

mantodea

/mantodea/

v0.1.3

2024-05-28

MIT

A template & package for specification-like documents.

Typst Community Contributors <@TYPST-COMMUNITY>

<https://github.com/typst-community/mantodea>

MANTODEA provides a template and tools for writing documents targeted at Typst developers, such as convenient bindings to Typst documentation pages, example rendering or source forge links. It is primarily used as a package and template for technical documents such as guidelines, documentation or specifications.

Table of contents

I. Manifest

II. Reference

II.1. Stability	3
II.2. Custom types	3
II.2.1. theme	3
II.3. Modules	3
II.3.1. mantodea	3
II.3.2. style	5
II.3.3. component	6
II.3.4. component/table-of-contents ..	7
II.3.5. component/title-page	8
II.3.6. theme	9
II.3.7. util	10
II.3.8. util/author	11
II.3.9. util/exmaple	13
II.3.10. util/link	15

III. Index

Part I.

Manifest

MANTODEA aims to be the following:

- simple to use
 - the default template should provide an good user experience
- unsurprising
 - parameters should have sensible names and behave as one would expect
- general
 - provided features are not for specific use cases, but rather baseline utilities for more specific use cases

If you think its behavior is surprising, you believe you found a bug or you think its defaults or parameters are not sufficient for your use case, please open an issue at [GitHub:typst-community/mantodea](https://github.com/typst-community/mantodea). Contributions are also welcome!

MANTODEA is a fork of **MANTYS**¹, a package for creating package documentation.

¹<https://github.com/jneug/typst-mantys>

Part II.

Reference

II.1. Stability

This package tries to adhere to semantic versioning², please report unintended API breakage at [GitHub:typst-community/mantodea](https://github.com/typst-community/mantodea).

II.2. Custom types

II.2.1. theme

Most functions take a theme argument, this type is a dictionary containign various style information to override, mainly colors and font configuration. The default can be found at `mantodea.theme.default` and serves as the defacto schema.

II.3. Modules

II.3.1. mantodea

The package entry point, containing the top level sub modules and the template main function.

```
#mantodea()
```

```
#mantodea(  
  <title>: "[Title]",  
  <subtitle>: "[Subtitle]",  
  <authors>: "John Doe <john@doe.com>",  
  <urls>: "https://github.com/typst-community/mantodea",  
  <date>: "datetime(year: 1970, month: 1, day: 1)",  
  <version>: "version(0, 1, 0)",  
  <abstract>: "lorem(100)",  
  <license>: "MIT",  
  <theme>: "theme.default"  
) → function
```

The main template function, this returns a function which is used in a show-all rule.

— Argument —

```
<title>: "[Title]"
```

```
str | content
```

The title for of this document.

²<https://semver.org>

2.3 Modules

Argument

`<subtitle>: "[Subtitle]"`

`str` | `content` | `none`

A subtitle shown below the title.

Argument

`<authors>: "John Doe <john@doe.com>"`

`str` | `content` | `array`

The authors of the document.

Argument

`<urls>: "https://github.com/typst-community/mantodea"`

`str` | `array` | `none`

One or more URLs relevant to this document.

Argument

`<date>: "datetime(year: 1970, month: 1, day: 1)"`

`datetime`

The date at which this document was created.

Argument

`<version>: "version(0, 1, 0)"`

`str` | `version`

The version of this document. A string can be passed explicitly to avoid the automatic `v` prefix.

Argument

`<abstract>: "lorem(100)"`

`str` | `content` | `none`

An abstract outlining the purpose and contents of this document.

Argument

`<license>: "MIT"`

`str` | `content` | `none`

The license of this document or a related piece of intellectual property.

Argument

`<theme>: "theme.default"`

`theme`

The theme to use for this document.

II.3.2. style

Styles for the template, namely the default style in use for the template body.

```
#default()
```

```
#default(<theme>: "_theme.default") → function
```

The default style applied over the whole document.

— Argument —

```
<theme>: "_theme.default"
```

theme

The theme to use for the document styling.

II.3.3. component

Parts of the template's components, while usable independently, they assume certain styles found in the style module.

```
#make-table-of-contents()
```

```
#make-table-of-contents(<title>: "heading(outlined: false, numbering: none)[Table of Contents]", <target>: "heading.where(outlined: true)", <columns>: 1, <theme>: "_theme.default") → content
```

Generate the default table of contents.

Argument

```
<title>: "heading(outlined: false, numbering: none)[Table of Contents]"      str |
content | none
```

The title to use, if this is a heading itself it will not be passed to outline.

