Sphinx-SimplePDF-DEMO

Version 1.0

DEMO PDF of Sphinx-SimplePDF

Build: 28.01.2026

Maintained by team useblocks

Table of Contents

Demo Sphinx-SimplePDF

This PDF contains examples and test-data for different documentation and layout elements.

Some of them are testing corner cases (e.g. a huge table), for which the PDF format is not the ideal one and therefore their representation is not perfect or even buggy.

1. Structural Elements

Table of Contents

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec lorem neque, interdum in ipsum nec, finibus dictum velit. Ut eu efficitur arcu, id aliquam erat. In sit amet diam gravida, imperdiet tellus eu, gravida nisl. Praesent aliquet odio eget libero elementum, quis rhoncus tellus tincidunt. Suspendisse quis volutpat ipsum. Sed lobortis scelerisque tristique. Aenean condimentum risus tellus, quis accumsan ipsum laoreet ut. Integer porttitor maximus suscipit. Mauris in posuere sapien. Aliquam accumsan feugiat ligula, nec fringilla libero commodo sed. Proin et erat pharetra.

Etiam turpis ante, luctus sed velit tristique, finibus volutpat dui. Nam sagittis vel ante nec malesuada. Praesent dignissim mi nec ornare elementum. Nunc eu augue vel sem dignissim cursus sed et nulla. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Pellentesque dictum dui sem, non placerat tortor rhoncus in. Sed placerat nulla at rhoncus iaculis.

1.1. Document Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed condimentum nulla vel neque venenatis, nec placerat lorem placerat. Cras purus eros, gravida vitae tincidunt id, vehicula nec nulla. Fusce aliquet auctor cursus. Phasellus ex neque, vestibulum non est vitae, viverra fringilla tortor. Donec vestibulum convallis justo, a faucibus lorem vulputate vel. Aliquam cursus odio eu felis sodales aliquet. Aliquam erat volutpat. Maecenas eget dictum mauris. Suspendisse arcu eros, condimentum eget risus sed, luctus efficitur arcu. Cras ut dictum mi. Nulla congue interdum lorem, semper semper enim commodo nec.

1.1.1. Document Subsection

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam efficitur in eros et blandit. Nunc maximus, nisl at auctor vestibulum, justo ex sollicitudin ligula, id faucibus urna orci tristique nisl. Duis auctor rutrum orci, in ornare lacus condimentum quis. Quisque arcu velit, facilisis quis interdum ac, hendrerit auctor mauris. Curabitur urna nibh, porttitor at ante sit amet, vestibulum interdum dolor. Duis dictum elit orci, tincidunt imperdiet sem pellentesque et. In vehicula pellentesque varius. Phasellus a turpis sollicitudin, bibendum massa et, imperdiet neque. Integer quis sapien in magna rutrum bibendum. Integer cursus ex sed magna vehicula finibus. Proin tempus orci quis dolor tempus, nec condimentum odio vestibulum. Etiam efficitur sollicitudin libero, tincidunt volutpat ligula interdum sed.

1.1.1.1. Document Subsubsection

Donec non rutrum lorem. Aenean sagittis metus at pharetra fringilla. Nunc sapien dolor, cursus sed nisi at, pretium tristique lectus. Sed pellentesque leo lectus, et convallis ipsum euismod a. Integer at leo vitae felis pretium aliquam fringilla quis odio. Sed pharetra enim accumsan feugiat pretium. Maecenas at pharetra tortor. Morbi semper eget mi vel finibus. Cras rutrum nulla eros, id feugiat arcu pellentesque ut. Sed finibus tortor ac nisi ultrices viverra. Duis feugiat malesuada sapien, at commodo ante porttitor ac. Curabitur posuere mauris mi, vel ornare orci scelerisque sit amet. Suspendisse nec fringilla dui.

1.1.1.1.1. Document Paragraph

Pellentesque nec est in odio ultrices elementum. Vestibulum et hendrerit sapien, quis vulputate turpis. Suspendisse potenti. Curabitur tristique sit amet lectus non viverra. Phasellus rutrum dapibus turpis sed imperdiet. Mauris maximus viverra ante. Donec eu egestas mauris. Morbi vulputate tincidunt euismod. Integer vel porttitor neque. Donec at lacus suscipit, lacinia lectus vel, sagittis lectus.

2. Structural Elements 2

Etiam turpis ante, luctus sed velit tristique, finibus volutpat dui. Nam sagittis vel ante nec malesuada. Praesent dignissim mi nec ornare elementum. Nunc eu augue vel sem dignissim cursus sed et nulla. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Pellentesque dictum dui sem, non placerat tortor rhoncus in. Sed placerat nulla at rhoncus iaculis.

2.1. Document Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed condimentum nulla vel neque venenatis, nec placerat lorem placerat. Cras purus eros, gravida vitae tincidunt id, vehicula nec nulla. Fusce aliquet auctor cursus. Phasellus ex neque, vestibulum non est vitae, viverra fringilla tortor. Donec vestibulum convallis justo, a faucibus lorem vulputate vel. Aliquam cursus odio eu felis sodales aliquet. Aliquam erat volutpat. Maecenas eget dictum mauris. Suspendisse arcu eros, condimentum eget risus sed, luctus efficitur arcu. Cras ut dictum mi. Nulla congue interdum lorem, semper semper enim commodo nec.

2.1.1. Document Subsection

This is a caption for a figure. Text should wrap around the caption.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam efficitur in eros et blandit. Nunc maximus, nisl at auctor vestibulum, justo ex sollicitudin ligula, id faucibus urna orci tristique nisl. Duis auctor rutrum orci, in ornare lacus condimentum quis. Quisque arcu velit, facilisis quis interdum ac, hendrerit auctor mauris. Curabitur urna nibh, porttitor at ante sit amet, vestibulum

interdum dolor. Duis dictum elit orci, tincidunt imperdiet sem pellentesque et. In vehicula pellentesque varius. Phasellus a turpis sollicitudin, bibendum massa et, imperdiet neque. Integer quis sapien in magna rutrum bibendum. Integer cursus ex sed magna vehicula finibus. Proin tempus orci quis dolor tempus, nec condimentum odio vestibulum. Etiam efficitur sollicitudin libero, tincidunt volutpat ligula interdum sed. Praesent congue sagittis nisl et suscipit. Vivamus sagittis risus et egestas commodo.Cras venenatis arcu in pharetra interdum. Donec quis metus porttitor tellus cursus lobortis. Quisque et orci magna. Fusce rhoncus mi mi, at vehicula massa rhoncus quis. Mauris augue leo, pretium eget molestie vitae, efficitur nec nulla. In hac habitasse platea dictumst. Sed sit amet imperdiet purus.

3. Paragraph Level Markup

Table of Contents

3.1. Inline Markup

Paragraphs contain text and may contain inline markup: *emphasis*, **strong emphasis**, `inline literals`, standalone hyperlinks (http://www.python.org), external hyperlinks (Python [5]), internal cross-references (example), external hyperlinks with embedded URIs (Python web site), footnote references (manually numbered [1], anonymous auto-numbered [3], labeled auto-numbered [2], or symbolic [*]), citation references ([12]), substitution references (≣), and inline hyperlink targets (see Targets below for a reference back to here). Character-level inline markup is also possible (although exceedingly ugly!) in *re* `Structured` *Text*. Problems are indicated by |problematic| text (generated by processing errors; this one is intentional).

Also with `sphinx.ext.autodoc`, which I use in the demo, I can link to `test_py_module.test.Foo`. It will link you right to my code documentation for it.

The default role for interpreted text is *Title Reference*. Here are some explicit interpreted text roles: a PEP reference (**PEP 287**); an RFC reference (**RFC 2822**); a subscript; a superscript; and explicit roles for *standard* **inline** `markup`.

GUI labels are a useful way to indicate that Some action is to be taken by the user. The GUI label should not run over `line-height` so as not to interfere with text from adjacent lines.

