

# Navigation

- [Sphinx-SimplePDF-DEMO documentation »](#)
- [Sphinx-SimplePDF-DEMO documentation](#)



# 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.

# Structural Elements

## Table of Contents

---

<b>Structural Elements</b>	<b>3</b>
● Document Section	4
● Document Subsection	4
● Document Subsubsection	4
● Document Paragraph	4
<b>Structural Elements 2</b>	<b>5</b>
● Document Section	5
● Document Subsection	5

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.



## 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.

### 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.

### 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.

### 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.

# 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.

## 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.

## 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.

# Paragraph Level Markup

## Table of Contents

---

<b>Paragraph Level Markup</b>	<b>7</b>
● Inline Markup	8
● Math	9
● Meta	10
● Blocks	10
● Literal Blocks	10
● Line Blocks	10
● Block Quotes	11
● Doctest Blocks	11
● Code Blocks	11
● Emphasized lines with line numbers	12
● Sidebar	12
● Code with Sidebar	13
● References	14
● Footnotes	14
● Citations	15
● Glossary	15

---

● Targets	15
● Directives	16
● Contents	16
● Centered text	16
● Images & Figures	16
● Images	16
● Figures	17
● Admonitions	18
● Topics, Sidebars, and Rubrics	20
● Target Footnotes	20
● Replacement Text	20
● Compound Paragraph	20
● Download Links	21

---

## Inline Markup

Paragraphs contain text and may contain inline markup: *emphasis* , **strong emphasis** , `code` , 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 `code` Text* . Problems are indicated by `|problematic|` text (generated by processing errors; this one is intentional).

Also with `code` , which I use in the demo, I can link to `code` . 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*.

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 so as not to interfere with text from adjacent lines.

Key-bindings indicate that the user is to press a button on the keyboard or mouse, for example and . Another useful markup to indicate a user action is to use this can be used to show short and long menus in software. For example, and 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:

If the 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?](https://www.google.com/search?hl=en&q=very%20long%20url%20example%20of%20a%20url%20that%20is%20extremely%20long%20you%20pr)

[hl=en&q=very%20long%20url%20example%20of%20a%20url%20that%20is%20extremely%20long%20you%20pr](https://www.google.com/search?hl=en&q=very%20long%20url%20example%20of%20a%20url%20that%20is%20extremely%20long%20you%20pr)

## Math

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

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

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

## Meta

## Blocks

### Literal Blocks

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

```
if
```

```
None
```

Or they can be quoted without indentation:

### 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...

## 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 brontosaurus 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)

## Doctest Blocks

```
>>>
```

```
>>>
```

## Code Blocks

Code Blocks can have captions.

```
"windows"
```

```
    "panes"
```

```
        "shell_command"
```

```
"shell_command"
```

```
"window_name"
```

```
"session_name"
```

