Allabaster

Release

July 01, 2016

Contents

Warning: This page is not reworked yet, and copied from Alabaster.

Allabaster is embeddable Alabaster

CHAPTER 1

Changes from Alabaster

- All these fancy sidebars and footers were stripped down, as well, as html head and body tags. Now theme generates not complete html pages, but html fragments. It's intended usage – to insert these fragments into other html pages.
- CSS code was modified to decrease chance of interference with css rules of the main site.
- Breadcrumbs were added to simplify navigation.

Alabaster is a visually (c)lean, responsive, configurable theme for the Sphinx documentation system. It is Python 2+3 compatible.

It began as a third-party theme, and is still maintained separately, but as of Sphinx 1.3, Alabaster is an install-time dependency of Sphinx and is selected as the default theme.

Live examples of this theme can be seen on this project's own website, paramiko.org, fabfile.org and pyinvoke.org.

For more documentation, please see http://alabaster.readthedocs.io.

Note: You can install the development version via `pip install alabaster==dev`.

1.1 Features

- Easy ability to install/use as a Python package (tip o' the hat to Dave & Eric's sphinx_rtd_theme for showing the way);
- Style tweaks compared to the source themes, such as better code-block alignment, Github button placement, page source link moved to footer, improved (optional) related-items sidebar item, and many more;
- Many customization hooks, including toggle of various sidebar & footer components; header/link/etc color control; etc;
- Improved documentation for all customizations (pre-existing & new).

1.2 Project background

Alabaster is a modified (with permission) version of Kenneth Reitz's "krTheme" Sphinx theme (it's the one used in his Requests project). Kenneth's theme was itself originally based on Armin Ronacher's Flask theme. Many thanks to both for their hard work.

1.3 Implementation notes

- Fabric #419 contains a lot of general exposition & thoughts as I developed Alabaster, specifically with a mind towards using it on two nearly identical 'sister' sites (single-version www 'info' site & versioned API docs site).
- Alabaster includes/requires a tiny Sphinx extension on top of the theme itself; this is just so we can inject dynamic metadata (like Alabaster's own version number) into template contexts. It doesn't add any additional directives or the like, at least not yet.

1.3.1 Changelog

- Fork from Alabaster.

1.3.2 Customization

Warning: This page is not reworked yet, and copied from Alabaster.

Alabaster's behavior & style can be customized in multiple ways:

- Various template-level or nontrivial-style settings can be configured via your `conf.py` in `html_theme_options`; see *Theme options*.
- As of Alabaster 0.7.8, you can provide your own CSS stylesheet overrides via a *custom stylesheet*. This is suitable for changes that only need minor CSS modifications.

  Note: Some theme options implemented prior to 0.7.8 would have been more suitable as local custom stylesheet overrides. Therefore:

  - We no longer accept feature requests which are more appropriately solved by using this functionality instead.
  - In future backwards-incompatible versions we may deprecate some of those options; as such we highly recommend leveraging the custom stylesheet whenever possible, even if an option is present below.
    - When in doubt, simply check the built-in stylesheet's template to see whether the option you're looking at is a basic variable insertion or something more complicated.)

Custom stylesheet

If you need to modify Alabaster's default CSS styles in a way not covered by the theme options from the next section, you may provide a custom CSS stylesheet as follows:

- Create a file named `custom.css` anywhere you prefer (typically `_static/`, but this is solely convention) containing your desired overrides to the CSS found in Alabaster's `static/alabaster_css_t`.
- Set the core Sphinx option html_static_path to either that file's path, or the directory it lives within.

Theme options

Alabaster's primary configuration route is the `html_theme_options` variable, set in `conf.py` alongside the rest, e.g.:

```
html_theme_options = {
    'logo': 'logo.png',
    'github_user': 'bitprophet',
    'github_repo': 'alabaster',
}
```

The following subsections detail available such options, including notes about behavior & default values.