Key-bindings indicate that the read is to press a button on the keyboard or mouse, for example `MMB` and `Shift`-`MMB`. Another useful markup to indicate a user action is to use `menuselection` this can be used to show short and long menus in software. For example, and `menuselection` can be seen here that breaks is too long to fit on this line. My ▸ Software ▸ Some menu ▸ Some sub menu 1 ▸ sub menu 2.

Let's test wrapping and whitespace significance in inline literals: `This is an example of --inline-literal --text, --including some-- strangely--hyphenated-words. Adjust-the-width-of-your-browser-window to see how the text is wrapped. -- ---- -------- Now note the spacing between the words of this sentence (words should be grouped in pairs).`

If the `--pep-references` option was supplied, there should be a live link to PEP 258 here.

Very long URLs should be wrapped so lines do not overflow and cause horizontal scrolling: https://www.google.com/search?hl=en&q=very%20long%20url%20example%20of%20a%20url%20that%20is%20extremely%20long%20you%20pr

3.2. Math

This is a test. Here is an equation: $X_{0:5} = (X_0, X_1, X_2, X_3, X_4)$. Here is another:

$$\nabla^2 f = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial f}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial f}{\partial \theta}\left(\sin\theta\,\frac{\partial f}{\partial \theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2 f}{\partial \phi^2} \tag{1}$$

You can add a link to equations like the one above (1) by using `:eq:`.

3.3. Meta

3.4. Blocks

3.4.1. Literal Blocks

Literal blocks are indicated with a double-colon ("::") at the end of the preceding paragraph (over there `-->`). They can be indented:

```
if literal_block:
    text = 'is left as-is'
    spaces_and_linebreaks = 'are preserved'
    markup_processing = None
```

Or they can be quoted without indentation:

```
>> Great idea!
>
> Why didn't I think of that?
```

3.4.2. Line Blocks

This is a line block. It ends with a blank line.
Each new line begins with a vertical bar ("|").
Line breaks and initial indents are preserved.
Continuation lines are wrapped portions of long lines; they begin with a space in place of the vertical bar.
The left edge of a continuation line need not be aligned with the left edge of the text above it.
This is a second line block.

Blank lines are permitted internally, but they must begin with a "|".
Take it away, Eric the Orchestra Leader!

> A one, two, a one two three four
>
> Half a bee, philosophically,
> must, *ipso facto*, half not be.
> But half the bee has got to be,
> *vis a vis* its entity. D'you see?

> But can a bee be said to be
> or not to be an entire bee,
> when half the bee is not a bee,
> due to some ancient injury?
>
> Singing...

3.4.3. Block Quotes

Block quotes consist of indented body elements:

> My theory by A. Elk. Brackets Miss, brackets. This theory goes as follows and begins now. All brontosauruses are thin at one end, much much thicker in the middle and then thin again at the far end. That is my theory, it is mine, and belongs to me and I own it, and what it is too.
>
> —Anne Elk (Miss)

3.4.4. Doctest Blocks

```
>>> print 'Python-specific usage examples; begun with ">>>"'
Python-specific usage examples; begun with ">>>"
>>> print '(cut and pasted from interactive Python sessions)'
(cut and pasted from interactive Python sessions)
```

3.4.5. Code Blocks

```
# parsed-literal test
curl -O http://someurl/release-1.0.tar-gz
```

Code Blocks can have captions.

```
{
"windows": [
    {
    "panes": [
        {
        "shell_command": [
            "echo 'did you know'",
            "echo 'you can inline'"
        ]
```

```
        },
        {
        "shell_command": "echo 'single commands'"
        },
        "echo 'for panes'"
    ],
    "window_name": "long form"
    }
],
"session_name": "shorthands"
}
```

3.4.5.1. Emphasized lines with line numbers

```
1 def some_function():
2     interesting = False
3     print 'This line is highlighted.'
4     print 'This one is not...'
5     print '...but this one is.'
```

3.5. Sidebar

Ch'ien / The Creative

Above CH'IEN THE CREATIVE, HEAVEN

Below CH'IEN THE CREATIVE, HEAVEN

The first hexagram is made up of six unbroken lines. These unbroken lines stand for the primal power, which is light-giving, active, strong, and of the spirit. The hexagram is consistently strong in character, and since it is without weakness, its essence is power or energy. Its image is heaven. Its energy is represented as unrestricted by any fixed conditions in space and is therefore conceived of as motion. Time is regarded as the basis of this motion. Thus the hexagram includes also the power of time and the power of persisting in time, that is, duration.

The power represented by the hexagram is to be interpreted in a dual sense in terms of its action on the universe and of its action on the world of men. In relation to the universe, the hexagram expresses the strong, creative action of the Deity. In relation to the human world, it denotes the creative action of the holy man or sage, of the ruler or leader of men, who through his power awakens and develops their higher nature.

3.5.1. Code with Sidebar

A code example

With a sidebar on the right.

Literal includes can also have captions.

```
"""Test Module for sphinx_rtd_theme."""


class Foo:
    """Docstring for class Foo.

    This text tests for the formatting of docstrings generated from output
    ``sphinx.ext.autodoc``. Which contain reST, but sphinx nests it in the
    ``<dl>``, and ``<dt>`` tags. Also, ``<tt>`` is used for class, method names
    and etc, but those will *always* have the ``.descname`` or
    ``.descclassname`` class.

    Term
        It is also possible to include definitions inside docstrings.
        They should be styled as a normal definition list.

    :Field List:
        It is also possible to include definitions inside docstrings.
        They should be styled as a normal definition list.

    .. [1] A footnote contains body elements, consistently indented by at
       least 3 spaces.

    .. [Citation] A citation contains body elements, consistently indented by at
       least 3 spaces.

```

```
 27     Normal ``<tt>`` (like the <tt> I just wrote here) needs to be
shown with
 28     the same style as anything else with ````this type of
markup````.
 29
 30     It's common for programmers to give a code example inside of
their
 31     docstring::
 32
 33         from test_py_module import Foo
 34
 35         myclass = Foo()
 36         myclass.dothismethod('with this argument')
 37         myclass.flush()
 38
 39         print(myclass)
 40
```

3.6. References

3.6.1. Footnotes

[1] (1,2)
A footnote contains body elements, consistently indented by at least 3 spaces.

This is the footnote's second paragraph.

[2] (1,2)
Footnotes may be numbered, either manually (as in [1]) or automatically using a "#"-prefixed label. This footnote has a label so it can be referred to from multiple places, both as a footnote reference ([2]) and as a hyperlink reference (label).

[3]
This footnote is numbered automatically and anonymously using a label of "#" only.

[*]
Footnotes may also use symbols, specified with a "*" label. Here's a reference to the next footnote: [†].

[†]
This footnote shows the next symbol in the sequence.

[4]
Here's an unreferenced footnote, with a reference to a nonexistent footnote: [5]_.

3.6.2. Citations

[Citation]
This is the citation I made, let's make this extremely long so that we can tell that it doesn't follow the normal responsive table stuff.

[12]
This citation has some `code blocks` in it, maybe some **bold** and *italics* too. Heck, lets put a link to a meta citation [13] too.

[13]
This citation will have two backlinks.

Here's a reference to the above, [Citation], and a [nonexistent] citation.

3.6.3. Glossary

This is a glossary with definition terms for thing like Writing:

Documentation
: Provides users with the knowledge they need to use something.

Reading
: The process of taking information into ones mind through the use of eyes.

Writing
: The process of putting thoughts into a medium for other people to read.

3.6.4. Targets

This paragraph is pointed to by the explicit "example" target. A reference can be found under Inline Markup, above. Inline hyperlink targets are also possible.

Section headers are implicit targets, referred to by name. See Targets, which is a subsection of References.

Explicit external targets are interpolated into references such as "Python [5]".

Targets may be indirect and anonymous. Thus this phrase may also refer to the Targets section.

Here's a `hyperlink reference without a target`_, which generates an error.