## Emphasized lines with line numbers

```
def
```

```
False
```

## 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.

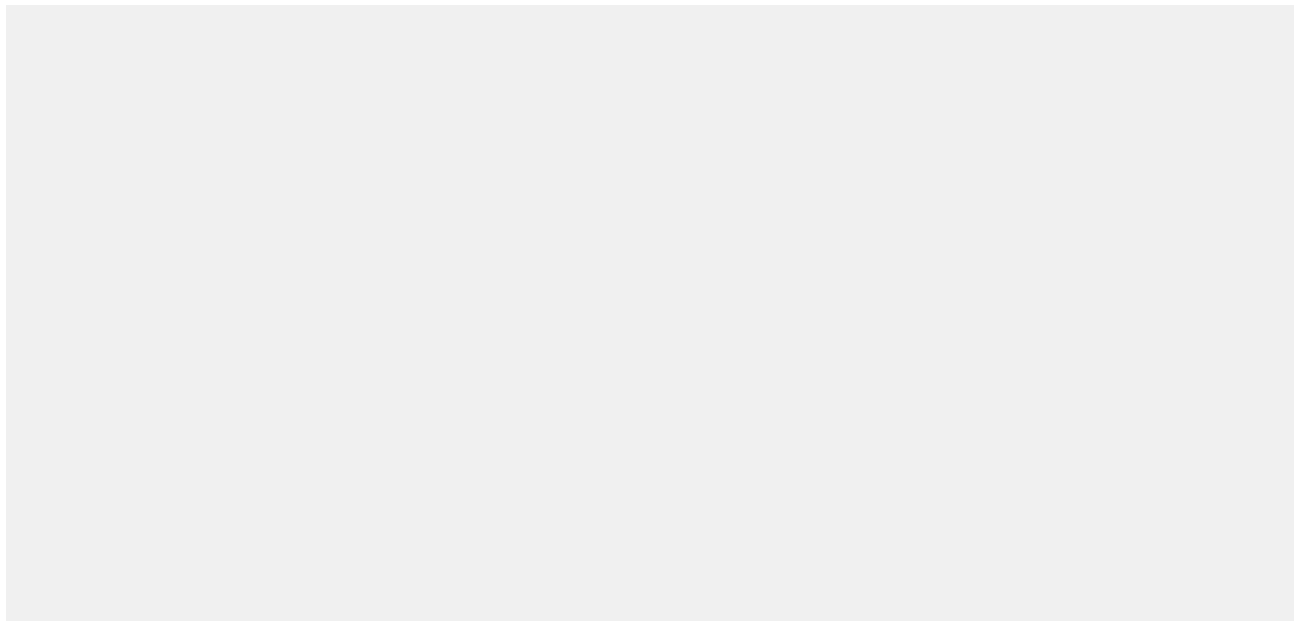
## Code with Sidebar

A code example

With a sidebar on the right.

Literal includes can also have captions.

```
# -*- coding: utf-8 -*-  
"""Test Module for sphinx_rtd_theme."""  
  
class Foo
```



## References

### 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: [\[5\]](#)\_ .



## 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  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.

## 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](#) .

## 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.

# Directives

## Contents

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

## Centered text

You can create a statement with centered text with `..text-align::center`

This is centered text!

## Images & Figures

### Images

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



## 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.

## 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:  $\rho_{\mu\nu} = 8\pi G(T_{\mu\nu} + \rho_{\Lambda} g_{\mu\nu})$ .

**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.

## 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

## Target Footnotes

[ 5 ] ( 1 , 2 , 3 )

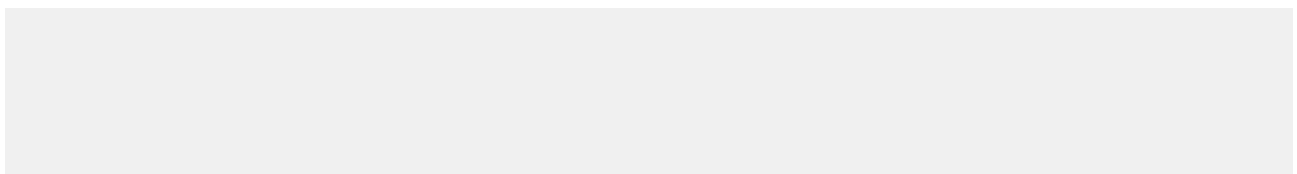
<http://www.python.org/>

## Replacement Text

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

## Compound Paragraph

This paragraph contains a literal block:



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.



## Download Links

⬇ This long long long long long long long long long long long long long long long long download link should be blue, normal weight text and should wrap white-spaces

# Lists & Tables

## Table of Contents

---

<b>Lists &amp; Tables</b>	<b>22</b>
● Lists	23
● Enumerated Lists	23
● Definition Lists	23
● Option Lists	24
● Field list	25
● Bullet Lists	27
● Simple	27
● Complex	27
● Second list level	28
● But deeper down the rabbit hole	29
● Hlists	29
● Numbered List	30
● Tables	30
● Grid Tables	30
● Giant Tables	31
● List Tables	32

---

- Tables with paragraphs

33

## Lists

### 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.

### Definition Lists

#### **Term**

Definition

#### **Term classifier**

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?!

## Option Lists

For listing command-line options:

*file*

command-line option "a"

options can have arguments and long descriptions

options can be long also

*file*

long options can also have arguments

The description can also start on the next line.

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

Multiple options are an "option group".

Commonly-seen: short & long options.

*file*

Multiple options with arguments.

