

---

# rinoh<sup>type</sup>

## Reference Guide

Release 0.5.4



# Contents

<b>5</b>	<b>Chapter 1: rinoh</b>
<b>7</b>	<b>Chapter 2: Sphinx Builder</b>
8	Legacy Configuration Variables
<b>11</b>	<b>Chapter 3: Style Sheets</b>
11	Included Style Sheets
11	Default Matcher
16	Element Style Classes
<b>89</b>	<b>Chapter 4: Standard Document Templates</b>
89	Article
91	Book
<b>95</b>	<b>Chapter 5: API Documentation</b>
95	Attribute ( <code>rinoh.attribute</code> )
96	Color ( <code>rinoh.color</code> )
96	Dimension ( <code>rinoh.dimension</code> )
97	Document ( <code>rinoh.document</code> )
98	Drawing Primitives ( <code>rinoh.draw</code> )
99	Flowable ( <code>rinoh.flowable</code> )
101	Fonts and Typefaces ( <code>rinoh.font</code> )
103	Images and Figures ( <code>rinoh.image</code> )
105	Index ( <code>rinoh.index</code> )
105	Language ( <code>rinoh.language</code> )
110	Number ( <code>rinoh.number</code> )
111	Paper sizes ( <code>rinoh.paper</code> )
112	Paragraph ( <code>rinoh.paragraph</code> )
115	Cross-References and Fields ( <code>rinoh.reference</code> )
117	Structure ( <code>rinoh.structure</code> )
120	Strings ( <code>rinoh.strings</code> )
120	Style ( <code>rinoh.style</code> )
122	Tables ( <code>rinoh.table</code> )

**123**    Templates (`rinoh.template`)

**133**    **Python Module Index**

**135**    **Index**

Render a structured document to PDF.

```
usage: rinoh [-h] [-f FORMAT] [-o OPTION=VALUE] [-t NAME or FILENAME]
             [-s NAME or FILENAME] [-O FILENAME or DIRECTORY] [-p PAPER] [-i]
             [--list-templates] [--list-stylesheets] [--list-fonts]
             [FILENAME]
             [--list-formats] [--list-options FRONTEND] [--version] [--docs]
             [input]
```

**input**

the document to render

**-h, --help**

show this help message and exit

**-f <format>, --format <format>**

the format of the input file (default: autodetect)

**-o <option=value>, --option <option=value>**

options to be passed to the input file reader

**-t <name or filename>, --template <name or filename>**

the document template or template configuration file to use (default: article)

**-s <name or filename>, --stylesheet <name or filename>**

the style sheet used to style the document elements (default: the template's default)

**-O <filename or directory>, --output <filename or directory>**

write the PDF output to FILENAME or to an existing DIRECTORY with a filename derived from the input filename (default: the current working directory)

**-p <paper>, --paper <paper>**

the paper size to render to (default: the template's default)

**-i, --install-resources**

automatically install missing resources (fonts, templates, style sheets) from PyPI

**--list-templates**

list the installed document templates and exit

**--list-stylesheets**

list the installed style sheets and exit

**--list-fonts** <filename>  
list the installed fonts or, if FILENAME is given, write a PDF file displaying all the fonts

**--list-formats**  
list the supported input formats and exit

**--list-options** <frontend>  
list the options supported by the given frontend and exit

**--version**  
show program's version number and exit

**--docs**  
open the online documentation in the default browser

`rinoh.frontent.sphinx` is a Sphinx extension module. It provides a Sphinx builder with the name *rinoh*. The builder recognizes the following `conf.py` options. Of these, only `rinoh_documents` (or `latex_documents`) is required.

## **rinoh\_documents**

Determines which PDF documents to build. Its format is a list of dictionaries, one for each document to be generated. The supported keys for these dictionaries are listed below. Two of them are required:

### **doc (required)**

String that specifies the name of the master document (typically `index`).

### **target (required)**

The name of the target PDF file without the extension.

The following keys allow overriding the content included on the document's title page and page headers/footers (depending on the template and its configuration). The values for text content can be plain text or `StyledText`.

### **logo (no default)**

Path (absolute or relative to the location of the `conf.py` file) to an image file, typically included on the title page.

### **title (default: "project documentation")**

The title of the document.

### **subtitle (default: "Release release")**

Subtitle of the document.

### **author (default: "author")**

The document's author.

### **date (default: determined from today and today\_fmt)**

The document (build) date.

The remaining keys control what is included in the document and the document template (configuration) to use.

### **toctree\_only (default: False)**

Must be `True` or `False`. If `true`, the document itself is not included in the output, only the documents referenced by it via TOC trees.

### **domain\_indices (default: True)**

If `true`, generate domain-specific indices in addition to the general index. It is equivalent to the `html_domain_indices` and `latex_domain_indices` configuration variables.

### **template (default: Book)**

Determines the template used to render the document. It takes:

- the filename of a template configuration file,
- a `TemplateConfiguration` instance,

- the name of an installed template (see `rinoh --list-templates`), or
- a `DocumentTemplate` subclass

`rinoh_documents` will also accept other keys than those listed above. These items will be available as document metadata and can be used in a custom document template.

Example:

```
rinoh_documents = [
    dict(doc='index', target='manual', toctree_only=False,
         template='manual.rtt', logo='logo.pdf'),
    dict(doc='ref', target='reference', title='Reference Manual',
         template='reference.rtt', stamp='DRAFT'),
]
```

### `rinoh_targets`

Limits rendering to the specified set of targets. This variable accepts either a list of strings or a string containing comma-separated values. The target names should match those set in `rinoh_documents`.

To render only the ‘manual’ target, you can set the following in `conf.py`:

```
rinoh_targets = ['manual']
```

The same can be achieved from the command line using the by using the string form with `sphinx-build -D`:

```
sphinx-build -b rinoh -D rinoh_targets="manual" source build
```

## 2.1 Legacy Configuration Variables

The configuration variables below are no longer supported. Instead, their functionality can now be configured per document in `rinoh_documents`.

### `rinoh_template`

This configuration variable is **no longer supported** since the document template can be specified in `rinoh_documents` entries.

### `rinoh_stylesheet`

This configuration variable is **no longer supported** since it is not obvious which style sheet will be used when the template configuration also specifies a style sheet. Please specify the style sheet to use in your template configuration file:

```
[TEMPLATE_CONFIGURATION]
name = My Book
template = book
stylesheet = my_stylesheet.rts
```

### `rinoh_paper_size`

This configuration variable is **no longer supported** since it was not obvious which paper size was being used when the template configuration (`rinoh_template`) also specified a paper size. Please specify the `paper_size` to use in your template configuration file:

```
[TEMPLATE_CONFIGURATION]
name = My Book
template = book

[VARIABLES]
```



```
paper_size = A5
```

**rinoh\_logo**

This configuration variable is **no longer supported** since the logo can be specified in the `rinoh_documents` entries.

**rinoh\_domain\_indices**

This configuration variable is **no longer supported** since the domain\_indices can be specified in the `rinoh_documents` entries.

**rinoh\_metadata**

This configuration variable is **no longer supported**. Metadata entries can now be added to `rinoh_documents` entries.



## 3.1 Included Style Sheets

These style sheets are included with rinohtype:

`rinohtype.stylesheets.sphinx`

Mostly a copy of the LaTeX style included with Sphinx

Entry point name: `sphinx`

`rinohtype.stylesheets.sphinx_article`

The Sphinx stylesheet adjusted for the article template

Entry point name: `sphinx_article`

`rinohtype.stylesheets.sphinx_base14`

The Sphinx stylesheet, but using the fonts from the PDF core set (yielding smaller PDF files)

Entry point name: `sphinx_base14`

Additional style sheets can be installed from PyPI. The installed style sheets can be listed using `rinohtype --list-stylesheets`.

## 3.2 Default Matcher

The default matcher provides the selectors for the styles used in the standard style sheets.

`rinohtype.stylesheets.matcher`

The default matcher defines the following styles

- title page rule: `HorizontalRuleStyle`
- title page logo: `FlowableStyle`
- title page title: `ParagraphStyle`
- title page subtitle: `ParagraphStyle`
- title page author: `ParagraphStyle`
- title page date: `ParagraphStyle`
- title page extra: `ParagraphStyle`
- front matter section title: `ParagraphStyle`
- body matter chapter label: `ParagraphStyle`
- body matter chapter number: `TextStyle`
- body matter chapter title: `ParagraphStyle`
- fallback: `TextStyle`
- italic: `TextStyle`

- **bold:** TextStyle
- **monospaced:** TextStyle
- **emphasis:** TextStyle
- **strong:** TextStyle
- **literal emphasis:** TextStyle
- **literal strong:** TextStyle
- **inline math:** TextStyle
- **quote:** TextStyle
- **file path:** TextStyle
- **keystrokes:** TextStyle
- **regular expression:** TextStyle
- **code with variable:** TextStyle
- **mail header:** TextStyle
- **MIME type:** TextStyle
- **newsgroup:** TextStyle
- **command:** TextStyle
- **make variable:** TextStyle
- **program:** TextStyle
- **man page:** TextStyle
- **window title:** TextStyle
- **UI control:** TextStyle
- **UI control accelerator:** TextStyle
- **menu cascade:** TextStyle
- **draft comment:** TextStyle
- **title reference:** TextStyle
- **error:** TextStyle
- **linked reference:** ReferenceStyle
- **external hyperlink:** TextStyle
- **broken hyperlink:** TextStyle
- **glossary inline definition:** TextStyle
- **body:** ParagraphStyle
- **code block:** ParagraphStyle
- **code block caption:** CaptionStyle
- **math block:** ParagraphStyle
- **attribution:** ParagraphStyle
- **centered:** ParagraphStyle
- **line block:** ParagraphStyle
- **block quote:** GroupedFlowablesStyle
- **chapter:** SectionStyle
- **content chapter:** SectionStyle
- **heading level 1:** HeadingStyle
- **unnumbered heading level 1:** HeadingStyle
- **heading level 2:** HeadingStyle
- **unnumbered heading level 2:** HeadingStyle
- **heading level 3:** HeadingStyle
- **unnumbered heading level 3:** HeadingStyle
- **heading level 4:** HeadingStyle
- **unnumbered heading level 4:** HeadingStyle
- **heading level 5:** HeadingStyle
- **unnumbered heading level 5:** HeadingStyle
- **other heading levels:** HeadingStyle
- **appendix:** SectionStyle
- **appendix heading level 1:** HeadingStyle

- appendix heading level 2: HeadingStyle
- appendix heading level 3: HeadingStyle
- appendix heading level 4: HeadingStyle
- appendix heading level 5: HeadingStyle
- title: ParagraphStyle
- prerequisites: GroupedFlowablesStyle
- prerequisites title: ParagraphStyle
- post requirement: GroupedFlowablesStyle
- abstract: GroupedFlowablesStyle
- abstract paragraph: ParagraphStyle
- example: GroupedFlowablesStyle
- example title: ParagraphStyle
- topic: GroupedFlowablesStyle
- topic title: ParagraphStyle
- rubric: ParagraphStyle
- sidebar: GroupedFlowablesStyle
- sidebar title: ParagraphStyle
- sidebar subtitle: ParagraphStyle
- list item label: ListItemLabelStyle
- list item body: GroupedFlowablesStyle
- list item paragraph: ParagraphStyle
- enumerated list: ListStyle
- enumerated list item: LabeledFlowableStyle
- enumerated list item label: ListItemLabelStyle
- nested enumerated list: ListStyle
- (table) enumerated list: ListStyle
- (table) enumerated list item: LabeledFlowableStyle
- (table) enumerated list item label: ListItemLabelStyle
- bulleted list: ListStyle
- compact bulleted list: ListStyle
- bulleted list item: LabeledFlowableStyle
- bulleted list item label: ListItemLabelStyle
- nested bulleted list: ListStyle
- (table) bulleted list: ListStyle
- (table) bulleted list item: LabeledFlowableStyle
- (table) bulleted list item label: ListItemLabelStyle
- steps list: ListStyle
- steps list title: ParagraphStyle
- steps list item: LabeledFlowableStyle
- steps list item label: ListItemLabelStyle
- unordered steps list: ListStyle
- unordered steps list title: ParagraphStyle
- unordered steps list item: LabeledFlowableStyle
- unordered steps list item label: ListItemLabelStyle
- choices list: ListStyle
- choices list item: LabeledFlowableStyle
- choices list item label: ListItemLabelStyle
- definition list: GroupedFlowablesStyle
- definition list item: LabeledFlowableStyle
- definition term: GroupedFlowablesStyle
- definition term paragraph: ParagraphStyle
- definition term classifier: TextStyle
- definition: GroupedFlowablesStyle

- definition paragraph: ParagraphStyle
- related links: GroupedFlowablesStyle
- related links section title: ParagraphStyle
- related links list: ListStyle
- related links list item: LabeledFlowableStyle
- related links list item label: ListItemLabelStyle
- related links list item paragraph: ReferencingParagraphStyle
- related link title reference: ReferenceStyle
- related link page reference: ReferenceStyle
- related link number reference: ReferenceStyle
- related link reference: ReferenceStyle
- versionmodified: TextStyle
- object description: LabeledFlowableStyle
- object signatures: GroupedFlowablesStyle
- object signature: ParagraphStyle
- object name: TextStyle
- additional name part: TextStyle
- object type: TextStyle
- object returns: TextStyle
- object parentheses: TextStyle
- object parameter list: TextStyle
- object parameter: TextStyle
- object parameter (no emphasis): TextStyle
- object brackets: TextStyle
- object optional parameter: TextStyle
- object annotation: TextStyle
- object description content: GroupedFlowablesStyle
- object description content paragraph: ParagraphStyle
- production list: GroupedFlowablesStyle
- production: LabeledFlowableStyle
- token name: ParagraphStyle
- token definition: ParagraphStyle
- field list: GroupedFlowablesStyle
- field list item: LabeledFlowableStyle
- field name: ParagraphStyle
- option list: GroupedFlowablesStyle
- option list item: LabeledFlowableStyle
- option: ParagraphStyle
- option string: TextStyle
- option argument: TextStyle
- admonition: AdmonitionStyle
- admonition title: ParagraphStyle
- admonition inline title: TextStyle
- attention admonition: AdmonitionStyle
- attention admonition title: ParagraphStyle
- attention admonition inline title: TextStyle
- caution admonition: AdmonitionStyle
- caution admonition title: ParagraphStyle
- caution admonition inline title: TextStyle
- danger admonition: AdmonitionStyle
- danger admonition title: ParagraphStyle
- danger admonition inline title: TextStyle
- error admonition: AdmonitionStyle

- error admonition title: ParagraphStyle
- error admonition inline title: TextStyle
- hint admonition: AdmonitionStyle
- hint admonition title: ParagraphStyle
- hint admonition inline title: TextStyle
- important admonition: AdmonitionStyle
- important admonition title: ParagraphStyle
- important admonition inline title: TextStyle
- note admonition: AdmonitionStyle
- note admonition title: ParagraphStyle
- note admonition inline title: TextStyle
- tip admonition: AdmonitionStyle
- tip admonition title: ParagraphStyle
- tip admonition inline title: TextStyle
- warning admonition: AdmonitionStyle
- warning admonition title: ParagraphStyle
- warning admonition inline title: TextStyle
- seealso admonition: AdmonitionStyle
- seealso admonition title: ParagraphStyle
- seealso admonition inline title: TextStyle
- header: ParagraphStyle
- footer: ParagraphStyle
- footnote: NoteStyle
- footnote marker: NoteMarkerStyle
- footnote paragraph: ParagraphStyle
- footnote label: ParagraphStyle
- citation: NoteStyle
- citation marker: NoteMarkerStyle
- citation paragraph: ParagraphStyle
- citation label: ParagraphStyle
- image: FlowableStyle
- inline image: InlineFlowableStyle
- figure: FigureStyle
- figure image: FlowableStyle
- figure caption: CaptionStyle
- figure legend: GroupedFlowablesStyle
- figure legend paragraph: ParagraphStyle
- table of contents section: SectionStyle
- table of contents title: HeadingStyle
- table of contents: TableOfContentsStyle
- local table of contents: TableOfContentsStyle
- toc level 1: TableOfContentsEntryStyle
- toc level 2: TableOfContentsEntryStyle
- toc level 3: TableOfContentsEntryStyle
- L3 toc level 3: TableOfContentsEntryStyle
- toc entry number reference field: ReferenceStyle
- list of figures section: SectionStyle
- list of figures: ListOfStyle
- list of figures entry: ListOfEntryStyle
- list of tables section: SectionStyle
- list of tables: ListOfStyle
- list of tables entry: ListOfEntryStyle
- table: TableStyle

- table with caption: `TableWithCaptionStyle`
- table caption: `CaptionStyle`
- choices table: `TableStyle`
- table cell: `TableCellStyle`
- table body cell background on even row: `TableCellBackgroundStyle`
- table body cell background on odd row: `TableCellBackgroundStyle`
- table body cell paragraph: `ParagraphStyle`
- table first column paragraph: `ParagraphStyle`
- table body cell list item number: `ParagraphStyle`
- table head cell: `TableCellStyle`
- table head cell background: `TableCellBackgroundStyle`
- table head cell paragraph: `ParagraphStyle`
- table cell left border: `TableCellBorderStyle`
- table cell top border: `TableCellBorderStyle`
- table cell right border: `TableCellBorderStyle`
- table cell bottom border: `TableCellBorderStyle`
- table top border: `TableCellBorderStyle`
- table bottom border: `TableCellBorderStyle`
- table left border: `TableCellBorderStyle`
- table right border: `TableCellBorderStyle`
- table head cell left border: `TableCellBorderStyle`
- table head cell right border: `TableCellBorderStyle`
- table head bottom border: `TableCellBorderStyle`
- table head left border: `TableCellBorderStyle`
- table head right border: `TableCellBorderStyle`
- table body top border: `TableCellBorderStyle`
- table body left border: `TableCellBorderStyle`
- table body right border: `TableCellBorderStyle`
- horizontal rule: `HorizontalRuleStyle`
- index: `IndexStyle`
- index section label: `ParagraphStyle`
- level 1 index entry: `ParagraphStyle`
- level 2 index entry: `ParagraphStyle`
- level 3 index entry: `ParagraphStyle`
- level 4 index entry: `ParagraphStyle`
- domain index entry name: `TextStyle`

### 3.3 Element Style Classes

These are the style classes corresponding to each type of document element. For each style class, the supported style attributes are listed along with the values that can be assigned to them in a style sheet.

