The Mantys Package

Template for documenting **Typst** Packages and Templates

v0.0.1   2023-07-05

A **Typst** template to create consistens and readable manuals for pakcages and templates.

Jonas Neugebauer

https://github.com/jneug/typst-mantys

Weit hinten, hinter den Wortbergen, fern der Länder Vokalien und Konsonantien leben die Blindtexte. Abgeschieden wohnen sie in Buchstabhausen an der Küste des Semantik, eines großen Sprachozeans. Ein kleines Bächlein namens Duden fließt durch ihren Ort und versorgt sie mit den nötigen Regelialien. Es ist ein paradiesmatisches Land, in dem einem gebratene Satzteile in den Mund fliegen. Nicht einmal von der allmächtigen Interpunktion werden die Blindtexte beherrscht – ein geradezu unorthographisches Leben.

Table of contents

Part I.
About

Mantys is a **Typst** package

> Mantys is in active development and its functionality is subject to change.

> Contributions are welcome.

Part II.
Usage

II.1. Loading the package

Currently the package needs to be installed into the local package repository.

Either download the current release from GitHub[1] and unpack the archive into yout system dependent local repository folder or clone it directly:

```shell
git clone https://github.com/jneug/typst-typopts.git typopts-0.0.3
```

After installing the package just import it inside your `typ` file:

```
#import "@local/typopts:0.0.3": options
```

II.2. Available functions

II.2.1. Describing commands

#argument(name, type, default: "__none__")

Sets a command with arguments.

name	Name of the command	content
func	Function to pass the value to.	v => v

Retrieves the value for the option by the given `name` and passes it to `func`, which is a function of on argument.

If no option `name` exists, the given `default` is passed on.

If `final: true`, the final value for the option is retrieved, otherwise the current value. If `loc` is given, the call is not wrapped inside a `locate` call and the given `location` is used.

II.2.2. Source code and examples

MANTYS provides several commands to handle source code snippets and show examples of functionality. The usual `#raw()` command still works, but theses command allow you to highlight code in different ways or add line numbers.

#sourcecode(title: none, file: none)[code]

If provided, the `title` and `file` argument are set as a titlebar above the content.

---

[1]https://github.com/jneug/typst-typopts/releases/latest

code	A #raw() block, that will be set inside a bordered block. The raw content is not modified and keeps its lang attribute, if set.	content
title: none	A title to show above the code in a titlebar.	content
file: none	A filename to show above the code in a titlebar.	content

If provided, the title and file argument are set as a titlebar above the content.

```
#sourcecode(title:"Some Rust code")[
```rust
fn main() {
  println!("Hello World!");
}
```

]
```

Some Rust code

```
fn main() {
println!("Hello World!");
}
```

> The ultimate goal is to provide an #example() command simmilar to the example environment in CNLTX, that is able to typeset the example code and the result in one go. As of now #example() requires you to give it both the source and result as separate arguments.

II.2.3. Utilities

```
#mty.box(header:  none,  footer:  none,  invert-headers:  true,  stroke-color:
rgb("#a60912"), bg-color: rgb("#ffffff"), width: 100%, padding: 8pt, radius:
4pt)[body]
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.