
Style guide for Sphinx-based documentations

Release 0.1dev

Benoit Bryon

February 10, 2017

1	Contents	3
1.1	Documentation style guide	3
1.2	Contributing	6
1.3	About	7
2	Indices and tables	11
3	References	13

This project is about coding standards for [Sphinx](http://sphinx.pocoo.org)¹-based documentations.
This project focuses on conventions about what [PEP 8](#) calls “style guide”.

¹ <http://sphinx.pocoo.org>

1.1 Documentation style guide

This document focuses on style-guide. It is a kind of coding standards applied to documentation files. It is not about documentation content.

1.1.1 RestructuredText with Sphinx directives

This documentation uses [Python-sphinx](http://sphinx.pocoo.org/)¹, which itself uses [reStructuredText](http://docutils.sourceforge.net/rst.html)² syntax.

1.1.2 Filenames

Use only lowercase alphanumeric characters and `-` (minus) symbol.

Suffix filenames with the `.txt` extension.

Note: Usage of `.rst` extension is not recommended because:

- RST files are human-readable text files. Most systems natively recognize the `.txt` extension and open these files with a text editor. This is a good choice.
- Some programs parse `.rst` with [rst2html](http://docutils.sourceforge.net/docs/user/tools.html#rst2html-py)³, which cannot interpret some Sphinx's directives such as `code-block`. So readers using such programs actually lose some content.

As an example, well known [Github](https://github.com)⁴ platform uses `rst2html` to render `.rst` files in its repository browser. Not only you lose content, you also lose features like links to lines.

- When you need to read HTML builds of the documentation, best practice is to export documentation as static builds with `sphinx-build` command, then host and serve these builds as static files. For public projects, you may have a look at services like [Read the docs](http://readthedocs.org)⁵.

¹ <http://sphinx.pocoo.org/>

² <http://docutils.sourceforge.net/rst.html>

³ <http://docutils.sourceforge.net/docs/user/tools.html#rst2html-py>

⁴ <https://github.com>

⁵ <http://readthedocs.org>

1.1.3 Whitespaces

Indentation

Indent with 2 spaces.

Except:

- `toctree` directive requires a 3 spaces indentation.

Blank lines

Two blank lines before overlined sections, i.e. before H1 and H2. One blank line before other sections. See *Headings* for an example.

One blank line to separate directives.

```
Some text before.  
  
.. note::  
  
    Some note.
```

Exception: directives can be written without blank lines if they are only one line long.

```
.. note:: A short note.
```

1.1.4 Line length

Limit all lines to a maximum of 79 characters.

1.1.5 Headings

Use the following symbols to create headings:

1. # with overline
2. * with overline
3. =
4. -
5. ^
6. "

As an example:

```
#####  
H1: document title  
#####  
  
Introduction text.  
  
*****  
Sample H2
```

```
*****  
  
Sample content.  
  
*****  
Another H2  
*****  
  
Sample H3  
=====
```

Sample H4

Sample H5
^^^^^^^^

Sample H6
""""""

And some text.

If you need more than heading level 4 (i.e. H5 or H6), then you should consider creating a new document. There should be only one H1 in a document.

Note: See also [Sphinx's documentation about sections](#) ⁶.

1.1.6 Code blocks

Use the `code-block` directive **and** specify the programming language. As an example:

```
.. code-block:: python  
  
import this
```

1.1.7 Links and references

Use links and references footnotes with the `target-notes` directive. As an example:

```
#####  
Some document  
#####  
  
Some text which includes links to `Example website`_ and many other links.  
  
`Example website`_ can be referenced multiple times.  
  
(... document content...)  
  
And at the end of the document...
```

⁶ <http://sphinx.pocoo.org/rest.html#sections>

```
*****
References
*****

.. target-notes::

.. _`Example website` : http://www.example.com/
```

1.1.8 References

1.2 Contributing

This document provides guidelines for people who want to contribute to the `documentation-style-guide-sphinx` project.

1.2.1 Create tickets

Please use the [bugtracker](#)¹ **before** starting some work:

- check if the bug or feature request has already been filed. It may have been answered too!
- else create a new ticket.
- if you plan to contribute, tell us, but don't wait for us! So that we are given an opportunity to discuss, join forces or give feedback as soon as possible.

1.2.2 Fork and branch

- Work in forks and branches.
- Prefix your branch with the ticket ID corresponding to the issue. As an example, if you are working on ticket #23 which is about headings convention, name your branch like `23-headings`.

1.2.3 Download and install

System requirements:

- [Python](#)² version 2.6 or 2.7.

Note: The provided Makefile uses `python` command. So you may use [Virtualenv](#)³ to make sure the active `python` is the adequate one.