3.7. Directives

3.7.1. Contents

These are just a sample of the many reStructuredText Directives. For others, please see: http://docutils.sourceforge.net/docs/ref/rst/directives.html.

3.7.2. Centered text

You can create a statement with centered text with `.. centered::`

This is centered text!

3.7.3. Images & Figures

3.7.3.1. Images

An image directive (also clickable – a hyperlink reference):

3.7.3.2. Figures

A figure is an image with a caption and/or a legend:

re	Revised, revisited, based on ‘re’ module.
Structured	Structure-enhanced text, structuredtext.
Text	Well it is, isn’t it?

This paragraph is also part of the legend.

A figure directive with center alignment

This caption should be centered.

3.7.4. Admonitions

Attention

Directives at large.

Caution

Don’t take any wooden nickels.

Danger

Mad scientist at work!

Error

Does not compute.

Hint

It’s bigger than a bread box.

Important

- Wash behind your ears.
- Clean up your room.
  - Including the closet.
  - The bathroom too.
    - Take the trash out of the bathroom.
    - Clean the sink.
- Call your mother.

- Back up your data.

Note

This is a note. Equations within a note: $G_{\mu\nu} = 8\pi G(T_{\mu\nu} + \rho_\Lambda g_{\mu\nu})$.

Tip

15% if the service is good.

Example
Thing1
Thing2
Thing3

Warning

Strong prose may provoke extreme mental exertion. Reader discretion is strongly advised.

And, by the way...

You can make up your own admonition too.

3.7.5. Topics, Sidebars, and Rubrics

Sidebar Title

Optional Subtitle

This is a sidebar. It is for text outside the flow of the main text.

This is a rubric inside a sidebar

Sidebars often appears beside the main text with a border and background color.

Topic Title

This is a topic.

This is a rubric

3.7.6. Target Footnotes

[5] (1,2,3)
http://www.python.org/

3.7.7. Replacement Text

I recommend you try Python, *the* best language around [5].

3.7.8. Compound Paragraph

This paragraph contains a literal block:

```
Connecting... OK
Transmitting data... OK
Disconnecting... OK
```

and thus consists of a simple paragraph, a literal block, and another simple paragraph. Nonetheless it is semantically *one* paragraph.

This construct is called a *compound paragraph* and can be produced with the "compound" directive.

3.8. Download Links

This long long long long long long long long long long long long long long long download link should be blue, normal weight text with a leading icon, and should wrap white-spaces

4. Lists & Tables

Table of Contents

4.1. Lists

4.1.1. Enumerated Lists

1. Arabic numerals.
    1. lower alpha)
        1. (lower roman)
            1. upper alpha.
                1. upper roman)
2. Lists that don't start at 1:
    1. Three
    2. Four
    1. C
    2. D
    1. iii
    2. iv
3. List items may also be auto-enumerated.

4.1.2. Definition Lists

Term

Definition

Termclassifier
Definition paragraph 1.

Definition paragraph 2.

Term
Definition

I have no clue why the definition list below is classified as a different style of definition list than the one above.

Is it the spaces in the term?
Maybe it was the multiple line paragraph in the line below that caused this?

Is it the paragraph above the list maybe?
I guess a lot of these lists don't have leading paragraphs?

Is it everything all at once?
Who knows?!

4.1.3. Option Lists

For listing command-line options:

`-a`
  command-line option “a”

`-b file`
  options can have arguments and long descriptions

`--long`
  options can be long also

`--input=file`
  long options can also have arguments

`--very-long-option`
  The description can also start on the next line.

  The description may contain multiple body elements, regardless of where it starts.

`-x, -y, -z`
  Multiple options are an “option group”.

`-v, --verbose`
  Commonly-seen: short & long options.

`-1 file, --one=file, --two file`
  Multiple options with arguments.

`/V`
  DOS/VMS-style options too

There must be at least two spaces between the option and the description.

4.1.4. Field list

Author:

David Goodger

Address:

123 Example Street Example, EX Canada A1B 2C3

Contact:

docutils-develop@lists.sourceforge.net

Authors:

Me; Myself; I

organization:

humankind

date:

$Date: 2012-01-03 19:23:53 +0000 (Tue, 03 Jan 2012) $

status:

This is a “work in progress”

revision:

$Revision: 7302 $

version:

1

copyright:

This document has been placed in the public domain. You may do with it as you wish. You may copy, modify, redistribute, reattribute, sell, buy, rent, lease, destroy, or improve it, quote it

at length, excerpt, incorporate, collate, fold, staple, or mutilate it, or do anything else to it that your or anyone else's heart desires.

field name:

This is a generic bibliographic field.

field name 2:

Generic bibliographic fields may contain multiple body elements.

Like this.

Dedication:

For Docutils users & co-developers.

abstract:

This document is a demonstration of the reStructuredText markup language, containing examples of all basic reStructuredText constructs and many advanced constructs.

4.1.5. Bullet Lists

4.1.5.1. Simple

- A simple list.
- There are no margins between list items.
- Simple lists do not contain multiple paragraphs. That's a complex list.
- In the case of a nested list
  - There are no margins between elements
    - Still no margins
      - Still no margins

4.1.5.2. Complex

- A bullet list
  - Nested bullet list.
  - Nested item 2.
- Item 2.

  Paragraph 2 of item 2.
  - Nested bullet list.
  - Nested item 2.
    - Third level.
    - Item 2.
  - Nested item 3.
- `inline literall`
- `inline literall`
- `inline literall`
- This item has multiple paragraphs.

  This item has multiple paragraphs.
- This item has multiple paragraphs.

  This item has multiple paragraphs.

4.1.5.3. Second list level

- here is a list in a second-level section.
- yahoo
- yahoo
  - yahoo
  - here is an inner bullet `oh`
    - one more `with an inline literally`. yahoo

heh heh. child. try to beat this embed:

```
  1 """Test Module for sphinx_rtd_theme."""
  2
  3
  4 class Foo:
  5     """Docstring for class Foo.
  6
  7     This text tests for the formatting of docstrings
generated from output
  8     ``sphinx.ext.autodoc``. Which contain reST, but sphinx
nests it in the
  9     ``<dl>``, and ``<dt>`` tags. Also, ``<tt>`` is used for
class, method names
 10     and etc, but those will *always* have the ``.descname``
or
```

- and another. yahoo
- yahoo
- `hi`

- how about an admonition?

> **Note**
>
> This is a note nested in a list.

- and hehe

4.1.5.3.1. But deeper down the rabbit hole

- I kept saying that, “deeper down the rabbit hole”. yahoo
  - I cackle at night yahoo.
- I’m so lonely here in GZ `guangzhou`
- A man of python destiny, hopes and dreams. yahoo
  - yahoo
    - yahoo `hi`
    - `destiny`

4.1.6. Hlists

- First item
- Second item
- Third item
- Forth item
- Fifth item
- Sixths item

Hlist with images

This is a short caption for a figure.

This is a long caption for a figure. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec porttitor dolor in odio posuere, vitae ornare libero mattis. In lobortis justo vestibulum nibh aliquet, non.

4.1.7. Numbered List

1. One,
2. Two.
3. Three with long text. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed feugiat sagittis neque quis eleifend. Duis rutrum lectus sit amet mattis suscipit.
    - 1. Using bullets and letters. (A)
    - 1. Using bullets and letters. (B)
    - 1. Using bullets and letters. (C)

4.2. Tables

4.2.1. Grid Tables

Here's a grid table followed by a simple table:

Header row, column 1 (header rows optional)	**Header 2**	**Header 3**	**Header 4**
body row 1, column 1	column 2	column 3	column 4
body row 2	Cells may span columns.		
body row 3	Cells may span rows.	• Table cells • contain • body elements.	
body row 4			
body row 5	Cells may also be empty: `-->`		

Inputs		**Output**
A	**B**	**A or B**
False	False	False
True	False	True
False	True	True

Inputs		Output
A	**B**	**A or B**
True	True	True

4.2.1.1. Giant Tables

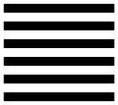
Header 1	Header 2	Header 3	Header 1	Header 2	Header 3	Header 1	Header 2	Header 3	Header 1	Header 2	Header 3
body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3
body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3
body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3
body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3	body row 1	column 2	column 3

4.2.2. List Tables

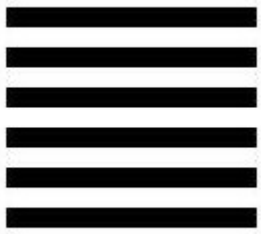
List tables can have captions like this one.