<code>draw.LineStyle</code>	Style class for Line
<code>draw.ShapeStyle</code>	Style class for Shape
<code>flowable.FlowableStyle</code>	Style class for Flowable
<code>flowable.LabeledFlowableStyle</code>	Style class for LabeledFlowable
<code>flowable.GroupedFlowablesStyle</code>	Style class for GroupedFlowables
<code>flowable.FloatStyle</code>	Style class for Float
<code>image.CaptionStyle</code>	Style class for Caption
<code>image.FigureStyle</code>	Style class for Figure



<code>index.IndexStyle</code>	Style class for Index
<code>paragraph.ParagraphStyle</code>	Style class for Paragraph
<code>text.TextStyle</code>	Style class for Text
<code>inline.InlineFlowableStyle</code>	Style class for InlineFlowable
<code>reference.ReferencingParagraphStyle</code>	Style class for ReferencingParagraph
<code>reference.NoteMarkerStyle</code>	Style class for NoteMarker
<code>structure.SectionStyle</code>	Style class for Section
<code>structure.HeadingStyle</code>	Style class for Heading
<code>structure.ListStyle</code>	Style class for List
<code>structure.ListItemLabelStyle</code>	Style class for ListItemLabel
<code>structure.TableOfContentsStyle</code>	Style class for TableOfContents
<code>structure.TableOfContentsEntryStyle</code>	Style class for TableOfContentsEntry
<code>structure.AdmonitionStyle</code>	Style class for Admonition
<code>structure.HorizontalRuleStyle</code>	Style class for HorizontalRule
<code>table.TableStyle</code>	Style class for Table
<code>table.TableCellStyle</code>	Style class for TableCell
<code>table.TableCellBorderStyle</code>	Style class for TableCellBorder
<code>table.TableCellBackgroundStyle</code>	Style class for TableCellBackground

### 3.3.1 LineStyle

**class** `rinoh.draw.LineStyle` ( *base=None, \*\*attributes* )

Style class for Line

#### **stroke**

Width and color used to draw the line

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma ( , )

*Default:* 1pt, #000

#### **hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool:* true or false

*Default:* false

### 3.3.2 ShapeStyle

**class** `rinoh.draw.ShapeStyle` ( *base=None, \*\*attributes* )

Style class for Shape

#### **fill\_color**

Color to fill the shape

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* #e5e5e5

#### **stroke**

Width and color used to draw the line (inherited from `LineStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by

a comma ( , )  
*Default:* 1pt, #000

**hide**

Suppress rendering this element (inherited from `Style`)  
*Accepts* `Bool`: true or false  
*Default:* false

### 3.3.3 FlowableStyle

```
class rinoh.flowable.FlowableStyle ( base=None, **attributes )
```

Style class for Flowable

**width**

Width to render the flowable at  
*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* auto

**horizontal\_align**

Horizontal alignment of the flowable  
*Accepts* `HorizontalAlignment`: left, right, center  
*Default:* left

**space\_above**

Vertical space preceding the flowable  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**space\_below**

Vertical space following the flowable  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_left**

Left margin  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_right**

Right margin  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding**

Padding  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_left**

Left padding  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_right**

Right padding

*Accepts* **Dimension**: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding\_top**

Top padding

*Accepts* **Dimension**: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding\_bottom**

Bottom padding

*Accepts* **Dimension**: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**keep\_with\_next**

Keep this flowable and the next on the same page

*Accepts* **Bool**: true or false  
*Default*: false

**border**

Border surrounding the flowable

*Accepts* **Stroke**: the width (**Dimension**) and color (**Color**) of the stroke, separated by a comma (, )  
*Default*: none

**border\_left**

Border left of the flowable

*Accepts* **Stroke**: the width (**Dimension**) and color (**Color**) of the stroke, separated by a comma (, )  
*Default*: none

**border\_right**

Border right of the flowable

*Accepts* **Stroke**: the width (**Dimension**) and color (**Color**) of the stroke, separated by a comma (, )  
*Default*: none

**border\_top**

Border above the flowable

*Accepts* **Stroke**: the width (**Dimension**) and color (**Color**) of the stroke, separated by a comma (, )  
*Default*: none

**border\_bottom**

Border below the flowable

*Accepts* **Stroke**: the width (**Dimension**) and color (**Color**) of the stroke, separated by a comma (, )  
*Default*: none

**background\_color**

Color of the area within the flowable's borders

*Accepts* **Color**: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)  
*Default*: none

**page\_break**

Type of page break to insert before rendering this flowable

*Accepts* **Break**: none, any, left, right, any restart, left restart, right

restart  
*Default:* None

**hide**  
Suppress rendering this element (inherited from `Style`)  
*Accepts* `Bool`: true or false  
*Default:* false

### 3.3.4 LabeledFlowableStyle

```
class rinoh.flowable.LabeledFlowableStyle ( base=None, **attributes )
    Style class for LabeledFlowable

    label_min_width
        Minimum label width
        Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)
        Default: 12pt

    label_max_width
        Maximum label width
        Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)
        Default: 80pt

    label_spacing
        Spacing between a label and the labeled flowable
        Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)
        Default: 3pt

    align_baselines
        Line up the baselines of the label and the labeled flowable
        Accepts Bool: true or false
        Default: true

    wrap_label
        Wrap the label at label_max_width
        Accepts Bool: true or false
        Default: false

    width
        Width to render the flowable at (inherited from FlowableStyle)
        Accepts FlowableWidth: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)
        Default: auto

    horizontal_align
        Horizontal alignment of the flowable (inherited from FlowableStyle)
        Accepts HorizontalAlignment: left, right, center
        Default: left

    space_above
        Vertical space preceding the flowable (inherited from FlowableStyle)
        Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)
        Default: 0

    space_below
        Vertical space following the flowable (inherited from FlowableStyle)
```

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool:* true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none*

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none*

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none*

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)*

*Default: none*

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break: none, any, left, right, any restart, left restart, right restart*

*Default: None*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool: true or false*

*Default: false*

### 3.3.5 GroupedFlowablesStyle

```
class rinoh.flowable.GroupedFlowablesStyle ( base=None, **attributes )
```

Style class for `GroupedFlowables`

**width**

Width to render the flowable at (inherited from `FlowableStyle`)

*Accepts FlowableWidth: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: fill (overrides `FlowableStyle` default)*

**title**

Title to precede the flowables

*Accepts StyledText: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)*

*Default: (no value)*

**flowable\_spacing**

Spacing between flowables

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default: left*

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false

*Default: false*

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool:* true or false

*Default:* false

### 3.3.6 FloatStyle

```
class rinoh.flowable.FloatStyle ( base=None, **attributes )
```

Style class for `Float`

**float**

Float the flowable to the top or bottom of the page



*Accepts* Bool: true or false  
*Default:* false

**width**

Width to render the flowable at (inherited from `FlowableStyle`)  
*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* auto

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)  
*Accepts* `HorizontalAlignment`: left, right, center  
*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* `Break`: none, any, left, right, any restart, left restart, right restart

*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default:* false

### 3.3.7 CaptionStyle

**class** rinoh.image.**CaptionStyle** ( *base=None, \*\*attributes* )

Style class for Caption

**number\_format**

How numbers are formatted (inherited from ParagraphStyle)

*Accepts* NumberFormat: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default:* number (overrides ParagraphStyle default)

**numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative) (inherited from ParagraphStyle)

*Accepts* Integer: a natural number (positive integer)

*Default:* 1 (overrides ParagraphStyle default)

**width**

Width to render the flowable at (inherited from ParagraphStyle)

*Accepts* FlowableWidth: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* fill

**indent\_first**

Indentation of the first line of text (inherited from ParagraphStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text (inherited from ParagraphStyle)

*Accepts* LineSpacing: a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: <spacing type>(<arg1>, <arg2>, ...)

*Default:* default

**text\_align**

Alignment of text to the margins (inherited from ParagraphStyle)

*Accepts* TextAlign: left, right, center, justify

*Default:* justify

**tab\_stops**

List of tab positions (inherited from ParagraphStyle)

*Accepts* TabStopList: a comma-separated list of tab stops. A tab stop is specified as <position> [align] [fill string], where position (Dimension) is required and align (TabAlign) and fill string (string enclosed in quotes) are optional.

*Default:* (no value)

**number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If None, only show this item's number label. (inherited from ParagraphStyle)

*Accepts* StyledText: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in

braces: 'text string' (style name)  
*Default:* '.'

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)  
*Accepts* `HorizontalAlignment`: left, right, center  
*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)  
*Accepts* `Bool`: true or false  
*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default:* None

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: Times

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default*: 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default*: upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default*: 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default*: #000

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: normal, small capital, oldstyle figures

*Default*: normal

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: normal, superscript, subscript

*Default*: normal

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: 2

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as 'en\_US' (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the <language ID>\_<region ID> format

*Default*: en\_US

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default*: false

### 3.3.8 FigureStyle

```
class rinoh.image.FigureStyle ( base=None, **attributes )
```

Style class for Figure

**float**

Float the flowable to the top or bottom of the page (inherited from `FloatStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**width**

Width to render the flowable at (inherited from `FlowableStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* auto

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false

*Default:* false



**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool:* true or false

*Default:* false

**title**

Title to precede the flowables (inherited from `GroupedFlowablesStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**flowable\_spacing**

Spacing between flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**3.3.9 IndexStyle**

`class rinoh.index.IndexStyle ( base=None, **attributes )`

Style class for `Index`

**initials**

Group index entries based on their first letter

*Accepts* `Bool`: true or false

*Default*: true

**width**

Width to render the flowable at (inherited from `GroupedFlowablesStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: fill

**title**

Title to precede the flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string ( ' or " ), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**flowable\_spacing**

Spacing between flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default*: left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false  
*Default*: false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )  
*Default*: none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )  
*Default*: none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )  
*Default*: none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )  
*Default*: none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default*: none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default*: none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* `Break`: none, any, left, right, any restart, left restart, right restart

*Default*: None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default*: false

### 3.3.10 ParagraphStyle

```
class rinoh.paragraph.ParagraphStyle ( base=None, **attributes )
```

Style class for `Paragraph`

**width**

Width to render the flowable at (inherited from `FlowableStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: fill (overrides `FlowableStyle` default)

**indent\_first**

Indentation of the first line of text

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text

*Accepts* `LineSpacing`: a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: `<spacing type>(<arg1>, <arg2>, ...)`

*Default*: default

**text\_align**

Alignment of text to the margins

*Accepts* `TextAlign`: left, right, center, justify

*Default*: justify

**tab\_stops**

List of tab positions

*Accepts* `TabStopList`: a comma-separated list of tab stops. A tab stop is specified as

`<position> [align] [fill string]`, where `position` (`Dimension`) is required and `align` (`TabAlign`) and `fill string` (string enclosed in quotes) are optional.

*Default:* (no value)

#### **number\_format**

How numbers are formatted (inherited from `NumberStyle`)

*Accepts* `NumberFormat`: `none`, `number`, `symbol`, `lowercase character`, `uppercase character`, `lowercase roman`, `uppercase roman`

*Default:* `None` (overrides `NumberStyle` default)

#### **number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If `None`, only show this item's number label.

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string ( ' or " ), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* ' . '

#### **numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative)

*Accepts* `Integer`: a natural number (positive integer)

*Default:* (no value)

#### **horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: `left`, `right`, `center`

*Default:* `left`

#### **space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default:* 0

#### **space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default:* 0

#### **margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default:* 0

#### **margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default:* 0

#### **padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default:* 0

#### **padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool: true or false*

*Default: false*

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)*

*Default: none*

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* Break: none, any, left, right, any restart, left restart, right restart

*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* Bool: true or false

*Default:* false

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* Bool: true or false

*Default:* false

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* Typeface: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default:* Times

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default:* 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default:* upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default:* 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default*: #000

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: normal, small capital, oldstyle figures

*Default*: normal

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: normal, superscript, subscript

*Default*: normal

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: 2

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as 'en\_US' (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the <language ID>\_<region ID> format

*Default*: en\_US

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)



**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**3.3.11 TextStyle**

```
class rinoh.text.TextStyle ( base=None, **attributes )
```

Style class for Text

**typeface**

Typeface to set the text in

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: Times

**font\_weight**

Thickness of character outlines relative to their height

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default*: 500

**font\_slant**

Slope style of the font

*Accepts* `FontSlant`: upright, oblique, italic

*Default*: upright

**font\_width**

Stretch of the characters

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default*: 5

**font\_size**

Height of characters

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 10pt

**font\_color**

Color of the font

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default*: #000

**font\_variant**

Variant of the font

*Accepts* `FontVariant`: normal, small capital, oldstyle figures

*Default*: normal

**position**

Vertical text position

*Accepts* `TextPosition`: normal, superscript, subscript  
*Default*: normal

**kerning**

Improve inter-letter spacing  
*Accepts* `Bool`: true or false  
*Default*: true

**ligatures**

Run letters together where possible  
*Accepts* `Bool`: true or false  
*Default*: true

**hyphenate**

Allow words to be broken over two lines  
*Accepts* `Bool`: true or false  
*Default*: true

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word  
*Accepts* `Integer`: a natural number (positive integer)  
*Default*: 2

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as 'en\_US'  
*Accepts* `Locale`: locale identifier in the <language ID>\_<region ID> format  
*Default*: en\_US

**before**

Item to insert before this one (inherited from `InlineStyle`)  
*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)  
*Default*: (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)  
*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)  
*Default*: (no value)

**hide**

Suppress rendering this element (inherited from `Style`)  
*Accepts* `Bool`: true or false  
*Default*: false

### 3.3.12 `InlineFlowableStyle`

```
class rinoh.inline.InlineFlowableStyle ( base=None, **attributes )
    Style class for InlineFlowable
```

**baseline**

The location of the flowable's baseline relative to the bottom edge  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0pt*

**width**

Width to render the flowable at (inherited from `FlowableStyle`)

*Accepts* `FlowableWidth`: `auto`, `fill` or a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: auto*

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: `left`, `right`, `center`

*Default: left*

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (`pt`, `in`, `pc`, `mm`, `cm`, `%`, `/2`, `/4`)

*Default: 0*

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `false`

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: `none`

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: `none`

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: `none`

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: `none`

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: `none`

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default*: `none`

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* `Break`: `none`, `any`, `left`, `right`, `any restart`, `left restart`, `right restart`

*Default*: `None`

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: `true` or `false`

*Default*: `false`

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text

string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)  
*Default:* (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)  
*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)  
*Default:* (no value)

### 3.3.13 ReferencingParagraphStyle

```
class rinoh.reference.ReferencingParagraphStyle ( base=None, **attributes )
```

Style class for `ReferencingParagraph`

**text**

The text content of this paragraph  
*Accepts* `ReferenceText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)  
*Default:* '{TITLE}'

**width**

Width to render the flowable at (inherited from `ParagraphStyle`)  
*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* fill

**indent\_first**

Indentation of the first line of text (inherited from `ParagraphStyle`)  
*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text (inherited from `ParagraphStyle`)  
*Accepts* `LineSpacing`: a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: <spacing type>(<arg1>, <arg2>, ...)  
*Default:* default

**text\_align**

Alignment of text to the margins (inherited from `ParagraphStyle`)  
*Accepts* `TextAlign`: left, right, center, justify  
*Default:* justify

**tab\_stops**

List of tab positions (inherited from `ParagraphStyle`)  
*Accepts* `TabStopList`: a comma-separated list of tab stops. A tab stop is specified as <position> [align] [fill string], where position (`Dimension`) is required and align (`TabAlign`) and fill string (string enclosed in quotes) are optional.  
*Default:* (no value)

**number\_format**

How numbers are formatted (inherited from `ParagraphStyle`)

*Accepts* `NumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default*: None

**number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If None, only show this item's number label. (inherited from `ParagraphStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: ' . '

**numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative) (inherited from `ParagraphStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: (no value)

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default*: left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool:* true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* Break: none, any, left, right, any restart, left restart, right restart

*Default:* None

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default:* false

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default:* Times

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default:* 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default:* upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default:* 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 10pt



**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default*: `#000`

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: `normal`, `small capital`, `oldstyle figures`

*Default*: `normal`

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: `normal`, `superscript`, `subscript`

*Default*: `normal`

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: `2`

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as `'en_US'` (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the `<language ID>_<region ID>` format

*Default*: `en_US`

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default*: false

**3.3.14 NoteMarkerStyle**

**class** `rinoh.reference.NoteMarkerStyle` ( *base=None, \*\*attributes* )

Style class for NoteMarker

**type**

How the reference should be displayed (inherited from `ReferenceStyle`)

*Accepts* `ReferenceType`: reference, number, title, page, custom

*Default*: number (overrides `ReferenceStyle` default)

**link**

Create a hyperlink to the reference target (inherited from `ReferenceStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**quiet**

If the given reference type does not exist for the target, resolve to an empty string instead of making a fuss about it (inherited from `ReferenceStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: Times

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default*: 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default*: upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default*: 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default*: `#000`

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: `normal`, `small capital`, `oldstyle figures`

*Default*: `normal`

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: `normal`, `superscript`, `subscript`

*Default*: `normal`

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: `2`

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as `'en_US'` (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the `<language ID>_<region ID>` format

*Default*: `en_US`

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**number\_format**

How numbers are formatted (inherited from `NumberStyle`)

*Accepts* `NumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default*: number

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default*: false

### 3.3.15 SectionStyle

```
class rinoh.structure.SectionStyle ( base=None, **attributes )
```

Style class for `Section`

**show\_in\_toc**

List this section in the table of contents

*Accepts* `Bool`: true or false

*Default*: true

**width**

Width to render the flowable at (inherited from `GroupedFlowablesStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: fill

**title**

Title to precede the flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**flowable\_spacing**

Spacing between flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default*: 0

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default*: left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false

*Default: false*

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default: none*

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default: none*

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default: none*

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default: none*

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default: none*

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default: none*

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default: None*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool:* true or false

*Default: false*

### 3.3.16 HeadingStyle

```
class rinoh.structure.HeadingStyle ( base=None, **attributes )
```

Style class for Heading

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false

*Default*: true (overrides `FlowableStyle` default)

**numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative) (inherited from `ParagraphStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: -1 (overrides `ParagraphStyle` default)

**width**

Width to render the flowable at (inherited from `ParagraphStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: fill

**indent\_first**

Indentation of the first line of text (inherited from `ParagraphStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text (inherited from `ParagraphStyle`)

*Accepts* `LineSpacing`: a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: <spacing type>(<arg1>, <arg2>, ...)