Argument

```
<target>: "heading.where(outlined: true)"      selector | function
```

The target to show, this is primarily used to restrict the search space of the default selector. If something other than a heading selector is passed, then this may not work.

Argument

```
<columns>: 1      int
```

The amount of columns to use for the outline.

Argument

```
<theme>: "_theme.default"      theme
```

The theme to use for this table of contents.

II.3.4. component/table-of-contents

Contains an outline implementation with improved alignment.

```
#make-table-of-contents()
```

```
#make-table-of-contents(<title>: "heading(outlined: false, numbering: none)[Table  
of Contents]", <target>: "heading.where(outlined: true)", <columns>: 1, <theme>:  
"_theme.default") → content
```

Generate the default table of contents.

—Argument—

```
<title>: "heading(outlined: false, numbering: none)[Table of Contents]"      str |  
content | none
```

The title to use, if this is a heading itself it will not be passed to outline.

—Argument—

```
<target>: "heading.where(outlined: true)"      selector | function
```

The target to show, this is primarily used to restrict the search space of the default selector. If something other than a heading selector is passed, then this may not work.

—Argument—

```
<columns>: 1      int
```

The amount of columns to use for the outline.

—Argument—

```
<theme>: "_theme.default"      theme
```

The theme to use for this table of contents.

II.3.5. component/title-page

Contains the title page component.

```
#make-table-of-contents()
```

```
#make-table-of-contents(<title>: "heading(outlined: false, numbering: none)[Table  
of Contents]", <target>: "heading.where(outlined: true)", <columns>: 1, <theme>:  
"_theme.default") → content
```

Generate the default table of contents.

— Argument —

```
<title>: "heading(outlined: false, numbering: none)[Table of Contents]"      str |  
content | none
```

The title to use, if this is a heading itself it will not be passed to outline.

— Argument —

```
<target>: "heading.where(outlined: true)"      selector | function
```

The target to show, this is primarily used to restrict the search space of the default selector. If something other than a heading selector is passed, then this may not work.

— Argument —

```
<columns>: 1      int
```

The amount of columns to use for the outline.

— Argument —

```
<theme>: "_theme.default"      theme
```

The theme to use for this table of contents.

II.3.6. theme

Theming related values, namely the default style dictionary used when not style is given.

II.3.7. util

General utilities.

```
#alert()
```

#alert(**<color>**: "purple", **<size>**: 0.8em, **..)[**<body>**] → **content****

Draw an alert box.

— Argument —

<color>: "purple"

color

The color to use for the box and text color.

— Argument —

<size>: 0.8em

size

The text size to use.

— Argument —

..

any

Additional arguments to pass to showybox.

— Argument —

<body>

content

The box body.

#hint

Draw a blue alert box. See alert.

#info

Draw a green alert box. See alert.

#warn

Draw a yellow alert box. See alert.

#error

Draw a red alert box. See alert.

II.3.8. util/author

Utilities for handling author rendering.

`#author()`

`#email()`

`#name()`

`#name(..<first>, <last>, <short>: 1) → content`

Display a person with the given options.

—Argument—

`..<first>`

`str`

The first and middle names.

—Argument—

`<last>`

`str | array`

The last names.

—Argument—

`<short>: 1`

`int`

The shortness level.

- 0: do not shorten any names
- 1: shorten middle names
- 2: shorten middle and first names

`#author(
 ..<first>,
 <last>,
 <short>: 1,
 <label>: none,
 <email>: none
) → content`

Show full author information, see also name.

—Argument—

`..<first>`

`str | none`

The first and middle names.

—Argument—

`<last>`

`str | array`

The last names.

—Argument—

`<short>: 1`

`int`

The shortness level.

- 0: do not shorten any names

- 1: shorten middle names
- 2: shorten middle and first names

—Argument—

`<label>: none``label | none`

The label to use for email attribution, see email.

—Argument—

`<email>: none``str | content | none`

The email to attribute the user to.

#email(`<mark>: auto`, `<label>`, `<body>`) → `content`

Display an email attribution, i.e. a mark that can be linked to from author.

—Argument—

`<mark>: auto``content | auto`

A symbol used for attribution, the linked authors will have this displayed as a superscript. Uses * numbering if auto.

—Argument—

`<label>``label`

The label to use in author for automatic attribution.

—Argument—

`<body>``str | content`

The body to attribute the authors to, this is often a generic email pattern such as "<name>@university.com", but doesn't have to be.

#marks

The marks used in author for automatic mark numbering.