*file*            *file*

DOS/VMS-style options too

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

## Field list

**Author :** David Goodger

**Address :** 123 Example Street Example, EX Canada A1B 2C3

**Contact :** [docutils-develop@lists.sourceforge.net](mailto: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.



## Bullet Lists

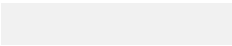
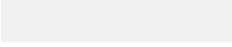
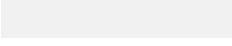
### Simple

- 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

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

This item has multiple paragraphs.

- This item has multiple paragraphs.

This item has multiple paragraphs.

## Second list level

- here is a list in a second-level section.

- yahoo

- yahoo

- yahoo

- here is an inner bullet

- one more . yahoo

heh heh. child. try to beat this embed:

```
# -*- coding: utf-8 -*-  
"""Test Module for sphinx_rtd_theme."""  
  
class Foo
```

- and another. yahoo

- yahoo

- 

- how about an admonition?

### **i** Note

This is a note nested in a list.

- and hehe

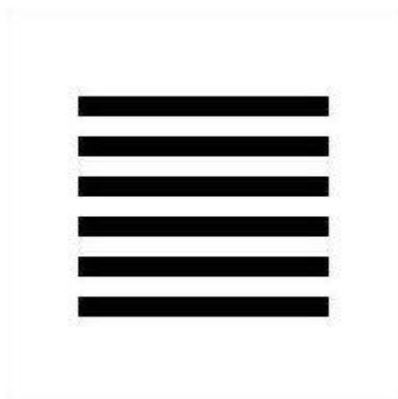
## But deeper down the rabbit hole

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

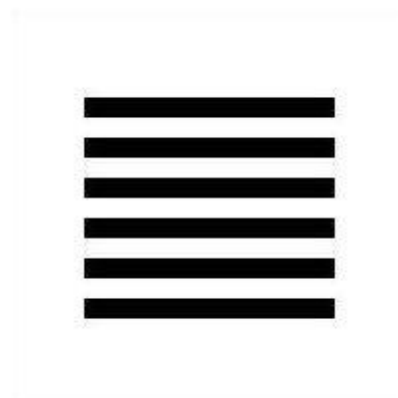
## Hlists

- |             |             |
|-------------|-------------|
| First item  | Forth item  |
| Second item | Fifth item  |
| Third 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.

## 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)

## Tables

### 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.	<ul style="list-style-type: none"> <li>• Table cells</li> </ul>	
body row 4			<ul style="list-style-type: none"> <li>• contain</li> <li>• body elements.</li> </ul>

<b>Header row, column 1 (header optional)</b>	<b>Header 2</b>	<b>Header 3</b>	<b>Header 4</b>
---	-----------------	-----------------	-----------------

body row 5

Cells may also be empty:

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

## Giant Tables

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



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

## 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.

	
<p>This is a short caption for a figure.</p>	<p>This is a long caption for a figure. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec porttitor dolor in odio posuere,</p>

vitae ornare libero mattis. In  
lobortis justo vestibulum nibh  
aliquet, non.

## Tables with paragraphs

Test to see that tables behave well with nested paragraphs.

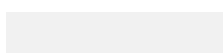
Precedence	Operator	Description
1	::	Scope resolution
2	()	Function call
	[]	Subscript
	.	Member access
	.{ }	Bit-field concatenation

# API documentation and generated content

## Table of Contents

---

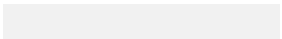



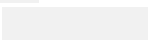
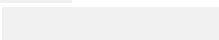
API documentation and generated content	34
● 	34
● C++ API	37
● JavaScript API	38
● Generated Index	39
● Optional parameter args	39
● Data	40



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  . Which contain reST, but sphinx nests it in the  , and  tags. Also,  is used for class, method names and etc, but those will *always* have the  or  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 `<code>` (like the `<tt>` I just wrote here) needs to be shown with the same style as anything else with `<code>`.

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

```
from test_py_module import
```

Here is a link to `<code>`. Here is a link to `<code>`.

`<code>` (*qux*, *spam* = *False*)

Start the Foo.

Parameters :

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

`<code>`

list of weak references to the object (if defined)

`<code>` (*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 `█` is not an integer.