List table	Header 1	Header 2	Header 3 long. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam sit amet mauris arcu.
Stub Row 1	Row 1	Column 2	Column 3 long. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam sit amet mauris arcu.
Stub Row 2	Row 2	Column 2	Column 3 long. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam sit amet mauris arcu.
Stub Row 3	Row 3	Column 2	Column 3 long. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam sit amet mauris arcu.

This is a list table with images in it.



This is a short caption for a figure.



This is a long caption for a figure. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec porttitor dolor in odio posuere, vitae ornare libero mattis. In lobortis justo vestibulum nibh aliquet, non.

4.2.3. Tables with paragraphs

Test to see that tables behave well with nested paragraphs.

Precedence	Operator	Description
1	::	Scope resolution
2	( )	Function call
	[ ]	Subscript
	.	Member access
	.{ }	Bit-field concatenation

4.2.4. Tables with non-breakable text

True	This text could be broken at all whitespaces occuring in the text
False	This is a code block line, normaly could also be broken at all whitespaces in text
True	This_is_a_non_breakable_line_due_to_no_whitespaces_in_text_at_all_which_is_not_readable_without_breaking_it_working_if_you_can_read_THIS

4.2.5. CSV Table

The following table is too big for the PDF. There is no way to get a nice looking picture of it.

CSV Table

Example CSV							
City	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Munich	1	3	7	7	6	4	8
Paris	1	4	8	8	7	5	9
Moscow	2	34	6	3	4	35	7
Madrid	3	7	-2	3	5	8	6

4.2.6. Tiny tables

ssp-tinier

CSV Table

Example CSV							
City	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Munich	1	3	7	7	6	4	8
Paris	1	4	8	8	7	5	9
Moscow	2	34	6	3	4	35	7
Madrid	3	7	-2	3	5	8	6

ssp-tiny

CSV Table

Example CSV							
City	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Munich	1	3	7	7	6	4	8
Paris	1	4	8	8	7	5	9
Moscow	2	34	6	3	4	35	7

Example CSV							
City	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Madrid	3	7	-2	3	5	8	6

4.2.7. Landscape tables

Landscape page orientation and ssp-tiny table class

CSV Table

Example CSV																											
City	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**
Munich	1	3	7	7	6	4	8	8	7	5	9	9	5	9	9	8	6	10	10	7	11	11	8	12	12	9	13
Paris	1	4	8	8	7	5	9	9	6	6	10	10	6	10	10	7	7	11	11	8	12	12	9	13	13	10	14
Moscow	2	34	6	3	4	35	7	4	5	36	8	5	36	8	5	6	37	9	6	38	10	7	39	11	8	40	12
Madrid	3	7	-2	3	5	8	6	4	6	9	6	5	9	7	5	7	10	7	6	11	8	7	12	9	8	13	10
Rome	1	65	-34	4		66	7	5	6	67	6	6	67	8	6	7	68	7	7	69	8	8	70	9	9	71	10
Barcelona	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20	-21	-22
Berlin	5	3	1	-1	-3	-5	-7	-9	-11	-13	-15	-17	-19	-21	-23	-25	-27	-29	-31	-33	-35	-37	-39	-41	-43	-45	-47
New York	6	3	0	-3	-6	-9	-12	-15	-18	-21	-24	-27	-30	-33	-36	-39	-42	-45	-48	-51	-54	-57	-60	-63	-66	-69	-72
Tokyo	7	8	9	10	11	12	13	14	15	3	17	18	13	14	15	16	4	18	19	5	19	20	6	20	21	7	21
Melbourn	8	2	-4	-10	-16	-22	-28	-34	-40	-46	5	-58	-21	-27	-33	-39	-45	6	-57	-44	7	-56	-43	8	-55	-42	9
San Francisco	9	2	-5	-12	5	-26	-33	-40	-47	3	-61	-68	-25	-32	-39	-46	4	-60	-67	5	-59	-66	6	-58	-65	7	-57
Rio	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
Accra	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75	-82	-89	-96	-103	-110	-117	-124	-131	-138	-145	-152	-159	-166	-173
Munich	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74	-81	-88	-95	-102	-109	-116	-123	-130	-137	-144	-151	-158	-165	-172
Paris	11	4	-3	-10	-17	-24	-31	-38	-45	-52	-59	-66	-73	-80	-87	-94	-101	-108	-115	-122	-129	-136	-143	-150	-157	-164	-171
Moscow	12	5	-2	-9	-16	-23	-30	-37	-44	-51	-58	-65	-72	-79	-86	-93	-100	-107	-114	-121	-128	-135	-142	-149	-156	-163	-170
Madrid	13	6	-1	-8	-15	-22	-29	-36	-43	-50	-57	-64	-71	-78	-85	-92	-99	-106	-113	-120	-127	-134	-141	-148	-155	-162	-169
Rome	14	7	0	-7	-14	-21	-28	-35	-42	-49	-56	-63	-70	-77	-84	-91	-98	-105	-112	-119	-126	-133	-140	-147	-154	-161	-168
Barcelona	15	8	1	-6	-13	-20	-27	-34	-41	-48	-55	-62	-69	-76	-83	-90	-97	-104	-111	-118	-125	-132	-139	-146	-153	-160	-167
Berlin	16	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75	-82	-89	-96	-103	-110	-117	-124	-131	-138	-145	-152	-159	-166
New York	17	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74	-81	-88	-95	-102	-109	-116	-123	-130	-137	-144	-151	-158	-165
Tokyo	18	11	4	-3	-10	-17	-24	-31	-38	-45	-52	-59	-66	-73	-80	-87	-94	-101	-108	-115	-122	-129	-136	-143	-150	-157	-164
Melbourn	19	12	5	-2	-9	-16	-23	-30	-37	-44	-51	-58	-65	-72	-79	-86	-93	-100	-107	-114	-121	-128	-135	-142	-149	-156	-163
San Francisco	20	13	6	-1	-8	-15	-22	-29	-36	-43	-50	-57	-64	-71	-78	-85	-92	-99	-106	-113	-120	-127	-134	-141	-148	-155	-162
Rio	21	14	7	0	-7	-14	-21	-28	-35	-42	-49	-56	-63	-70	-77	-84	-91	-98	-105	-112	-119	-126	-133	-140	-147	-154	-161
Accra	22	15	8	1	-6	-13	-20	-27	-34	-41	-48	-55	-62	-69	-76	-83	-90	-97	-104	-111	-118	-125	-132	-139	-146	-153	-160

Example CSV																											
City	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**
Munich	23	16	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75	-82	-89	-96	-103	-110	-117	-124	-131	-138	-145	-152	-159
Paris	24	17	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74	-81	-88	-95	-102	-109	-116	-123	-130	-137	-144	-151	-158
Moscow	25	18	11	4	-3	-10	-17	-24	-31	-38	-45	-52	-59	-66	-73	-80	-87	-94	-101	-108	-115	-122	-129	-136	-143	-150	-157
Madrid	26	19	12	5	-2	-9	-16	-23	-30	-37	-44	-51	-58	-65	-72	-79	-86	-93	-100	-107	-114	-121	-128	-135	-142	-149	-156
Rome	27	20	13	6	-1	-8	-15	-22	-29	-36	-43	-50	-57	-64	-71	-78	-85	-92	-99	-106	-113	-120	-127	-134	-141	-148	-155
Barcelona	28	21	14	7	0	-7	-14	-21	-28	-35	-42	-49	-56	-63	-70	-77	-84	-91	-98	-105	-112	-119	-126	-133	-140	-147	-154
Berlin	29	22	15	8	1	-6	-13	-20	-27	-34	-41	-48	-55	-62	-69	-76	-83	-90	-97	-104	-111	-118	-125	-132	-139	-146	-153
New York	30	23	16	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75	-82	-89	-96	-103	-110	-117	-124	-131	-138	-145	-152
Tokyo	31	24	17	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74	-81	-88	-95	-102	-109	-116	-123	-130	-137	-144	-151

5. API documentation and generated content

Table of Contents

5.1. `test_py_module`

Test Module for sphinx_rtd_theme.

class test_py_module.test.Foo(*qux*, *spam=False*)

Docstring for class Foo.

