Sphinx Github Webpage Tutorials

Release 1.00

Wenqiang Feng

February 16, 2019

CONTENTS

Welcome to my Sphinx gihub webpage tutorials! In those tutorials, you will learn how to use `Sphinx` to create `.html` and `.pdf` and how to hookup your `Sphinx` webpage to github. The PDF version can be downloaded from HERE.

CHAPTER

ONE

PREFACE

1.1 About this tutorial

This document is a summary of my valueable experiences in using Python decumentation `Sphinx` with `Github` webpage. **You may download and distribute it. Please be aware, however, that the note contains typos as well as inaccurate or incorrect description.**

In this repository, I try to use the detailed demo code and examples to show how to use `Sphinx` to generate the `.html` and `.pdf` documents and how to hookup them automatically on `Github`. If you find your work wasn't cited in this note, please feel free to let me know.

Although I am by no means a python programming and Sphinx expert, I decided that it would be useful for me to share what I learned about Sphinx in the form of easy tutorials with detailed example. I hope those tutorials will be a valuable tool for your studies.

The tutorials assume that the reader has a preliminary knowledge of `python` programing, `LaTex` and `Linux`. And this document is generated automatically by using sphinx.

1.1.1 About the authors

- **Wenqiang Feng**
  - Data Scientist and PhD in Mathematics
  - University of Tennessee at Knoxville
  - Email: von198@gmail.com

- **Biography**

  Wenqiang Feng is Data Scientist within DST's Applied Analytics Group. Dr. Feng's responsibilities include providing DST clients with access to cutting-edge skills and technologies, including Big Data analytic solutions, advanced analytic and data enhancement techniques and modeling.

Dr. Feng has deep analytic expertise in data mining, analytic systems, machine learning algorithms, business intelligence, and applying Big Data tools to strategically solve industry problems in a cross-functional business. Before joining DST, Dr. Feng was an IMA Data Science Fellow at The Institute for Mathematics and its Applications (IMA) at the University of Minnesota. While there, he helped startup companies make marketing decisions based on deep predictive analytics.

Dr. Feng graduated from University of Tennessee, Knoxville, with Ph.D. in Computational Mathematics and Master's degree in Statistics. He also holds Master's degree in Computational Mathematics from Missouri University of Science and Technology (MST) and Master's degree in Applied Mathematics from the University of Science and Technology of China (USTC).

- **Declaration**

  The work of Wenqiang Feng was supported by the IMA, while working at IMA. However, any opinion, finding, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the IMA, UTK and DST.

1.2 Motivation for this tutorial

`Sphinx` is an awesome Python documentation package, and it has excellent facilities for the documentation of software projects in a range of languages. I was impressed and attracted by Sphinx in the first using. And I foud that:

1. It supports **several popular output formats**: `HTML` (including Windows HTML Help), `LaTeX` (for printable PDF versions), ePub, Texinfo, manual pages, plain text.
2. It has **easy publishing routes**: Github.
3. Is has **extensive cross-references**: semantic markup and automatic links for functions, classes, citations, glossary terms and similar pieces of information
4. It has **hierarchical structure**: easy definition of a document tree, with automatic links to siblings, parents and children.
5. It has **automatic indices**: general index as well as a language-specific module indices
6. It has awesome **code handling**: automatic highlighting using the Pygments highlighter
7. Is has abundant **extensions**: automatic testing of code snippets, inclusion of docstrings from Python modules (API docs), and more
8. It has abundant **contributed extensions**: more than 50 extensions contributed by users in a second repository; most of them installable from PyPI

1.3 Feedback and suggestions

Your comments and suggestions are highly appreciated. I am more than happy to receive corrections, suggestions or feedbacks through email (Wenqiang Feng: von198@gmail.com) for improvements.

CHAPTER

TWO

INTRODUCTION

2.1 Sphnix: Python Documentation Generator

The following descriptions are from Sphinx:

Sphinx is a tool that makes it easy to create intelligent and beautiful documentation, written by Georg Brandl and licensed under the BSD license.

It was originally created for the Python documentation, and it has excellent facilities for the documentation of software projects in a range of languages. Of course, this site is also created from reStructuredText sources using Sphinx! The following features should be highlighted:

1. **Output formats:** `HTML` (including Windows HTML Help), `LaTeX` (for printable PDF versions), ePub, Texinfo, manual pages, plain text
2. **OExtensive cross-references:** semantic markup and automatic links for functions, classes, citations, glossary terms and similar pieces of information
3. **OHierarchical structure:** easy definition of a document tree, with automatic links to siblings, parents and children
4. **OAutomatic indices:** general index as well as a language-specific module indices
5. **OCode handling:** automatic highlighting using the Pygments highlighter
6. **OExtensions:** automatic testing of code snippets, inclusion of docstrings from Python modules (API docs), and more
7. **OContributed extensions:** more than 50 extensions contributed by users in a second repository; most of them installable from PyPI

Sphinx uses reStructuredText as its markup language, and many of its strengths come from the power and straightforwardness of reStructuredText and its parsing and translating suite, the Docutils.

2.2 reStructured Text

The following descriptions are from reStructuredText:

reStructuredText (RST, ReST, or reST) is a file format for textual data used primarily in the Python programming language community for technical documentation.

2.3 Latex Document Preparation System

The following descriptions are from LaTex:

LaTeX (a shortening of Lamport TeX) is a document preparation system. When writing, the writer uses plain text as opposed to the formatted text found in WYSIWYG ("what you see is what you get") word processors like Microsoft Word, LibreOffice Writer and Apple Pages.

LaTeX is widely used in academia for the communication and publication of scientific documents in many fields, including mathematics, statistics, computer science, engineering, chemistry, physics, economics, linguistics, quantitative psychology, philosophy, and political science.

More information can be get from LaTeX .

CHAPTER

THREE

PACKAGES INSTALLATION

Warning: It's been 10 years since I abandoned `Windows` operating systems. So I am a noob for `Windows` operating systems and I really do not know how to install `some packages` on `Windows` operating systems.

3.1 Python Installation

1. Install `pip`:

```
sudo easy_install pip
```

2. In stall `python`:

```
pip install python
```

3.2 Sphinx Installation

```
pip install -U Sphinx
```

3.3 Latex Installation

You can download the `MacTex` from: https://www.tug.org/mactex/ and install it for Mac system. Or you can use the following command to install `TexLive` on Linux system:

```
sudo apt update && sudo apt install texlive-full
```

CHAPTER

FOUR

SPHINX CONFIGURATION

Warning: It's been 10 years since I abandoned `Windows` operating systems. So I am a noob for `Windows` operating systems and I really do not know how to install `some packages` on `Windows` operating systems.

4.1 Python Installation

1. Install `pip`:

```
sudo easy_install pip
```

2. In stall `python`:

```
pip install python
```

4.2 Sphinx Installation

```
pip install -U Sphinx
```

4.3 Latex Installation

You can download the `MacTex` from: https://www.tug.org/mactex/ and install it for Mac system. Or you can use the following command to install `TexLive` on Linux system:

```
sudo apt update && sudo apt install texlive-full
```