*Default*: default

**text\_align**

Alignment of text to the margins (inherited from `ParagraphStyle`)

*Accepts* `TextAlign`: left, right, center, justify

*Default*: justify

**tab\_stops**

List of tab positions (inherited from `ParagraphStyle`)

*Accepts* `TabStopList`: a comma-separated list of tab stops. A tab stop is specified as <position> [align] [fill string], where position (`Dimension`) is required and align (`TabAlign`) and fill string (string enclosed in quotes) are optional.

*Default*: (no value)

**number\_format**

How numbers are formatted (inherited from `ParagraphStyle`)

*Accepts* `NumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default*: None

**number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If None, only show this item's number label. (inherited from `ParagraphStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default: ' . '*

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none



**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default:* None

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts Bool:* true or false

*Default:* false

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: Times

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default*: 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default*: upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default*: 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default*: #000

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: normal, small capital, oldstyle figures

*Default*: normal

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: normal, superscript, subscript

*Default*: normal

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: `2`

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as `'en_US'` (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the `<language ID>_<region ID>` format

*Default*: `en_US`

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: `true` or `false`

*Default*: `false`

### 3.3.17 ListStyle

```
class rinoh.structure.ListStyle ( base=None, **attributes )
```

Style class for `List`

**ordered**

This list is ordered or unordered

*Accepts* `Bool`: `true` or `false`

*Default*: `false`

**bullet**

Bullet to use in unordered lists

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (`'` or `"`), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: `'•'`

**width**

Width to render the flowable at (inherited from `GroupedFlowablesStyle`)

*Accepts* FlowableWidth: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* fill

#### **title**

Title to precede the flowables (inherited from GroupedFlowablesStyle)

*Accepts* StyledText: a list of styled text strings, separated by spaces. A styled text string is a quoted string ( ' or " ), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

#### **flowable\_spacing**

Spacing between flowables (inherited from GroupedFlowablesStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **horizontal\_align**

Horizontal alignment of the flowable (inherited from FlowableStyle)

*Accepts* HorizontalAlignment: left, right, center

*Default:* left

#### **space\_above**

Vertical space preceding the flowable (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **space\_below**

Vertical space following the flowable (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **margin\_left**

Left margin (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **margin\_right**

Right margin (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **padding**

Padding (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **padding\_left**

Left padding (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

#### **padding\_right**

Right padding (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool:* true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default:* false

**number\_format**

How numbers are formatted (inherited from `NumberStyle`)

*Accepts* `NumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default:* number

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default:* false

### 3.3.18 ListItemLabelStyle

```
class rinoh.structure.ListItemLabelStyle ( base=None, **attributes )
```

Style class for `ListItemLabel`

**width**

Width to render the flowable at (inherited from `ParagraphStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* fill

**indent\_first**

Indentation of the first line of text (inherited from `ParagraphStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text (inherited from `ParagraphStyle`)

*Accepts* `LineSpacing`: a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: <spacing type>(<arg1>, <arg2>, ...)

*Default:* default

**text\_align**

Alignment of text to the margins (inherited from `ParagraphStyle`)

*Accepts* `TextAlign`: left, right, center, justify

*Default*: justify

**tab\_stops**

List of tab positions (inherited from `ParagraphStyle`)

*Accepts* `TabStopList`: a comma-separated list of tab stops. A tab stop is specified as `<position> [align] [fill string]`, where `position` (`Dimension`) is required and `align` (`TabAlign`) and `fill string` (string enclosed in quotes) are optional.

*Default*: (no value)

**number\_format**

How numbers are formatted (inherited from `ParagraphStyle`)

*Accepts* `NumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default*: None

**number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If None, only show this item's number label. (inherited from `ParagraphStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: ' . '

**numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative) (inherited from `ParagraphStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: (no value)

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default*: left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*  
*Default: 0*

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*  
*Default: 0*

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*  
*Default: 0*

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*  
*Default: 0*

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*  
*Default: 0*

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*  
*Default: 0*

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool: true or false*  
*Default: false*

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*  
*Default: none*

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*  
*Default: none*

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*  
*Default: none*

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*  
*Default: none*



**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default*: none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default*: none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* `Break`: none, any, left, right, any restart, left restart, right restart

*Default*: None

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: Times

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default*: 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default*: upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default*: 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default*: #000

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: normal, small capital, oldstyle figures

*Default*: normal

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: normal, superscript, subscript

*Default*: normal

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default*: true

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: 2

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as 'en\_US' (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the <language ID>\_<region ID> format

*Default*: en\_US

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default*: false

### 3.3.19 TableOfContentsStyle

```
class rinoh.structure.TableOfContentsStyle ( base=None, **attributes )
```

Style class for `TableOfContents`

**depth**

The number of section levels to include in the table of contents

*Accepts* `Integer`: a natural number (positive integer)

*Default*: 3

**width**

Width to render the flowable at (inherited from `GroupedFlowablesStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: fill

**title**

Title to precede the flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**flowable\_spacing**

Spacing between flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default*: left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool: true or false*

*Default: false*

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break:* none, any, left, right, any restart, left restart, right restart

*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool:* true or false

*Default:* false

**indent\_first**

Indentation of the first line of text (inherited from `ParagraphStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text (inherited from `ParagraphStyle`)

*Accepts LineSpacing:* a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: `<spacing type>(<arg1>, <arg2>, ...)`

*Default:* default

**text\_align**

Alignment of text to the margins (inherited from `ParagraphStyle`)

*Accepts TextAlign:* left, right, center, justify

*Default:* justify

**tab\_stops**

List of tab positions (inherited from `ParagraphStyle`)

*Accepts* `TabStopList`: a comma-separated list of tab stops. A tab stop is specified as `<position> [align] [fill string]`, where `position` (`Dimension`) is required and `align` (`TabAlign`) and `fill string` (string enclosed in quotes) are optional.

*Default*: (no value)

**number\_format**

How numbers are formatted (inherited from `ParagraphStyle`)

*Accepts* `NumberFormat`: `none`, `number`, `symbol`, `lowercase character`, `uppercase character`, `lowercase roman`, `uppercase roman`

*Default*: `None`

**number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If `None`, only show this item's number label. (inherited from `ParagraphStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: `'.'`

**numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative) (inherited from `ParagraphStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default*: (no value)

**label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: `'text string' (style name)`

*Default*: (no value)

**custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: `true` or `false`

*Default*: `false`

**typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: `Times`

**font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* FontWeight: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black  
*Default:* 500

**font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* FontSlant: upright, oblique, italic

*Default:* upright

**font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* FontWidth: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

*Default:* 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* Color: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default:* #000

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* FontVariant: normal, small capital, oldstyle figures

*Default:* normal

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* TextPosition: normal, superscript, subscript

*Default:* normal

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* Bool: true or false

*Default:* true

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* Bool: true or false

*Default:* true

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* Bool: true or false

*Default:* true

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* Integer: a natural number (positive integer)

*Default: 2*

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as 'en\_US' (inherited from `TextStyle`)

*Accepts Locale:* locale identifier in the <language ID>\_<region ID> format

*Default:* en\_US

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts StyledText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

### 3.3.20 TableOfContentsEntryStyle

```
class rinoh.structure.TableOfContentsEntryStyle ( base=None, **attributes )
```

Style class for `TableOfContentsEntry`

**text**

The text content of this paragraph (inherited from `ReferencingParagraphStyle`)

*Accepts ReferenceText:* a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* '{NUMBER}' '\t' '{TITLE}' '\t' '{PAGE}' (overrides `ReferencingParagraphStyle` default)

**width**

Width to render the flowable at (inherited from `ParagraphStyle`)

*Accepts FlowableWidth:* auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* fill

**indent\_first**

Indentation of the first line of text (inherited from `ParagraphStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0pt

**line\_spacing**

Spacing between the baselines of two successive lines of text (inherited from `ParagraphStyle`)

*Accepts LineSpacing:* a predefined line spacing type or the name of a line spacing type followed by comma-separated arguments enclosed in braces: <spacing type>(<arg1>, <arg2>, ...)

*Default:* default

**text\_align**

Alignment of text to the margins (inherited from `ParagraphStyle`)



*Accepts* TextAlign: left, right, center, justify  
*Default:* justify

**tab\_stops**

List of tab positions (inherited from ParagraphStyle)

*Accepts* TabStopList: a comma-separated list of tab stops. A tab stop is specified as <position> [align] [fill string], where position (Dimension) is required and align (TabAlign) and fill string (string enclosed in quotes) are optional.

*Default:* (no value)

**number\_format**

How numbers are formatted (inherited from ParagraphStyle)

*Accepts* NumberFormat: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman

*Default:* None

**number\_separator**

Characters inserted between the number label of this element's parent and this element's own number label. If None, only show this item's number label. (inherited from ParagraphStyle)

*Accepts* StyledText: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* ' . '

**numbering\_level**

At which section level to restart numbering (positive: absolute, negative: relative) (inherited from ParagraphStyle)

*Accepts* Integer: a natural number (positive integer)

*Default:* (no value)

**horizontal\_align**

Horizontal alignment of the flowable (inherited from FlowableStyle)

*Accepts* HorizontalAlignment: left, right, center

*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_left**

Left margin (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_right**

Right margin (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool:* true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default*: none

#### **background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts* `Color`: HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)

*Default*: none

#### **page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* `Break`: none, any, left, right, any restart, left restart, right restart

*Default*: None

#### **label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

#### **label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

#### **custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default*: false

#### **typeface**

Typeface to set the text in (inherited from `TextStyle`)

*Accepts* `Typeface`: the name of a typeface. See `rinoh --list-fonts` for a list of the available typefaces.

*Default*: Times

#### **font\_weight**

Thickness of character outlines relative to their height (inherited from `TextStyle`)

*Accepts* `FontWeight`: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

*Default*: 500

#### **font\_slant**

Slope style of the font (inherited from `TextStyle`)

*Accepts* `FontSlant`: upright, oblique, italic

*Default*: upright

#### **font\_width**

Stretch of the characters (inherited from `TextStyle`)

*Accepts* `FontWidth`: ultra-condensed, extra-condensed, condensed,

semi-condensed, normal, medium, semi-expanded, expanded,  
extra-expanded, ultra-expanded  
*Default:* 5

**font\_size**

Height of characters (inherited from `TextStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 10pt

**font\_color**

Color of the font (inherited from `TextStyle`)

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default:* #000

**font\_variant**

Variant of the font (inherited from `TextStyle`)

*Accepts* `FontVariant`: normal, small capital, oldstyle figures

*Default:* normal

**position**

Vertical text position (inherited from `TextStyle`)

*Accepts* `TextPosition`: normal, superscript, subscript

*Default:* normal

**kerning**

Improve inter-letter spacing (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default:* true

**ligatures**

Run letters together where possible (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default:* true

**hyphenate**

Allow words to be broken over two lines (inherited from `TextStyle`)

*Accepts* `Bool`: true or false

*Default:* true

**hyphen\_chars**

Minimum number of characters in a hyphenated part of a word (inherited from `TextStyle`)

*Accepts* `Integer`: a natural number (positive integer)

*Default:* 2

**hyphen\_lang**

Language to use for hyphenation. Accepts locale codes such as 'en\_US' (inherited from `TextStyle`)

*Accepts* `Locale`: locale identifier in the <language ID>\_<region ID> format

*Default:* en\_US

**before**

Item to insert before this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default: (no value)*

**after**

Item to insert after this one (inherited from `InlineStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default: (no value)*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default: false*

### 3.3.21 **AdmonitionStyle**

```
class rinoh.structure.AdmonitionStyle ( base=None, **attributes )
```

Style class for `Admonition`

**inline\_title**

Show the admonition's title inline with the body text, if possible

*Accepts* `Bool`: true or false

*Default: true*

**width**

Width to render the flowable at (inherited from `GroupedFlowablesStyle`)

*Accepts* `FlowableWidth`: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: fill*

**title**

Title to precede the flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default: (no value)*

**flowable\_spacing**

Spacing between flowables (inherited from `GroupedFlowablesStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* `HorizontalAlignment`: left, right, center

*Default: left*

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)*

*Default: 0*

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool: true or false*

*Default: false*

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none*

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none*

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by*

a comma (, )  
*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)  
*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )  
*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)  
*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )  
*Default:* none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)  
*Accepts Color:* HEX string with optional alpha component (`#RRGGBB`, `#RRGGBBAA`, `#RGB` or `#RGBA`)  
*Default:* none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)  
*Accepts Break:* none, any, left, right, any restart, left restart, right restart  
*Default:* None

**hide**

Suppress rendering this element (inherited from `Style`)  
*Accepts Bool:* true or false  
*Default:* false

### 3.3.22 HorizontalRuleStyle

```
class rinoh.structure.HorizontalRuleStyle ( base=None, **attributes )  
    Style class for HorizontalRule
```

**width**

Width to render the flowable at (inherited from `FlowableStyle`)  
*Accepts FlowableWidth:* auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* auto

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)  
*Accepts HorizontalAlignment:* left, right, center  
*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)  
*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)  
*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool:* true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)



*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: none*

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)*

*Default: none*

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break: none, any, left, right, any restart, left restart, right restart*

*Default: None*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool: true or false*

*Default: false*

**stroke**

Width and color used to draw the line (inherited from `LineStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (, )*

*Default: 1pt, #000*

### 3.3.23 TableStyle

```
class rinoh.table.TableStyle ( base=None, **attributes )
```

Style class for Table

**column\_widths**

Absolute or relative widths of each column

*Accepts ColumnWidths: a whitespace-delimited list of column widths; Dimensions (absolute width), integers (relative width) and/or 'auto' (automatic width)*

*Default: none*

**split\_minimum\_rows**

The minimum number of rows to display when the table is split across pages

*Accepts Integer: a natural number (positive integer)*

*Default: (no value)*

**repeat\_head**

Repeat the head when the table is split across pages

*Accepts* Bool: true or false

*Default:* false

**width**

Width to render the flowable at (inherited from `FlowableStyle`)

*Accepts* FlowableWidth: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* auto

**horizontal\_align**

Horizontal alignment of the flowable (inherited from `FlowableStyle`)

*Accepts* HorizontalAlignment: left, right, center

*Default:* left

**space\_above**

Vertical space preceding the flowable (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**space\_below**

Vertical space following the flowable (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_left**

Left margin (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**margin\_right**

Right margin (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts* `Stroke`: the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (, )

*Default*: none

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts* `Color`: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)

*Default*: none

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts* `Break`: none, any, left, right, any restart, left restart, right restart

*Default*: None

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default: false*

### 3.3.24 TableCellStyle

```
class rinoh.table.TableCellStyle ( base=None, **attributes )
```

Style class for TableCell

#### **vertical\_align**

Vertical alignment of the cell contents within the available space

*Accepts* VerticalAlign: top, middle, bottom

*Default: middle*

#### **width**

Width to render the flowable at (inherited from GroupedFlowablesStyle)

*Accepts* FlowableWidth: auto, fill or a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: fill*

#### **title**

Title to precede the flowables (inherited from GroupedFlowablesStyle)

*Accepts* StyledText: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default: (no value)*

#### **flowable\_spacing**

Spacing between flowables (inherited from GroupedFlowablesStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

#### **horizontal\_align**

Horizontal alignment of the flowable (inherited from FlowableStyle)

*Accepts* HorizontalAlignment: left, right, center

*Default: left*

#### **space\_above**

Vertical space preceding the flowable (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

#### **space\_below**

Vertical space following the flowable (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

#### **margin\_left**

Left margin (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

#### **margin\_right**

Right margin (inherited from FlowableStyle)

*Accepts* Dimension: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default: 0*

**padding**

Padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_left**

Left padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_right**

Right padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_top**

Top padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**padding\_bottom**

Bottom padding (inherited from `FlowableStyle`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 0

**keep\_with\_next**

Keep this flowable and the next on the same page (inherited from `FlowableStyle`)

*Accepts Bool:* true or false

*Default:* false

**border**

Border surrounding the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_left**

Border left of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_right**

Border right of the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_top**

Border above the flowable (inherited from `FlowableStyle`)

*Accepts Stroke:* the width (`Dimension`) and color (`Color`) of the stroke, separated by a comma (,)

*Default:* none

**border\_bottom**

Border below the flowable (inherited from `FlowableStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none*

**background\_color**

Color of the area within the flowable's borders (inherited from `FlowableStyle`)

*Accepts Color: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)*

*Default: none*

**page\_break**

Type of page break to insert before rendering this flowable (inherited from `FlowableStyle`)

*Accepts Break: none, any, left, right, any restart, left restart, right restart*

*Default: None*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool: true or false*

*Default: false*

### 3.3.25 TableCellBorderStyle

```
class rinoh.table.TableCellBorderStyle ( base=None, **attributes )
```

Style class for `TableCellBorder`

**stroke**

Width and color used to draw the line (inherited from `LineStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none (overrides LineStyle default)*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts Bool: true or false*

*Default: false*

### 3.3.26 TableCellBackgroundStyle

```
class rinoh.table.TableCellBackgroundStyle ( base=None, **attributes )
```

Style class for `TableCellBackground`

**fill\_color**

Color to fill the shape (inherited from `ShapeStyle`)

*Accepts Color: HEX string with optional alpha component (#RRGGBB, #RRGGBBAA, #RGB or #RGBA)*

*Default: none (overrides ShapeStyle default)*

**stroke**

Width and color used to draw the line (inherited from `LineStyle`)

*Accepts Stroke: the width (Dimension) and color (Color) of the stroke, separated by a comma (,)*

*Default: none (overrides LineStyle default)*

**hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: `true` or `false`

*Default*: `false`





Rinohtype includes a number of document templates. These are configurable and therefore should cater for most documents.