New 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

## 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**

## JavaScript API

• Link to 

`class module_a . submodule . ModTopLevel ( )`

• Link to 

• Link to 

`ModTopLevel . mod_child_1 ( )`

• Link to 

`ModTopLevel . mod_child_2 ( )`

• Link to 

• Link to 

```
class module_b . submodule . ModNested ( )
```

```
ModNested . nested_child_1 ( )
```

• Link to 

```
ModNested . nested_child_2 ( )
```

• Link to 

## Generated Index

Part of the sphinx build process in generate and index file: [Index](#) .

## 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

## 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  .

# Sphinx-Needs objects

Requirement : **Sphinx-Needs Theme extension support** [REQ\\_001](#)

status: done

tags: sphinx , extension

links incoming: [SPEC\\_001](#)

The   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

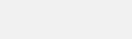
style: green\_border,  
break

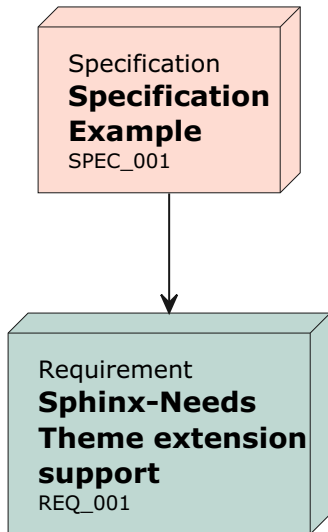


# Sphinx-Needs tables

ID	Title	Status	Tags
REQ_001	Sphinx-Needs Theme extension support	done	sphinx ; extension
SPEC_001	Specification Example	open	sphinx ; example

# Sphinx-Needs needflow

Using  to render image.



## CSV Table

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

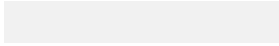
CSV Table

<b>Example CSV</b>															
<b>City</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>
Munich	1	3	7	7	6	4	8	8	7	5	9	9	5	9	9
Paris	1	4	8	8	7	5	9	9	6	6	10	10	6	10	10
Moscow	2	34	6	3	4	35	7	4	5	36	8	5	36	8	5
Madrid	3	7	-2	3	5	8	6	4	6	9	6	5	9	7	5
Rome	1	65	-34	4		66	7	5	6	67	6	6	67	8	6
Barcelona	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
Berlin	5	3	1	-1	-3	-5	-7	-9	-11	-13	-15	-17	-19	-21	-23
New York	6	3	0	-3	-6	-9	-12	-15	-18	-21	-24	-27	-30	-33	-36
Tokyo	7	8	9	10	11	12	13	14	15	3	17	18	13	14	15
Melbourn	8	2	-4	-10	-16	-22	-28	-34	-40	-46	5	-58	-21	-27	-33
San Francisco	9	2	-5	-12	5	-26	-33	-40	-47	3	-61	-68	-25	-32	-39
Rio	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Accra	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75	-82	-89

<b>Example CSV</b>															
<b>City</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>
Munich	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74	-81	-88
Paris	11	4	-3	-10	-17	-24	-31	-38	-45	-52	-59	-66	-73	-80	-87
Moscow	12	5	-2	-9	-16	-23	-30	-37	-44	-51	-58	-65	-72	-79	-86
Madrid	13	6	-1	-8	-15	-22	-29	-36	-43	-50	-57	-64	-71	-78	-85
Rome	14	7	0	-7	-14	-21	-28	-35	-42	-49	-56	-63	-70	-77	-84
Barcelona	15	8	1	-6	-13	-20	-27	-34	-41	-48	-55	-62	-69	-76	-83
Berlin	16	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75	-82
New York	17	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74	-81
Tokyo	18	11	4	-3	-10	-17	-24	-31	-38	-45	-52	-59	-66	-73	-80
Melbourn	19	12	5	-2	-9	-16	-23	-30	-37	-44	-51	-58	-65	-72	-79
San Francisco	20	13	6	-1	-8	-15	-22	-29	-36	-43	-50	-57	-64	-71	-78
Rio	21	14	7	0	-7	-14	-21	-28	-35	-42	-49	-56	-63	-70	-77
Accra	22	15	8	1	-6	-13	-20	-27	-34	-41	-48	-55	-62	-69	-76

