sphinx-revealjs Documentation

Release 1.0.1

Kazuya Takei

Jan 30, 2021

CONTENTS

`sphinx-revealjs` is Sphinx extension to generate Reveal.js presentation documents from **standard** reStructuredText.

CHAPTER

ONE

BASIC FEATURES

- Nested sections
- Speaker note
- Syntax highlight for Reveal.js (not used pygments)
- Customize slides and sections by conf.py or source reST

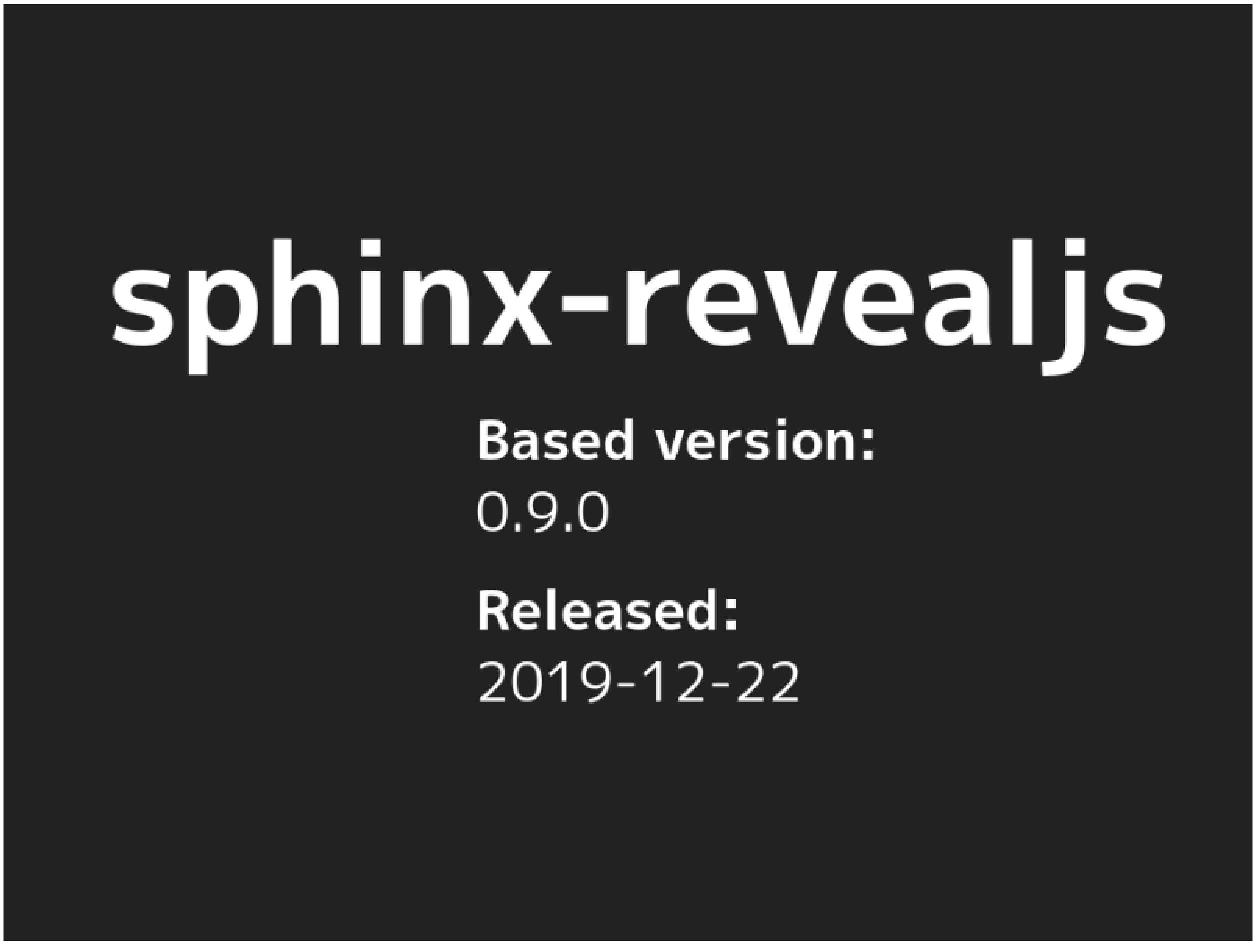
Note: In update from 0.x to 1.x, it has breaking changes. See *Migration guide*.

