

CSE1019: Fundamentals of Open Source Software

Lab 4 : Documentation

Software Engineering Lab

Except where otherwise noted, the contents of this document are Copyright 2016 Gwanggyu Choi, Junggyu Shin. All rights reserved. Any redistribution, reproduction, transmission, or storage of part or all of the contents in any form is prohibited without the author's expressed written permission.

Learn by run: 문서화

- reST로 소스코드 문서화 하기
- 파이썬 코드 작성
 - 가위바위보를 하는 프로그램 작성하기
 - 10게임을 진행하여 누가 승리했는지 알려주는 프로그램
- 파이썬 소스코드 문서화
 - 자신이 작성한 소스코드의 함수의 기능 설명
 - 입력 / 출력
 - 코드 라이선스
- Sphinx로 소스코드로부터 문서 생성하기
 - Sphinx 설치
 - Sphinx를 이용한 HTML 문서 생성
- Markdown으로 프로그램 소개 작성
- 최종 결과물
 - 간단한 파이썬 프로그램+주석
 - Sphinx로 생성한 html 문서
 - Markdown (.md 확장자) 으로 작성한 설명

문서화

- 오픈소스에서는 다수의 사람들이 프로젝트에 참여한다. 참여하는 인원에게 모두를 말이나 1:1 설명으로 이해 시키려면 많은 자원이 투자되기 때문에 커뮤니케이션 도구로 문서를 사용한다.
- 개발한 코드 혹은 개발 중인 코드에 대한 설명을 문서로 만드는 것을 문서화라고 한다.
- 최근엔 time - to -market의 속도가 빨라져 문서화조차 최소화 하는 Agile 개발 방법론을 사용한다.
- 따라서 문서화 작업을 최소화 시키기 위해 코드를 개발함과 동시에 같은 파일내에 문서화를 위한 문법을 입력하고 프레임워크를 자동으로 문서화 시킨다.

reST

reStructuredText

Markup Syntax and Parser Component of Docutils

- 일반 텍스트 문서 양식을 편집하는데 사용하는 markup 언어의 일종
- 문법 그 자체로도 내용파악과 사용이 쉽다
- 확장자는 .rst

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

lab(se);

reST 기본문법

구조

```
# raw-form
이것은 문단 입니다. 매우 짧은
첫번째 문단 입니다.

    이것은 두번째 문단입니다. 한단계 들여쓰기 되어 있습니다.
    문단은 빈 줄로 구분됩니다.

이것은 마지막 문단입니다.

# results in
이것은 문단 입니다. 매우 짧은 첫번째 문단 입니다.

    이것은 두번째 문단입니다. 한단계 들여쓰기 되어 있습니다. 문단은 빈 줄로 구분됩니다.
이것은 마지막 문단입니다.
```

문단구분은 아래와 같이 빈줄로 구분한다. 들여쓰기를 하면 들여쓰기된 단락이 생성된다.

reST 기본문법

문자 스타일

```
# Italics
*이탈릭체로 나옵니다.*

# Bold
**굵은 글씨로 나옵니다.**
```

문자의 스타일을 지정할 수 있다.

http://www.incodom.kr/reStructuredText#h_c107cbc2ba8f522b7d84a866c349f854

reST 기본문법

목록생성

```
1. 목록입니다.  
2. 목록입니다.  
#. 목록입니다.
```

일반 적인 목록

```
항목1  
    항목1의 정의 입니다.  
    항목1의 정의  
  
항목2  
    항목2의 정의 입니다.  
    항목2의 정의
```

사전형 목록

```
-a          옵션 a 설명  
-b file     옵션 b 설명  
--opt_a     옵션 opt_a 설명
```

옵션 목록

http://www.incodom.kr/reStructuredText#h_c107cbc2ba8f522b7d84a866c349f854

reST 기본문법

표 생성하기

```
+-----+-----+-----+
| Header 1 | Header 2 | Header 3 |
+-----+-----+-----+
| body row 1 | column 2 | column 3 |
+-----+-----+-----+
| body row 2 | Cells may span columns. |
+-----+-----+-----+
| body row 3 | Cells may | - Cells |
+-----+-----+-----+
| body row 4 | span rows. | - contain |
+-----+-----+-----+
| body row 4 | | - blocks. |
+-----+-----+-----+
```