Can be used to add new or alter existing marks, marks come in the form of arbitrary content which is placed as a superscript.

II.3.9. util/exmample

Utilities for rendering examples with and without evaluation of Typst code.

`#code-result()` `#frame()`

`#frame(<theme>: "_theme.default", ..<args>) → content`

Show a source code frame.

—Argument—

`<theme>: "_theme.default"`

theme

The theme to use for this code frame.

—Argument—

`..<args>`

any

The args to pass to showybox.

`#code-result(
 <side-by-side>: false,
 <scope>: "(:)",
 <breakable>: false,
 <theme>: "_theme.default",
 <result>: auto
) [<source>] → content`

Shows example code and its corresponding output in a frame.

—Argument—

`<side-by-side>: false`

bool

Whether or not the example source and output should be shown side by side.

—Argument—

`<scope>: "(:)"`

dictionary

The scope to pass to eval.

—Argument—

`<breakable>: false`

bool

If the frame can brake over multiple pages.

—Argument—

`<theme>: "_theme.default"`

theme

The theme to use for this code example.

—Argument—

`<result>: auto`

content

The content to render as the example result. Evaluated eval.

2.3 Modules

Argument

`<source>`

content

A raw element containing the source code to evaluate.

II.3.10. util/link

Utilities for linking to Typst resources or source forges.

<code>#codeberg()</code>	<code>#github()</code>	<code>#package-repo()</code>
<code>#footlink()</code>	<code>#gitlab()</code>	<code>#type()</code>
<code>#forge()</code>	<code>#package()</code>	

#footlink(<url>, <label>) → content

Creates a link with a footnote.

— Argument —

<url>

str

The url for the link and the footnote.

— Argument —

<label>

str

The label for the link.

#forge(<base>, ..<relative>, <label>: auto) → content

Creates a link with footnote to a source forge and optional repo.

— Argument —

<base>

str

The forge base url.

— Argument —

..<<relative>

str

An optional author or repository path.

— Argument —

<label>: auto

str | auto

The label to use for the forge, uses forges if auto.

#github(<repo>) → content

Creates a link with a footnote to a GitHub repository.

— Argument —

<repo>

str

The repository path.

#gitlab(<repo>) → content

Creates a link with a footnote to a GitLab repository.

— Argument —

<repo>

str

The repository path.

#codeberg(**<repo>**) → **content**

Creates a link with a footnote to a Codeberg repository.

Argument
<repo> **str**
 The repository path.

#package(**<name>**, **<version>**: **none**) → **content**

Creates a link with a footnote to the Typst Universe page of a package.

Argument
<name> **str**
 The name of the package.

Argument
<version>: **none** **version | none**
 The version of the package, this has no effect on the link itself.

#package-repo(**<name>**, **<version>**, **<namespace>**: **"preview"**) → **content**

Creates a link with a footnote to the Typst package repository.

Argument
<name> **str**
 Name of the package.

Argument
<version> **version**
 The version of the package.

Argument
<namespace>: **"preview"** **str**
 The package namespace to use.

#type(**<value>**, **<with-footnote>**: **false**, **<theme>**: **"_theme.default"**) → **content**

Creates a link with a footnote to the documentation of a built-in Typst type.

Argument
<value> **str | any**
 The type or value to which to link to:

- if this is a **str**, it is inferred to be the type representation
- if this is a type, it is inferred to be the expected type
- if this is a value, it is inferred to be a value of the expected type

Argument
<theme>: **"_theme.default"** **theme**
 The theme to use for this type.

#typst

2.3 Modules

The type links supported by `type`.

Can be used to add new or alter existing type links.

The default contains the types built in types, including the special types which contain no page like any, linking to the types page instead.

#**forges**

The forges supported by `forge`.

Can be used to add new, or alter existing matchers, matchers come in the form of a function which takes a url and either returns a `str` or `none` directly, or a boolean. `false` or `none` means the matcher did not match, `true` means use the matcher key, a `str` means use this string directly.

The default matchers include GitHub, GitLab and Codeberg as well as a crude git-subdomain matcher which returns the top and second level domain.

Part III.

Index

A

`#alert` 10

`#author` 11

C

`#code-result` 13

`#codeberg` 16

D

`#default` 5

E

`#email` 12

F

`#footlink` 15

`#forge` 15

`#frame` 13

G

`#github` 15

`#gitlab` 15

M

`#make-table-of-contents` . 6,
7, 8

`#mantodea` 3

N

`#name` 11

P

`#package` 16

`#package-repo` 16

T

`#type` 16