CHAPTER TWO

DEMO

- Source: Sphinx document
- Created: Reveal.js presentation

```
===============
sphinx-revealjs
===============

.. revealjs_slide::

    {
        controls: true,
        progress: true,
        history: true,
        center: true,
        transition: "slide",
        dependencies: [
            { src: "{{ pathto('_static/revealjs/plugin/notes/notes.js', 1) }}", async: true },
            { src: "{{ pathto('_static/revealjs/plugin/highlight/highlight.js', 1) }}", async: true, callback: function()
        ]
    }

:Based version: 0.9.0
:Released: 2019-12-22

Overview
========

What is this?
-------------

Sphinx extension to build Revealjs presentation

Features
--------

.. This is reST comment. Render into speaker note section

* Convert sections from reStructuredText directly
* Select theme from default themes


Convert sections from reStructuredText directly
===============================================

Adjust section structure
```



sphinx-revealjs

Based version:
0.9.0

Released:
2019-12-22

CHAPTER

THREE

CONTENTS

3.1 Setup

3.1.1 Requirements

`sphinx-revealjs` requires Python 3.6+ and Sphinx.

Current development environment

- Python: `3.9`
- Sphinx: `2.4.4`

3.1.2 Installation

You can install `sphinx-revealjs` from PyPI.

```
$ pip install sphinx-revealjs
```

`sphinx-revealjs` specify `Sphinx` and `docutils` expressly as dependencies. You get `Sphinx` by this command only.

3.1.3 Configuration

`sphinx-revealjs` does not provide `revealjs` builder instead of `html` builder. To use builder, edit your `conf.py`.

```
extensions = [
    "sphinx_revealjs",
]
```

if you want to configure more, edit conf.py with seeing *Configurations*.

3.1.4 Build

Run `make` command to build presentations. Files are generated to **revealjs** folder.

```
$ make revealjs
```

3.2 Migration guide

3.2.1 To 0.x -> 1.x

From version 1.x, this bundle Reveal.js 4.x, and implement for it. Due it, documentations for old version does not work to build correctly.

You have to lock version, or migrate source for next version.

revealjs_script_plugins

Reveal.js 4.x has big changes for usage of plugins from 3.x.

`sphinx-revealjs` is also adjust for this changes, and need update `revealjs_script_plugins`.

```
revealjs_script_plugins = [
    {
+        "name": "RevealNotes",
+        "src": "revealjs4/plugin/notes/notes.js",
-        "src": "revealjs/plugin/notes/notes.js",
    },
    {
+        "name": "RevealHighlight",
+        "src": "revealjs4/plugin/highlight/highlight.js",
-        "src": "revealjs/plugin/highlight/highlight.js",
-        "options": """
-            {async: true, callback: function() { hljs.initHighlightingOnLoad(); } }
-        """
    },
]
```

- Changed structure from src and options to src and name.
  - For 4.x, to use plugin for core,
    add class name of it not source path,
    and need to preload source by `script` tag.
  - Class name is defined in plugin source.
    You need find from source or ref documents (official plugin only)
  - In adding, does not accept options for plugins.

MORE: See Using Plugins from Reveal.js document

revealjs_css_files

If you use highlight plugin and specify bundled stylesheet file, change path of stylesheet. Style files is migrated to highlight plugin folder.

- Before: `revealjs/lib/css/zenburn.css`
- After: `revealjs4/plugin/highlight`

3.3 Configurations

`sphinx-revealjs` can build multiple presentations. You can configure in `conf.py` for all presentations.

3.3.1 Style Configurations

revealjs_static_path

Type `list`

Optional

Default `[]` (empty)

Example `["_static"]`

List of static files directory (same as `html_static_path`)

revealjs_css_files

Type `list`

Optional

Default `[]` (empty)

Example `["custom.css"]`

List of using custom css (same of `html_css_files`).

If you want to customize presentation by CSS, write external css and use it.

revealjs_style_theme

Type `str`

Optional

Default `black`

Example `moon`, `custom.css`

Theme name of stylesheet for Reveal.js.

- If value does not have suffix `.css`,
  use bundled Reveal.js theme(included `revealjs/css/theme`).

revealjs_google_fonts

Type dict

Optional

Default []

Example []

List of using fonts from Google Fonts. If this value is set, render link and style tags into html.

revealjs_generic_font

Type str

Optional

Default sans-serif

Example serif, monospace

If you use revealjs_google_fonts, set last of font-family style.

3.3.2 Presentation Configurations

revealjs_script_files

Type List[str]

Optional

Default []

Example ["presentation.js"]

List of sources that render as script tags.

There is bundled Reveal.js script at revealjs/js/reveal.js.

Example:

```
<div>
  <!-- Presentation body -->
</div>
<!-- here!! -->
<script src="_static/revealjs/js/revealjs.js"></script>
<script src="_static/presentation.js"></script>
```

revealjs_script_conf

Type str

Optional

Default None

Raw JavaScript code for configuration of Reveal.js.