Execute:

```
git clone git@github.com:benoitbryon/documentation-style-guide-sphinx.git
cd documentation-style-guide-sphinx/
make install
```

If you cannot execute the Makefile, read it and adapt the few commands it contains in the `install` section to your needs.

¹ <https://github.com/benoitbryon/documentation-style-guide-sphinx/issues>

² <http://python.org>

³ <http://virtualenv.org>

1.2.4 Edit documents

They said “Eat your own dog food”, so follow [style guide for Sphinx-based documentations](#) ⁴.

In your commit messages, reference the ticket with some `refs #TICKET-ID` syntax.

1.2.5 Test and build

Tests and builds will automatically be triggered before commit:

- tests include the build of documentation and README as HTML.
- a [Git pre-commit hook](#) ⁵ is installed during *Download and install*.

If you want to run them manually, use the provided Makefile:

- run tests with `make tests`
- build documentation with `make documentation-build`
- build README with `make README-build`

1.2.6 Share

- Push your code
- Submit a pull request

1.2.7 References

1.3 About

This section contains information about the `documentation-project-style-guide-sphinx` project itself.

1.3.1 Vision

About motivations and goals of the `documentation-style-guide-sphinx` project.

Context

In software development, documentation matters.

[Sphinx](#) ¹ is a great documentation builder. It is community standard for [Python](#) ² projects, but is not tied to Python and can be used to document almost everything. One of his strength is to use the [reStructuredText](#) ³ syntax, which focuses on simplicity and readability for humans.

As of may 2012, [Sphinx's reStructuredText primer](#) ⁴ and [docutils' reStructuredText documentation](#) ³ focus on syntax. They give examples, but they are quite permissive.

⁴ <http://documentation-style-guide-sphinx.readthedocs.org/>

⁵ <http://git-scm.com/book/en/Customizing-Git-Git-Hooks>

¹ <http://sphinx.pocoo.org>

² <http://python.org>

³ <http://docutils.sourceforge.net/rst.html>

⁴ <http://sphinx.pocoo.org/rest.html>

When developers contribute to projects, they read and write documentation. There are conventions about writing code, but there is no about writing documentation. So documentations, despite they are based on a common toolkit, are really heterogeneous:

- the more developers, the more heterogeneous.
- every project may use its own conventions.

Goal

```
Feature: public sphinx style guide
```

```
In order to improve readability of documentation and make it consistent  
across the wide spectrum of Sphinx-based documentations  
As member of a development team  
I want to follow style conventions.
```

```
Scenario: Adopt documentation-related conventions for a project
```

```
  Given a project
```

```
  And a team
```

```
  When the team chose Sphinx as documentation builder for project
```

```
  And the team needs conventions about the documentation
```

```
  Then team members follow the rules provided at
```

```
  http://documentation-style-guide-sphinx.readthedocs.org/
```

```
  And reference it in their project's documentation.
```

Priorities

As explained in [Status](#), priorities of this release are:

- write, try and maintain a public draft,
- communicate about this project.

Related work

Style guide is mostly about syntax. It is not enough.

- Because writing documentation is not so easy, we need guidelines about documentation content, i.e. best-practices. See [documentation-best-practices](#)⁵ project.
- Because we often adapt patterns while documenting, we need helpers to generate documentation content, i.e. templates.
- We also need tools to automate some tasks, generate more content from code, to run tests...

References

1.3.2 Status

Notes about the project status, until next release.

⁵ <http://documentation-best-practices.readthedocs.org>

This is a proposal

Although applied on private or small projects, this style guide should be considered as a draft. It is not industry standard and have not been supported by some “big” projects.

However, you can already use it or provide feedback!

Status would change if, as examples:

- additional users give feedback, i.e. if the project is not used by only its author ;)
- additional contributors join the project, which should be equivalent to the previous point.
- another project replaces this one.

Planning

- Use the style guide in some projects managed by authors, contributors and early adopters.
- Communicate and get feedback. Propose it to some visible projects such as:
 - Read the docs ¹
 - Sphinx ²
 - Python ³
 - Lettuce ⁴
- Depending on the feedback and contributions, reconsider this project’s relevance, adapt goals and planning.

References

1.3.3 Authors and contributors

This project is released under the [fair license](#). Here are project’s authors and contributors:

- Benoît Bryon <benoit@marmelune.net>

1.3.4 License

Copyright [documentation-style-guide-sphinx authors and contributors](#).

Usage of the works is permitted provided that this instrument is retained with the works, so that any entity that uses the works is notified of this instrument.

DISCLAIMER: THE WORKS ARE WITHOUT WARRANTY.

¹ <http://readthedocs.org>

² <http://sphinx.pocoo.org>

³ <http://python.org>

⁴ <http://lettuce.it>

Indices and tables

- `genindex`
- `modindex`
- `search`

References

P

Python Enhancement Proposals

PEP 8, 1