표를 만들 수 있다.

http://www.incodom.kr/reStructuredText#h_c107cbc2ba8f522b7d84a866c349f854

reST 기본문법

섹션 나누기

```
=====
Document Title
=====

-----
Subtitle
-----

Chapter 1 Title
-----

Section 1.1 Title
-----

Subsection 1.1.1 Title
~~~~~

Section 1.2 Title
-----

Chapter 2 Title
=====
```

http://www.incodom.kr/reStructuredText#h_c107cbc2ba8f522b7d84a866c349f854

reST 기본문법

이미지 넣기

```
.. image:: images.png
   :height: 100
   :width: 200
   :scale: 50
   :alt: 이미지 설명
```

문서에 이미지를 넣는다

http://www.incodom.kr/reStructuredText#h_c107cbc2ba8f522b7d84a866c349f854

reST 기본문법

- reST에 대한 자세한 문법은
 - <http://www.sphinx-doc.org/en/stable/rest.html>
- reST문법에 대한 데모는 아래 주소에서 실행가능하다
 - <http://rst.ninjs.org/>

Sphinx



Sphinx는 파이썬 프로젝트를 위한 문서작성 도구로 만들어 졌고, 현재 C/C++도 지원하고 있다. 앞으로 다른 언어도 지원할 예정이라고 한다.

<http://www.sphinx-doc.org/en/stable/index.html>

Sphinx 설치 (Window)

1. **pip** 설치 과정이 조금 복잡하므로 아래 블로그를 참조하시면 됩니다

- <http://be4rpoooh02.tistory.com/8>

2. **pip**로 **Sphinx** 설치하기

- `sudo pip install sphinx`

Sphinx 설치 (Mac OS)

1.easy_install 설치 및 easy_install로 pip 설치

- `curl -O http://python-distribute.org/distribute_setup.py`
- `sudo python distribute_setup.py`
- `sudo easy_install pip`

2.pip로 Sphinx 설치하기

- `sudo pip install sphinx`

Sphinx 설치 (linux/ubuntu)

1. apt-get 으로 pip 설치

- `sudo apt-get update`
- `sudo apt-get upgrade`
- `sudo apt-get install python-pip`

2.pip로 Sphinx 설치하기

- `sudo pip install sphinx`

Sphinx 실행해보기

```
[choigwanggyuui-iMac:oss choigwanggyu$ mkdir sphinx_test  
[choigwanggyuui-iMac:oss choigwanggyu$ cd sphinx_test/  
[choigwanggyuui-iMac:sphinx_test choigwanggyu$ sphinx-quickstart
```

1. Sphinx를 적용시킬 폴더 생성

- mkdir sphinx_test
- cd sphinx_test

2. Sphinx-quickstart 명령어를 통해 html 파일을 생성

- sphinx-quickstart

<http://alifejeon.blogspot.kr/2016/02/sphinx-python-api.html>

Sphinx 실행해보기

```
Enter the root path for documentation.
```

```
[> Root path for the documentation [.]:
```

```
You have two options for placing the build directory for Sphinx output.
```

```
Either, you use a directory "_build" within the root path, or you separate  
"source" and "build" directories within the root path.
```

```
[> Separate source and build directories (y/n) [n]: y
```

```
Inside the root directory, two more directories will be created; "_templates"  
for custom HTML templates and "_static" for custom stylesheets and other static  
files. You can enter another prefix (such as ".") to replace the underscore.
```

```
[> Name prefix for templates and static dir [_]:
```

1. 문서화할 폴더가 현재 위치에 있는가? . 은 현재위치
2. 설정파일들과 결과물의 위치를 나눌것인가
3. 폴더는 앞에 _를 붙여 줄것인가

Sphinx 실행해보기

Sphinx has the notion of a "version" and a "release" for the software. Each version can have multiple releases. For example, for Python the version is something like 2.5 or 3.0, while the release is something like 2.5.1 or 3.0a1. If you don't need this dual structure, just set both to the same value.