This text tests for the formatting of docstrings generated from output `sphinx.ext.autodoc`. Which contain reST, but sphinx nests it in the `<dl>`, and `<dt>` tags. Also, `<tt>` is used for class, method names and etc, but those will *always* have the `.descname` or `.descclassname` class.

Term

It is also possible to include definitions inside docstrings. They should be styled as a normal definition list.

Field List:

It is also possible to include definitions inside docstrings.
They should be styled as a normal definition list.

[1]
A footnote contains body elements, consistently indented by at least 3 spaces.

[Citation]
A citation contains body elements, consistently indented by at least 3 spaces.

Normal `<tt>` (like the <tt> I just wrote here) needs to be shown with the same style as anything else with ``` ``this type of markup`` ```.

It's common for programmers to give a code example inside of their docstring:

```
from test_py_module import Foo

myclass = Foo()
myclass.dothismethod('with this argument')
myclass.flush()

print(myclass)
```

Here is a link to `capitalize()`. Here is a link to `__init__()`.

__init__(*qux*, *spam=False*)
Start the Foo.

Parameters:

- **qux** (*string*) – The first argument to initialize class.
- **spam** (*bool*) – Spam me yes or no...

__weakref__
list of weak references to the object

add(*val1*, *val2*)
Return the added values.

Parameters:

- **val1** (*int*) – First number to add.
- **val2** (*int*) – Second number to add.

Return type:

int

The parameters of this method are described in the parameter list.

another_function(*a*, *b*, ***kwargs*)

Here is another function.

Parameters:

- **a** (*int*) – The number of green hats you own.
- **b** (*int*) – The number of non-green hats you own.
- **kwargs** (*float*) – Additional keyword arguments. Each keyword parameter should specify the name of your favorite cuisine. The values should be floats, specifying the mean price of your favorite dish in that cooking style.

Returns:

A 2-tuple. The first element is the mean price of all dishes across cuisines. The second element is the total number of hats you own: $a + b$.

Return type:

tuple

Raises:

ValueError – When `a` is not an integer.

Added in version 1.0: This was added in 1.0

Changed in version 2.0: This was changed in 2.0

Deprecated since version 3.0: This is deprecated since 3.0

bar = *1*

Doc comment for class attribute Foo.bar. It can have multiple lines.

baz = *2*

Docstring for class attribute Foo.baz.

capitalize(*myvalue*)

Return a string as uppercase.

Parameters:

myvalue (*string*) – String to change

Return type:

string

flox *= 1.5*

Doc comment for Foo.flox. One line only.

qux

Doc comment for instance attribute qux.

spam

Docstring for instance attribute spam.

test_py_module.test.add_numbers(*a: int*, *b: int = 0*) → int

Add two numbers together

Parameters:

- **a** – The first number
- **b** – The second number

Here is some more text.

test_py_module.test.subtract_numbers(*a: int*, *b: int = 0*) → int

Subtract two numbers

Parameters:

- **a** – The first number
- **b** – The second number

5.2. C++ API

type MyType

Some type

const MyType Foo(const MyType bar)
Some function type thing

template<typename T, std::size_t N>
class std::array
Some cpp class

float Sphinx::version
The description of Sphinx::version.

int version
The description of version.

typedef std::vector<int> List
The description of List type.

enum MyEnum
An unscoped enum.

enumerator A

enum class MyScopedEnum
A scoped enum.

enumerator B

protected enum struct MyScopedVisibilityEnum : std::underlying_type<MySpecificEnum>::type
A scoped enum with non-default visibility, and with a specified underlying type.

enumerator B

5.3. JavaScript API

- Link to `ModTopLevel()`

class module_a.submodule.ModTopLevel()
- Link to `mod_child_1()`
- Link to `ModTopLevel.mod_child_1()`

ModTopLevel.mod_child_1()
- Link to `mod_child_2()`

ModTopLevel.mod_child_2()
- Link to `module_a.submodule.ModTopLevel.mod_child_1()`

- Link to `ModTopLevel()`

class module_b.submodule.ModNested()

ModNested.nested_child_1()

- Link to `nested_child_2()`

ModNested.nested_child_2()

- Link to `nested_child_1()`

5.4. Generated Index

Part of the sphinx build process in generate and index file: Index.

5.5. Optional parameter args

At this point optional parameters cannot be generated from code. However, some projects will manually do it, like so:

This example comes from django-payments module docs.

class payments.dotpay.DotpayProvider(*seller_id*, *pin*[, *channel=0*[, *lock=False*], *lang='pl'*])

This backend implements payments using a popular Polish gateway, Dotpay.pl.

Due to API limitations there is no support for transferring purchased items.

Parameters:

- **seller_id** – Seller ID assigned by Dotpay
- **pin** – PIN assigned by Dotpay
- **channel** – Default payment channel (consult reference guide)
- **lang** – UI language
- **lock** – Whether to disable channels other than the default selected above

5.6. Data

test_py_module.test.Data_item_1
test_py_module.test.Data_item_2

test_py_module.test.Data_item_3

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce congue elit eu hendrerit mattis.

Some data link `Data_item_1`.

6. Sphinx-Needs objects

Requirement: **Sphinx-Needs Theme extension support** REQ_001

status: done
tags: sphinx, extension
links incoming: SPEC_001

The `Sphinx-Needs Theme` for PDF shall support all possible Sphinx extensions an their outcome.

SPEC_001	Specification Example	Specification

status: open
tags: sphinx, example

links outgoing: REQ_001

A specification example with an image.

layout: complete

style: green_border, break

Requirement: **Sphinx-Needs Theme extension support with code examples** REQ_002

status: done
tags: sphinx, extension

The `Sphinx-Needs Theme` for PDF shall support also code examples or inline codes with long text

This_is_a_non_breakable_line_due_to_no_whitespaces_in_text_at_all_which_is_not_readable_without_breaking_it_working_if_you_can_read_THIS

7. Sphinx-Needs tables

ID	Title	Status	Tags
REQ_001	Sphinx-Needs Theme extension support	done	sphinx; extension
REQ_002	Sphinx-Needs Theme extension support with code examples	done	sphinx; extension
REQ_021	Sphinx-Needs Theme extension support with code examples from imported needs	done	sphinx; extension; imported
SPEC_001	Specification Example	open	sphinx; example

This is the same table, but with datatables style. This normally adds a scrollbar to tables if extending the normal layout size

ID	Title	Status	Tags	Content
REQ_001	Sphinx-Needs Theme extension support	done	sphinx; extension	The ``Sphinx-Needs Theme`` for PDF shall support all possible Sphinx extensions an their outcome.
	Sphinx-	done	sphinx;	The ``Sphinx-Needs Theme`` for PDF shall support also code examples or inline codes with long text

ID	Title	Status	Tags	Content
REQ_002	Needs Theme extension support with code examples		extension	This_is_a_non_breakable_line_due_to_no_whitespaces_in_text_at_all_which_is_not_readable_without_breaking_it_working_if_you_can_read_THIS
REQ_021	Sphinx-Needs Theme extension support with code examples from imported needs	done	sphinx; extension; imported	The ``Sphinx-Needs Theme`` for PDF shall support also text even if imported. This_is_a_non_breakable_line_due_to_no_whitespaces_in_text_at_all_which_is_not_readable_without_breaking_it_working_if_you_can_read_THIS
SPEC_001	Specification Example	open	sphinx; example	A specification example with an image. .. image:: /_static/example.jpg :align: center

8. Sphinx-Needs needflow

Using `plantuml` to render image.