<b>Example CSV</b>															
<b>City</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>
Munich	23	16	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68	-75
Paris	24	17	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67	-74
Moscow	25	18	11	4	-3	-10	-17	-24	-31	-38	-45	-52	-59	-66	-73
Madrid	26	19	12	5	-2	-9	-16	-23	-30	-37	-44	-51	-58	-65	-72
Rome	27	20	13	6	-1	-8	-15	-22	-29	-36	-43	-50	-57	-64	-71
Barcelona	28	21	14	7	0	-7	-14	-21	-28	-35	-42	-49	-56	-63	-70
Berlin	29	22	15	8	1	-6	-13	-20	-27	-34	-41	-48	-55	-62	-69
New York	30	23	16	9	2	-5	-12	-19	-26	-33	-40	-47	-54	-61	-68
Tokyo	31	24	17	10	3	-4	-11	-18	-25	-32	-39	-46	-53	-60	-67

# Breadcrumb Level 1



## Breadcrumb Level 3

Breadcrumb Level 4

Breadcrumb Level 5

Breadcrumb Level 6

# Long Sticky Nav

## Table of Contents

---

<b>Long Sticky Nav</b>	<b>49</b>
● Example Menu 1	51
● Example Menu 2	51
● Example Menu 3	51
● Example Menu 4	51
● Example Menu 5	51
● Example Menu 6	51
● Example Menu 7	51
● Example Menu 8	52
● Example Menu 9	52
● Example Menu 10	52
● Example Menu 11	52
● Example Menu 12	52
● Example Menu 13	52
● Example Menu 14	52
● Example Menu 15	52
● Example Menu 16	53

---

- Example Menu 17 53

---

- Example Menu 18 53

---

- Example Menu 19 53

---

- Example Menu 20 53

---

- Example Submenu 1 53
  - Submenu 1 53
    - Subsubmenu 1 53

---

    - Subsubmenu 2 53

---

  - Submenu 2 54
    - Subsubmenu 1 5

---

  - Submenu 3 54

---

  - Submenu 4 54

---

  - Submenu 5 54

---

- Example Submenu 2 54
  - Submenu 1 5
    - Subsubmenu 1 5

---

  - Submenu 2 3
    - Subsubmenu 1 3

---

  - Submenu 3 3

---

  - Submenu 4 3

---

  - Submenu 5 3

---



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.

## Example Menu 1

Just a place holder...

## Example Menu 2

Just a place holder...

## Example Menu 3

Just a place holder...

## Example Menu 4

Just a place holder...

## Example Menu 5

Just a place holder...

## Example Menu 6

Just a place holder...

## Example Menu 7

Just a place holder...

## Example Menu 8

Just a place holder...

## Example Menu 9

Just a place holder...

## Example Menu 10

Just a place holder...

## Example Menu 11

Just a place holder...

## Example Menu 12

Just a place holder...

## Example Menu 13

Just a place holder...

## Example Menu 14

Just a place holder...

## Example Menu 15

Just a place holder...

## Example Menu 16

Just a place holder...

## Example Menu 17

Just a place holder...

## Example Menu 18

Just a place holder...

## Example Menu 19

Just a place holder...

## Example Menu 20

Just a place holder...

## Example Submenu 1

Just a place holder...

### Submenu 1

Just a place holder...

### Subsubmenu 1

Just a place holder...

### Subsubmenu 2

Just a place holder...

## Submenu 2

Just a place holder...

### Subsubmenu 1

Just a place holder...

## Submenu 3

Just a place holder...

## Submenu 4

Just a place holder...

## Submenu 5

Just a place holder...

## Example Submenu 2

Just a place holder...

## Submenu 1

Just a place holder...

### Subsubmenu 1

Just a place holder...

## Submenu 2

Just a place holder...

## Subsubmenu 1

Just a place holder...

## Submenu 3

Just a place holder...

## Submenu 4

Just a place holder...