If this value is set, render `script` tag after source script tags.

Example:

```
revealjs_script_conf = """
{
    controls: false,
    transition: 'zoom',
}
"""
```

```
<div>
  <!-- Presentation body -->
</div>
<script src="_static/revealjs/js/revealjs.js"></script>
<!-- here!! -->
<script>
  let revealjsConfig = {};
  revealjsConfig = Object.assign(revealjsConfig, {
    controls: false,
    transition: 'zoom',
  });
  revealjs.initialize(revealjsConfig);
</script>
```

revealjs_script_plugins

Type `List[Dict]`

Optional

Default `[]`

List of plugin configurations. If this value is set, render `script` tag after source script tags.

There are bundled Reveal.js plugins at `revealjs/plugin`.

Example:

```
revealjs_script_plugins = [
    "src": "revealjs/plugin/highlight/highlight.js",
    "name": "RevealHighlight",
    "options: """
      {async: true, callback: function() { hljs.initHighlightingOnLoad(); } }
    """,
]
```

```
<!-- For revealjs 3.x -->
<div>
  <!-- Presentation body -->
</div>
<script src="_static/revealjs/js/revealjs.js"></script>
<!-- here!! -->
<script>
  let revealjsConfig = {};
  plugin_0 = {async: true, callback: function() { hljs.
↪initHighlightingOnLoad(); } };
```

(continues on next page)

(continued from previous page)

```
  plugin_0.src = "_static/revealjs/plugin/highlight/highlight.js"
  revealjsConfig.dependencies.push(plugin_0);
  revealjs.initialize(revealjsConfig);
</script>
```

```
<!-- For revealjs 4.x -->
<div>
  <!-- Presentation body -->
</div>
<script src="_static/revealjs/js/revealjs.js"></script>
<script src="_static/revealjs/plugin/highlight/highlight.js"></script>
<!-- here!! -->
<script>
  let revealjsConfig = {};
  revealjsConfig.plugins = [RevealHighlight,];
  revealjs.initialize(revealjsConfig);
</script>
```

3.4 Customize slide from document

Sphinx can manage multiple documents, so that `sphinx-revealjs` can build multiple presentation slides.

If you want to configure one presentation from some, write `revealjs_slide` directive into reST document.

3.4.1 Directive usage

Write `revealjs_slide` directive on directly below of title header.

```
Presentation title
==================

.. revealjs_slide::
    :theme: moon

Section
-------

Content
```

3.4.2 Directive attributes

Note: Directive based customize has options less than conf based because implementation restrict.

theme

Override `revealjs_style_theme`.

google_font

Override `revealjs_google_fonts`, but it can specify only one.

conf

Override `revealjs_script_conf`, but single line only.

3.5 Customize sections

To change behavior of sections, `sphinx-revealjs` provide some directives.

3.5.1 revealjs_section

To change behavior per section, write directive per section.

Usage

Write `revealjs_slide` directive on directly below of section title header.

```
Title
=====

Section
-------

.. revealjs_section::
    :data-background-color: #009900
```

Attributes

This directive can accept attribute as same as Reveal.js `section` tags.

3.5.2 revealjs_break

If you want to transition section with keeping title, `revealjs_break` can use.

Usage

Write `revealjs_break` to point of want to split section.

```
Title
=====

Section
-------

Content 1

.. revealjs_break::

Content 2(next slide)
```

Attributes

It accepts attributes as same as `revealjs_section`.

And it accepts `notitle` as unique feature.

notitle If it is set in directive, next section page does not display title.

3.6 Content directives

`sphinx-revealjs` provides features for contents of section.

3.6.1 revealjs_fragments

Note: There are cases not working regular.

Inject `fragment` attribute into objects.

Usage

Write block as directive that you want to present as fragments.

```
.. revealjs_fragments::

    * First
    * Second
    * Third
```

See demo

Reference

- https://github.com/hakimel/reveal.js/#fragments

CHAPTER

FOUR

CONCEPT AND MOTIVATION

Goal of this library is to provide presentation platform for self-branding of engineer using Sphinx. Using static site hosting service, you can show own presentations to anyone.

Core motivation is that I want to play presentation by this library.

CHAPTER

FIVE

LICENSES

This library is licensed Apache License version 2.0.

About license of directly dependencies, please see each software projects or documentations.

- docutils:
  - https://docutils.sourceforge.io/
- Sphinx:
  - https://www.sphinx-doc.org/
  - https://github.com/sphinx-doc/sphinx
- Reveal.js:
  - https://revealjs.com/#/
  - https://github.com/hakimel/reveal.js

CHAPTER

SIX

INDICES AND TABLES

- genindex
- modindex
- search