9. Sphinx-Needs needimport

Requirement: **Sphinx-Needs Theme extension support with code examples from imported needs** REQ_021

status: done
tags: sphinx, extension, imported
duration: 0
completion: 0

The `Sphinx-Needs Theme` for PDF shall support also text even if imported. This_is_a_non_breakable_line_due_to_no_whitespaces_in_text_at_all_which_is_not_readable_without_breaking_it_working_if_you_can_read_THIS

10. Breadcrumb Level 1

10.1. `Breadcrumb Level 2`

10.1.1. Breadcrumb Level 3

10.1.1.1. Breadcrumb Level 4

10.1.1.1.1. Breadcrumb Level 5

10.1.1.1.1.1. Breadcrumb Level 6

11. Long Sticky Nav

Table of Contents

This section demonstrates how the ‘sticky_navigation’ setting behaves when the menu is very long. When this section is selected, it will make the menu and the main area scroll when you are at the top of the page.

11.1. Example Menu 1

Just a place holder...

11.2. Example Menu 2

Just a place holder...

11.3. Example Menu 3

Just a place holder...

11.4. Example Menu 4

Just a place holder...

11.5. Example Menu 5

Just a place holder...

11.6. Example Menu 6

Just a place holder...

11.7. Example Menu 7

Just a place holder...

11.8. Example Menu 8

Just a place holder...

11.9. Example Menu 9

Just a place holder...

11.10. Example Menu 10

Just a place holder...

11.11. Example Menu 11

Just a place holder...

11.12. Example Menu 12

Just a place holder...

11.13. Example Menu 13

Just a place holder...

11.14. Example Menu 14

Just a place holder...

11.15. Example Menu 15

Just a place holder...

11.16. Example Menu 16

Just a place holder...

11.17. Example Menu 17

Just a place holder...

11.18. Example Menu 18

Just a place holder...

11.19. Example Menu 19

Just a place holder...

11.20. Example Menu 20

Just a place holder...

11.21. Example Submenu 1

Just a place holder...

11.21.1. Submenu 1

Just a place holder...

11.21.1.1. Subsubmenu 1

Just a place holder...

11.21.1.2. Subsubmenu 2

Just a place holder...

11.21.2. Submenu 2

Just a place holder...

11.21.2.1. Subsubmenu 1

Just a place holder...

11.21.3. Submenu 3

Just a place holder...

11.21.4. Submenu 4

Just a place holder...

11.21.5. Submenu 5

Just a place holder...

11.22. Example Submenu 2

Just a place holder...

11.22.1. Submenu 1

Just a place holder...

11.22.1.1. Subsubmenu 1

Just a place holder...

11.22.2. Submenu 2

Just a place holder...

11.22.2.1. Subsubmenu 1

Just a place holder...

11.22.3. Submenu 3

Just a place holder...

11.22.4. Submenu 4

Just a place holder...

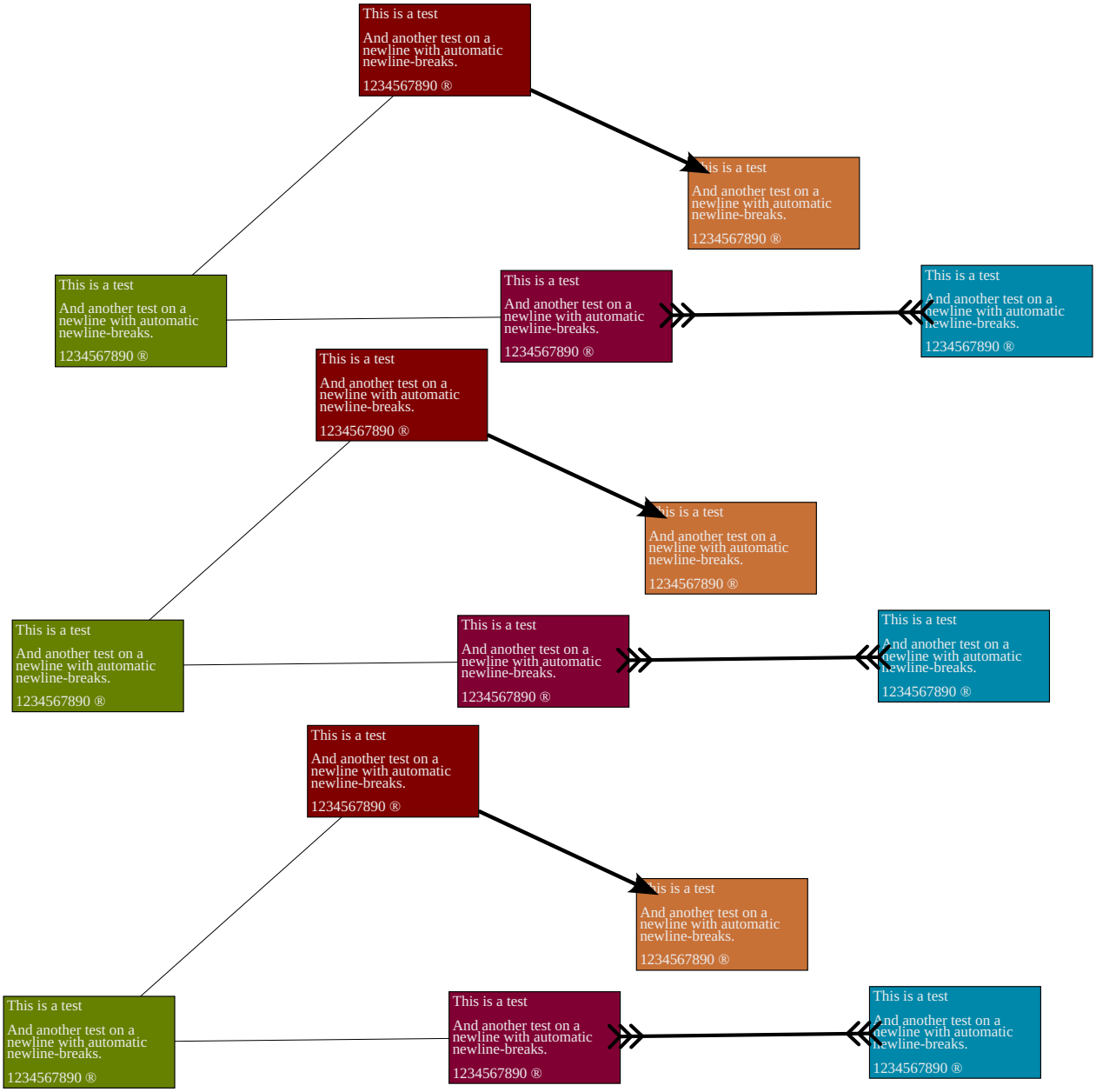
11.22.5. Submenu 5

Just a place holder...

