ultrad.com.br

Let there be documentation!

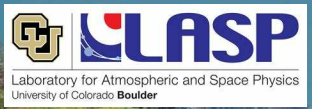
[python] [git] [readthedocs]

Illa R. Losada

LASP and CU Boulder

PCM

19 May 2021



What?

What?

Definition of documentation

(Dictionary definition)

1. The act or an instance of the supplying of documents or supporting references or records.
2. The documents or references so supplied.

What?

Let's discuss this:

Good documentation
(From O'Reilly)

What?

Let's discuss this:

Good documentation
(From O'Reilly)

What?

Let's discuss this:

Good documentation
(From O'Reilly)

- Write documentation that is inviting and clear

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project
- Write documentation that is skimmable

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project
- Write documentation that is skimmable
- Write documentation that offers examples of how to use the software

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project
- Write documentation that is skimmable
- Write documentation that offers examples of how to use the software
- Write documentation that has repetition, when useful

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project
- Write documentation that is skimmable
- Write documentation that offers examples of how to use the software
- Write documentation that has repetition, when useful
- Write documentation that is up-to-date

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project
- Write documentation that is skimmable
- Write documentation that offers examples of how to use the software
- Write documentation that has repetition, when useful
- Write documentation that is up-to-date
- Write documentation that is easy to contribute to

What?

Let's discuss this:

Good documentation

(From O'Reilly)

- Write documentation that is inviting and clear
- Write documentation that is comprehensive, detailing all aspects of the project
- Write documentation that is skimmable
- Write documentation that offers examples of how to use the software
- Write documentation that has repetition, when useful
- Write documentation that is up-to-date
- Write documentation that is easy to contribute to
- Write documentation that is easy to find

Why?

Why?

My reasons:

Why?

My reasons:

- For myself

Why?

My reasons:

- For myself
- For collaboration

Why?

My reasons:

- For myself
- For collaboration
- For new users

Why?

My reasons:

- For myself
- For collaboration
- For new users

What the internet says:

Why?

My reasons:

- For myself
- For collaboration
- For new users

What the internet says:

- Building code's value

Why?

My reasons:

- For myself
- For collaboration
- For new users

What the internet says:

- Building code's value
- Collaboration

Why?

My reasons:

- For myself
- For collaboration
- For new users

What the internet says:

- Building code's value
- Collaboration
- Reduced training time

Why?

My reasons:

- For myself
- For collaboration
- For new users

What the internet says:

- Building code's value
- Collaboration
- Reduced training time
- Robustness of code

Why?

My reasons:

- For myself
- For collaboration
- For new users

What the internet says:

- Building code's value
- Collaboration
- Reduced training time
- Robustness of code

What are your reasons?

How?

How? Web

Pencil-code webpage: `http://pencil-code.nordita.org/doc.php`

How? Web

Pencil-code webpage: `http://pencil-code.nordita.org/doc.php`

Pdf documentation

- Full manual (313 pages)
- Quick start guide (7 pages)

How? Web

Pencil-code webpage: `http://pencil-code.nordita.org/doc.php`

Pdf documentation

- Full manual (313 pages)
- Quick start guide (7 pages)

Problems:

- Difficult to maintain up-to-date
- Not comprehensive

How? Web

Pencil-code webpage: `http://pencil-code.nordita.org/doc.php`

Pdf documentation

- Full manual (313 pages)
- Quick start guide (7 pages)

Problems:

- Difficult to maintain up-to-date
- Not comprehensive

Tutorials:

- Great to learn about specific problems
- Hands-on approach

How? Web

Pencil-code webpage: `http://pencil-code.nordita.org/doc.php`

Pdf documentation

- Full manual (313 pages)
- Quick start guide (7 pages)

Problems:

- Difficult to maintain up-to-date
- Not comprehensive

Tutorials:

- Great to learn about specific problems
- Hands-on approach

Problems:

- Only some available

How? Git wiki

`https://github.com/pencil-code/pencil-code/wiki`

How? Git wiki

`https://github.com/pencil-code/pencil-code/wiki`

Content:

- Mostly about coding
- Python section with some examples
- IDL - Python guide

How? Git wiki

`https://github.com/pencil-code/pencil-code/wiki`

Content:

- Mostly about coding
- Python section with some examples
- IDL - Python guide

Problems:

- Keeping it up-to-date
- Writing extra documentation

How? Read the docs!

How? Read the docs!

Why?

How? Read the docs!

Why?

- Automatic integration of git and documentation

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code
- Automatic update

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code
- Automatic update
- Most standard tool nowadays

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code
- Automatic update
- Most standard tool nowadays

How?

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code
- Automatic update
- Most standard tool nowadays

How?

- Online configuration: basic

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code
- Automatic update
- Most standard tool nowadays

How?

- Online configuration: basic
- All documents are read from git, even the configuration

How? Read the docs!

Why?

- Automatic integration of git and documentation
- Automatic parsing of the code
- Automatic update
- Most standard tool nowadays

How?

- Online configuration: basic
- All documents are read from git, even the configuration
- All is setup within git and automatically available

Example case: Python for Pencil

https://pencil-python.readthedocs.io

Configuration files:

- `pencil-code/.readthedocs.yaml`
- Also needs a python setup: `pencil-code/python/pencil/setup.py`

Doc directory:

- `pencil-code/python/docs`

Required files inside docs:

- `conf.py`
- `index.rst`
- `requirements.txt`

Moving forward

Moving forward: proposal

All documentation in read the docs

Moving forward: proposal

All documentation in read the docs

- Python

Moving forward: proposal

All documentation in read the docs

- Python
- Pencil

Moving forward: proposal

All documentation in read the docs

- Python
- Pencil
- Scripts

Moving forward: proposal

All documentation in read the docs

- Python
- Pencil
- Scripts
- IDL

Moving forward: proposal

All documentation in read the docs

- Python
- Pencil
- Scripts
- IDL
- Manuals

Moving forward: proposal

All documentation in read the docs

- Python
- Pencil
- Scripts
- IDL
- Manuals
- Tutorials

Let's discuss!