

MM-DOCS

Full documentation

 **Publication Date** : 2025-06-03

This box will disappear when printing

[mkdocs-print-site-plugin](#)

Content

- [Home](#)
- [Demonstration](#)
- [Extra](#)
 - [Variables](#)
 - [Revision](#)
- [Download](#)

MM-DOCS

This project is used to produce all forms of documentation for the service using Markdown and other repository friendly markup languages.

The documentation is created using the mm-docs docker bundle:

- <https://github.com/majkinetor/mm-docs>

Features

- Write documentation in extended Markdown, HTML, PlantUML and other markup languages
- Beautiful responsive design using material theme
- Navigational bar, TOC, search, internationalization, and various rich presentation functionalities
- Rich diagrams and interface mockups using PlantUML
- Macros can be written in python
- PDF export for each page or full site

- Single page with entire documentation
- Live reload during editing
- GitLab/GitHub repository connection for editing within a browser
- Automated AppVeyor build and deploy to GitHub pages using Ubuntu image
- Cross platform for hosting and development

Prerequisites


- Docker
- PowerShell (optional, developer friendliness)
 - [Invoke-Build](#) module

Quick start

To edit documentation:

- Clone/use [new project template](#)
- Add new documents somewhere in the `source\docs` hierarchy.
- Add document path to TOC (file: `source\mkdocs.yml` ; yml key: `nav`) to make page visible in the navigation section, otherwise, the page is available via direct link.

Start development server - run in PowerShell `Invoke-Build Run` . The command is syntax sugar for running docker containers. To build static site run `Invoke-Build Build` .

 **Last Change:** 2025-06-03 11:43

 **Publication Date:** 2025-06-03 11:44

demo

mm-docs system

test

Demonstration page

[mm-docs](#) demonstration page.

For more details about included tools see:

- [Mkdocs](#)
- [Mkdocs Material Extensions](#)
- [PlantUML](#)
- [MathJax](#)

- [Jinja](#)

Customization

- [Color palettes](#)
- [Language](#)
- [Extending theme](#)

Text

Unicode: Мислим дакле постојим

Basic effects

Output	Code
Bold text	<code>**Bold text**</code> or <code>__Bold text__</code>
<i>Italic text</i>	<code>*Italic text*</code> or <code>_Italic text_</code>
Strike Through	<code>~~Strike Through~~</code>
Colored text	<code>==Colored text==</code>
Inline code	<code>`Inline code`</code>

Subscript and superscript:

- H~2~0
- A^T^A

- H₂O
- A^TA

Buttons

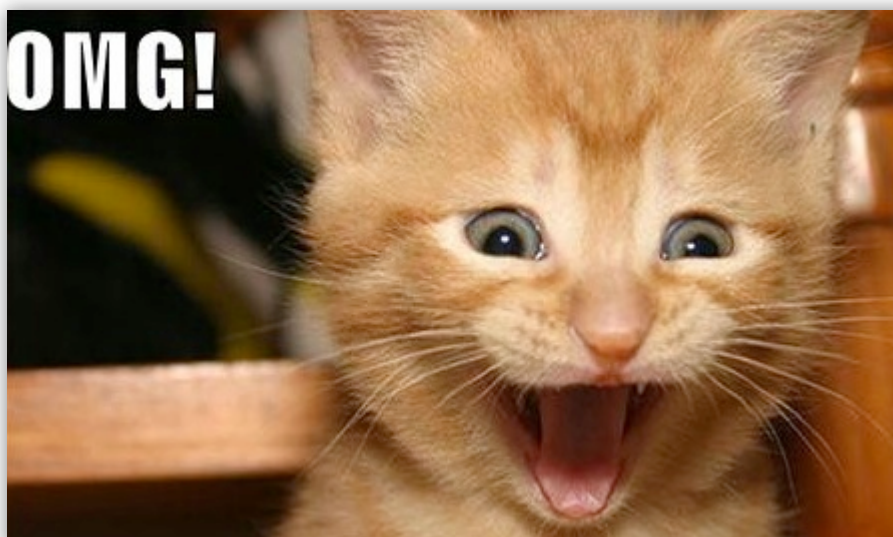
Adding button

```
[Subscribe to our newsletter](#){ .md-button }
```

[Subscribe to our newsletter](#) ↗

Local files and images

```
![omg-lolcat](files/omg-lolcat.jpg)
```



To align image and add some text next to it you must put in div with class `result` :

Aligned image

```
<div class="result" markdown>  
![omg-lolcat](files/omg-lolcat.jpg){ align=left }  
</div>
```



Some OMG cat is displayed here.

