

# Pankyll Theme Newspaper

Christian Külker

2020-04-26

## Contents

<b>1 Prerequisites</b> . . . . .	<b>1</b>
<b>2 Installation</b> . . . . .	<b>1</b>
<b>3 Usage</b> . . . . .	<b>2</b>
<b>4 Features Of The Theme</b> . . . . .	<b>2</b>
<b>5 Pankyll Features Supported By The Theme</b> . . . . .	<b>2</b>
<b>6 Features In Detail</b> . . . . .	<b>3</b>
6.1 Inline Code Highlight . . . . .	3
<b>7 Feature: Description - Optional</b> . . . . .	<b>3</b>

This page describes briefly the aim and content for [pankyll-theme-newspaper](#). The goal of [pankyll-theme-newspaper](#) is to show the configuration and usage of [pankyll-theme-newspaper](#) by providing working example with a directory tree that can be used as base to build up a site. The lightweight theme's main feature is to use columns to display standard text. After one is used to read columns, one can read columns faster, as the eye do not lose track of the line so often. However this [theme](#) do not look good on short texts. The length of a paragraph should be considerably.

- [source](#)
- [git-url](#)

## 1 Prerequisites

The [pankyll-theme-newspaper](#) needs the static site generator [Pankyll](#).

## 2 Installation

Assuming you have the directory `project` to create a `Pankyll` site.

Place the content of the [git repository](#) under the directory `project/themes` for example like so:

```
cd project
mkdir themes
cd themes
git clone --recursive
  https://github.com/ckuelker/pankyl-theme-newspaper.git
```

## 3 Usage

The [Pankyll theme Newspaper](#) provides three document types (1) `title`, (2) `blog` and (3) `doc`. The document type `title` is probably used only once for the project's title page. However in this repository it is also used for other pages. The page type `title` has navigational content listed on the right side in from of a table of content. While standard news articles should have the type `blog` all other, presumably automatically created, pages should either have no type assignment (the default assignment is `doc`) or the assignment `doc`. The `doc` page type has a either a keyword listing or a siblings page listing at the bottom. As mentioned above the main feature of the [theme](#) are columns. This columns require larger text bodies of several hundreds words before looking considerably good. In case there is a paragraph that is rather short the [Markdown](#) environment for citation can be used. This environment has 'greater as' signs at the left page border.

## 4 Features Of The Theme

- [HTML](#) meta keywords via [Markdown](#) front matter
- Column Based Layout
- Imprint possible
- Inline Code Highlight
- Document types: title, doc, blog

## 5 Pankyll Features Supported By The Theme

- [Markdown](#) front matter
- Title from `cfg.yaml`, [Markdown](#) front matter, ..
- Author from [Markdown](#) front matter

- Date from [Markdown](#) front matter or git
- [Git](#) info
- Keywords (tags and categories in the example)
- [PDF](#) download
- Code fence syntax highlighting
- Plain code fences
- Breadcrumbs

## 6 Features In Detail

### 6.1 Inline Code Highlight

The 'Inline Code Highlight Feature' brings inline `code` to the foreground by changing size and the background color to grey. For this to work the [HTML](#) head needs to source in (a) `jquery` and (b) the `js/parent.js` `jquery` script that changes the class of the [HTML](#) `<code>` tag to `<code class="code-inline">` which is then referenced by `css/default.css`. This works in the following some but not all environments.

#### Environments:

- text paragraph (p)
- list (li)

#### jQuery:

```
$( "p" ).find("code").addClass("code-inline");
$( "li" ).find("code").addClass("code-inline");
```

#### CSS:

```
.code-inline {
  background-color: rgba(0,0,0,.05);
  font-size: 90%;
  color: rgba(0,0,0,.9);
  padding: .2em .4em;
}
```

## 7 Feature: Description - Optional

```
description:
  en_US:
```

```
text: Pankyll Documentation
de_DE:
text: Pankyll Dokumentation
ja_JP:
text: パンキュルのドキュメンテーション
```