Variables and feature toggles

- logo: Relative path (from $PROJECT/_static/) to a logo image, which will appear in the upper left corner above the name of the project.
  - If logo is not set, your project name setting (from the top level Sphinx config) will be used in a text header instead. This preserves a link back to your homepage from inner doc pages.
- logo_name: Set to true to insert your site's project name under the logo image as text. Useful if your logo doesn't include the project name itself. Defaults to false.
- logo_text_align: Which CSS text-align value to use for logo text (if there is any.)
- body_text_align: Which CSS text-align value to use for body text (if there is any.)
- description: Text blurb about your project, to appear under the logo.
- description_font_style: Which CSS font-style to use for description text. Defaults to normal.
- github_user, github_repo: Used by github_button and github_banner (see below); does nothing if both of those are set to false.
- github_button: true or false (default: true) - whether to link to your Github.
  - If true, requires that you set github_user and github_repo.
  - See also these other related options, which behave as described in Github Buttons' documentation:
    * github_type: Defaults to watch.
    * github_count: Defaults to true.
- github_banner: true or false (default: false) - whether to apply a 'Fork me on Github' banner in the top right corner of the page.
  - If true, requires that you set github_user and github_repo.
  - May also submit a string file path (as with logo, relative to $PROJECT/_static/) to be used as the banner image instead of the default.
- travis_button: true, false or a Github-style "account/repo" string - used to display a Travis-CI build status button in the sidebar. If true, uses your github_(user|repo) settings; defaults to false.
- codecov_button: true, false or a Github-style "account/repo" string - used to display a Codecov build status button in the sidebar. If true, uses your github_(user|repo) settings; defaults to false.
- gratipay_user: Set to your Gratipay username if you want a Gratipay 'Donate' section in your sidebar.
  - This used to be gittip_user before that service changed its name to Gratipay; we've left the old setting in place as an alias for backwards compatibility reasons. It may be removed in the future.
  - If both options are given, gratipay_user wins.
- analytics_id: Set to your Google Analytics ID (e.g. UA-#######-##) to enable tracking.

- touch_icon: Path to an image (as with logo, relative to $PROJECT/_static/) to be used for an iOS application icon, for when pages are saved to an iOS device's home screen via Safari.
- canonical_url: If set, is used as the base URL (set before the relative path/pagename) for a <link rel="canonical"> canonical URL header tag.

  Note: This value must end with a trailing slash.

- extra_nav_links: Dictionary mapping link names to link targets; these will be added in a UL below the main sidebar navigation (provided you've enabled navigation.html.) Useful for static links outside your Sphinx doctree.
- sidebar_includehidden: Boolean determining whether the TOC sidebar should include hidden Sphinx toctree elements. Defaults to true so you can use :hidden: in your index page's root toctree & avoid having 2x copies of your navigation on your landing page.
- **sidebar_collapse: Boolean determining whether all TOC entries that** are not ancestors of the current page are collapsed. You can read more about this in the Sphinx toctree docs.
- show_powered_by: Boolean controlling display of the Powered by Sphinx N.N.N. & Alabaster M.M.M section of the footer. When true, is displayed next to the copyright information; when false, is hidden.
- show_related: Boolean controlling whether the 'next/previous/related' secondary navigation elements are hidden or displayed. Defaults to false since on many sites these elements are superfluous.
- page_width: CSS width specifier controlling default content/page width. Defaults to 940px.
- sidebar_width: CSS width specifier controlling default sidebar width. Defaults to 220px.
- fixed_sidebar: Makes the sidebar 'fixed' or pinned in place, so that the main body of the page scrolls but the sidebar remains visible. (Applies only to desktop window sizes; the mobile view is unaffected.) Defaults to false.

Style colors

These should be fully qualified CSS color specifiers such as #004B6B or #444. The first few items in the list are "global" colors used as defaults for many of the others; update these to make sweeping changes to the colorscheme. The more granular settings can be used to override as needed.

- gray_1: Dark gray.
- gray_2: Light gray.
- gray_3: Medium gray.
- pink_1: Light pink.
- pink_2: Medium pink.
- body_text: Main content text.
- footer_text: Footer text (includes links.)
- link: Non-hovered body links.
- link_hover: Body links, hovered.
- sidebar_header: Sidebar headers. Defaults to gray_1.
- sidebar_text: Sidebar paragraph text.