## 4.1 Article

The article template consists of a title page, an optional table of contents and the body text. By default, it uses a single numbering style for all pages and the same template for even and odd pages.

```
class rinoh.templates.article.Article ( document_tree, configuration=None,  
backend=None )
```

### **stylesheet**

The stylesheet to use for styling document elements (inherited from `DocumentTemplate`)

*Accepts* `StyleSheet`: the name of an Section 3.1 or the filename of a stylesheet file (with the `.rts` extension)

*Default*: `sphinx_article` (= `rinoh.stylesheets.sphinx_article`) (overrides `DocumentTemplate` default)

### **table\_of\_contents**

Show or hide the table of contents

*Accepts* `Bool`: `true` or `false`

*Default*: `true`

### **abstract\_location**

Where to place the abstract

*Accepts* `AbstractLocation`: `title`, `front_matter`

*Default*: `front_matter`

### **parts**

The parts making up this document (inherited from `DocumentTemplate`)

*Accepts* `PartsList`: a space-separated list of document part template names

*Default*: `title front_matter contents` (overrides `DocumentTemplate` default)

### **language**

The main language of the document (inherited from `DocumentTemplate`)

*Accepts* `Language`: the code of one of the supported languages

*Default*: `EN` (English)

### **strings**

Strings to override standard element names (inherited from `DocumentTemplate`)

*Accepts Strings:* strings need to be entered in INI sections named after the `StringCollection` subclasses

*Default:* none

**title**

base: None

Type

`TitlePartTemplate`

**front\_matter**

base: None

Overrides these defaults:

- `page_number_format = continue`

Type

`ArticleFrontMatter`

**contents**

base: None

Overrides these defaults:

- `page_number_format = continue`

Type

`ContentsPartTemplate`

**page**

base: None

Overrides these defaults:

- `page_size = $(paper_size)`

Type

`PageTemplate`

**title\_page**

base: `page`

Overrides these defaults:

- `top_margin = 8cm`

Type

`TitlePageTemplate`

**front\_matter\_page**

base: `page`

Type

`PageTemplate`

**contents\_page**

base: `page`

Type

`PageTemplate`

**Configuration**

alias of `rinoh.template.ArticleConfiguration`

### **ConfigurationFile**

alias of `rinoh.template.ArticleConfigurationFile`

### **class rinoh.templates.article.AbstractLocation**

Where to place the article's abstract

Accepts: title, front matter

## **4.2 Book**

The book template consists of a title page, the table of contents, the body text and an index. The front matter pages are numbered using lowercase roman numerals. The template uses different templates for even and odd pages.

**class** `rinoh.templates.book.Book` ( *document\_tree*, *configuration=None*, *backend=None* )

### **stylesheet**

The stylesheet to use for styling document elements (inherited from `DocumentTemplate`)

*Accepts* `StyleSheet`: the name of an Section 3.1 or the filename of a stylesheet file (with the `.rts` extension)

*Default*: `sphinx` (= `rinoh.stylesheets.sphinx`) (overrides `DocumentTemplate` default)

### **parts**

The parts making up this document (inherited from `DocumentTemplate`)

*Accepts* `PartsList`: a space-separated list of document part template names

*Default*: `title front_matter contents back_matter` (overrides `DocumentTemplate` default)

### **language**

The main language of the document (inherited from `DocumentTemplate`)

*Accepts* `Language`: the code of one of the supported languages

*Default*: `EN` (English)

### **strings**

Strings to override standard element names (inherited from `DocumentTemplate`)

*Accepts* `Strings`: strings need to be entered in INI sections named after the `StringCollection` subclasses

*Default*: `none`

### **cover**

base: `None`

Overrides these defaults:

- `drop_if_empty` = `False`
- `page_number_format` = `None`
- `end_at_page` = `left`

Type

`FixedDocumentPartTemplate`

### **title**

base: `None`

Overrides these defaults:

- `page_number_format` = `number`

- `end_at_page = left`

Type

`TitlePartTemplate`

### **front\_matter**

base: None

Overrides these defaults:

- `flowables = [TableOfContentsSection(), ListOfFiguresSection(), ListOfTablesSection()]`
- `page_number_format = lowercase roman`
- `end_at_page = left`

Type

`FixedDocumentPartTemplate`

### **contents**

base: None

Overrides these defaults:

- `page_number_format = number`
- `end_at_page = left`

Type

`ContentsPartTemplate`

### **back\_matter**

base: None

Overrides these defaults:

- `page_number_format = continue`
- `end_at_page = left`

Type

`BackMatterTemplate`

### **page**

base: None

Overrides these defaults:

- `page_size = $(paper_size)`
- `left_margin = 1in`
- `right_margin = 1in`
- `top_margin = 1in`
- `bottom_margin = 1in`

Type

`PageTemplate`

### **cover\_page**

base: page

Type

`PageTemplate`

### **title\_page**

base: page

Type

TitlePageTemplate

#### **front\_matter\_page**

base: page

Overrides these defaults:

- header\_footer\_distance = 0
- header\_text = None

Type

PageTemplate

#### **front\_matter\_right\_page**

base: front\_matter\_page

Overrides these defaults:

- footer\_text = '\t' '\t' '{PAGE\_NUMBER}'
- chapter\_header\_text = None
- chapter\_footer\_text = '\t' '\t' '{PAGE\_NUMBER}'
- chapter\_title\_height = 2.5in
- chapter\_title\_flowables = [<rinoh.paragraph.Paragraph object at 0x7fec2eaf3280>]

Type

PageTemplate

#### **front\_matter\_left\_page**

base: front\_matter\_page

Overrides these defaults:

- footer\_text = '{PAGE\_NUMBER}'

Type

PageTemplate

#### **contents\_page**

base: page

Overrides these defaults:

- header\_footer\_distance = 0

Type

PageTemplate

#### **contents\_right\_page**

base: contents\_page

Overrides these defaults:

- header\_text = '\t' '\t' '{DOCUMENT\_TITLE}' ', ' '{DOCUMENT\_SUBTITLE}'
- footer\_text = '{SECTION\_NUMBER(2)}' '. ' '{SECTION\_TITLE(2)}' '\t' '\t' '{PAGE\_NUMBER}'
- chapter\_header\_text = None
- chapter\_footer\_text = '\t' '\t' '{PAGE\_NUMBER}'
- chapter\_title\_height = 2.4in
- chapter\_title\_flowables = [<rinoh.paragraph.Paragraph object at 0x7fec2ead6820>, <rinoh.paragraph.Paragraph object at 0x7fec2eae7130>]

Type

PageTemplate

#### **contents\_left\_page**

base: contents\_page

Overrides these defaults:

- header\_text = '{DOCUMENT\_TITLE}' ', ' '{DOCUMENT\_SUBTITLE}'
- footer\_text = '{PAGE\_NUMBER}' '\t' '\t' '{SectionTitles.chapter}' ' ' '{SECTION\_NUMBER(1)}' ' '. ' '{SECTION\_TITLE(1)}'

Type

PageTemplate

#### **back\_matter\_page**

base: page

Overrides these defaults:

- columns = 2
- header\_footer\_distance = 0

Type

PageTemplate

#### **back\_matter\_right\_page**

base: back\_matter\_page

Overrides these defaults:

- header\_text = '\t' '\t' '{DOCUMENT\_TITLE}' ', ' '{DOCUMENT\_SUBTITLE}'
- footer\_text = '{SECTION\_TITLE(1)}' '\t' '\t' '{PAGE\_NUMBER}'
- chapter\_header\_text = None
- chapter\_footer\_text = '\t' '\t' '{PAGE\_NUMBER}'
- chapter\_title\_height = 2.5in
- chapter\_title\_flowables = [<rinoh.paragraph.Paragraph object at 0x7fec2eaf3280>]

Type

PageTemplate

#### **back\_matter\_left\_page**

base: back\_matter\_page

Overrides these defaults:

- header\_text = '{DOCUMENT\_TITLE}' ', ' '{DOCUMENT\_SUBTITLE}'
- footer\_text = '{PAGE\_NUMBER}' '\t' '\t' '{SECTION\_TITLE(1)}'

Type

PageTemplate

#### **Configuration**

alias of rinoh.template.BookConfiguration

#### **ConfigurationFile**

alias of rinoh.template.BookConfigurationFile

---

**Note**

The API documentation is still incomplete.

---

### 5.1 **Attribute** (`rinoh.attribute`)

**class** `rinoh.attribute.AcceptNoneAttributeType`

Accepts 'none' (besides other values)

**class** `rinoh.attribute.OptionSet`

Accepts the values listed in `values`

Accepts:

**class** `rinoh.attribute.Attribute` ( *accepted\_type, default\_value, description* )

Descriptor used to describe a style attribute

**class** `rinoh.attribute.OverrideDefault` ( *default\_value* )

Overrides the default value of an attribute defined in a superclass

**class** `rinoh.attribute.AttributesDictionary` ( *base=None, \*\*attributes* )

**class** `rinoh.attribute.RuleSet` ( *name, base=None, source=None, \*\*kwargs* )

**find\_source** ( *name* )

Find top-most ruleset where configuration *name* is defined

**class** `rinoh.attribute.RuleSetFile` ( *filename, base=None, source=None, \*\*kwargs* )

**property location**

Textual representation of this source

**property root**

Directory path for resolving paths relative to this source

**class** `rinoh.attribute.Bool`

Expresses a binary choice

**class** `rinoh.attribute.Integer`

Accepts natural numbers

**exception** `rinoh.attribute.ParseError`

## 5.2 Color (`rinoh.color`)

`class rinoh.color.Color ( red, green, blue, alpha=1 )`

Color defined by component values

Parameters

- **red** (*float*) – red component (0 .. 1)
- **blue** (*float*) – blue component (0 .. 1)
- **green** (*float*) – green component (0 .. 1)
- **alpha** (*float*) – opacity (0 .. 1)

property **rgba**

classmethod **parse\_string** ( *string*, *source* )

classmethod **doc\_format** ( )

`class rinoh.color.HexColor ( string )`

Color defined as a hexadecimal string

Parameters

**string** (*str*) – #RGBA or #RRGGBBAA (# and A are optional)

`class rinoh.color.Gray ( luminance, alpha=1 )`

Shade of gray

Parameters

- **luminance** (*float*) – brightness (0 .. 1)
- **alpha** (*float*) – opacity (0 .. 1)

## 5.3 Dimension (`rinoh.dimension`)

Classes for expressing dimensions: lengths, widths, line thickness, etc.

Each dimension is expressed in terms of a unit. Several common units are defined here as constants. To create a new dimension, multiply number with a unit:

```
height = 100*PT
width = 50*PERCENT
```

Fractional dimensions are evaluated within the context they are defined in. For example, the width of a `Flowable` is evaluated with respect to the total width available to it.

`class rinoh.dimension.Dimension ( value=0, unit=None )`

A simple dimension

Parameters

- **value** (*int or float*) – the magnitude of the dimension
- **unit** (*DimensionUnit*) – the unit this dimension is expressed in. Default: `PT`.

**grow** ( *value* )

Grow this dimension (in-place)

The `value` is interpreted as a magnitude expressed in the same unit as this dimension.



Parameters

**value** (*int or float*) – the amount to add to the magnitude of this dimension

Returns

*Dimension* – this (growable) dimension itself

```
rinoh.dimension.PT = DimensionUnit(1.0, 'pt')
```

PostScript points

```
rinoh.dimension.PICA = DimensionUnit(12.0, 'pc')
```

computer pica

```
rinoh.dimension.INCH = DimensionUnit(72.0, 'in')
```

imperial/US inch

```
rinoh.dimension.MM = DimensionUnit(2.8346456692913384, 'mm')
```

millimeter

```
rinoh.dimension.CM = DimensionUnit(28.346456692913385, 'cm')
```

centimeter

```
rinoh.dimension.PERCENT = FractionUnit(100, '%')
```

fraction of 100

```
rinoh.dimension.QUARTERS = FractionUnit(4, '1/4')
```

fraction of 4

## 5.4 Document (`rinoh.document`)

```
class rinoh.document.DocumentTree ( flowables, options=None, style=None, source=None )
```

Holds the document's contents as a tree of flowables

Parameters

- **flowables** (*list [Flowable]*) – the list of top-level flowables
- **options** (*Reader*) – frontend-specific options

```
class rinoh.document.Document ( document_tree, stylesheet, language, strings=None, backend=None )
```

Renders a document tree to pages

Parameters

- **document\_tree** (*DocumentTree*) – a tree of the document's contents
- **stylesheet** (*StyleSheet*) – style sheet used to style document elements
- **language** (*Language*) – the language to use for standard strings
- **strings** (*Strings*) – overrides localized strings provided by *language*
- **backend** – the backend used for rendering the document

```
render ( filename_root=None, file=None )
```

Render the document repeatedly until the output no longer changes due to cross-references that need some iterations to converge.

### 5.4.1 Pages

**class** `rinoh.document.Page` ( *document\_part*, *number*, *paper*, *orientation*=*'portrait'* )

A single page in a document.

A `Page` is a `Container`, so other containers can be added as children.

Parameters

- **document\_part** (*DocumentPart*) – the document part this page is part of
- **number** (*int*) – the 1-based index of this page in the document part
- **paper** (*Paper*) – determines the dimensions of this page
- **orientation** (*PageOrientation*) – the orientation of this page

**property** `page`

Returns the page itself.

**render** ( )

Render the contents of this container to its canvas.

Note that the rendered contents need to be `place()`d on the parent container's canvas before they become visible.

**place** ( )

Place this container's canvas onto the parent container's canvas.

**class** `rinoh.document.PageOrientation`

Accepts: `portrait`, `landscape`

**class** `rinoh.document.PageType`

Accepts: `left`, `right`, `any`

**class** `rinoh.document.PageNumberFormat`

How (or if) page numbers are displayed

Accepts: `none`, `number`, `symbol`, `lowercase character`, `uppercase character`, `lowercase roman`, `uppercase roman`, `continue`

## 5.5 Drawing Primitives (`rinoh.draw`)

**class** `rinoh.draw.Stroke` ( *width*, *color* )

The display properties of a line

Parameters

- **width** (*Dimension*) – the width of the line
- **color** (*Color*) – the color of the line

**class** `rinoh.draw.Line` ( *start*, *end*, *style*=*None*, *parent*=*None* )

Draws a line

Parameters

- **start** (*2-tuple*) – coordinates for the start point of the line
- **end** (*2-tuple*) – coordinates for the end point of the line

**style\_class**

alias of `rinoh.draw.LineStyle`

```
class rinoh.draw.Shape ( style=None, parent=None )
    Base class for closed shapes

    style_class
        alias of rinoh.draw.ShapeStyle

class rinoh.draw.Polygon ( points, style=None, parent=None )

class rinoh.draw.Rectangle ( bottom_left, width, height, style=None, parent=None )
```

## 5.6 Flowable (rinoh.flowable)

```
class rinoh.flowable.Flowable ( align=None, width=None, id=None, style=None,
parent=None, source=None )
    A document element that can be “flowed” into a container on the page.
    A flowable can adapt to the width of the container, or it can horizontally align itself in the
    container.

    Parameters
        • align (HorizontalAlignment) – horizontal alignment of the flowable
        • width (FlowableWidth or DimensionBase) – the width of the flowable.

class rinoh.flowable.FlowableState ( flowable, _initial=True )
    Stores a flowable’s rendering state, which can be copied.
    This enables saving the rendering state at certain points in the rendering process, so
    rendering can later be resumed at those points, if needed.
```

### 5.6.1 No-Output Flowables

These flowables do not directly place anything on the page. All except `DummyFlowable` do have side-effects however. Some of these side-effects affect the rendering of the document in an indirect way.

```
class rinoh.flowable.DummyFlowable ( id=None, parent=None )
    A flowable that does not directly place anything on the page.
    Subclasses can produce side-effects to affect the output in another way.

class rinoh.flowable.AnchorFlowable ( id=None, parent=None )
    A dummy flowable that registers a destination anchor.
    Places a destination for the flowable’s ID at the current cursor position.

class rinoh.flowable.SetMetadataFlowable ( parent=None, **metadata )
    A dummy flowable that stores metadata in the document.
    The metadata is passed as keyword arguments. It will be available to other flowables
    during the rendering stage.

class rinoh.flowable.WarnFlowable ( message, parent=None )
    A dummy flowable that emits a warning during the rendering stage.

    Parameters
        message (str) – the warning message to emit

class rinoh.flowable.PageBreak ( page_break='any' )
```

## 5.6.2 Labeled Flowables

```
class rinoh.flowable.LabeledFlowable ( label, flowable, id=None, style=None, parent=None )
```

A flowable with a label.