12. Images

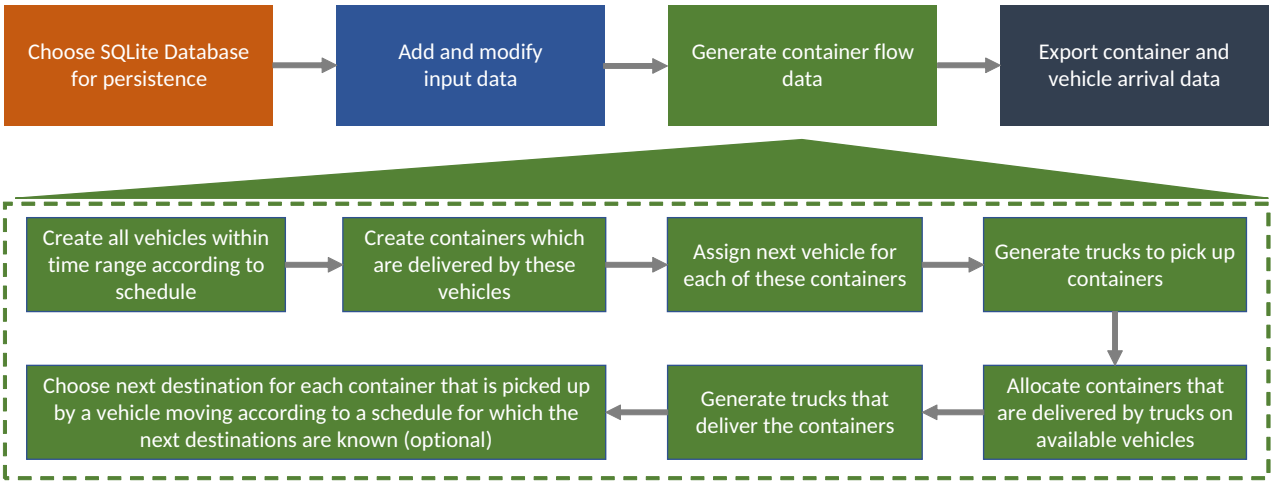
12.1. SVG

This is a SVG file with the dimensions 3000x3000 pixels.



12.1.1. Another svg

This svg files has no embedded fonts. The final look may be based on the fonts available on the machine used for the build.



12.2. PNG

This jpg file has the dimensions 2600x2176. File size is > 5 MB.

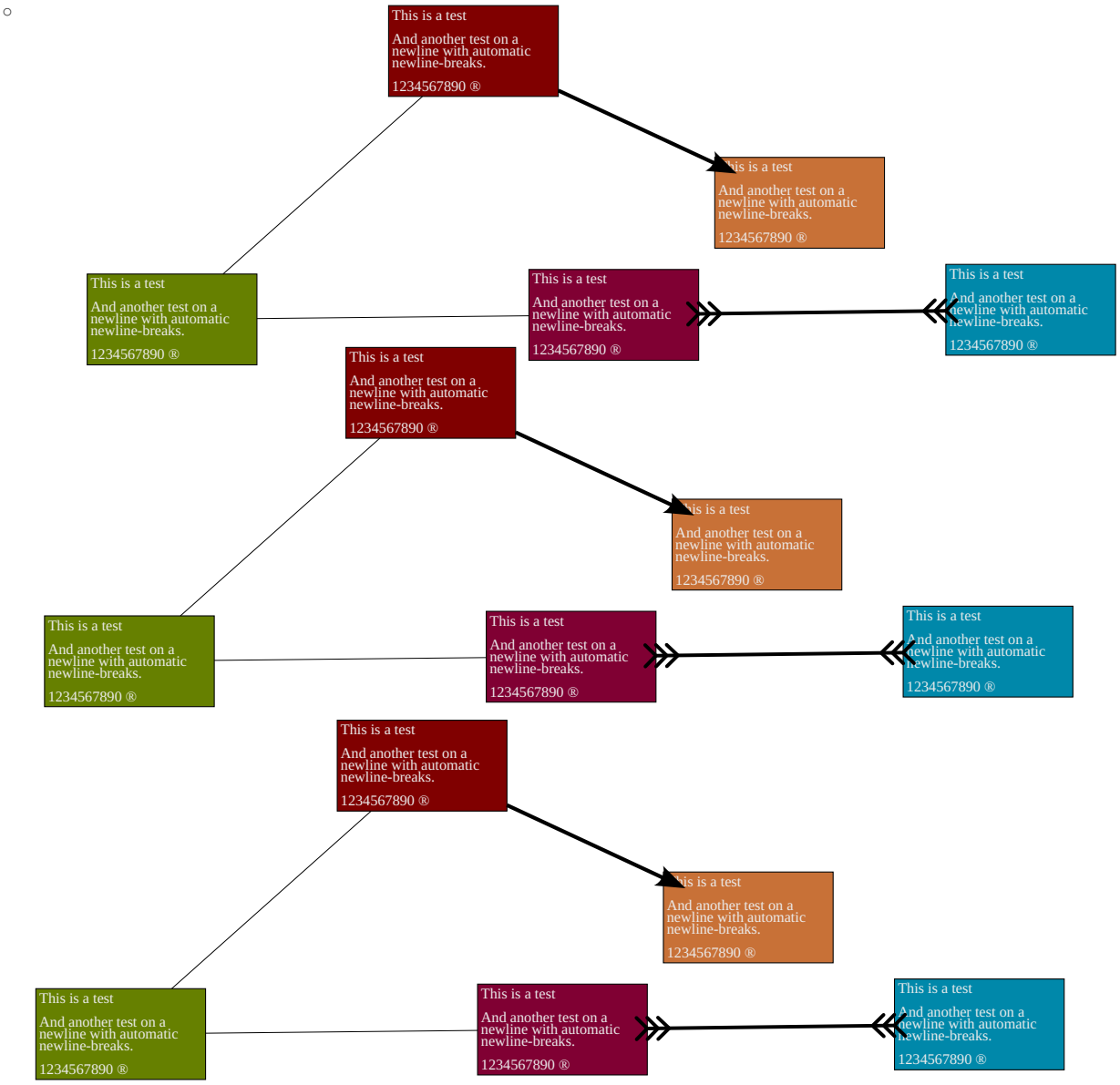
12.3. JPG

This jpg file has the dimensions 2600x2176.

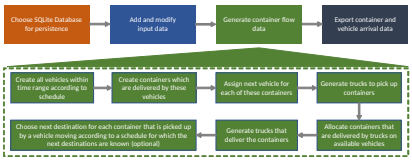
12.4. Images in lists

- An image on level 1

- 
  - An image on level 2



12.5. Images in tables

Image 1	Image 2
	

SimplePDF Debug output

This is some build environment specific output. It shall help to identify problems during the build process.

You see this output because **simplepdf_debug=True** is set on the **conf.py** file.

Sphinx

Version: 8.2.3
Srcdir: /home/docs/checkouts/readthedocs.org/user_builds/sphinx-simplepdf/checkouts/126/demo
Confdir: /home/docs/checkouts/readthedocs.org/user_builds/sphinx-simplepdf/checkouts/126/demo
Outdir: /home/docs/checkouts/readthedocs.org/user_builds/sphinx-simplepdf/checkouts/126/demo/_build/simplepdf

Extensions

Used Sphinx extension can be also found in the packages list of Python, which also includes the used version.

sphinx_simplepdf
sphinxcontrib.plantuml
sphinx_needs
sphinx.ext.autodoc
sphinx.ext.imgmath

SimplePDF Configs

simplepdf_vars: {'cover-overlay': 'rgba(26, 150, 26, 0.7)', 'primary-opaque': 'rgba(26, 150, 26, 0.7)', 'cover-bg': 'url(frog.jpg) no-repeat center', 'primary': '#1a961a', 'secondary': '#379683', 'cover': '#ffffff', 'white': '#ffffff', 'links': '#1a961a', 'top-left-content': '"Header left"', 'top-center-content': '"Header center"', 'top-right-content': '"Header right"', 'bottom-left-content': 'counter(page)', 'bottom-center-content': '"Bottom center"', 'bottom-right-content': 'string(heading)'}
simplepdf_file_name: Sphinx-SimplePDF-DEMO.pdf
simplepdf_debug: True
simplepdf_weasyprint_timeout: None
simplepdf_weasyprint_retries: 0
simplepdf_weasyprint_flags: None
simplepdf_weasyprint_filter: []
simplepdf_use_weasyprint_api: None
simplepdf_theme: simplepdf_theme
simplepdf_theme_options: {}
simplepdf_sidebars: {'**': ['localtoc.html']}

Python

Executable: /home/docs/.asdf/installs/python/3.12.10/bin/python3.12
Operating System: Linux (Release: 6.8.0-1029-aws)

Packages

This chapter shows a list of installed packages in the current Python environment, which was used to build this PDF. The second value is the version number.