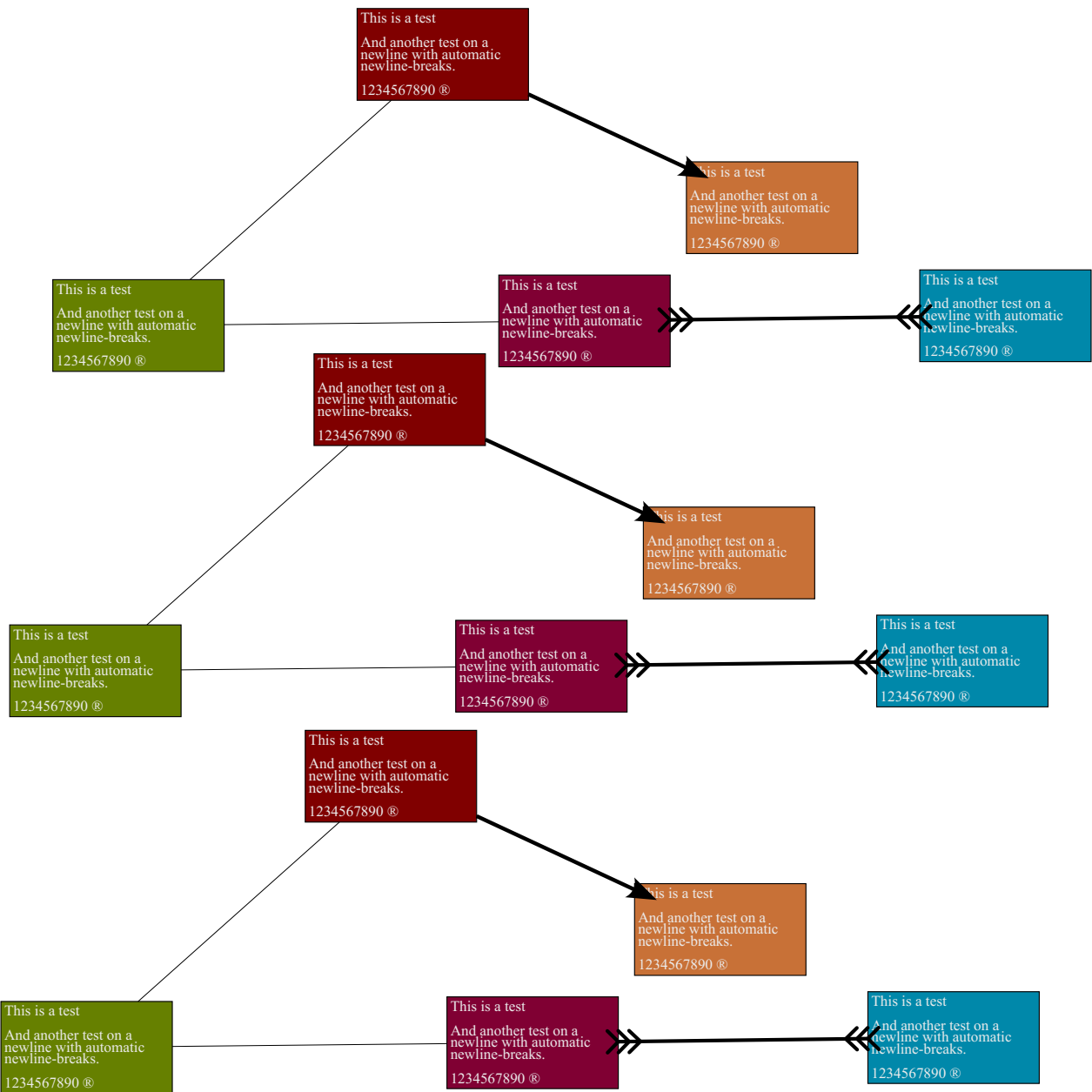
## Submenu 5

Just a place holder...

