

Documenting Software

Basics, Sphinx, and Jupyter Book

Jason DeBacker

January 17, 2025

Roadmap

Intro

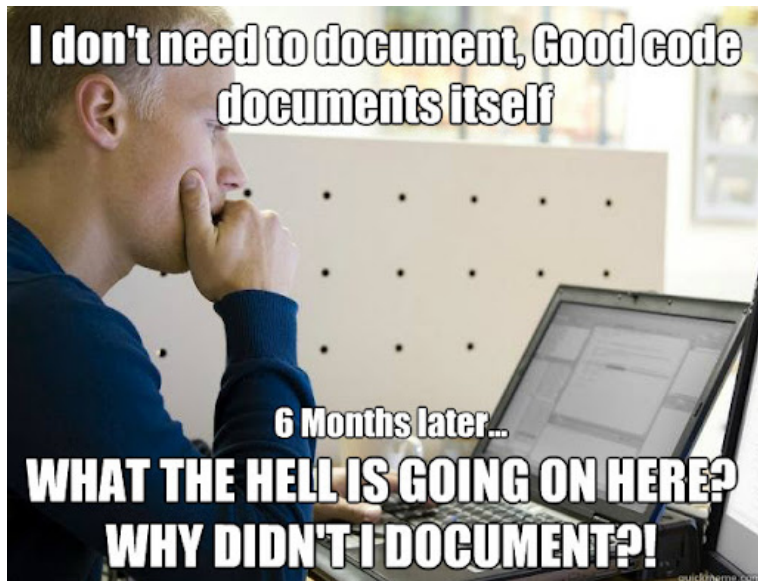
In-line comments

Docstrings

Sphinx

Jupyter Book

Documenting Code



Documenting Code

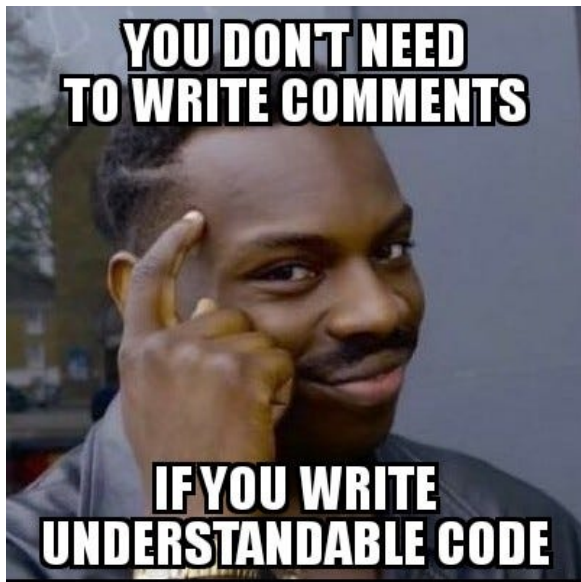
Junior: Where is documentation?

Team Lead:

I AM DOCUMENTATION

@ithumor

Documenting Code



Why document code?

- Selfish:
 - Helps you understand your code – be considerate to your future self
 - Reduces errors
 - Aids coding copilot
- Selfless:
 - Helps others understand your code
 - Encourages collaboration
 - Helps with reproducibility

Levels of documentation for software

1. In-line comments
2. Docstrings
3. Sphinx documentation for Python code
4. Jupyter Book documentation for a software/research project

Roadmap

Intro

In-line comments

Docstrings

Sphinx

Jupyter Book

The most basic form of documentation: in-line comments

- These are comments that are written directly in the code
- They are useful for explaining what a block of code does
- Typically very short
- In Python, they are denoted by a `#` symbol

In-line comment example

```
1 # define a function that adds two numbers
2 def add_numbers(a, b):
3     # make computation
4     result = a + b
5     return result # return the result
```

Roadmap

Intro

In-line comments

Docstrings

Sphinx

Jupyter Book

One step up from in-line comments: docstrings

- Docstrings are multi-line strings that are written at the beginning of a function or class
- They are enclosed in triple quotes (either single or double)
- They are used to describe what the function or class does
- They are useful for providing information about the inputs and outputs of a function
- There are some agreed-upon conventions for writing docstrings in Python
 - [Numpydoc](#)
 - [Google](#)

Docstring example (Google style)

```
1 def add_numbers(a, b):
2     """
3     A function to add two numbers
4
5     Args:
6         a (float): first number
7         b (float): second number
8
9     Returns:
10        result (float): sum of a and b
11    """
12    result = a + b
13    return result
```

Roadmap

Intro

In-line comments

Docstrings

Sphinx

Jupyter Book

Sphinx

- Sphinx is a tool that makes it easy to create nicely formatted documentation for code
- It was originally created for the Python documentation
- It can be used to document software written in any language
- It can be used to create documentation in many formats, including HTML, PDF, and ePub

Sphinx syntax

- Sphinx uses reStructuredText (reST) as its markup language
- reST is a lightweight markup language
- It is similar to Markdown, but more flexible

Sphinx docs

- Sphinx can be used on it's own to create documentation from docstrings and other documentation files in a project
- This is one reason it's helpful to write docstrings and do so in a conventional way (e.g., Google or Numpydoc)
- Sphinx can also be used to create documentation for a project that is not written in Python
- But Sphinx is also integrated with Jupyter Book

Roadmap

Intro

In-line comments

Docstrings

Sphinx

Jupyter Book

Jupyter Book

- **Jupyter Book** is a tool for creating publication-quality books and documents from computational material
- It is built on top of Sphinx, allowing you to use **MyST**, Markdown, and reStructuredText to write documentation
- It can be used to create documentation for a software project, a research project, or a course
- It can be used to create documentation from Jupyter notebooks

Jupyter Book examples

- Jupyter Book documentation
- QuantEcon
- Numpy Tutorials
- And, this course!

Building a Jupyter Book

- Necessary files:
 - `_config.yml` : configures book (title, logo, how handles Jupyter Notebooks, extensions)
 - `_toc.yml` : sets the table of contents, or structure of the book
 - Some content: (`.md`, `.rst`, `.myst`, `.ipynb`) files referenced in `_toc.yml`
 - Note: content must be in directories at or below the level of the `_config.yml` file (or you need to have a symlink to files in other directories)
- Install the Jupyter Book package: e.g.,
`pip install jupyter-book`
- Compile: `jb build path_w_config` (replace last piece with path to the directory with config file)