The flowable and the label are rendered side-by-side. If the label exceeds the *label\_max\_width* style attribute value, the flowable is rendered below the label.

Parameters

- **label** (*Flowable*) – the label for the flowable
- **flowable** (*Flowable*) – the flowable to label

**style\_class**

alias of `rinoh.flowable.LabeledFlowableStyle`

**prepare** ( *flowable\_target* )

Determine number labels and register references with the document

**render** ( *container*, *last\_descender*, *state*, *label\_column\_width=None*, *space\_below=0*, *\*\*kwargs* )

Renders the flowable's content to *container*, with the flowable's top edge lining up with the container's cursor. *descender* is the descender height of the preceding line or *None*.

```
class rinoh.flowable.LabeledFlowableState ( flowable, content_flowable_state, _initial=True )
```

## 5.6.3 Grouping Flowables

```
class rinoh.flowable.GroupedFlowables ( align=None, width=None, id=None, style=None, parent=None, source=None )
```

Groups a list of flowables and renders them one below the other.

Makes sure that a flowable for which *keep\_with\_next* is enabled is not separated from the flowable that follows it.

Subclasses should implement `flowables()`.

**style\_class**

alias of `rinoh.flowable.GroupedFlowablesStyle`

**flowables** ( *container* )

Generator yielding the `Flowables` to group

**render** ( *container*, *descender*, *state*, *first\_line\_only=False*, *\*\*kwargs* )

Renders the flowable's content to *container*, with the flowable's top edge lining up with the container's cursor. *descender* is the descender height of the preceding line or *None*.

```
class rinoh.flowable.GroupedFlowablesState ( groupedflowables, flowables, first_flowable_state=None, _initial=True, _index=0 )
```

```
class rinoh.flowable.StaticGroupedFlowables ( flowables, align=None, width=None, id=None, style=None, parent=None, source=None )
```

Groups a static list of flowables.

Parameters

**flowables** (*iterable*[*Flowable*]) – the flowables to group

**flowables** ( *container* )

Generator yielding the `Flowables` to group

**build\_document** ( *flowable\_target* )

Set document metadata and populate front and back matter

**prepare** ( *flowable\_target* )

Determine number labels and register references with the document

**class** `rinoh.flowable.GroupedLabeledFlowables` ( *align=None, width=None, id=None, style=None, parent=None, source=None* )

Groups a list of labeled flowables, lining them up.

**render** ( *container, descender, state, \*\*kwargs* )

Renders the flowable's content to *container*, with the flowable's top edge lining up with the container's cursor. *descender* is the descender height of the preceding line or *None*.

#### 5.6.4 Floating Flowables

**class** `rinoh.flowable.Float` ( *align=None, width=None, id=None, style=None, parent=None, source=None* )

A flowable that can optionally be placed elsewhere on the page.

If this flowable's *float* style attribute is set to `True`, it is not flowed in line with the surrounding flowables, but it is instead flowed into another container pointed to by the former's `Container.float_space` attribute.

This is typically used to place figures and tables at the top or bottom of a page, instead of in between paragraphs.

**flow** ( *container, last\_descender, state=None, \*\*kwargs* )

Flow this flowable into *container* and return the vertical space consumed.

The flowable's contents are preceded by a vertical space with a height as specified in its style's *space\_above* attribute. Similarly, the flowed content is followed by a vertical space with a height given by the *space\_below* style attribute.

#### 5.6.5 Styling Properties

**class** `rinoh.flowable.HorizontalAlignment`

Controls horizontal placement

Accepts: `left, right, center`

**class** `rinoh.flowable.FlowableWidth`

Controls the width of a flowable

Accepts: `auto, fill` or a numeric value followed by a unit (`pt, in, pc, mm, cm, %, /2, /4`)

### 5.7 Fonts and Typefaces (`rinoh.font`)

Classes for fonts and typefaces.

**class** `rinoh.font.Font` ( *filename, weight, slant, width* )

A collection of glyphs in a particular style

This is a base class for classes that parse different font formats. See `rinoh.font.type1` and `rinoh.font.opentype`.

#### Parameters

- **filename** (*str*) – filename of the font file to load
- **weight** (*FontWeight*) – weight of the font
- **slant** (*FontSlant*) – slant of the font
- **width** (*FontWidth*) – width of the font

#### encoding

If no encoding is set for the `Font`, glyphs are addressed by glyph ID (and thus support more than 256 glyphs).

#### `get_glyph ( char, variant )`

Return the glyph for a particular character

If the glyph of requested font variant is not present in the font, the *normal* variant is returned instead. If that is not present either, an exception is raised.

#### Parameters

- **char** (*str of length 1*) – the character for which to find the glyph
- **variant** (*FontVariant*) – the variant of the glyph to return

#### Returns

*GlyphMetrics* – the requested glyph

#### Raises

**MissingGlyphException** – when the requested glyph is not present in the font

#### `get_ligature ( glyph, successor_glyph )`

Return the ligature to replace the given glyphs

If no ligature is defined in the font for the given glyphs, return `None`.

#### Parameters

- **glyph** (*GlyphMetrics*) – the first of the glyphs to combine
- **successor\_glyph** (*GlyphMetrics*) – the second of the glyphs to combine

#### Returns

*GlyphMetrics* or `None` – the ligature to replace the given glyphs

#### `get_kerning ( a, b )`

Look up the kerning for two glyphs

#### Parameters

- **a** (*GlyphMetrics*) – the first of the glyphs
- **b** (*GlyphMetrics*) – the second of the glyphs

#### Returns

*float* – the kerning value in font units

#### `class rinoh.font.Typeface ( name, *fonts )`

A set of fonts that share common design features

The fonts collected in a typeface differ in weight, width and/or slant.

#### Parameters

**\*fonts** (*Font*) – the fonts that make up this typeface

#### `fonts ( )`

Generator yielding all fonts of this typeface

Yields

*Font* – the next font in this typeface

**get\_font** ( *weight*=400, *slant*='upright', *width*=5 )

Return the font matching or closest to the given style

If a font with the given weight, slant and width is available, return it. Otherwise, return the font that is closest in style.

Parameters

- **weight** (*FontWeight*) – weight of the font
- **slant** (*FontSlant*) – slant of the font
- **width** (*FontWidth*) – width of the font

Returns

*Font* – the requested font

**class rinoh.font.style.FontWeight**

Accepts: hairline, thin, ultra-light, extra-light, light, regular, normal, book, roman, medium, semi-bold, demi-bold, bold, extra-bold, ultra-bold, black, heavy, extra-black, ultra-black

**class rinoh.font.style.FontSlant**

Accepts: upright, oblique, italic

**class rinoh.font.style.FontWidth**

Accepts: ultra-condensed, extra-condensed, condensed, semi-condensed, normal, medium, semi-expanded, expanded, extra-expanded, ultra-expanded

**class rinoh.font.style.FontVariant**

Accepts: normal, small capital, oldstyle figures

**class rinoh.font.style.TextPosition**

Accepts: normal, superscript, subscript

## 5.8 Images and Figures (**rinoh.image**)

**class rinoh.image.Image** ( *filename\_or\_file*, *scale*=1.0, *width*=None, *height*=None, *dpi*=None, *rotate*=0, *limit\_width*=<rinoh.dimension.Fraction object>, *align*=None, *id*=None, *style*=None, *parent*=None, *source*=None )

**class rinoh.image.InlineImage** ( *filename\_or\_file*, *scale*=1.0, *width*=None, *height*=None, *dpi*=None, *rotate*=0, *baseline*=None, *id*=None, *style*=None, *parent*=None, *source*=None )

**arguments**

alias of `rinoh.image.InlineImageArgs`

**class rinoh.image.BackgroundImage** ( *filename\_or\_file*, *scale*=1.0, *width*=None, *height*=None, *dpi*=None, *rotate*=0, *limit\_width*=<rinoh.dimension.Fraction object>, *align*=None, *id*=None, *style*=None, *parent*=None, *source*=None )

Image to place in the background of a page

Takes the same arguments as `Image`.

**arguments**

alias of `rinoh.image.ImageArgs`

```
class rinoh.image.ImageArgs ( base=None, **attributes )
```

**limit\_width**

Limit the image width when none of `scale`, `width` and `height` are given and the image would be wider than the container

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 100%

**align**

How to align the image within the page

*Accepts HorizontalAlignment:* left, right, center

*Default:* left

**file\_or\_filename**

Path to the image file (inherited from ImageArgsBase)

*Accepts ImagePath:* path to an image file enclosed in quotes

*Default:* none

**scale**

Scaling factor for the image (inherited from ImageArgsBase)

*Accepts Scale:* fit, fill or a float larger than 0

*Default:* fit

**width**

The width to scale the image to (inherited from ImageArgsBase)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* none

**height**

The height to scale the image to (inherited from ImageArgsBase)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* none

**dpi**

Overrides the DPI value set in the image (inherited from ImageArgsBase)

*Accepts Integer:* a natural number (positive integer)

*Default:* (no value)

**rotate**

Angle in degrees to rotate the image (inherited from ImageArgsBase)

*Accepts Integer:* a natural number (positive integer)

*Default:* (no value)

```
class rinoh.image.Scale
```

Scaling factor for images

Can be a numerical value where a value of 1 keeps the image's original size or one of values.

*Accepts:* fit, fill or a float larger than 0

```
class rinoh.image.Caption ( content, custom_label=None, align=None, width=None, id=None, style=None, parent=None, source=None )
```

**style\_class**

alias of `rinoh.image.CaptionStyle`



```

class rinoh.image.Figure ( flowables, align=None, width=None, id=None, style=None,
parent=None, source=None )

    style_class
        alias of rinoh.image.FigureStyle

class rinoh.image.ListOfFiguresSection

    list_class
        alias of rinoh.image.ListOfFigures

class rinoh.image.ListOfFigures ( local=False, id=None, style=None, parent=None )

```

## 5.9 Index (rinoh.index)

```

class rinoh.index.IndexSection ( title=None, flowables=None, style=None )

class rinoh.index.Index ( id=None, style=None, parent=None )

    style_class
        alias of rinoh.index.IndexStyle

    flowables ( container )
        Generator yielding the Flowables to group

class rinoh.index.IndexLabel ( content, custom_label=None, align=None, width=None,
id=None, style=None, parent=None, source=None )

class rinoh.index.IndexTerm ( *levels )

class rinoh.index.InlineIndexTarget ( index_terms, *args, **kwargs )

    to_string ( flowable_target )
        Return the text content of this styled text.

class rinoh.index.IndexTarget ( index_terms, parent=None )

    flow ( container, last_descender, state=None, **kwargs )
        Flow this flowable into container and return the vertical space consumed.
        The flowable's contents are preceded by a vertical space with a height as specified in its
        style's space_above attribute. Similarly, the flowed content is followed by a vertical space
        with a height given by the space_below style attribute.

```

## 5.10 Language (rinoh.language)

```

class rinoh.language.Language ( code, name )
    Collects localized strings for a particular language

    Parameters
        • code (str) – short code identifying the language
        • name (str) – native name of the language

    languages
        Dictionary mapping codes to Languages

```

The following languages are supported:

```
rinoh.language.EN = Language('en', 'English')
```

Localized strings for English

SectionTitles

Contents

Table of Contents

List\_of\_figures

List of Figures

List\_of\_tables

List of Tables

Chapter

Chapter

Index

Index

AdmonitionTitles

Attention

Attention!

Caution

Caution!

Danger

!DANGER!

Error

Error

Hint

Hint

Important

Important

Note

Note

Tip Tip

Warning

Warning

Seealso

See also

```
rinoh.language.FR = Language('fr', 'Français')
```

Localized strings for Français

SectionTitles

Contents

Table des Matières

List\_of\_figures

Liste des Figures

List\_of\_tables  
Liste des Tableaux

Chapter  
Chapitre

Index  
Index

AdmonitionTitles

Attention  
Attention!

Caution  
Prudence!

Danger  
!DANGER!

Error  
Erreur

Hint  
Conseil

Important  
Important

Note  
Note

Tip Astuce

Warning  
Avertissement

Seealso  
Voir aussi

`rinoh.language.IT = Language('it', 'Italiano')`

Localized strings for Italiano

SectionTitles

Contents  
Contenuti

List\_of\_figures  
Elenco delle Figure

List\_of\_tables  
Elenco delle Tabelle

Chapter  
Capitolo

Index  
Indice

AdmonitionTitles

Attention  
     Attenzione!  
 Caution  
     Prudenza!  
 Danger  
     !PERICOLO!  
 Error  
     Errore  
 Hint  
     Consiglio  
 Important  
     Importante  
 Note  
     Nota  
 Tip Suggestimento  
 Warning  
     Avvertimento  
 Seealso  
     Vedi anche

`rinoh.language.NL = Language('nl', 'Nederlands')`

Localized strings for Nederlands  
 SectionTitles

Contents  
     Inhoudsopgave  
 List\_of\_figures  
     Lijst van Figuren  
 List\_of\_tables  
     Lijst van Tabellen

Chapter  
     Hoofdstuk

Index  
     Index  
 AdmonitionTitles

Attention  
     Opgelet!

Caution  
     Pas op!

Danger  
     !GEVAAR!

Error  
     Fout

Hint  
Hint  
Important  
Belangrijk  
Note  
Noot  
Tip Tip  
Warning  
Waarschuwing  
Seealso  
Zie ook

`rinoh.language.PL = Language('pl', 'Polski')`

Localized strings for Polski  
SectionTitles

Contents  
Spis Treści  
List\_of\_figures  
Spis Ilustracji  
List\_of\_tables  
Spis Tabel

Chapter  
Rozdział

Index  
Skorowidz  
AdmonitionTitles

Attention  
Uwaga!

Caution  
Ostrożnie!

Danger  
!NIEBEZPIECZEŃSTWO!

Error  
Błąd

Hint  
Wskazówka

Important  
Ważne

Note  
Notatka

Tip Porada

Warning

Ostrzeżenie

Seealso

Zobacz również

## 5.11 Number (`rinoh.number`)

**class** `rinoh.number.NumberFormat`

How (or if) numbers are displayed

Instead of a numbering format, it's possible to supply a custom label (`StyledText`) to show instead of the number.

Accepts: `none`, `number`, `symbol`, `lowercase character`, `uppercase character`, `lowercase roman`, `uppercase roman`

**class** `rinoh.number.Label ( custom_label=None )`

Mixin class that formats a label

Parameters

**custom\_label** (`StyledText`) – a frontend can supply a custom label to use instead of an automatically determined label (typically a number)

**class** `rinoh.number.LabelStyle ( base=None, **attributes )`

Style class for `Label`

**label\_prefix**

Text to prepend to the label

Accepts `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

Default: (no value)

**label\_suffix**

Text to append to the label

Accepts `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

Default: (no value)

**custom\_label**

Use a custom label if specified

Accepts `Bool`: `true` or `false`

Default: `false`

**hide**

Suppress rendering this element (inherited from `Style`)

Accepts `Bool`: `true` or `false`

Default: `false`

**class** `rinoh.number.NumberStyle ( base=None, **attributes )`

Style class for `Number`

**number\_format**

How numbers are formatted

Accepts `NumberFormat`: `none`, `number`, `symbol`, `lowercase character`,

uppercase character, lowercase roman, uppercase roman  
*Default:* number

#### **label\_prefix**

Text to prepend to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string ( ' or " ), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

#### **label\_suffix**

Text to append to the label (inherited from `LabelStyle`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string ( ' or " ), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

#### **custom\_label**

Use a custom label if specified (inherited from `LabelStyle`)

*Accepts* `Bool`: true or false

*Default:* false

#### **hide**

Suppress rendering this element (inherited from `Style`)

*Accepts* `Bool`: true or false

*Default:* false

`rinoh.number.format_number ( number, format )`

Format *number* according the given *format* (`NumberFormat`)

## **5.12 Paper sizes (rinoh.paper)**

The `Paper` class and a number of predefined paper formats.

`class rinoh.paper.Paper ( name, width, height )`

Defines a paper size.