# Images

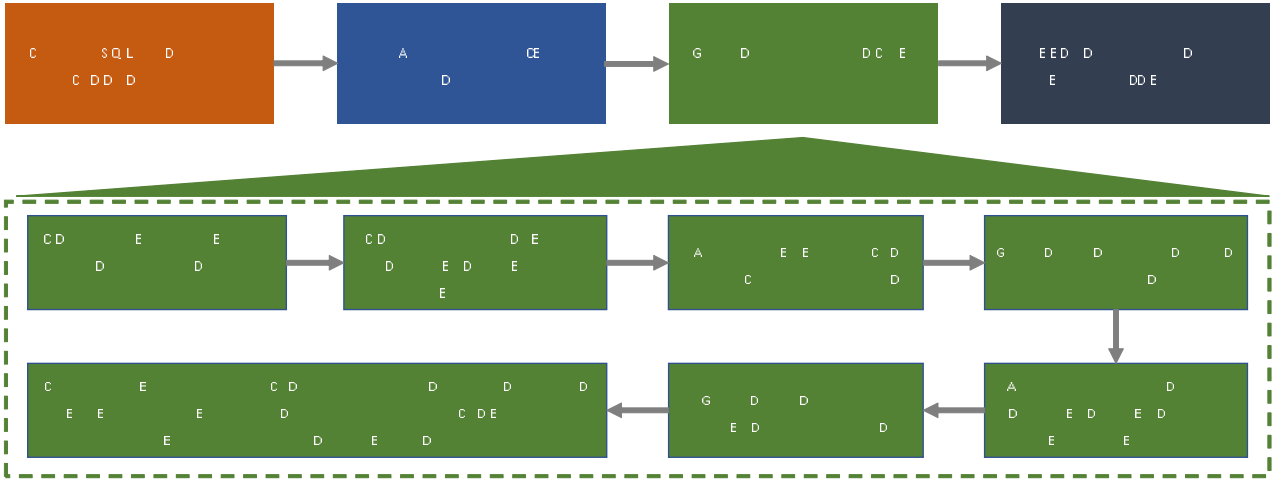
## SVG

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



## 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.



## PNG

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



JPG

This jpg file has the dimensions 2600x2176.





