
read_the_docks_test

Release 0.0.2

Paul K.

Jan 12, 2021

CONTENTS:

1 Erstellen einer Doku für readTheDocks mit Sphinx: 3

1.1 (Links to) StructuredText: 3

1.2 Lokal: 3

1.3 Diese Seite als “raw code”: 4

2 Bilder Integrieren 5

2.1 Bilder werden so hinzugefügt: 5

3 The reStructuredText Cheat Sheet: Syntax Reminders 7

3.1 Section Structure 7

3.2 Body Elements 7

3.3 Inline Markup 10

3.4 Directive Quick Reference 10

3.5 Interpreted Text Role Quick Reference 11

4 Indices and tables 13

Bibliography 15

Link zum [Code](#) (GitLab).

ERSTELLEN EINER DOKU FÜR READTHEDOCKS MIT SPHINX:

1.1 (Links to) StructuredText:

- <https://github.com/DevDungeon/reStructuredText-Documentation-Reference>

1.2 Lokal:

1.2.1 (Python Environment)

- **im Git-Repository:**
 - install sphinx:

```
pip install sphinx
```

- “docs” erstellt automatisch einen Ordner “docs” in dem die Doku-Files und Make-Files (make.bat) liegen werden

```
sphinx-quickstart docs
```

- ==>
 - standards verwenden & Projektnamen (+ aktuelle Projektversion) eingeben (... Min. 33:27 (kurz) oder 17:00 (lang - mit erklärung) ..)
- Um aus den .txt oder .rst (restructuredtext) files die fertige Doku-Seite (i.d.R. html) zu machen:

```
cd docs  
make html
```

- make html wird auch automatisch von readthedocks ausgeführt - aber so können wir das Ergebnis lokal anschauen..
- ==>
 - das index.html file ist im /docs/build/html ordner verfügbar und kann z.B. mit Firefox geöffnet werden

1.3 Diese Seite als “raw code”:

```
=====
Erstellen einer Doku für readTheDocks mit Sphinx:
=====

(Links to) StructuredText:
=====
* https://github.com/DevDungeon/reStructuredText-Documentation-Reference

Lokal:
=====

(Python Environment)
-----

- im Git-Repository:
    * install sphinx:

.. code-block::

    pip install sphinx

* "docs" erstellt automatisch einen Ordner "docs" in dem die Doku-Files und Make-
  ↳Files (make.bat) liegen werden

.. code-block::

    sphinx-quickstart docs

* ==>
    + standards verwenden & Projektnamen (+ aktuelle Projektversion) eingeben,
  ↳ (... Min. 33:27 (kurz) oder 17:00 (lang - mit erklärung) .. )

* Um aus den .txt oder .rst (restructuredtext) files die fertige Doku-Seite (i.d.R.
  ↳html) zu machen:

.. code-block::

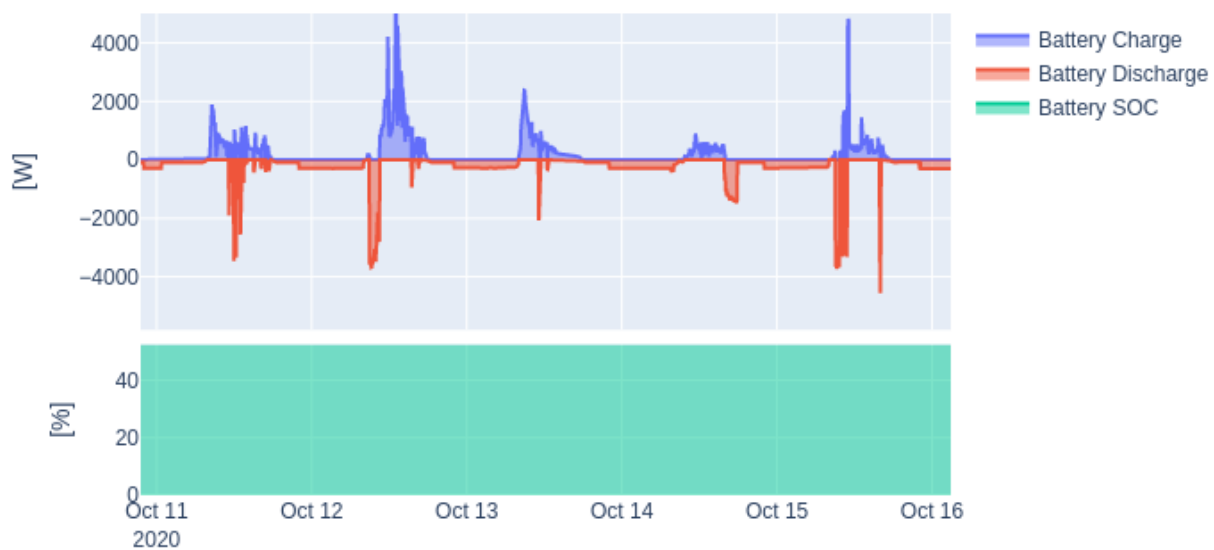
    cd docs
    make html

* make html wird auch automatisch von readthedocks ausgeführt - aber so können wir
  ↳das Ergebnis lokal anschauen..

* ==>
    + das index.html file ist im /docs/build/html    ordner verfügbar und
  ↳kann z.B. mit Firefox geöffnet werden
```