Some more text here...

Hotkeys

Output	Code
	<code>++enter++</code>
	<code>++"PgDown"++</code>
	<code>++"Non Existent Key"++</code>

Links, footnotes and comments

Links

Output	Code
An inline-style link	<code>[An inline-style link](https://www.google.com)</code>
An inline-style link with title	<code>[An inline-style link with title](https://www.google.com "Google's Homepage")</code>
Reference-style link	<code>[Reference-style link][arbitrary reference text] [arbitrary reference text]: https://www.mozilla.org</code>
Relative reference to a repository file	<code>[Relative reference to a repository file](../about/overview)</code>
Numbers for reference-style link definitions	<code>[Numbers for reference-style link definitions][1] [1]: http://slashdot.org</code>
Use the link text itself	<code>Use the [link text itself] [link text itself]: http://www.reddit.com</code>

Magic links and emails: turned to links as recognized

Output	Code
http://www.google.com	<code>http://www.google.com</code>
majkinetor@gmail.com	<code>majkinetor@gmail.com</code>

Output	Code
www.google.com	<code>www.google.com</code>

Footnotes

```
A text with a short footnote[^short]
[^short]: https://en.wikipedia.org/wiki/Note_(typography)
```

A text with a short footnote¹

```
A text with a long footnote[^long]
[^long]:
    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod
    massa, nec semper lorem quam in massa.

    ...
code
```

A text with a long footnote²

Comments

```
The text contains 2 comments bellow this line which should not be visible.
[comment]: # (Developed using Visual Studio Code with plantuml extension: cinst
visualstudiocode; code --install-extension jebbs.plantuml)
[comment]: # (PlantUML version may influence diagrams. This document is developped with
1.2017.15: cinst plantuml --version 1.2017.15)
```

The text contains 2 comments bellow this line which should not be visible.

Citations

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa. Inline:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

```
code in citations
```

Another citation line

HTML

HTML is allowed without restrictions. Custom styles are added in `docs_css\extra.css` file.

```
<div style='background-color: #fdd; padding:1em; text-align: center ' >
  <font size="20">
    <span class="yellow bred pl1 pr1">cookies</span>
  </font>
</div>
```

cookies

Critic and comments

Here is some ~~incorrect~~ Markdown. I am adding this here. Here is some more ~~text that I am removing~~text. And here is even more text that I am adding. - Paragraph was deleted and replaced with some spaces.

Spaces were removed and a paragraph was added.

And here is a comment on **some text**/* This works quite well. I just wanted to comment on it. */. Substitutions is are great!

General block handling:

- ~~test remove~~
- ~~test remove~~

- test add
- test add

Emoji and icons

Use more than 7.000 icons and thousands of emojis in your project documentation with zero effort. Custom icons can be added and used in `mkdocs.yml`, documents and templates.

- [More info](#)
- [List of emoji codes](#)

Examples:

- `:tada:` 🎉
- `:jack_o_lantern:` 🎃
- `:alien:` 👽
- `:robot_face:` 🤖
- `:smile:` 😊
- `:heart:` ❤️
- `:thumbsup:` 👍



- 🕸 – `:material-account-circle:` (icon in `.icons/material/account-circle.svg`)
- 😄 – `:fontawesome-regular-face-laugh:` (icon in `.icons/fontawesome/regular/face-laugh.svg`)
- [🔗 In a link](#)

Lists

Ordered list

1. List item 1
2. List item 2
 - a. List item 2.1
 - b. List item 2.2
 - i. List item 2.2.1
 - ii. List item 2.2.2
 - c. List item 2.3
3. List item 4

Unordered, mixed

- List item 1
- List item 2
 - List item 2.1
 - List item 2.2
 - a. List item 2.2.1
 - b. List item 2.2.2