- `sidebar_link`: Sidebar links (there is no hover variant.) Applies to both header & text links. Defaults to `gray_1`.
- `sidebar_link_underscore`: Sidebar links' underline (technically a bottom-border).
- `sidebar_search_button`: Background color of the search field's 'Go' button.
- `sidebar_list`: Foreground color of sidebar list bullets & unlinked text.
- `sidebar_hr`: Color of sidebar horizontal rule dividers. Defaults to `gray_3`.
- `anchor`: Foreground color of section anchor links (the 'paragraph' symbol that shows up when you mouseover page section headers.)
- `anchor_hover_fg`: Foreground color of section anchor links (as above) when moused over. Defaults to `gray_1`.
- `anchor_hover_bg`: Background color of above.
- `note_bg`: Background of `.. note::` blocks. Defaults to `gray_2`.
- `note_border`: Border of same.
- `seealso_bg`: Background of `.. seealso::` blocks. Defaults to `gray_2`.
- `seealso_border`: Border of same.
- `warn_bg`: Background of `.. warn::` blocks. Defaults to `pink_1`.
- `warn_border`: Border of same. Defaults to `pink_2`.
- `footnote_bg`: Background of footnote blocks.
- `footnote_border`: Border of same. Defaults to `gray_2`.
- `pre_bg`: Background of preformatted text blocks (including code snippets.) Defaults to `gray_2`.
- `narrow_sidebar_bg`: Background of 'sidebar' when narrow window forces it to the bottom of the page.
- `narrow_sidebar_fg`: Text color of same.
- `narrow_sidebar_link`: Link color of same. Defaults to `gray_3`.
- `code_highlight`: Color of highlight when using `:emphasize-lines:` in a code block.

Fonts

- `font_family`: Font family of body text. Defaults to `'goudy old style', 'minion pro', 'bell mt', Georgia, 'Hiragino Mincho Pro', serif`.
- `font_size`: Font size of body text. Defaults to `17px` (`1.0625em`).
- `head_font_family`: Font family of headings. Defaults to `'Garamond', 'Georgia', serif`.
- `code_font_size`: Font size of code block text. Defaults to `0.9em`.
- `code_font_family`: Font family of code block text. Defaults to `'Consolas', 'Menlo', 'Deja Vu Sans Mono', 'Bitstream Vera Sans Mono', monospace`.
- `caption_font_size`: Font size of caption block text. Defaults to `font-size`.
- `caption_font_family`: Font family of caption block text. Defaults to `font-family`.

1.3.3 Installation

Warning: This page is not reworked yet, and copied from Alabaster.

The bare minimum required to install Alabaster is as follows:

- If you're on **Sphinx 1.2 or older**:
  - `pip install alabaster` or equivalent.
  - Add the following to your `conf.py` so Alabaster's theme location & mini-extension are located & loaded:

```
import alabaster

html_theme_path = [alabaster.get_path()]
extensions = ['alabaster']
html_theme = 'alabaster'
```

  - If you've installed Alabaster by hand (without using `pip`) and/or are doing funky things to your PYTHONPATH, you may need to replace the `alabaster.get_path()` call with your own explicit string, as per the Sphinx config docs.
- If you have **Sphinx 1.3 or above**:
  - You already have Alabaster installed as a dependency! No need to manually install it or explicitly load it.

Note: If you distribute your documentation via the excellent Read the Docs, you may need to explicitly enable Alabaster (as RTD defaults to using its own theme) by adding this line to your `conf.py`:

```
html_theme = ['alabaster']
```

- **Either way**, add an explicit `html_sidebars` setting so Alabaster's customized sidebar templates are loaded:

```
html_sidebars = {
    '**': [
        'about.html',
        'navigation.html',
        'relations.html',
        'searchbox.html',
        'donate.html',
    ]
}
```

That's it! You now have the standard Alabaster theme set up. Read on for more core configuration concerns, or see Customization for feature/style options.

Sidebars

Feel free to adjust `html_sidebars` as desired - the theme is designed assuming you'll always have `about.html` activated, but otherwise it doesn't care much.

- See the Sphinx docs for details on how this setting behaves.
- Alabaster provides `about.html` (logo, github button + blurb), `donate.html` (Gratipay blurb/button) and `navigation.html` (a more flexible version of the builtin `localtoc`/`globaltoc` templates). `searchbox.html` comes with Sphinx itself.

Static path for images and/or custom stylesheet

If you’re using any of the image-related options listed on Customization (`logo` or `touch-icon`) or a *custom stylesheet*, you’ll also want to tell Sphinx where to get these files from. If so, add a line like this (changing the path if necessary; see the Sphinx docs for ‘html_static_path’) to your `conf.py`:

```
html_static_path = ['_static']
```