BILDER INTEGRIEREN

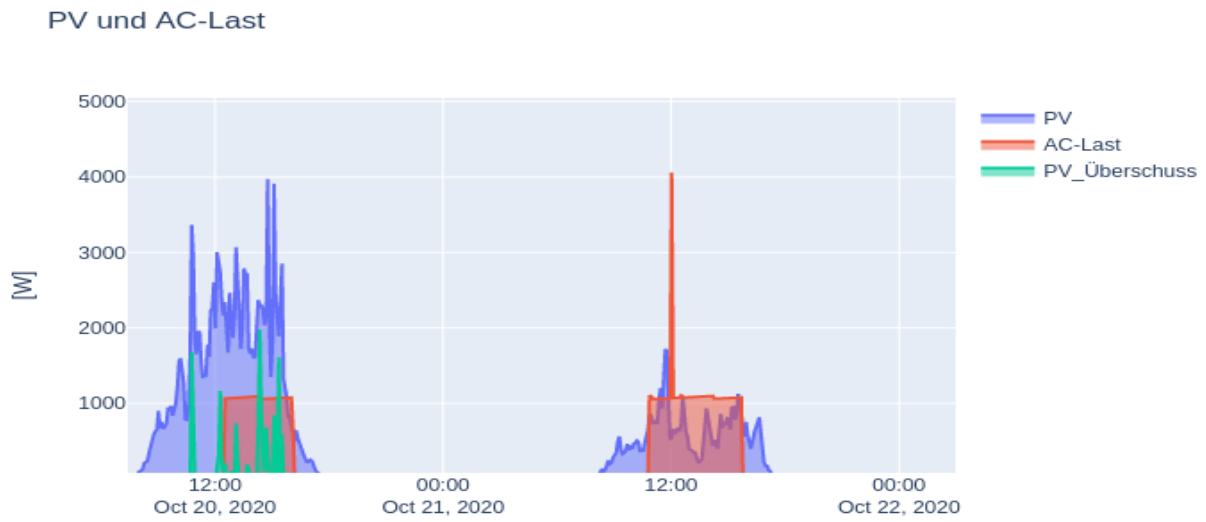
Leistung und State of Charge



2.1 Bilder werden so hinzugefügt:

```
.. image:: images/example2.png

.. oder komplizierter:
.. image:: images/example2.png
   :height: 100
   :width: 200
   :alt: alternate text
```



THE RESTRUCTUREDTEXT CHEAT SHEET: SYNTAX REMINDERS

Info See <<http://docutils.sf.net/rst.html>> for introductory docs.

Author David Goodger <goodger@python.org>

Date \$Date: 2013-02-20 02:10:53 +0100 (Mi, 20. Feb 2013) \$

Revision \$Revision: 7612 \$

Description This is a “docinfo block”, or bibliographic field list

Note: If you are reading this as HTML, please read [cheatsheet.txt](#) instead to see the input syntax examples!

3.1 Section Structure

Section titles are underlined or overlined & underlined.