Parameters

- **name** (*str*) – the name of this paper type
- **width** (*Dimension*) – the (portrait) width of this paper type
- **height** (*Dimension*) – the (portrait) height of this paper type

`rinoh.paper.A0 = Paper('A0', width=841*MM, height=1189*MM)`

`rinoh.paper.A1 = Paper('A1', width=594*MM, height=841*MM)`

`rinoh.paper.A2 = Paper('A2', width=420*MM, height=594*MM)`

`rinoh.paper.A3 = Paper('A3', width=297*MM, height=420*MM)`

`rinoh.paper.A4 = Paper('A4', width=210*MM, height=297*MM)`

`rinoh.paper.A5 = Paper('A5', width=148*MM, height=210*MM)`

`rinoh.paper.A6 = Paper('A6', width=105*MM, height=148*MM)`

`rinoh.paper.A7 = Paper('A7', width=74*MM, height=105*MM)`

```

rinoh.paper.A8 = Paper('A8', width=52*MM, height=74*MM)
rinoh.paper.A9 = Paper('A9', width=37*MM, height=52*MM)
rinoh.paper.A10 = Paper('A10', width=26*MM, height=37*MM)
rinoh.paper.LETTER = Paper('letter', width=8.5*INCH, height=11*INCH)
rinoh.paper.LEGAL = Paper('legal', width=8.5*INCH, height=14*INCH)
rinoh.paper.JUNIOR_LEGAL = Paper('junior legal', width=8*INCH, height=5*INCH)
rinoh.paper.LEDGER = Paper('ledger', width=17*INCH, height=11*INCH)
rinoh.paper.TABLOID = Paper('tabloid', width=11*INCH, height=17*INCH)

```

### 5.13 Paragraph (`rinoh.paragraph`)

```

class rinoh.paragraph.ParagraphBase ( custom_label=None, align=None, width=None,
id=None, style=None, parent=None, source=None )

```

Base class for paragraphs

A paragraph holds styled text and other line elements to be flowed into a container.

**style\_class**

alias of `rinoh.paragraph.ParagraphStyle`

**prepare** (*flowable\_target* )

Determine number labels and register references with the document

**render** (*container, descender, state, space\_below=0, first\_line\_only=False* )

Typeset the paragraph

The paragraph is typeset in the given container starting below the current cursor position of the container. When the end of the container is reached, the rendering state is preserved to continue setting the rest of the paragraph when this method is called with a new container.

Parameters

- **container** (*Container*) – the container to render to
- **descender** (*float or None*) – descender height of the preceding line
- **state** (*ParagraphState*) – the state where rendering will continue
- **first\_line\_only** (*bool*) – typeset only the first line

```

class rinoh.paragraph.Paragraph ( content, custom_label=None, align=None, width=None,
id=None, style=None, parent=None, source=None )

```

```

class rinoh.paragraph.ParagraphState ( paragraph, words, nested_flowable_state=None,
_first_word=None, _initial=True )

```

#### 5.13.1 Styled Text (`rinoh.text`)

```

class rinoh.text.StyledText ( id=None, style=None, parent=None, source=None )

```

Base class for text that has a `TextStyle` associated with it.

**to\_string** (*flowable\_target* )

Return the text content of this styled text.



**is\_script** ( *container* )  
Returns *True* if this styled text is super/subscript.

**script\_level** ( *container* )  
Nesting level of super/subscript.

**height** ( *container* )  
Font size after super/subscript size adjustment.

**y\_offset** ( *container* )  
Vertical baseline offset (up is positive).

**property items**  
The list of items in this StyledText.

**style\_class**  
alias of `rinoh.text.TextStyle`

**class** `rinoh.text.SingleStyledText` ( *text*, *style=None*, *parent=None*, *source=None* )  
Text with a single style applied

**class** `rinoh.text.MixedStyledText` ( *text\_or\_items*, *id=None*, *style=None*, *parent=None*, *source=None* )  
Concatenation of styled text

Parameters

- **text\_or\_items** (*str*, *StyledText* or *iterable of these*) – mixed styled text
- **style** – see *Styled*
- **parent** – see *DocumentElement*

**prepare** ( *flowable\_target* )  
Determine number labels and register references with the document

**append** ( *item* )  
Append *item* (*StyledText* or *str*) to the end of this mixed-styled text.  
The parent of *item* is set to this mixed-styled text.

**property items**  
The list of items in this StyledText.

**copy** ( *parent=None* )  
Return a shallow copy of the list.

**class** `rinoh.text.Locale`  
Selects a language

### 5.13.2 Inline Elements (`rinoh.inline`)

**class** `rinoh.inline.InlineFlowable` ( *baseline=None*, *id=None*, *style=None*, *parent=None*, *source=None* )

**style\_class**  
alias of `rinoh.inline.InlineFlowableStyle`

### 5.13.3 Styling Properties

**class** `rinoh.paragraph.TextAlign`  
Text justification  
Accepts: left, right, center, justify

#### Line Spacing

**class** `rinoh.paragraph.LineSpacing`  
Base class for line spacing types  
Line spacing is defined as the distance between the baselines of two consecutive lines of text in a paragraph.

**advance** ( *line, last\_descender, container* )  
Return the distance between the descender of the previous line and the baseline of the current line.

**class** `rinoh.paragraph.DefaultSpacing`  
The default line spacing as specified by the font.

**advance** ( *line, last\_descender, container* )  
Return the distance between the descender of the previous line and the baseline of the current line.

**class** `rinoh.paragraph.ProportionalSpacing` ( *factor* )  
Line spacing proportional to the line height

Parameters

**factor** (*float*) – amount by which the line height is multiplied to obtain the line spacing

**advance** ( *line, last\_descender, container* )  
Return the distance between the descender of the previous line and the baseline of the current line.

**class** `rinoh.paragraph.FixedSpacing` ( *pitch, minimum=ProportionalSpacing(1.0)* )  
Fixed line spacing, with optional minimum spacing

Parameters

- **pitch** (*Dimension*) – the distance between the baseline of two consecutive lines of text
- **minimum** (*LineSpacing*) – the minimum line spacing to prevents lines with large fonts (or inline elements) from overlapping; set to `None` if no minimum is required, set to `None`

**advance** ( *line, last\_descender, container* )  
Return the distance between the descender of the previous line and the baseline of the current line.

**class** `rinoh.paragraph.Leading` ( *leading* )  
Line spacing determined by the space in between two lines

Parameters

**leading** (*Dimension*) – the space between the bottom of a line and the top of the next line of text

**advance** ( *line*, *last\_descender*, *container* )

Return the distance between the descender of the previous line and the baseline of the current line.

The following standard line spacings have been predefined:

`rinoh.paragraph.DEFAULT = DefaultSpacing()`

The default line spacing as specified by the font.

`rinoh.paragraph.STANDARD = ProportionalSpacing(1.2)`

Line spacing of 1.2 times the line height.

`rinoh.paragraph.SINGLE = ProportionalSpacing(1.0)`

Line spacing equal to the line height (no leading).

`rinoh.paragraph.DOUBLE = ProportionalSpacing(2.0)`

Line spacing of double the line height.

### Tabulation

`class rinoh.paragraph.TabAlign`

Alignment of text with respect to a tab stop

Accepts: left, right, center

`class rinoh.paragraph.TabStop ( position, align='left', fill=None )`

Horizontal position for aligning text of successive lines

Parameters

- **position** (Dimension or Fraction) – tab stop position
- **align** (*TabAlign*) – the alignment of text with respect to the tab stop position
- **fill** (*str*) – string pattern to fill the empty tab space with

`get_position ( line_width )`

Return the absolute position of this tab stop.

`class rinoh.paragraph.TabStopList ( iterable=(), / )`

List of tab stop positions (with alignment and fill string)

### 5.13.4 Rendering Internals

`class rinoh.paragraph.Glyph ( glyph, width, char )`

`class rinoh.paragraph.GlyphsSpan ( span, chars_to_glyphs, glyphs_and_widths=[] )`

### 5.13.5 Miscellaneous Internals

`class rinoh.paragraph.HyphenatorStore`

## 5.14 Cross-References and Fields (`rinoh.reference`)

`class rinoh.reference.Reference ( target_id, *args, **kwargs )`

`class rinoh.reference.ReferenceField ( type=None, custom_title=None, style=None, parent=None, source=None )`

```

class rinoh.reference.ReferenceText ( id=None, style=None, parent=None, source=None )

class rinoh.reference.ReferenceType
    Accepts: reference, number, title, page, custom

class rinoh.reference.ReferenceParagraph ( target_id_or_flowable, id=None,
style=None, parent=None )

    style_class
        alias of rinoh.reference.ReferenceParagraphStyle

    target_style ( document )
        Filter selection on the style attribute of the target flowable

    target_has_class ( document )
        Filter selection on a class of the target flowable

    target_has_classes ( document )
        Filter selection on a set of classes of the target flowable

    target_is_of_type ( document )
        Filter selection on the type of the target flowable

class rinoh.reference.Note ( flowable, id=None, style=None, parent=None )

    style_class
        alias of rinoh.reference.NoteStyle

    flow ( container, last_descender, state=None, footnote=False, **kwargs )
        Flow this flowable into container and return the vertical space consumed.
        The flowable's contents are preceded by a vertical space with a height as specified in its
        style's space_above attribute. Similarly, the flowed content is followed by a vertical space
        with a height given by the space_below style attribute.

class rinoh.reference.NoteMarkerBase ( custom_label=None, **kwargs )

    style_class
        alias of rinoh.reference.NoteMarkerStyle

    prepare ( flowable_target )
        Determine number labels and register references with the document

class rinoh.reference.NoteMarkerByID ( target_id, *args, **kwargs )

class rinoh.reference.NoteMarkerWithNote ( referenceable, type='number', **kwargs )

    prepare ( flowable_target )
        Determine number labels and register references with the document

class rinoh.reference.Field ( type, id=None, style=None, parent=None, source=None )

```

## 5.15 Structure (`rinoh.structure`)

### 5.15.1 Sections

**class** `rinoh.structure.Section` ( *flowables*, *align=None*, *width=None*, *id=None*, *style=None*, *parent=None*, *source=None* )

A subdivision of a document

A section usually has a heading associated with it, which is optionally numbered.

**class** `rinoh.structure.Heading` ( *content*, *custom\_label=None*, *align=None*, *width=None*, *id=None*, *style=None*, *parent=None*, *source=None* )

The title for a section

Parameters

**title** (*StyledText*) – this heading’s text

**style\_class**

alias of `rinoh.structure.HeadingStyle`

**prepare** ( *container* )

Determine number labels and register references with the document

**flow** ( *container*, *last\_descender*, *state=None*, *\*\*kwargs* )

Flow this flowable into *container* and return the vertical space consumed.

The flowable’s contents are preceded by a vertical space with a height as specified in its style’s *space\_above* attribute. Similarly, the flowed content is followed by a vertical space with a height given by the *space\_below* style attribute.

**class** `rinoh.structure.SectionTitles` ( *\*\*strings* )

Collection of localized titles for common sections

**contents**

Title for the table of contents section

Type

`String`

**list\_of\_figures**

Title for the list of figures section

Type

`String`

**list\_of\_tables**

Title for the list of tables section

Type

`String`

**chapter**

Label for top-level sections

Type

`String`

**index**  
 Title for the index section  
 Type  
 String

### 5.15.2 Lists

```
class rinoh.structure.List ( list_items, start_index=1, id=None, style=None, parent=None )

    style_class
        alias of rinoh.structure.ListStyle

class rinoh.structure.DefinitionList ( flowables, align=None, width=None, id=None,
style=None, parent=None, source=None )
```

### 5.15.3 Table of Contents

```
class rinoh.structure.TableOfContentsSection

class rinoh.structure.TableOfContents ( local=False, id=None, style=None, parent=None )

    style_class
        alias of rinoh.structure.TableOfContentsStyle

    flowables ( container )
        Generator yielding the Flowables to group

class rinoh.structure.TableOfContentsEntry ( flowable, id=None, style=None,
parent=None )

    style_class
        alias of rinoh.structure.TableOfContentsEntryStyle
```

### 5.15.4 Admonitions

```
class rinoh.structure.Admonition ( flowables, title=None, type=None, id=None, style=None,
parent=None )

    style_class
        alias of rinoh.structure.AdmonitionStyle

    flowables ( container )
        Generator yielding the Flowables to group

class rinoh.structure.AdmonitionTitles ( **strings )
    Collection of localized titles for common admonitions

    attention
        Title for attention admonitions
        Type
        String
```

**caution**  
 Title for caution admonitions  
 Type  
     String

**danger**  
 Title for danger admonitions  
 Type  
     String

**error**  
 Title for error admonitions  
 Type  
     String

**hint**  
 Title for hint admonitions  
 Type  
     String

**important**  
 Title for important admonitions  
 Type  
     String

**note**  
 Title for note admonitions  
 Type  
     String

**tip**  
 Title for tip admonitions  
 Type  
     String

**warning**  
 Title for warning admonitions  
 Type  
     String

**seealso**  
 Title for see-also admonitions  
 Type  
     String

### 5.15.5 Horizontal Rule

**class** rinoh.structure.**HorizontalRule** ( *align=None, width=None, id=None, style=None, parent=None, source=None* )

**style\_class**

alias of `rinoh.structure.HorizontalRuleStyle`

**render** ( *container, descender, state, \*\*kwargs* )

Renders the flowable's content to *container*, with the flowable's top edge lining up with the container's cursor. *descender* is the descender height of the preceding line or *None*.

## 5.16 Strings (`rinoh.strings`)

**class** rinoh.strings.**String** ( *description* )

Descriptor used to describe a configurable string

Parameters

**description** (*str*) – a short description for this string

**class** rinoh.strings.**StringCollection** ( *\*\*strings* )

A collection of related configurable strings

Subclasses

- `AdmonitionTitles`
- `SectionTitles`
- `UserStrings`

**class** rinoh.strings.**UserStrings** ( *\*\*strings* )

Collection of user-specified strings

Unlike other string collections, these are not limited to a predefined set.

Attributes:

**class** rinoh.strings.**Strings** ( *\*string\_collections* )

Stores several `StringCollections`

**class** rinoh.strings.**StringField** ( *strings\_class, key, style=None, parent=None, source=None* )

Styled text that will be substituted with a configured string

The displayed string is either the localized string as determined by the language set for the document or the user-supplied string passed to the `TemplateConfiguration`.

## 5.17 Style (`rinoh.style`)

**class** rinoh.style.**Styled** ( *id=None, style=None, parent=None, source=None* )

A document element who's style can be configured.

Parameters

**style** (*str, Style*) – the style to associate with this element. If *style* is a string, the corresponding style is lookup up in the document's style sheet by means of selectors.



**style\_class**

The `Style` subclass that corresponds to this `Styled` subclass.

**property has\_id**

Filter selection on an ID of this `Styled`

**property has\_class**

Filter selection on a class of this `Styled`

**property has\_classes**

Filter selection on a set of classes of this `Styled`

**class** `rinoh.style.Style` ( *base=None, \*\*attributes* )

Dictionary storing style attributes.

The style attributes associated with this `Style` are specified as class attributes of type `Attribute`.

Style attributes can also be accessed as object attributes.

**hide**

Suppress rendering this element

*Accepts Bool: true or false*

*Default: false*

### 5.17.1 Style Sheet

**class** `rinoh.style.StyleSheet` ( *name, matcher=None, base=None, source=None, description=None, pygments\_style=None, \*\*user\_options* )

Dictionary storing a collection of related styles by name.

Styles stored in a `StyleSheet` can refer to their base style by name.

Parameters

- **name** (*str*) – a label for this style sheet
- **matcher** (*StyledMatcher*) – the matcher providing the selectors the styles contained in this style sheet map to. If no matcher is given and *base* is specified, the *base*'s matcher is used. If *base* is not set, the default matcher is used.
- **base** (*StyleSheet* or *str*) – the style sheet to extend
- **description** (*str*) – a short string describing this style sheet
- **pygments\_style** (*str*) – the Pygments style to use for styling code blocks

**get\_selector** ( *name* )

Find a selector mapped to a style in this or a base style sheet.

Parameters

**name** (*str*) – a style name

Returns

*Selector* – the selector mapped to the style *name*

Raises

**KeyError** – if the style *name* was not found in this or a base style sheet

**class** `rinoh.style.StyleSheetFile` ( *filename, base=None, source=None, \*\*kwargs* )

Loads styles defined in a *.rts* file (INI format).

Parameters

**filename** (*str*) – the path to the style sheet file

`StyleSheetFile` takes the same optional arguments as `StyleSheet`. These can also be specified in the `[STYLESHEET]` section of the style sheet file. If an argument is specified in both places, the one passed as an argument overrides the one specified in the style sheet file.

**class** `rinoh.style.StyledMatcher` ( *mapping\_or\_iterable=None, \*\*kwargs* )

Dictionary mapping labels to selectors.

This matcher can be initialized in the same way as a `dict` by passing a mapping, an iterable and/or keyword arguments.

**update** ( *\*\*F [, E]* ) → `None`. Update `D` from `dict/iterable E` and `F`.

If `E` is present and has a `.keys()` method, then does: for `k` in `E`: `D[k] = E[k]` If `E` is present and lacks a `.keys()` method, then does: for `k, v` in `E`: `D[k] = v` In either case, this is followed by: for `k` in `F`: `D[k] = F[k]`

## 5.18 Tables (`rinoh.table`)

**class** `rinoh.table.Table` ( *body, head=None, align=None, width=None, column\_widths=None, id=None, style=None, parent=None* )

**style\_class**

alias of `rinoh.table.TableStyle`

**prepare** ( *flowable\_target* )

Determine number labels and register references with the document

**render** ( *container, last\_descender, state, space\_below=0, \*\*kwargs* )

Renders the flowable's content to *container*, with the flowable's top edge lining up with the container's cursor. *descender* is the descender height of the preceding line or `None`.