```
[> Project version []: 1.0  
[> Project release [1.0]: 1.0
```

If the documents are to be written in a language other than English, you can select a language here by its language code. Sphinx will then translate text that it generates into that language.

For a list of supported codes, see
<http://sphinx-doc.org/config.html#confval-language>.

```
[> Project language [en]: en
```

The file name suffix for source files. Commonly, this is either ".txt" or ".rst". Only files with this suffix are considered documents.

```
[> Source file suffix [.rst]:
```

1. 프로젝트 버전에 몇 인가?
2. 문서화 하는 언어는 무엇인가? en은 영어
3. 문서화의 결과물은 어떤 확장자로 출력 할 것인가?

Sphinx 실행해보기

```
Sphinx can also add configuration for epub output:
```

```
[> Do you want to use the epub builder (y/n) [n]:
```

```
Please indicate if you want to use one of the following Sphinx extensions:
```

```
[> autodoc: automatically insert docstrings from modules (y/n) [n]: y
```

```
[> doctest: automatically test code snippets in doctest blocks (y/n) [n]: y
```

```
[> intersphinx: link between Sphinx documentation of different projects (y/n) [n]: y
```

```
[> todo: write "todo" entries that can be shown or hidden on build (y/n) [n]: y
```

```
[> coverage: checks for documentation coverage (y/n) [n]: y
```

```
[> imgmath: include math, rendered as PNG or SVG images (y/n) [n]:
```

```
[> mathjax: include math, rendered in the browser by MathJax (y/n) [n]:
```

```
[> ifconfig: conditional inclusion of content based on config values (y/n) [n]:
```

```
[> viewcode: include links to the source code of documented Python objects (y/n) [n]: y
```

```
[> githubpages: create .nojekyll file to publish the document on GitHub pages (y/n) [n]:
```

1. 기타등등 자신의 프로젝트에 맞게 선택하여 준다

Sphinx 실행해보기

A Makefile and a Windows command file can be generated for you so that you only have to run e.g. 'make html' instead of invoking sphinx-build directly.

```
[> Create Makefile? (y/n) [y]: y
```

```
[> Create Windows command file? (y/n) [y]: n
```

```
Creating file ./source/conf.py.
```

```
Creating file ./source/index.rst.
```

```
Creating file ./Makefile.
```

Finished: An initial directory structure has been created.

1. Mac , Linux는 위에만 y
2. Windows 는 아래에만 y

Sphinx 실행해보기

```
[choigwanggyuui-iMac:sphinx_test choigwanggyu$ ls
Makefile      build      source
[choigwanggyuui-iMac:sphinx_test choigwanggyu$ make html
Running Sphinx v1.5a1
making output directory...
loading pickled environment... not yet created
loading intersphinx inventory from https://docs.python.org/objects.inv...
intersphinx inventory has moved: https://docs.python.org/objects.inv -> https://docs.python.org/2/objects.inv
building [mo]: targets for 0 po files that are out of date
building [html]: targets for 1 source files that are out of date
updating environment: 1 added, 0 changed, 0 removed
reading sources... [100%] index
looking for now-outdated files... none found
pickling environment... done
checking consistency... done
preparing documents... done
writing output... [100%] index
generating indices... genindex
writing additional pages... search
copying static files... done
copying extra files... done
dumping search index in English (code: en) ... done
dumping object inventory... done
build succeeded.

Build finished. The HTML pages are in build/html.
```

1. make html을 입력
2. sphinx의 기본 html 생성완료

Sphinx 실행해보기

결과물

Table Of Contents

[Welcome to oss lab4's
documentation!](#)

[Indices and tables](#)

This Page

[Show Source](#)

Quick search

Welcome to oss lab4's documentation!

Indices and tables

- [Index](#)
- [Module Index](#)
- [Search Page](#)

©2016, zecag189. | Powered by [Sphinx 1.5a1](#) & [Alabaster 0.7.9](#) | [Page source](#)