- List item 2.3
- List item 4

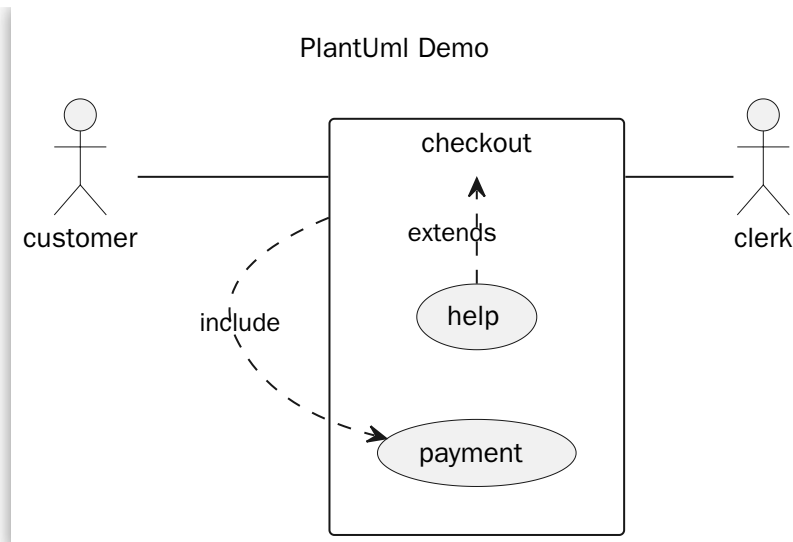
Tables

Col1	Col2
row1.1	row1.2
row2.1	row2.2-1 row2.2-2

Plantuml

Add `plantuml` in a block code header to set code section as PlantUml source. Use `format` to specify output type:

```
```plantuml format="svg"
title PlantUml Demo
left to right direction
skinparam packageStyle rectangle
actor customer
actor clerk
rectangle checkout {
 customer -- (checkout)
 (checkout) .> (payment) : include
 (help) .> (checkout) : extends
 (checkout) -- clerk
}
...
```
```

