
Socrates Documentation

Release 0.6.0

Honza Pokorny

July 23, 2011

CONTENTS

Socrates is a simple static site generator. It's geared towards blogs. You write your posts in your favorite plain text to HTML language (e.g. Markdown, textile) and save them as text files on your harddrive. Socrates then takes them, and creates a full HTML site for you. For free, you will get a home page which lists latest posts, single post pages, category pages, archive pages, an about page and an atom feed.

FEATURES

- Familiar Django and Jinja2 templates
- Simple install via pip
- Markdown, reStructuredText, Textile support
- YAML configuration
- Atom feed
- Github pages compatible

Contents:

1.1 Installation

First, set up your virtual environment and pip install Socrates.

```
virtualenv env --no-site-packages
source env/bin/activate
pip install socrates
```

Or, get the latest code from Github:

```
pip install -e git://github.com/honza/socrates.git#egg=socrates
```

The next step is to install your templates. At the moment, you can choose between [Django templates](#) and [Jinja2 templates](#). The default theme uses Django.

```
pip install django
pip install jinja2
```

If you're going to use a processor other than Markdown, you have to install that, too.

textile

```
pip install textile
```

reStructuredText

```
pip install docutils pygments
```

1.2 Usage

First, you need to create a new blog:

```
$ socrates -i blog
```

This will create a `blog` directory with a simple blog structure:

```
blog
  posts
    2010-your-post.md
  layout
    index.html
    single.html
    category.html
    ...
  media
    style.css
  config.yaml
  about.md
```

The `posts` directory is where you will place your posts files. Anything prefixed with `_` or `.` will be ignored. `layout` is your basic theme or a template. `config.yaml` is a site-wide configuration file. Don't forget to update the `about` file with relevant information.

You can also create a new blog in the current working directory:

```
$ socrates -i
```

When you are ready to generate your site, you run:

```
$ socrates -g blog
```

Or,

```
$ socrates -g
```

for current directory.

This will place all the generated files in `blog/deploy`. You can then take that directory and upload it to your server.

1.3 Configuration

This is a reference for your blog's main configuration file. Any values that you add in there will be made available in your templates' context.

author

Blog author. Defaults to *author*.

site_name

Site name. Defaults to *Socrates site*.

posts_per_page

Number of posts displayed per page. Used for pagination. Defaults to *10*. Setting this to *0* will display all posts on one page.

url

Your site's URL. Defaults to *http://example.com*.

date_format

Python strftime formatted date format. Defaults to *%B %d, %Y*.

text_processor

Which X to html processor to use; *markdown, textile, rst, html, extension*. Defaults to *markdown*. The 'extension' setting will decide on the processor to be used based on the post's file extension:

•Markdown

- .md
- .markdown
- .mkdn

•reStructuredText

- .rst

•HTML

- .html
- .htm
- .txt

•Textile

- .textile

templates

'django' or 'jinja2'. Defaults to *django*.

append_slash

Whether a slash should be appended to post urls. Defaults to *true*.

url_include_day

Whether to include the day with the month and year in the generated directories and urls. Defaults to *false*.

initial_header_level

By default, the first heading in your document will be `<h2>`. Only available for reStructuredText posts. Defaults to *2*.

skip_archives

If set to *true*, it won't bother generating archives. Defaults to *false*.

skip_categories

If set to *true*, it won't bother generating categories. Defaults to *false*.

pygments

Additional settings for the Pygments HTML Parser. It passes the arguments directly to the `HtmlFormatter` class when it's instantiated, so these settings include all of the available settings for `HtmlFormatter`

sample:

```
pygments:
  linenos: true
  noclasses: false
  style: 'pastie'
```

style

The style option has many default built in styles for your code blocks. The ones that ship with Pygments are: *monokai, manni, perldoc, borland, colorful, default, murphy, vs, trac, tango, fruity, autumn, bw, emacs, vim, pastie, friendly, native*

inline_css

Whether or not you want pygments to output a `pygments.css` file in your build directory for css. If set to `false` it will output the file.

1.4 Writing posts

All of your posts will typically be contained in the `posts` directory in your main blog directory. Your post file can be called anything you want, and as long as you're not mixing and matching different text processors, the file extension can be anything, too. I use the following naming convention:

```
2011-07-29-name-of-post.rst
```

This way my posts are automatically ordered by publish date when I run `ls` in the `posts` directory.

1.4.1 Markdown, textile and HTML

When you're writing your posts in Markdown (or textile or HTML), you need to add a bit of text to the top of your file to provide Socrates with some metadata about your post.

```
-----  
title: Title of your post  
date: 2011-07-29 13:00:00  
categories:  
  - Photos  
  - Vacation  
-----
```

The text between the two horizontal line is written in [YAML](#) syntax. Note that the horizontal line should have at least 79 characters.

1.4.2 reStructuredText

If you want to write your post in reStructuredText, you should use the `rst` native way to specify document metadata. Include this at the top of your post file:

```
:title: Title of your post  
:date: 2011-07-29 13:00:00  
:categories: Photos, Vacation
```

This way, your posts can be processed by the native [Docutils](#) utility functions such as `rst2html.py` or `rst2latex.py`.

1.4.3 Options

title

The title of the post

date

Publish date; YYYY-MM-DD HH:MM

categories

A list of categories

template

You can override the default template that the post is going to be rendered with.

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*