3.2 Body Elements

Grid table:

Paragraphs are flush-left, separated by blank lines. Block quotes are indented.	Literal block, preceded by “::”: Indented or: > Quoted
<pre>>>> print 'Doctest block' Doctest block</pre>	
Line blocks preserve line breaks & indents. [new in 0.3.6] Useful for addresses, verse, and adornment-free lists; long lines can be wrapped with continuation lines.	

Simple tables:

List Type	Examples (syntax in the text source)
Bullet list	<ul style="list-style-type: none">• items begin with “-“, “+”, or “*”
Enumerated list	<ol style="list-style-type: none">1. items use any variation of “1.”, “A)”, and “(i)”2. also auto-enumerated
Definition list	Term is flush-left [optional classifier] Definition is indented, no blank line between
Field list	field name field body
Option list	-o at least 2 spaces between option & description

Explicit Markup	Examples (visible in the text source)
Footnote	
Citation	
Hyperlink Target	
Anonymous Target	
	
Directive (“::”)	
Substitution Def	
Comment	
Empty Comment	(“..” on a line by itself, with blank lines before & after, used to separate indentation contexts)

3.3 Inline Markup

emphasis; **strong emphasis**; *interpreted text*; *interpreted text with role*; inline literal text; standalone hyperlink, <http://docutils.sourceforge.net>; named reference, `reStructuredText`; anonymous reference; footnote reference,¹; citation reference, `[CIT2002]`; like an inline directive; inline internal target.

3.4 Directive Quick Reference

See <http://docutils.sf.net/docs/ref/rst/directives.html> for full info.

Directive Name	Description (Docutils version added to, in [brackets])
attention	Specific admonition; also “caution”, “danger”, “error”, “hint”, “important”, “note”, “tip”, “warning”
admonition	Generic titled admonition: <code>.. admonition:: By The Way</code>
image	<code>.. image:: picture.png</code> ; many options possible
figure	Like “image”, but with optional caption and legend
topic	<code>.. topic:: Title</code> ; like a mini section
sidebar	<code>.. sidebar:: Title</code> ; like a mini parallel document
parsed-literal	A literal block with parsed inline markup
rubric	<code>.. rubric:: Informal Heading</code>
epigraph	Block quote with class=“epigraph”
highlights	Block quote with class=“highlights”
pull-quote	Block quote with class=“pull-quote”
compound	Compound paragraphs [0.3.6]
container	Generic block-level container element [0.3.10]
table	Create a titled table [0.3.1]
list-table	Create a table from a uniform two-level bullet list [0.3.8]
csv-table	Create a table from CSV data [0.3.4]
contents	Generate a table of contents
sectnum	Automatically number sections, subsections, etc.
header, footer	Create document decorations [0.3.8]
target-notes	Create an explicit footnote for each external target
math	Mathematical notation (input in LaTeX format)
meta	HTML-specific metadata
include	Read an external reST file as if it were inline
raw	Non-reST data passed untouched to the Writer
replace	Replacement text for substitution definitions
unicode	Unicode character code conversion for substitution defs
date	Generates today’s date; for substitution defs
class	Set a “class” attribute on the next element
role	Create a custom interpreted text role [0.3.2]
default-role	Set the default interpreted text role [0.3.10]
title	Set the metadata document title [0.3.10]

¹ Manually numbered or [#] auto-numbered (even [#labelled]) or [*] auto-symbol

3.5 Interpreted Text Role Quick Reference

See <http://docutils.sf.net/docs/ref/rst/roles.html> for full info.

Role Name	Description
emphasis	Equivalent to <i>emphasis</i>
literal	Equivalent to <code>literal</code> but processes backslash escapes
math	Mathematical notation (input in LaTeX format)
PEP	Reference to a numbered Python Enhancement Proposal
RFC	Reference to a numbered Internet Request For Comments
raw	For non-reST data; cannot be used directly (see docs) [0.3.6]
strong	Equivalent to strong
sub	Subscript
sup	Superscript
title	Title reference (book, etc.); standard default role

INDICES AND TABLES

- `genindex`
- `search`

BIBLIOGRAPHY

[CIT2002] A citation.