**class** `rinoh.table.TableWithCaption` ( *flowables, align=None, width=None, id=None, style=None, parent=None, source=None* )

**style\_class**

alias of `rinoh.table.TableWithCaptionStyle`

**class** `rinoh.table.TableSection` ( *rows, style=None, parent=None* )

**prepare** ( *flowable\_target* )

Determine number labels and register references with the document

**class** `rinoh.table.TableHead` ( *rows, style=None, parent=None* )

**class** `rinoh.table.TableBody` ( *rows, style=None, parent=None* )

**class** `rinoh.table.TableRow` ( *cells, style=None, parent=None* )

**prepare** ( *flowable\_target* )

Determine number labels and register references with the document

**get\_rowspanned\_columns** ( )

Return a dictionary mapping column indices to the number of columns spanned.

```

class rinoh.table.TableCell ( flowables, rowspan=1, colspan=1, id=None, style=None,
parent=None )

    style_class
        alias of rinoh.table.TableCellStyle

class rinoh.table.TableCellBorder ( rendered_cell, cell_height, position, style=None )

    style_class
        alias of rinoh.table.TableCellBorderStyle

class rinoh.table.TableCellBackground ( bottom_left, width, height, style=None,
parent=None )

    style_class
        alias of rinoh.table.TableCellBackgroundStyle

class rinoh.table.VerticalAlign
    Accepts: top, middle, bottom

class rinoh.table.ListOfTables ( local=False, id=None, style=None, parent=None )

class rinoh.table.ListOfTablesSection

    list_class
        alias of rinoh.table.ListOfTables

```

## 5.19 Templates (rinoh.template)

### 5.19.1 Documents

Document templates are created by subclassing `DocumentTemplate`, just like the Chapter 4 shipped with `rinoh`type.

```

class rinoh.template.DocumentTemplate ( document_tree, configuration=None,
backend=None )

```

Template for documents

Parameters

- **document\_tree** (*DocumentTree*) – a tree of the document’s contents
- **configuration** (*TemplateConfiguration*) – configuration for this template
- **backend** – the backend used for rendering the document

**language**

The main language of the document

*Accepts Language:* the code of one of the supported languages

*Default:* EN (English)

**strings**

Strings to override standard element names

*Accepts Strings:* strings need to be entered in INI sections named after the `StringCollection` subclasses

*Default:* none

**stylesheet**

The stylesheet to use for styling document elements

*Accepts* `StyleSheet`: the name of an Section 3.1 or the filename of a stylesheet file (with the `.rts` extension)

*Default*: `sphinx (= rinoh.stylesheets.sphinx)`

#### **parts**

The parts making up this document

*Accepts* `PartsList`: a space-separated list of document part template names

*Default*: (empty list)

#### **Configuration**

alias of `rinoh.template.DocumentTemplateConfiguration`

#### **ConfigurationFile**

alias of `rinoh.template.DocumentTemplateConfigurationFile`

Document templates can be customized by setting values for the configuration attributes defined in a `DocumentTemplate` subclass in a `TemplateConfiguration`. An template configuration can be passed as *configuration* on template instantiation. However, it is better to make use of the `document` method, however.

**class** `rinoh.template.TemplateConfiguration` ( *name*, *base=None*, *source=None*, *template=None*, *description=None*, *\*\*options* )

Stores a configuration for a `DocumentTemplate`

Parameters

- **name** (*str*) – a label for this template configuration
- **base** (*TemplateConfiguration*) – the template configuration to extend
- **template** (*DocumentTemplateMeta* or *str*) – the document template to configure
- **description** (*str*) – a short string describing this style sheet
- **\*\*options** – configuration values for the configuration attributes defined by the `document template`

#### **template = None**

The `DocumentTemplate` subclass to configure

**document** ( *document\_tree*, *backend=None* )

Create a `DocumentTemplate` object based on the given document tree and this template configuration

Parameters

- **document\_tree** (*DocumentTree*) – tree of the document's contents
- **backend** – the backend to use when rendering the document

**class** `rinoh.template.PartsList` ( *\*parts* )

Stores the names of the document part templates making up a document

Parameters

**\*parts** (*list[str]*) – the names of the document parts

### 5.19.2 Document Parts

**class** `rinoh.template.DocumentPart` ( *template*, *document*, *flowables*, *last\_number\_format* )

Part of a document.

Parameters

- **template** (*DocumentPartTemplate*) – the template that determines the

- contents and style of this document part
- **document** (*Document*) – the document this part belongs to
- **flowables** (*list [Flowable]*) – the flowables to render in this document part

#### **configuration\_class**

alias of `rinoh.template.DocumentPartTemplate`

#### **add\_page (page)**

Append *page* (*Page*) to this *DocumentPart*.

#### **new\_page (page\_number, new\_chapter, \*\*kwargs)**

Called by `render()` with the *Chain*'s that need more  
:class:`Container`'s. This method should create a new  
:class:`Page` which contains a container associated with *chain*.

The document part templates which are listed by name in `DocumentTemplate.parts` are looked up as attributes of the `DocumentTemplate` subclass. They are instances of `DocumentPartTemplate` subclasses:

#### **class rinoh.template.DocumentPartTemplate (base=None, \*\*attributes)**

A template that produces a document part

The document part is created given a set of flowables, and page templates. The latter are looked up in the `TemplateConfiguration` where this part template was.

#### **page\_number\_prefix**

Text to place in front of the page number.

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

#### **page\_number\_format**

The format for page numbers in this document part.

*Accepts* `PageNumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman, continue

*Default*: number

#### **end\_at\_page**

The type of page to end this document part on

*Accepts* `PageType`: left, right, any

*Default*: any

#### **drop\_if\_empty**

Exclude this part from the document if it is empty (no flowables)

*Accepts* `Bool`: true or false

*Default*: true

The following document part templates are used in the standard document templates:

#### **class rinoh.template.TitlePartTemplate (base=None, \*\*attributes)**

The title page of a document.

#### **drop\_if\_empty**

Exclude this part from the document if it is empty (no flowables) (inherited from `DocumentPartTemplate`)

*Accepts* `Bool`: true or false

*Default*: false (overrides `DocumentPartTemplate` default)

**page\_number\_prefix**

Text to place in front of the page number. (inherited from `DocumentPartTemplate`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**page\_number\_format**

The format for page numbers in this document part. (inherited from `DocumentPartTemplate`)

*Accepts* `PageNumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman, continue

*Default*: number

**end\_at\_page**

The type of page to end this document part on (inherited from `DocumentPartTemplate`)

*Accepts* `PageType`: left, right, any

*Default*: any

**class** `rinoh.template.ContentsPartTemplate` (*base=None, \*\*attributes*)

The body of a document.

Renders all of the content present in the `DocumentTree` passed to the `DocumentTemplate`.

**page\_number\_prefix**

Text to place in front of the page number. (inherited from `DocumentPartTemplate`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**page\_number\_format**

The format for page numbers in this document part. (inherited from `DocumentPartTemplate`)

*Accepts* `PageNumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman, continue

*Default*: number

**end\_at\_page**

The type of page to end this document part on (inherited from `DocumentPartTemplate`)

*Accepts* `PageType`: left, right, any

*Default*: any

**drop\_if\_empty**

Exclude this part from the document if it is empty (no flowables) (inherited from `DocumentPartTemplate`)

*Accepts* `Bool`: true or false

*Default*: true

**class** `rinoh.template.FixedDocumentPartTemplate` (*base=None, \*\*attributes*)

A document part template that renders a fixed list of flowables

**flowables**

The list of flowables to include in this document part

*Accepts* `FlowablesList`: Python source code that represents a list of `Flowables`

*Default:* []

**page\_number\_prefix**

Text to place in front of the page number. (inherited from `DocumentPartTemplate`)

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* (no value)

**page\_number\_format**

The format for page numbers in this document part. (inherited from `DocumentPartTemplate`)

*Accepts* `PageNumberFormat`: none, number, symbol, lowercase character, uppercase character, lowercase roman, uppercase roman, continue

*Default:* number

**end\_at\_page**

The type of page to end this document part on (inherited from `DocumentPartTemplate`)

*Accepts* `PageType`: left, right, any

*Default:* any

**drop\_if\_empty**

Exclude this part from the document if it is empty (no flowables) (inherited from `DocumentPartTemplate`)

*Accepts* `Bool`: true or false

*Default:* true

### 5.19.3 Page Templates

The document templates make use of page templates:

`class rinoh.template.PageTemplate ( base=None, **attributes )`

**header\_footer\_distance**

Distance of the header and footer to the content area

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 14pt

**columns**

The number of columns for the body text

*Accepts* `Integer`: a natural number (positive integer)

*Default:* 1

**column\_spacing**

The spacing between columns

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 1cm

**header\_text**

The text to place in the page header

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default:* '{SECTION\_NUMBER(1)}' ' ' '{SECTION\_TITLE(1)}'

**footer\_text**

The text to place in the page footer

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: '\t' '{PAGE\_NUMBER}' '/' '{NUMBER\_OF\_PAGES}'

**chapter\_header\_text**

The text to place in the header on a page that starts a new chapter

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**chapter\_footer\_text**

The text to place in the footer on a page that starts a new chapter

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**chapter\_title\_flowables**

Generator that yields the flowables to represent the chapter title

*Accepts* `FlowablesList`: Python source code that represents a list of `Flowables`

*Default*: none

**chapter\_title\_height**

The height of the container holding the chapter title

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 150pt

**page\_size**

The format of the pages in the document (inherited from `PageTemplateBase`)

*Accepts* `Paper`: the name of a Section 5.12 or <width> \* <height> where width and height are `Dimensions`

*Default*: A4

**page\_orientation**

The orientation of pages in the document (inherited from `PageTemplateBase`)

*Accepts* `PageOrientation`: portrait, landscape

*Default*: portrait

**left\_margin**

The margin size on the left of the page (inherited from `PageTemplateBase`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 3cm

**right\_margin**

The margin size on the right of the page (inherited from `PageTemplateBase`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 3cm

**top\_margin**

The margin size at the top of the page (inherited from `PageTemplateBase`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 3cm



**bottom\_margin**

The margin size at the bottom of the page (inherited from `PageTemplateBase`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 3cm

**background**

An image to place in the background of the page (inherited from `PageTemplateBase`)

*Accepts* `BackgroundImage`: filename of an image file enclosed in quotes, optionally followed by space-delimited keyword arguments (`<keyword>=<value>`) that determine how the image is displayed

*Default*: none

**after\_break\_background**

An image to place in the background after a page break (inherited from `PageTemplateBase`)

*Accepts* `BackgroundImage`: filename of an image file enclosed in quotes, optionally followed by space-delimited keyword arguments (`<keyword>=<value>`) that determine how the image is displayed

*Default*: none

```
class rinoh.template.TitlePageTemplate ( base=None, **attributes )
```

**show\_date**

Show or hide the document's date

*Accepts* `Bool`: true or false

*Default*: true

**show\_author**

Show or hide the document's author

*Accepts* `Bool`: true or false

*Default*: true

**extra**

Extra text to include on the title page below the title

*Accepts* `StyledText`: a list of styled text strings, separated by spaces. A styled text string is a quoted string (' or "), optionally followed by a style name enclosed in braces: 'text string' (style name)

*Default*: (no value)

**page\_size**

The format of the pages in the document (inherited from `PageTemplateBase`)

*Accepts* `Paper`: the name of a Section 5.12 or `<width> * <height>` where width and height are `Dimensions`

*Default*: A4

**page\_orientation**

The orientation of pages in the document (inherited from `PageTemplateBase`)

*Accepts* `PageOrientation`: portrait, landscape

*Default*: portrait

**left\_margin**

The margin size on the left of the page (inherited from `PageTemplateBase`)

*Accepts* `Dimension`: a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default*: 3cm

**right\_margin**

The margin size on the right of the page (inherited from `PageTemplateBase`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**top\_margin**

The margin size at the top of the page (inherited from `PageTemplateBase`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**bottom\_margin**

The margin size at the bottom of the page (inherited from `PageTemplateBase`)

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**background**

An image to place in the background of the page (inherited from `PageTemplateBase`)

*Accepts BackgroundImage:* filename of an image file enclosed in quotes, optionally followed by space-delimited keyword arguments (`<keyword>=<value>`) that determine how the image is displayed

*Default:* none

**after\_break\_background**

An image to place in the background after a page break (inherited from `PageTemplateBase`)

*Accepts BackgroundImage:* filename of an image file enclosed in quotes, optionally followed by space-delimited keyword arguments (`<keyword>=<value>`) that determine how the image is displayed

*Default:* none

The base class for these collects the common options:

```
class rinoh.template.PageTemplateBase ( base=None, **attributes )
```

**page\_size**

The format of the pages in the document

*Accepts Paper:* the name of a Section 5.12 or `<width> * <height>` where width and height are Dimensions

*Default:* A4

**page\_orientation**

The orientation of pages in the document

*Accepts PageOrientation:* portrait, landscape

*Default:* portrait

**left\_margin**

The margin size on the left of the page

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**right\_margin**

The margin size on the right of the page

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**top\_margin**

The margin size at the top of the page

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**bottom\_margin**

The margin size at the bottom of the page

*Accepts Dimension:* a numeric value followed by a unit (pt, in, pc, mm, cm, %, /2, /4)

*Default:* 3cm

**background**

An image to place in the background of the page

*Accepts BackgroundImage:* filename of an image file enclosed in quotes, optionally followed by space-delimited keyword arguments (<keyword>=<value>) that determine how the image is displayed

*Default:* none

**after\_break\_background**

An image to place in the background after a page break

*Accepts BackgroundImage:* filename of an image file enclosed in quotes, optionally followed by space-delimited keyword arguments (<keyword>=<value>) that determine how the image is displayed

*Default:* none



# Python Module Index

## r

### rinoh

- rinoh.attribute, 95
- rinoh.color, 96
- rinoh.dimension, 96
- rinoh.document, 97
- rinoh.draw, 98
- rinoh.flowable, 99
- rinoh.font, 101
- rinoh.font.style, 103
- rinoh.image, 103
- rinoh.index, 105
- rinoh.inline, 113
- rinoh.language, 105
- rinoh.number, 110
- rinoh.paper, 111
- rinoh.paragraph, 112
- rinoh.reference, 115
- rinoh.strings, 120
- rinoh.structure, 117
- rinoh.style, 120
- rinoh.stylesheets, 11
- rinoh.table, 122
- rinoh.template, 123
- rinoh.templates.article, 89
- rinoh.templates.book, 91
- rinoh.text, 112



# Index

## Symbols

- docs
  - rinoh command line option, 6
- format <format>
  - rinoh command line option, 5
- help
  - rinoh command line option, 5
- install-resources
  - rinoh command line option, 5
- list-fonts <filename>
  - rinoh command line option, 6
- list-formats
  - rinoh command line option, 6
- list-options <frontend>
  - rinoh command line option, 6
- list-stylesheets
  - rinoh command line option, 5
- list-templates
  - rinoh command line option, 5
- option <option=value>
  - rinoh command line option, 5
- output <filename or directory>
  - rinoh command line option, 5
- paper <paper>
  - rinoh command line option, 5
- stylesheet <name or filename>
  - rinoh command line option, 5
- template <name or filename>
  - rinoh command line option, 5
- version
  - rinoh command line option, 6
- f <format>
  - rinoh command line option, 5
- h
  - rinoh command line option, 5
- i
  - rinoh command line option, 5
- O <filename or directory>
  - rinoh command line option, 5
- o <option=value>
  - rinoh command line option, 5
- p <paper>

rinoh command line option, 5  
-s <name or filename>  
rinoh command line option, 5  
-t <name or filename>  
rinoh command line option, 5

## A

A0 (in module rinoh.paper), 111  
A1 (in module rinoh.paper), 111  
A10 (in module rinoh.paper), 112  
A2 (in module rinoh.paper), 111  
A3 (in module rinoh.paper), 111  
A4 (in module rinoh.paper), 111  
A5 (in module rinoh.paper), 111  
A6 (in module rinoh.paper), 111  
A7 (in module rinoh.paper), 111  
A8 (in module rinoh.paper), 112  
A9 (in module rinoh.paper), 112  
abstract\_location (rinoh.templates.article.Article attribute), 89  
AbstractLocation (class in rinoh.templates.article), 91  
AcceptNoneAttributeType (class in rinoh.attribute), 95  
add\_page() (rinoh.template.DocumentPart method), 125  
Admonition (class in rinoh.structure), 118  
AdmonitionStyle (class in rinoh.structure), 77  
AdmonitionTitles (class in rinoh.structure), 118  
advance() (rinoh.paragraph.DefaultSpacing method), 114  
advance() (rinoh.paragraph.FixedSpacing method), 114  
advance() (rinoh.paragraph.Leading method), 115  
advance() (rinoh.paragraph.LineSpacing method), 114  
advance() (rinoh.paragraph.ProportionalSpacing method), 114  
after (rinoh.image.CaptionStyle attribute), 31  
after (rinoh.inline.InlineFlowableStyle attribute), 45  
after (rinoh.paragraph.ParagraphStyle attribute), 41  
after (rinoh.reference.NoteMarkerStyle attribute), 51  
after (rinoh.reference.ReferencingParagraphStyle attribute), 49  
after (rinoh.structure.HeadingStyle attribute), 59  
after (rinoh.structure.ListItemLabelStyle attribute), 67  
after (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
after (rinoh.structure.TableOfContentsStyle attribute), 72  
after (rinoh.text.TextStyle attribute), 42  
after\_break\_background (rinoh.template.PageTemplate attribute), 129  
after\_break\_background (rinoh.template.PageTemplateBase attribute), 131  
after\_break\_background (rinoh.template.TitlePageTemplate attribute), 130  
align (rinoh.image.ImageArgs attribute), 104  
align\_baselines (rinoh.flowable.LabeledFlowableStyle attribute), 20  
AnchorFlowable (class in rinoh.flowable), 99  
append() (rinoh.text.MixedStyledText method), 113  
arguments (rinoh.image.BackgroundImage attribute), 103  
arguments (rinoh.image.InlineImage attribute), 103  
Article (class in rinoh.templates.article), 89  
attention (rinoh.structure.AdmonitionTitles attribute), 118  
Attribute (class in rinoh.attribute), 95  
AttributesDictionary (class in rinoh.attribute), 95

