



Markdownreveal Documentation

Release 0.1.0

The Markdownreveal contributors.

Apr 02, 2017

Contents

1	What is Markdownreveal?	1
2	Contents	3
2.1	Usage	3
2.2	Subcommands	5
2.3	Style	6
2.4	Developers	7
2.5	Software License and Disclaimer	8
2.6	Indices and tables	9

CHAPTER 1

What is Markdownreveal?

This tool allows you to create and visualize presentations with simple Markdown notation. It is based on reveal.js and is able to generate the required HTML and refresh your browser view for easier and faster presentation creation.

You can also have a look at our [online presentation example](#) and [its corresponding source code](#).

Usage

Requirements

In order to use Markdown reveal you need:

- Python 3.5 (or higher).
- [Pandoc](#).

And optionally:

- Docker (for exporting to PDF).

Note: Windows and Mac platforms are theoretically supported, but not currently tested by developers. For those platforms you may not even need to manually install Pandoc, as it should be automatically installed.

Installation

Installation is very straight-forward:

```
pip install markdownreveal
```

This will install all the required dependencies and will provide you with the *markdownreveal* command.

First steps

Let us create our first presentation! First, we need to create a simple Markdown file to write our presentation. Let us, for example, create a file *presentation.md* and write the following content:

```
% Presentation title
% Author
% YYYY-MM-DD

# First section

## Subsection

This is my first Markdownreveal presentation.
```

Now, simply execute:

```
markdownreveal presentation.md
```

If everything went well a new tab should be opened in your browser showing you the presentation.

Now, while *markdownreveal* is running, edit your *presentation.md* file and save the changes. Markdownreveal will automatically refresh your browser view for you!

Navigation

Here is a short list of the keys you can use to navigate these presentations:

- Use *SPACE* for next slide.
- Use *MAYUS + SPACE* for next slide.
- Use *ESC* to visualize the slides grid.
- Use arrows to navigate along the grid.
- Use *S* to open the presenter window.

For more information on navigation options, refer to the [official documentation](#).

Notation

The presentation should start with the title, author and date:

```
% Presentation title
% Author
% YYYY-MM-DD
```

You can create vertical sections in your presentation using titles:

```
# New section
```

New slides in a section using subtitles:

```
## Subtitle
```

Simple paragraphs with text lines:

```
This is a paragraph.
```

Simple lists:

```
- List item.
- Another one.
```

Numbered lists (note you can use *I.* for automatic numbering):

```
1. First item.
1. Second item.
```

Force the creation of a new slide:

```
---
```

Code (with optional syntax highlighting):

```
```python
print('Hello world!')
```
```

Images (with optional width):

```
![Alt text](./figures/yourfigure.png){width=70%}
```

Equations (using LaTeX notation):

```
$$ c = \sqrt{a^2 + b^2} $$
```

For more information, refer to the [official documentation](#).

Subcommands

List subcommands

Markdownreveal provides a couple of subcommands for convenience. If you want to list those commands, you can use the `--help` option:

```
markdownreveal --help
```

Note: There is one subcommand called `show`, which is the default when no subcommand is specified. This subcommand will, as you already know, open your web browser to visualize your presentation.

Share your presentation

If you want to share your presentation with somebody, or if you want to upload it to a server, you can use the `zip` subcommand:

```
markdownreveal zip presentation.md
```

This will create a ZIP file containing your presentation. Note that the presentation is a static webpage, so in order to view it you just need to open `index.html` with your web browser. You can also upload it to your own server if you prefer so.

GitHub pages

If you happen to be using GitHub to host your presentation code, then sharing could not be simpler. You can use the subcommand `upload` to upload your presentation to GitHub pages:

```
markdownreveal upload presentation.md
```

Warning: Note that executing this command will reset the branch `gh-pages`, so make sure your repository holds your presentation only or you are not using that branch.

Export to PDF

You can also export your presentation to PDF. This functionality, however, requires [Docker](#), so you need to have it installed and configured in your system.

In order to export your presentation to PDF, use the `pdf` subcommand and provide the URL where your presentation is being served (either the server where you uploaded it or the local server that is spawned when you run Markdownreveal locally and the presentation is opened in your browser:

```
markdownreveal pdf http://localhost:8123/#/
```

A PDF file will be created in the current directory with the presentation.