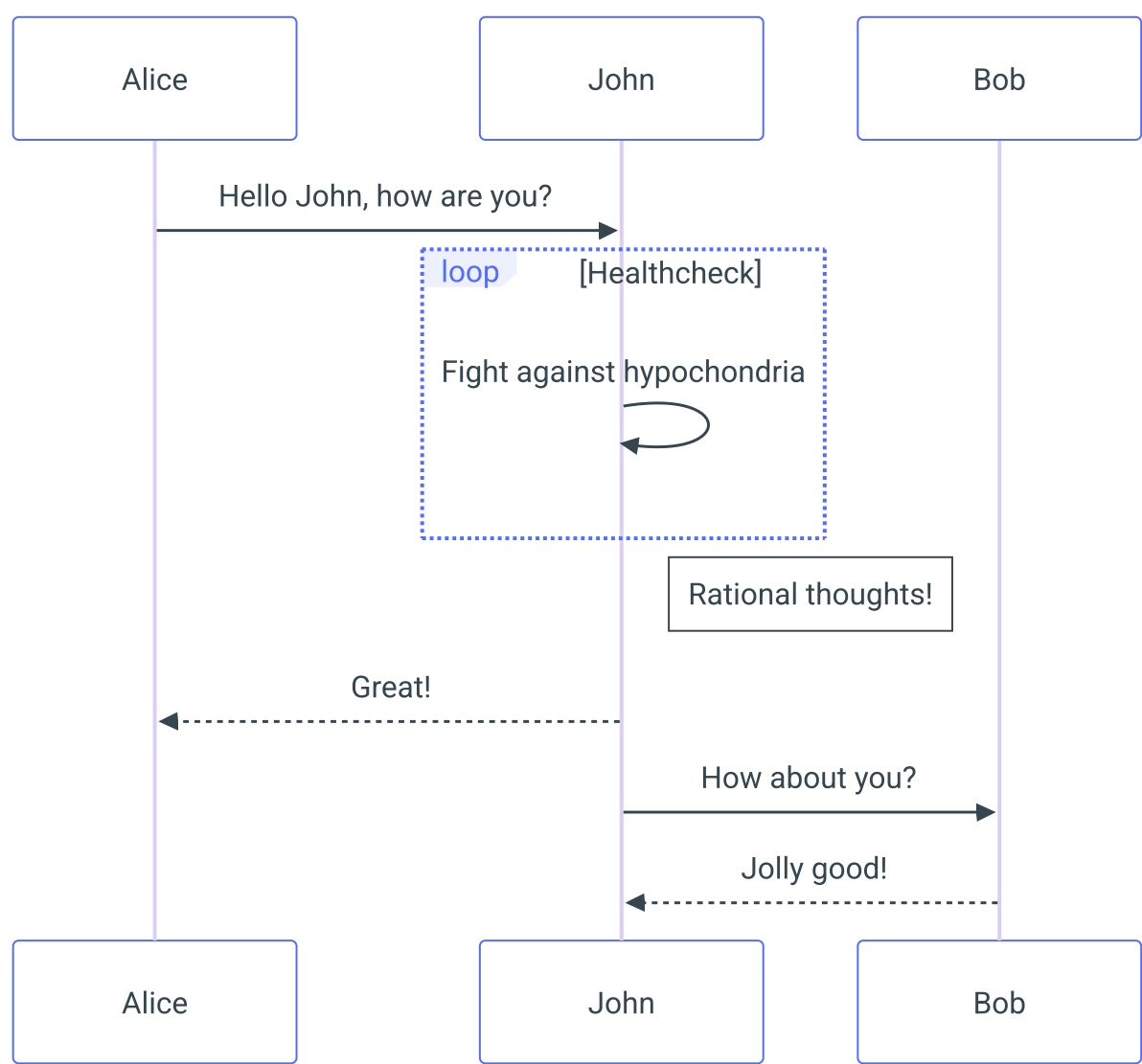


Mermaid

```

```mermaid
sequenceDiagram
 Alice->>John: Hello John, how are you?
 loop Healthcheck
 John->>John: Fight against hypochondria
 end
 Note right of John: Rational thoughts!
 John-->>Alice: Great!
 John->>Bob: How about you?
 Bob-->>John: Jolly good!
```

```



MathJax

The block code:

```
...
$$
\frac{n!}{k!(n-k)!} = \binom{n}{k}
$$
...
```

$$\frac{n!}{k!(n-k)!} = \binom{n}{k}$$

Inline math expressions: $p(x|y) = \frac{p(y|x)p(x)}{p(y)}$


Inline math expressions: $p(x|y) = \frac{p(y|x)p(x)}{p(y)}$

NOTE:



- This doesn't render corretly in PDF output, see [this](#) issue for explanation.

Admonition



[More info](#)

 **Note**


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

 **Collapsed closed** 

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

 **Collapsed open** 

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

 **Extreme danger**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

some code

Some citation

Code Highlight

[More info](#)

```
import tensorflow as tf
```

With tabs:

Bash

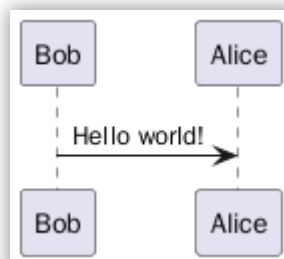
```
#!/bin/bash

echo "Hello world!"
```

PowerShell

```
Write-Host "Hello world!"
```

PlantUML



C

```
#include <stdio.h>

int main(void) {
    printf("Hello world!\n");
}
```

C++

```
#include <iostream>

int main() {
    std::cout << "Hello world!" << std::endl;
    return 0;
}
```

C#

```
using System;

class Program {
    static void Main(string[] args) {
        Console.WriteLine("Hello world!");
    }
}
```

Abbreviations

The following are defined in single file, `_inc\abbr.md` and included on each page.

- `IT, XML, JSON`

Smart symbols

- `(tm)` ™
- `(c)` ©
- `c/o` %
- `+/-` ±
- `-->` →
- `<--` ←
- `<-->` ↔
- `=/=` ≠
- `1/4` ¼
- `1st` 1st

Task list

- ✓ Lorem ipsum dolor sit amet, consectetur adipiscing elit
- ✓ Curabitur elit nibh, euismod et ullamcorper at, iaculis feugiat est
- ✓ Vestibulum convallis sit amet nisi a tincidunt
 - ✓ In hac habitasse platea dictumst
 - ✓ Praesent sed risus massa
- ✓ Nulla vel eros venenatis, imperdiet enim id, faucibus nisi

Macros

[More details.](#)

| | |
|----------------------|----------------------------------|
| | |
| Variable from config | <code>extra.version = 0.1</code> |

| | |
|--|---|
| Variable from this page | <code>acme = Acme Company Ltd</code> |
| Macro (defined in <code>main.py</code>) | <code>bar(1) = 9.3 - barbaz(3) = 9</code> |
| Python variable (defined in <code>main.py</code>) | <code>baz = John Doe</code> |
| Dictionary variable from config | <code>company.name = Acme</code> |

Jinja templates

You can do all the work you want with Jinja2, including defining pure macros, conditionals and for loops:

```
{% macro input(name, value='', type='text', size=20) -%}
  <input type="{{ type }}" name="{{ name }}" value="{{ value }}" size="{{ size }}">
{%- endmacro %}

{% input('username') %}
{% input('password', type='password') %}
```

Includes

You may use the include directive from jinja2, directly in your markdown code e.g.:

```
## Paragraph
{%include 'snippet.md' %}
{%include 'html/content1.html' %}
```

The root directory for your included files is in `docs_dir`.