Important packages

PIL: unknown
sphinx: 8.2.3
sphinx_simplepdf: 1.7.0
weasyprint: 68.0

Other packages

__future__: unknown
__hello__: unknown
__phello__: unknown
_aix_support: unknown
_asyncio: unknown
_bisect: unknown
_blake2: unknown
_brotli: unknown
_bz2: unknown
_cffi_backend: unknown
_codecs_cn: unknown
_codecs_hk: unknown
_codecs_iso2022: unknown
_codecs_jp: unknown
_codecs_kr: unknown
_codecs_tw: unknown
_collections_abc: unknown

_compat_pickle: unknown
_compression: unknown
_contextvars: unknown
_crypt: unknown
_csv: unknown
_ctypes: unknown
_ctypes_test: unknown
_curses: unknown
_curses_panel: unknown
_datetime: unknown
_dbm: unknown
_decimal: unknown
_elementtree: unknown
_gdbm: unknown
_hashlib: unknown
_heapq: unknown
_json: unknown
_lsprof: unknown
_lzma: unknown
_markupbase: unknown
_md5: unknown
_multibytecodec: unknown
_multiprocessing: unknown
_opcode: unknown
_osx_support: unknown
_pickle: unknown
_posixshmem: unknown
_posixsubprocess: unknown
_py_abc: unknown
_pydatetime: unknown
_pydecimal: unknown
_pyio: unknown
_pylong: unknown
_queue: unknown
_random: unknown
_sass: unknown
_sha1: unknown
_sha2: unknown
_sha3: unknown
_sitebuiltins: unknown
_socket: unknown
_sqlite3: unknown
_ssl: unknown

_statistics: unknown
_strptime: unknown
_struct: unknown
_sysconfigdata__linux_x86_64-linux-gnu: unknown
_testbuffer: unknown
_testcapi: unknown
_testclinic: unknown
_testimportmultiple: unknown
_testinternalcapi: unknown
_testmultiphase: unknown
_testsinglephase: unknown
_threading_local: unknown
_tkinter: unknown
_uuid: unknown
_weakrefset: unknown
_xxinterpchannels: unknown
_xxsubinterpreters: unknown
_xxtestfuzz: unknown
_zoneinfo: unknown
abc: unknown
aifc: unknown
alabaster: 1.0.0
antigravity: unknown
argparse: unknown
array: unknown
ast: unknown
asyncio: unknown
audioop: unknown
babel: 2.17.0
base64: unknown
bdb: unknown
binascii: unknown
bisect: unknown
brotli: 1.2.0
bs4: unknown
bz2: unknown
calendar: unknown
certifi: 2026.1.4
cffi: 2.0.0
cgi: unknown
cgitb: unknown
charset_normalizer: 3.4.4
chunk: unknown

cmath: unknown
cmd: unknown
code: unknown
codecs: unknown
codeop: unknown
collections: unknown
colorsys: unknown
compileall: unknown
concurrent: unknown
conf: unknown
configparser: unknown
contextlib: unknown
contextvars: unknown
copy: unknown
copyreg: unknown
cProfile: unknown
crypt: unknown
cssselect2: 0.8.0
csv: unknown
ctypes: unknown
curses: unknown
dataclasses: unknown
datetime: unknown
dbm: unknown
decimal: unknown
difflib: unknown
dis: unknown
distlib: 0.3.9
doctest: unknown
docutils: 0.21.2
email: unknown
encodings: unknown
ensurepip: unknown
enum: unknown
fcntl: unknown
filecmp: unknown
fileinput: unknown
filelock: 3.18.0
fnmatch: unknown
fontTools: 4.61.1
fractions: unknown
ftplib: unknown
functools: unknown

genericpath: unknown
getopt: unknown
getpass: unknown
gettext: unknown
glob: unknown
graphlib: unknown
grp: unknown
gzip: unknown
hashlib: unknown
heapq: unknown
hmac: unknown
html: unknown
http: unknown
idlelib: unknown
idna: 3.11
imagesize: 1.4.1
imaplib: unknown
imghdr: unknown
importlib: unknown
inspect: unknown
io: unknown
ipaddress: unknown
jinja2: 3.1.6
json: unknown
jsonschema_rs: 0.37.4
keyword: unknown
lib2to3: unknown
linecache: unknown
locale: unknown
logging: unknown
lzma: unknown
mailbox: unknown
mailcap: unknown
markupsafe: 3.0.3
math: unknown
mimetypes: unknown
mmap: unknown
modulefinder: unknown
multiprocessing: unknown
netrc: unknown
nis: unknown
nntplib: unknown
ntpath: unknown

nturl2path: unknown
numbers: unknown
opcode: unknown
operator: unknown
optparse: unknown
os: unknown
ossaudiodev: unknown
packaging: 26.0
pathlib: unknown
pdb: unknown
pickle: unknown
pickletools: unknown
pip: 25.0.1
pipes: unknown
pkgutil: unknown
platform: unknown
platformdirs: 3.11.0
plistlib: unknown
poplib: unknown
posixpath: unknown
pprint: unknown
profile: unknown
pstats: unknown
pty: unknown
py_compile: unknown
pyclbr: unknown
pycparser: 3.0
pydoc: unknown
pydoc_data: unknown
pydyf: 0.12.1
pyexpat: unknown
pygments: 2.19.2
pyphen: 0.17.2
pysassc: unknown
queue: unknown
quopri: unknown
random: unknown
re: unknown
readline: unknown
reprlib: unknown
requests: 2.32.5
requests_file: 2.1.0
resource: unknown

rlcompleter: unknown
roman_numerals: 4.1.0
runpy: unknown
sass: unknown
sasstests: unknown
sassutils: unknown
sched: unknown
secrets: unknown
select: unknown
selectors: unknown
shelve: unknown
shlex: unknown
shutil: unknown
signal: unknown
site: unknown
smtplib: unknown
sndhdr: unknown
snowballstemmer: 3.0.1
socket: unknown
socketserver: unknown
soupsieve: 2.8.3
sphinx_data_viewer: 0.1.5
sphinx_needs: 6.2.0
spwd: unknown
sqlite3: unknown
sre_compile: unknown
sre_constants: unknown
sre_parse: unknown
ssl: unknown
stat: unknown
statistics: unknown
string: unknown
stringprep: unknown
struct: unknown
subprocess: unknown
sunau: unknown
symtable: unknown
sysconfig: unknown
syslog: unknown
tabnanny: unknown
tarfile: unknown
telnetlib: unknown
tempfile: unknown

termios: unknown
test: unknown
test_py_module: unknown
textwrap: unknown
this: unknown
threading: unknown
timeit: unknown
tinycss2: 1.5.1
tinyhtml5: 2.0.0
tkinter: unknown
token: unknown
tokenize: unknown
tomllib: unknown
trace: unknown
traceback: unknown
tracemalloc: unknown
tty: unknown
turtle: unknown
turtledemo: unknown
typeguard: 4.4.4
types: unknown
typing: unknown
typing_extensions: 4.15.0
unicodedata: unknown
unittest: unknown
urllib: unknown
urllib3: 2.6.3
uu: unknown
uuid: unknown
venv: unknown
virtualenv: 20.21.1
warnings: unknown
wave: unknown
weakref: unknown
webbrowser: unknown
webencodings: 0.5.1
wsgiref: unknown
xdrlib: unknown
xml: unknown
xmlrpc: unknown
xxlimited: unknown
xxlimited_35: unknown
xxsubtype: unknown

zipapp: unknown
zipfile: unknown
zipimport: unknown
zlib: unknown
zoneinfo: unknown
zopfli: 0.4.0