Note: Docker was chosen for convenience and security. Exporting the presentation to PDF requires tools that otherwise would need to be installed as binaries (insecure) or compiled (inconvenient).

Currently the export process uses [DeckTape](#) and a patched [PhantomJS](#) version. If, at some point, DeckTape is able to work with a non-patched PhantomJS version, we may consider adding support for export without Docker.

For exporting the presentation, Docker is given access only to a temporary folder where it can write the PDF output. Then Markdownreveal copies this file to the current directory once the export is finished.

Clean local files

Markdownreveal downloads `reveal.js` and style files and saves them locally for future use. If you want to remove those files, you can make use of the `clean` subcommand:

```
markdownreveal clean
```

Style

Footer and header

You can very easily add a footer and/or a header to your slides with Markdownreveal. Try to create a file `config.yaml` and write the following:

```
footer: 'My footer | www.mywebsite.com'
```

You should now see a footer in all your slides. Use `header` to add a header too or instead.

Using reveal.js themes

We think the default Markdownreveal theme is okay, but to each their own... If you would rather use the default reveal.js themes, you can do so in a very easy way. In your *config.yaml* file, try to add:

```
style: ''
```

What you see now is the *white* reveal.js style. You can also change that parameter if you want. Try:

```
theme: 'moon'
```

Warning: Note that when using reveal.js themes, *footer* and *header* will not work as they are not defined in the default reveal.js CSS files.

Creating your own style

You can definitely create your own style too! To do so, create a folder named *style* in your presentation root directory. You can add the following files in there:

- *background.svg*: an image to be used as a background.
- *logo.svg*: a logo to display on your slides on a corner.
- *warmup.svg*: an image to be displayed before the presentation title.
- *custom.css*: your CSS rules for your own style.
- *config.yaml*: to tune the style default configuration (footer, header...).

You can check the [Markdownreveal style repository](#) to get an idea on how to create the CSS file. Note that your custom style will be based on reveal.js's *white* theme.

Using your style repository

If you are doing many presentation, or if you work for a company, you may find it useful to create your own style repository. Simply upload your files to a repository, as [we do ourselves with Markdownreveal](#) and then edit your *config.yaml* file:

```
style: 'https://github.com/markdownreveal/style-default/archive/master.tar.gz'
```

Put there an URL pointing to your style files. Note that they need to be contained in a *tar.gz* file.

Developers

Installing dependencies

To install the required dependencies for developing Markdownreveal, you can make use of the provided *requirements.txt* file:

```
pip install -r requirements.txt
```

Running tests

Running the tests locally is very simple, using `Tox` from the top level path of the project:

```
tox
```

That single command will run all the tests for all the supported Python versions available in your system or environment.

For faster results you may want to run all the tests just against a single Python version. This command will run all tests against Python 3.5 only:

```
tox -e py35
```

Note that those tests include style and static analysis checks. If you just want to run all the behavior tests (not recommended):

```
pytest -n 8
```

If you just want to run a handful of behavior tests (common when developing new functionality), just run:

```
pytest -k keyword
```

Note: Before submitting your changes for review, make sure all tests pass with `tox`, as the continuous integration system will run all those checks as well.

Generating documentation

Documentation is generated with Sphinx. In order to generate the documentation locally you need to run `make` from the `docs` directory:

```
make html
```

Software License and Disclaimer

Copyright (c) The Markdownreveal contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of Markdownreveal nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Indices and tables

- [genindex](#)
- [search](#)

A

abstract, 1

C

clean, 6

D

default

style, 6

dependencies

developers, 7

developers, 7

dependencies, 7

documentation, 8

tests, 7

disclaimer, 8

documentation

developers, 8

E

export, 6

F

first steps, 3

footer

header, 6

G

github, 5

H

header

footer, 6

I

installation, 3

L

license, 8

list, 5

local, 6

N

navigation, 4

notation, 4

O

own

style, 7

P

pages, 5

pdf, 6

R

repository

style, 7

requirements, 3

S

share, 5

software license, 8

style, 6

default, 6

own, 7

repository, 7

subcommands, 5

T

tests

developers, 7

U

usage, 3