You can also include [parts of the file](#).

Environment variables

Environment variables can be used inside of configuration file `mkdocs.yaml`. [More...](#)

```
site_name: !ENV CI_PROJECT_PATH
```

1. [https://en.wikipedia.org/wiki/Note_\(typography\)](https://en.wikipedia.org/wiki/Note_(typography)) ←
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod massa, nec semper lorem quam in massa.

code



Extra

Variables

Context

```
[(0, 'Page', "Page(title='Home', url='/index.html')"), (1, 'Page', "Page(title='Demonstration', url='/demo.html')"), (2, 'Section', "title = 'Extra', children = [Page(title='Variables', url='/vars.html'), Page(title='Revision', url='/revision.html')], _Section__active = False"), (3, 'Page', "Page(title='Download', url='/download.html')")]
```

Page

| Variable | Type | Content |
|----------------------|-----------------|---|
| file | <i>File</i> | src_uri = 'vars.md', src_dir = '/docs/source/docs', dest_dir = '/tmp/mkdocs_g7r_h44', use_directory_urls = False, inclusion [<i>InclusionLevel</i>], name = 'vars', dest_uri = 'vars.html', page [<i>Page</i>], url = 'vars.html', abs_src_path = '/docs/source/docs/vars.md' |
| title | <i>str</i> | 'Variables' |
| children | <i>NoneType</i> | None |
| previous_page | <i>Page</i> | Page(title='Demonstration', url='/demo.html') |
| next_page | <i>Page</i> | Page(title='Revision', url='/revision.html') |
| _Page__active | <i>bool</i> | False |
| update_date | <i>str</i> | '2025-06-03' |
| canonical_url | <i>str</i> | 'http://0.0.0.0:8000/vars.html' |
| abs_url | <i>str</i> | '/vars.html' |

| | | |
|---------------------------|-----------------|--|
| edit_url | <i>str</i> | 'https://github.com/majkinetor/mm-docs-template/edit/master/source/docs/vars.md' |
| markdown | <i>str</i> | "# {{ page.title }}\n\n## Context\n\n{{ context(navigation) }}\n\n## Page\n\n{{ context(page) pretty }}\n\n## Config\n\n{{ context(config) pretty }}\n\nFrontmatter var - foo: {{ page.meta.foo }}\n\n{% include 'footer.md' %}" |
| _title_from_render | <i>NoneType</i> | None |
| content | <i>NoneType</i> | None |
| toc | <i>list</i> | [] |
| meta | <i>dict</i> | title = 'Variables', foo = 'bar' |
| parent | <i>Section</i> | title = 'Extra', children = [Page(title='Variables', url='/vars.html'), Page(title='Revision', url='/revision.html')], _Section__active = False |


Config

| Variable | Type | Content |
|-------------------------|-----------------|---|
| config_file_path | <i>str</i> | '/docs/source/mkdocs.yml' |
| site_name | <i>str</i> | 'MM-DOCS' |
| nav | <i>list</i> | [{'Home': 'index.md'}, {'Demonstration': 'demo.md'}, {'Extra': [{'Variables': 'vars.md'}, {'Revision': 'revision.md'}]}, {'Download': 'download.md'}] |
| pages | <i>NoneType</i> | None |
| exclude_docs | <i>NoneType</i> | None |
| draft_docs | <i>NoneType</i> | None |
| not_in_nav | <i>NoneType</i> | None |
| site_url | <i>str</i> | 'http://0.0.0.0:8000/' |
| site_description | <i>str</i> | 'MM-DOCS Template' |
| site_author | <i>str</i> | 'majkinetor' |
| theme | <i>Theme</i> | Theme(name='material', dirs=['/docs/source/overrides', '/usr/lib/python3.12/site-packages/material/templates', '/usr/lib/python3.12/site-packages/mkdocs/templates'], |

