" href="/css/hljs/highlight-default.min.css">
{% else %}
  <link rel="stylesheet" href="//SOME_SERVER/styles/default.min.css">
{% endif %}
```

For this to work locally you have (1) copy `highlight-default.min.css` (obtained from some web site) to `themes/NAME/static/css/highlight-default.min.css` and (2) set `css_local: True` in `cfg.yaml`.

5 js_local

This option is basically the same as `css_local`, but just for [javascript](#) files. See section [css_local](#) for details.