
LLNL RtD Template

LLNL Open Source

Aug 20, 2020

CONTENTS:

- 1 Steps for Setting up Docs** **3**
- 1.1 A. Create Repo/Directory 3
- 1.2 B. Install Sphinx 3
- 1.3 C. Build and Configure Doc Files 3
- 1.4 D. Establish Webhook Integration 4
- 1.5 E. Create and Populate .rst Files 4

- 2 Resources** **7**

This documentation site and the files associated with the [.github repo](#) are designed to be used as a template for developers who are new to Read the Docs (RtD) documentation.

STEPS FOR SETTING UP DOCS

The process of creating a Read the Docs (RtD) site is widely documented (see also [Resources](#)), though your local environment will introduce variables. The examples that follow use `pipenv` as one variation.

The basic steps are:

- A. Create repo on GitHub or create local directory to existing repo
- B. Install Sphinx
- C. Build and configure doc files
- D. Set up GitHub and RtD webhook integration
- E. Create `.rst` files

1.1 A. Create Repo/Directory

The instructions below assume that developers understand step A.

1.2 B. Install Sphinx

1. Create a `/docs` folder in your project. Then `cd docs` to complete most of the remaining steps (exceptions are noted below).
2. Install RtD template from GitHub using Sphinx (`pipenv install sphinx-rtd-theme`).
3. Initiate Sphinx quickstart (`pipenv run sphinx-quickstart`). Answer the questions that follow installation, such as selecting No for setting up build and source separately.
4. Move the makefile into the `/docs` folder if it's not there already.
5. Confirm the Sphinx version (`pipenv run sphinx-build --version`).
6. Confirm that docs can build locally (`pipenv run make html`). You must be in the `/docs` directory for this to work.

1.3 C. Build and Configure Doc Files

1. Update the `conf.py` file. For instance, you can change the RtD theme (`html_theme`). In the General Configuration section, add the name of master `.rst` file:

```
# Tells Sphinx the name of the master .rst file.
master_doc = 'index'
```

2. Create the **.gitignore** file (touch `.gitignore`) so your commits will include only source files, not the HTML-rendered files (instructions).
3. Create the **requirements.txt** file to specify dependency versions (instructions). This file usually contains information like this:

```
# These dependencies should be installed using pip in order
# to build the documentation.

sphinx==2.0.1
sphinxcontrib-programoutput==0.14
sphinx-rtd-theme==0.4.3
python-levenshtein
```

4. Create the **.readthedocs.yml** file in the *top directory* of the repo. Paste in the **V2** content.
5. Commit configuration changes, then open and merge PR before moving on.

1.4 D. Establish Webhook Integration

This process generates a **readthedocs.io** URL for your project, which you can then link to in your repo's README file.

1. Log into the [RtD website](#) via your GitHub account. You will have to go through an account verification process the first time.
2. My Projects > Import a Project > select your repo. You may need to refresh the list or filter out an organization.
3. Enter repo name and GitHub URL if not pre-populated. Confirm **.git** is appended to the end of the URL.
4. Follow [webhook creation instructions](#). When you try to build the docs for the first time, you may get an error: *fatal: could not read Username for 'https://github.com': terminal prompts disabled*. So you must configure the webhook to recognize GitHub's 2FA via `git config --global --add url."git@github.com:".insteadOf "https://github.com/"`.
5. Add doc build status to your repo's README file. Open the file and paste in the badge syntax (where **name** is your repo):

```
[[Documentation Status] (https://readthedocs.org/projects/name/badge/?
↪version=latest)] (https://name.readthedocs.io/en/latest/?badge=latest)
```

6. Confirm that docs still build locally, then open and merge PR.
7. On your RtD dashboard, select the project and initiate a build to test: Projects > your repo > Builds > Build Version.
8. You may want to change some of RtD's advanced settings, such as specifying which version is considered "latest" or which branch is the default. Projects > your repo > Admin > Advanced Settings.

1.5 E. Create and Populate .rst Files

All documentation files should contain your repo's copyright date(s) and license information at the top, set off by two periods (.).

1. Create the **index.rst** file (or whatever file name you specified as the master file in step *CI* above).
2. Set up table of contents (TOC), specifying how many levels deep you want to expose. For example, this site's TOC looks like this:

```
.. toctree::  
    :maxdepth: 2  
    :caption: Contents:  
  
    setup_steps  
    resources
```

3. Create subordinate .rst files with reference to the link you established in the TOC. The page doesn't have to have the same title as the file name. For example, this page is titled *Steps for Setting up Docs* while the TOC linkage is based on the file name of **setup_steps.rst**:

```
.. _setup_steps:
```

4. Now you can begin a cycle of adding/editing files, building locally, and pushing to GitHub. The configuration settings above should trigger automatic RtD builds with every commit or PR, but you can always manually build the docs site from your RtD dashboard.

RESOURCES

These sites are good places to learn about Read the Docs (RtD), Sphinx, and all the things you can do in .rst files:

- [RtD dashboard](#) (you must be logged in)
- [RtD documentation](#) (in other words, RtD's RtD site)
- [Sphinx](#) (Python Document Generator)
- [Sphinx Getting Started](#)
- [Sphinx tutorial](#)
- [reStructuredText Primer](#)
- [Graphviz syntax](#)