| | | |
|---------------------|-----------------|---|
| | | static_templates={'404.html', 'sitemap.xml'}, name='material', locale=Locale('en'), language='en', direction=None, features=['navigation.expand', 'navigation.top', 'navigation.indexes', 'navigation.footer', 'content.action.edit', 'search.highlight', 'search.suggest'], font={'text': 'Roboto', 'code': 'Roboto Mono'}, icon=None, favicon='assets/images/favicon.png', palette=[{'scheme': 'default', 'toggle': {'icon': 'material/lightbulb-outline', 'name': 'Switch to dark mode'}}, {'scheme': 'slate', 'toggle': {'icon': 'material/lightbulb', 'name': 'Switch to light mode'}})] |
| docs_dir | str | '/docs/source/docs' |
| site_dir | str | '/tmp/mkdocs_g7r_h4a4' |
| copyright | str | '© majkinetor 2019' |
| google_analytics | NoneType | None |
| dev_addr | _IpAddressValue | _IpAddressValue(host='0.0.0.0', port=8000) |
| use_directory_urls | bool | False |
| repo_url | str | 'https://github.com/majkinetor/mm-docs-template' |
| repo_name | str | 'GitHub' |
| edit_uri_template | NoneType | None |
| edit_uri | str | 'edit/master/source/docs/' |
| extra_css | list | ['_extra/css/extra.css', '_extra/css/pdf.css'] |
| extra_javascript | list | ['_extra/js/mathjax.js', '_extra/js/tablesorth.js', 'https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-mml-cthtml.js', 'https://unpkg.com/tablesorth@5.3.0/dist/tablesorth.min.js'] |
| extra_templates | list | [] |
| markdown_extensions | list | ['toc', 'tables', 'fenced_code', 'abbr', 'admonition', 'attr_list', 'codehilite', 'def_list', 'footnotes', 'md_in_html', 'meta', 'sane_lists', 'pymdownx.arithmatex', 'pymdownx.betterem', 'pymdownx.caret', 'pymdownx.critic', 'pymdownx.details', 'pymdownx.emoji', 'pymdownx.inlinehilite', 'pymdownx.keys', 'pymdownx.magiclink', 'pymdownx.mark', 'pymdownx.superfences', 'pymdownx.smartsymbols', 'pymdownx.tabbed', 'pymdownx.tasklist', 'pymdownx.tilde', 'markdown_include.include', 'plantuml_markdown'] |

| | | |
|---------------|------------------|--|
| mdx_configs | dict | toc [dict], pymdownx.arithmatex [dict], pymdownx.betterem [dict], pymdownx.emoji [dict], pymdownx.superfences [dict], pymdownx.tabbed [dict], pymdownx.tasklist [dict], markdown_include.include [dict] |
| strict | bool | False |
| remote_branch | str | 'gh-pages' |
| remote_name | str | 'origin' |
| extra | LegacyConfig | {'url_prefix': 'http://localhost:8000', 'version': 0.1, 'company': {'name': 'Acme'}, 'social': [{'icon': 'dummy value to show social override'}], 'polyfills': ['https://unpkg.com/iframe-worker/shim']} |
| plugins | PluginCollection | material/offline [OfflinePlugin], exclude [Exclude], macros [MacrosPlugin], material/search [SearchPlugin], material/tags [TagsPlugin], pdf-export [PdfExportPlugin], print-site [PrintSitePlugin] |
| hooks | dict | |
| watch | list | [] |
| validation | Validation | {'nav': {'omitted_files': 20, 'not_found': 30, 'absolute_links': 20}, 'links': {'not_found': 30, 'absolute_links': 20, 'unrecognized_links': 20, 'anchors': 20}} |

Frontmatter var - foo: bar

 Last Change: 2025-06-03 11:43

 Publication Date: 2025-06-03 11:44

Revisions

Build Date: 2025-06-03 11:44:29

[View revision.json](#)

| Date | Path |
|---------------------|---------|
| 2025-06-03 11:43:37 | demo.md |
| 2025-06-03 11:43:37 | tags.md |

| Date | Path |
|---------------------|-------------------|
| 2025-06-03 11:43:37 | vars.md |
| 2025-06-03 11:43:37 | index.md |
| 2025-06-03 11:43:37 | download.md |
| 2025-06-03 11:43:37 | about/overview.md |

Download

You can consume this documentation off-line via:

1. PDF that you can [download](#)
2. [Standalone single HTML page](#) that you can save
3. [Entire site in ZIP archive](#) that you can unpack and use locally starting from `index.html`

 **Last Change:** 2025-06-03 11:43

 **Publication Date:** 2025-06-03 11:44