## Images in lists

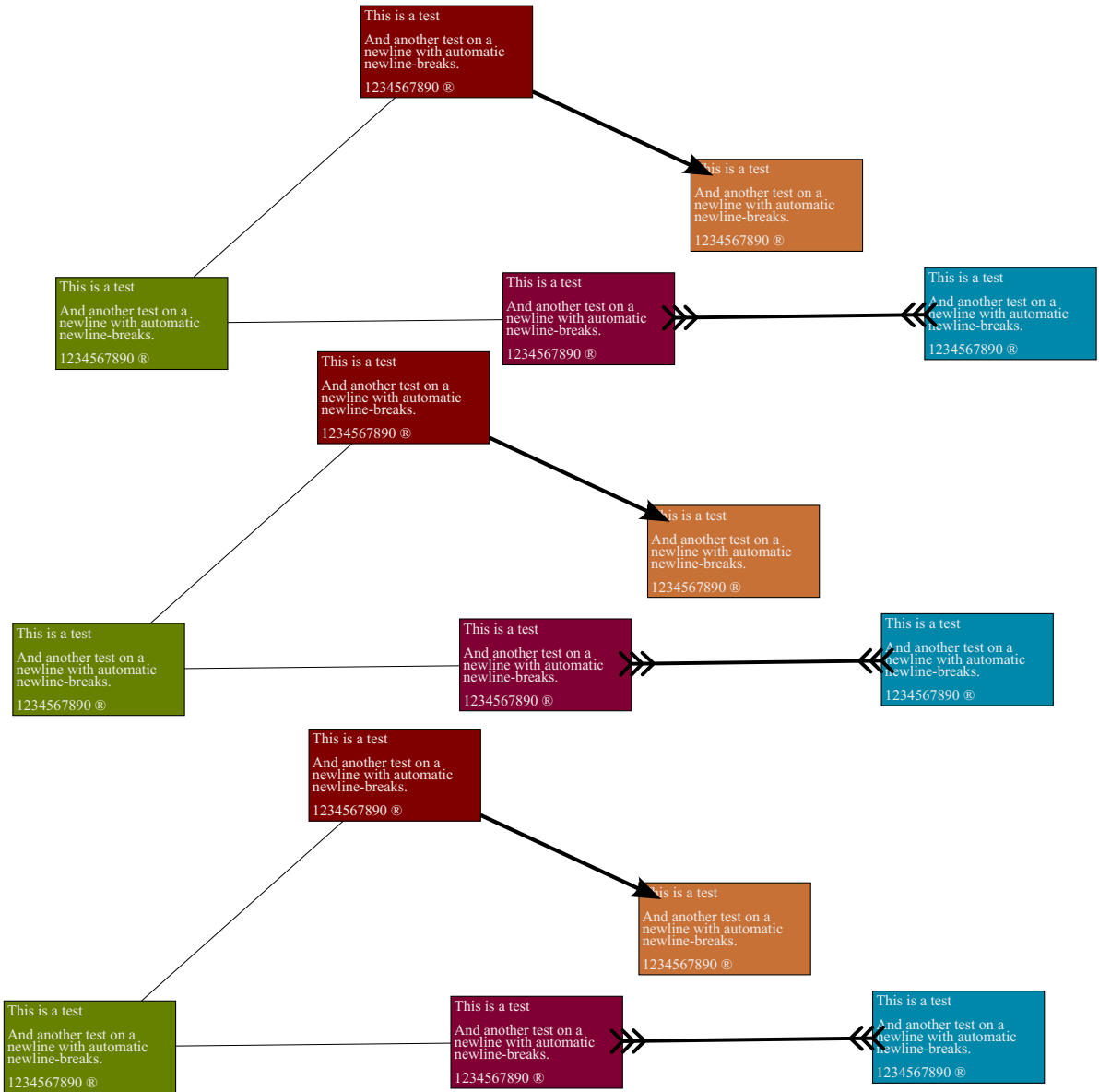
- An image on level 1





◦ An image on level 2

o



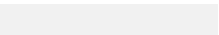
## Images in tables

Image 1





# Table of Contents

<b>1. Structural Elements</b>	<b>2</b>
• 1.1. Document Section	4
<b>2. Structural Elements 2</b>	<b>5</b>
• 2.1. Document Section	5
<b>3. Paragraph Level Markup</b>	<b>6</b>
• 3.1. Inline Markup	8
• 3.2. Math	9
• 3.3. Meta	10
• 3.4. Blocks	10
• 3.5. Sidebar	12
• 3.6. References	14
• 3.7. Directives	16
• 3.8. Download Links	21
<b>4. Lists &amp; Tables</b>	<b>21</b>
• 4.1. Lists	23
• 4.2. Tables	30
<b>5. API documentation and generated content</b>	<b>33</b>
• 5.1. 	34
• 5.2. C++ API	37
• 5.3. JavaScript API	38
• 5.4. Generated Index	39
• 5.5. Optional parameter args	39
• 5.6. Data	40
<b>6. Sphinx-Needs objects</b>	<b>40</b>
<b>7. Sphinx-Needs tables</b>	<b>43</b>
<b>8. Sphinx-Needs needflow</b>	<b>44</b>
• 8.1. CSV Table	44

---

<b>9. Breadcrumb Level 1</b>	<b>47</b>
• 9.1. <span style="background-color: #e0e0e0; display: inline-block; width: 100px; height: 1em;"></span>	48
<b>10. Long Sticky Nav</b>	<b>48</b>
• 10.1. Example Menu 1	51
• 10.2. Example Menu 2	51
• 10.3. Example Menu 3	51
• 10.4. Example Menu 4	51
• 10.5. Example Menu 5	51
• 10.6. Example Menu 6	51
• 10.7. Example Menu 7	51
• 10.8. Example Menu 8	52
• 10.9. Example Menu 9	52
• 10.10. Example Menu 10	52
• 10.11. Example Menu 11	52
• 10.12. Example Menu 12	52
• 10.13. Example Menu 13	52
• 10.14. Example Menu 14	52
• 10.15. Example Menu 15	52
• 10.16. Example Menu 16	53
• 10.17. Example Menu 17	53
• 10.18. Example Menu 18	53
• 10.19. Example Menu 19	53
• 10.20. Example Menu 20	53
• 10.21. Example Submenu 1	53
• 10.22. Example Submenu 2	54
<b>11. Images</b>	<b>55</b>
• 11.1. SVG	56
• 11.2. PNG	57
• 11.3. JPG	58
• 11.4. Images in lists	59
• 11.5. Images in tables	61

## Navigation

• [Sphinx-SimplePDF-DEMO documentation](#) »

• [Sphinx-SimplePDF-DEMO documentation](#)

© Copyright 2022, team useblocks. Created using [Sphinx](#) 5.11.