## B

back\_matter (rinoh.templates.book.Book attribute), 92  
back\_matter\_left\_page (rinoh.templates.book.Book attribute), 94



back\_matter\_page (rinoh.templates.book.Book attribute), 94  
 back\_matter\_right\_page (rinoh.templates.book.Book attribute), 94  
 background (rinoh.template.PageTemplate attribute), 129  
 background (rinoh.template.PageTemplateBase attribute), 131  
 background (rinoh.template.TitlePageTemplate attribute), 130  
 background\_color (rinoh.flowable.FloatStyle attribute), 26  
 background\_color (rinoh.flowable.FlowableStyle attribute), 19  
 background\_color (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
 background\_color (rinoh.flowable.LabeledFlowableStyle attribute), 22  
 background\_color (rinoh.image.CaptionStyle attribute), 29  
 background\_color (rinoh.image.FigureStyle attribute), 33  
 background\_color (rinoh.index.IndexStyle attribute), 36  
 background\_color (rinoh.inline.InlineFlowableStyle attribute), 44  
 background\_color (rinoh.paragraph.ParagraphStyle attribute), 38  
 background\_color (rinoh.reference.ReferencingParagraphStyle attribute), 47  
 background\_color (rinoh.structure.AdmonitionStyle attribute), 79  
 background\_color (rinoh.structure.HeadingStyle attribute), 57  
 background\_color (rinoh.structure.HorizontalRuleStyle attribute), 81  
 background\_color (rinoh.structure.ListItemLabelStyle attribute), 65  
 background\_color (rinoh.structure.ListStyle attribute), 61  
 background\_color (rinoh.structure.SectionStyle attribute), 54  
 background\_color (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 background\_color (rinoh.structure.TableOfContentsStyle attribute), 69  
 background\_color (rinoh.table.TableCellStyle attribute), 86  
 background\_color (rinoh.table.TableStyle attribute), 83  
 BackgroundImage (class in rinoh.image), 103  
 baseline (rinoh.inline.InlineFlowableStyle attribute), 42  
 before (rinoh.image.CaptionStyle attribute), 31  
 before (rinoh.inline.InlineFlowableStyle attribute), 44  
 before (rinoh.paragraph.ParagraphStyle attribute), 40  
 before (rinoh.reference.NoteMarkerStyle attribute), 51  
 before (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 before (rinoh.structure.HeadingStyle attribute), 59  
 before (rinoh.structure.ListItemLabelStyle attribute), 67  
 before (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 before (rinoh.structure.TableOfContentsStyle attribute), 72  
 before (rinoh.text.TextStyle attribute), 42  
 Book (class in rinoh.templates.book), 91  
 Bool (class in rinoh.attribute), 95  
 border (rinoh.flowable.FloatStyle attribute), 26  
 border (rinoh.flowable.FlowableStyle attribute), 19  
 border (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
 border (rinoh.flowable.LabeledFlowableStyle attribute), 21  
 border (rinoh.image.CaptionStyle attribute), 29  
 border (rinoh.image.FigureStyle attribute), 33  
 border (rinoh.index.IndexStyle attribute), 35  
 border (rinoh.inline.InlineFlowableStyle attribute), 44  
 border (rinoh.paragraph.ParagraphStyle attribute), 38  
 border (rinoh.reference.ReferencingParagraphStyle attribute), 47  
 border (rinoh.structure.AdmonitionStyle attribute), 78  
 border (rinoh.structure.HeadingStyle attribute), 56  
 border (rinoh.structure.HorizontalRuleStyle attribute), 80  
 border (rinoh.structure.ListItemLabelStyle attribute), 64  
 border (rinoh.structure.ListStyle attribute), 61  
 border (rinoh.structure.SectionStyle attribute), 54  
 border (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
 border (rinoh.structure.TableOfContentsStyle attribute), 68  
 border (rinoh.table.TableCellStyle attribute), 85

border (rinoh.table.TableStyle attribute), 83  
border\_bottom (rinoh.flowable.FloatStyle attribute), 26  
border\_bottom (rinoh.flowable.FlowableStyle attribute), 19  
border\_bottom (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
border\_bottom (rinoh.flowable.LabeledFlowableStyle attribute), 22  
border\_bottom (rinoh.image.CaptionStyle attribute), 29  
border\_bottom (rinoh.image.FigureStyle attribute), 33  
border\_bottom (rinoh.index.IndexStyle attribute), 36  
border\_bottom (rinoh.inline.InlineFlowableStyle attribute), 44  
border\_bottom (rinoh.paragraph.ParagraphStyle attribute), 38  
border\_bottom (rinoh.reference.ReferencingParagraphStyle attribute), 47  
border\_bottom (rinoh.structure.AdmonitionStyle attribute), 79  
border\_bottom (rinoh.structure.HeadingStyle attribute), 57  
border\_bottom (rinoh.structure.HorizontalRuleStyle attribute), 81  
border\_bottom (rinoh.structure.ListItemLabelStyle attribute), 65  
border\_bottom (rinoh.structure.ListStyle attribute), 61  
border\_bottom (rinoh.structure.SectionStyle attribute), 54  
border\_bottom (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
border\_bottom (rinoh.structure.TableOfContentsStyle attribute), 69  
border\_bottom (rinoh.table.TableCellStyle attribute), 85  
border\_bottom (rinoh.table.TableStyle attribute), 83  
border\_left (rinoh.flowable.FloatStyle attribute), 26  
border\_left (rinoh.flowable.FlowableStyle attribute), 19  
border\_left (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
border\_left (rinoh.flowable.LabeledFlowableStyle attribute), 21  
border\_left (rinoh.image.CaptionStyle attribute), 29  
border\_left (rinoh.image.FigureStyle attribute), 33  
border\_left (rinoh.index.IndexStyle attribute), 35  
border\_left (rinoh.inline.InlineFlowableStyle attribute), 44  
border\_left (rinoh.paragraph.ParagraphStyle attribute), 38  
border\_left (rinoh.reference.ReferencingParagraphStyle attribute), 47  
border\_left (rinoh.structure.AdmonitionStyle attribute), 78  
border\_left (rinoh.structure.HeadingStyle attribute), 57  
border\_left (rinoh.structure.HorizontalRuleStyle attribute), 80  
border\_left (rinoh.structure.ListItemLabelStyle attribute), 64  
border\_left (rinoh.structure.ListStyle attribute), 61  
border\_left (rinoh.structure.SectionStyle attribute), 54  
border\_left (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
border\_left (rinoh.structure.TableOfContentsStyle attribute), 68  
border\_left (rinoh.table.TableCellStyle attribute), 85  
border\_left (rinoh.table.TableStyle attribute), 83  
border\_right (rinoh.flowable.FloatStyle attribute), 26  
border\_right (rinoh.flowable.FlowableStyle attribute), 19  
border\_right (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
border\_right (rinoh.flowable.LabeledFlowableStyle attribute), 21  
border\_right (rinoh.image.CaptionStyle attribute), 29  
border\_right (rinoh.image.FigureStyle attribute), 33  
border\_right (rinoh.index.IndexStyle attribute), 35  
border\_right (rinoh.inline.InlineFlowableStyle attribute), 44  
border\_right (rinoh.paragraph.ParagraphStyle attribute), 38  
border\_right (rinoh.reference.ReferencingParagraphStyle attribute), 47  
border\_right (rinoh.structure.AdmonitionStyle attribute), 78  
border\_right (rinoh.structure.HeadingStyle attribute), 57  
border\_right (rinoh.structure.HorizontalRuleStyle attribute), 80  
border\_right (rinoh.structure.ListItemLabelStyle attribute), 64  
border\_right (rinoh.structure.ListStyle attribute), 61  
border\_right (rinoh.structure.SectionStyle attribute), 54  
border\_right (rinoh.structure.TableOfContentsEntryStyle attribute), 74

- border\_right (rinoh.structure.TableOfContentsStyle attribute), 69
- border\_right (rinoh.table.TableCellStyle attribute), 85
- border\_right (rinoh.table.TableStyle attribute), 83
- border\_top (rinoh.flowable.FloatStyle attribute), 26
- border\_top (rinoh.flowable.FlowableStyle attribute), 19
- border\_top (rinoh.flowable.GroupedFlowablesStyle attribute), 24
- border\_top (rinoh.flowable.LabeledFlowableStyle attribute), 22
- border\_top (rinoh.image.CaptionStyle attribute), 29
- border\_top (rinoh.image.FigureStyle attribute), 33
- border\_top (rinoh.index.IndexStyle attribute), 35
- border\_top (rinoh.inline.InlineFlowableStyle attribute), 44
- border\_top (rinoh.paragraph.ParagraphStyle attribute), 38
- border\_top (rinoh.reference.ReferencingParagraphStyle attribute), 47
- border\_top (rinoh.structure.AdmonitionStyle attribute), 79
- border\_top (rinoh.structure.HeadingStyle attribute), 57
- border\_top (rinoh.structure.HorizontalRuleStyle attribute), 81
- border\_top (rinoh.structure.ListItemLabelStyle attribute), 64
- border\_top (rinoh.structure.ListStyle attribute), 61
- border\_top (rinoh.structure.SectionStyle attribute), 54
- border\_top (rinoh.structure.TableOfContentsEntryStyle attribute), 74
- border\_top (rinoh.structure.TableOfContentsStyle attribute), 69
- border\_top (rinoh.table.TableCellStyle attribute), 85
- border\_top (rinoh.table.TableStyle attribute), 83
- bottom\_margin (rinoh.template.PageTemplate attribute), 129
- bottom\_margin (rinoh.template.PageTemplateBase attribute), 131
- bottom\_margin (rinoh.template.TitlePageTemplate attribute), 130
- build\_document() (rinoh.flowable.StaticGroupedFlowables method), 101
- bullet (rinoh.structure.ListStyle attribute), 59

## C

- Caption (class in rinoh.image), 104
- CaptionStyle (class in rinoh.image), 27
- caution (rinoh.structure.AdmonitionTitles attribute), 119
- chapter (rinoh.structure.SectionTitles attribute), 117
- chapter\_footer\_text (rinoh.template.PageTemplate attribute), 128
- chapter\_header\_text (rinoh.template.PageTemplate attribute), 128
- chapter\_title\_flowables (rinoh.template.PageTemplate attribute), 128
- chapter\_title\_height (rinoh.template.PageTemplate attribute), 128
- CM (in module rinoh.dimension), 97
- Color (class in rinoh.color), 96
- column\_spacing (rinoh.template.PageTemplate attribute), 127
- column\_widths (rinoh.table.TableStyle attribute), 81
- columns (rinoh.template.PageTemplate attribute), 127
- Configuration (rinoh.template.DocumentTemplate attribute), 124
- Configuration (rinoh.templates.article.Article attribute), 90
- Configuration (rinoh.templates.book.Book attribute), 94
- configuration\_class (rinoh.template.DocumentPart attribute), 125
- ConfigurationFile (rinoh.template.DocumentTemplate attribute), 124
- ConfigurationFile (rinoh.templates.article.Article attribute), 91
- ConfigurationFile (rinoh.templates.book.Book attribute), 94
- contents (rinoh.structure.SectionTitles attribute), 117
- contents (rinoh.templates.article.Article attribute), 90
- contents (rinoh.templates.book.Book attribute), 92
- contents\_left\_page (rinoh.templates.book.Book attribute), 94
- contents\_page (rinoh.templates.article.Article attribute), 90
- contents\_page (rinoh.templates.book.Book attribute), 93
- contents\_right\_page (rinoh.templates.book.Book attribute), 93

- ContentsPartTemplate (class in rinoh.template), 126
- copy() (rinoh.text.MixedStyledText method), 113
- cover (rinoh.templates.book.Book attribute), 91
- cover\_page (rinoh.templates.book.Book attribute), 92
- custom\_label (rinoh.image.CaptionStyle attribute), 30
- custom\_label (rinoh.number.LabelStyle attribute), 110
- custom\_label (rinoh.number.NumberStyle attribute), 111
- custom\_label (rinoh.paragraph.ParagraphStyle attribute), 39
- custom\_label (rinoh.reference.NoteMarkerStyle attribute), 52
- custom\_label (rinoh.reference.ReferencingParagraphStyle attribute), 48
- custom\_label (rinoh.structure.HeadingStyle attribute), 57
- custom\_label (rinoh.structure.ListItemLabelStyle attribute), 65
- custom\_label (rinoh.structure.ListStyle attribute), 62
- custom\_label (rinoh.structure.TableOfContentsEntryStyle attribute), 75
- custom\_label (rinoh.structure.TableOfContentsStyle attribute), 70

## D

- danger (rinoh.structure.AdmonitionTitles attribute), 119
- DEFAULT (in module rinoh.paragraph), 115
- DefaultSpacing (class in rinoh.paragraph), 114
- DefinitionList (class in rinoh.structure), 118
- depth (rinoh.structure.TableOfContentsStyle attribute), 67
- Dimension (class in rinoh.dimension), 96
- doc\_format() (rinoh.color.Color class method), 96
- Document (class in rinoh.document), 97
- document() (rinoh.template.TemplateConfiguration method), 124
- DocumentPart (class in rinoh.template), 124
- DocumentPartTemplate (class in rinoh.template), 125
- DocumentTemplate (class in rinoh.template), 123
- DocumentTree (class in rinoh.document), 97
- DOUBLE (in module rinoh.paragraph), 115
- dpi (rinoh.image.ImageArgs attribute), 104
- drop\_if\_empty (rinoh.template.ContentsPartTemplate attribute), 126
- drop\_if\_empty (rinoh.template.DocumentPartTemplate attribute), 125
- drop\_if\_empty (rinoh.template.FixedDocumentPartTemplate attribute), 127
- drop\_if\_empty (rinoh.template.TitlePartTemplate attribute), 125
- DummyFlowable (class in rinoh.flowable), 99

## E

- EN (in module rinoh.language), 106
- encoding (rinoh.font.Font attribute), 102
- end\_at\_page (rinoh.template.ContentsPartTemplate attribute), 126
- end\_at\_page (rinoh.template.DocumentPartTemplate attribute), 125
- end\_at\_page (rinoh.template.FixedDocumentPartTemplate attribute), 127
- end\_at\_page (rinoh.template.TitlePartTemplate attribute), 126
- error (rinoh.structure.AdmonitionTitles attribute), 119
- extra (rinoh.template.TitlePageTemplate attribute), 129

## F

- Field (class in rinoh.reference), 116
- Figure (class in rinoh.image), 105
- FigureStyle (class in rinoh.image), 31
- file\_or\_filename (rinoh.image.ImageArgs attribute), 104
- fill\_color (rinoh.draw.ShapeStyle attribute), 17
- fill\_color (rinoh.table.TableCellBackgroundStyle attribute), 86
- find\_source() (rinoh.attribute.RuleSet method), 95
- FixedDocumentPartTemplate (class in rinoh.template), 126

FixedSpacing (class in rinoh.paragraph), 114  
 Float (class in rinoh.flowable), 101  
 float (rinoh.flowable.FloatStyle attribute), 24  
 float (rinoh.image.FigureStyle attribute), 31  
 FloatStyle (class in rinoh.flowable), 24  
 flow() (rinoh.flowable.Float method), 101  
 flow() (rinoh.index.IndexTarget method), 105  
 flow() (rinoh.reference.Note method), 116  
 flow() (rinoh.structure.Heading method), 117  
 Flowable (class in rinoh.flowable), 99  
 flowable\_spacing (rinoh.flowable.GroupedFlowablesStyle attribute), 22  
 flowable\_spacing (rinoh.image.FigureStyle attribute), 34  
 flowable\_spacing (rinoh.index.IndexStyle attribute), 34  
 flowable\_spacing (rinoh.structure.AdmonitionStyle attribute), 77  
 flowable\_spacing (rinoh.structure.ListStyle attribute), 60  
 flowable\_spacing (rinoh.structure.SectionStyle attribute), 52  
 flowable\_spacing (rinoh.structure.TableOfContentsStyle attribute), 67  
 flowable\_spacing (rinoh.table.TableCellStyle attribute), 84  
 flowables (rinoh.template.FixedDocumentPartTemplate attribute), 126  
 flowables() (rinoh.flowable.GroupedFlowables method), 100  
 flowables() (rinoh.flowable.StaticGroupedFlowables method), 101  
 flowables() (rinoh.index.Index method), 105  
 flowables() (rinoh.structure.Admonition method), 118  
 flowables() (rinoh.structure.TableOfContents method), 118  
 FlowableState (class in rinoh.flowable), 99  
 FlowableStyle (class in rinoh.flowable), 18  
 FlowableWidth (class in rinoh.flowable), 101  
 Font (class in rinoh.font), 101  
 font\_color (rinoh.image.CaptionStyle attribute), 30  
 font\_color (rinoh.paragraph.ParagraphStyle attribute), 40  
 font\_color (rinoh.reference.NoteMarkerStyle attribute), 51  
 font\_color (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 font\_color (rinoh.structure.HeadingStyle attribute), 58  
 font\_color (rinoh.structure.ListItemLabelStyle attribute), 66  
 font\_color (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 font\_color (rinoh.structure.TableOfContentsStyle attribute), 71  
 font\_color (rinoh.text.TextStyle attribute), 41  
 font\_size (rinoh.image.CaptionStyle attribute), 30  
 font\_size (rinoh.paragraph.ParagraphStyle attribute), 40  
 font\_size (rinoh.reference.NoteMarkerStyle attribute), 50  
 font\_size (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 font\_size (rinoh.structure.HeadingStyle attribute), 58  
 font\_size (rinoh.structure.ListItemLabelStyle attribute), 66  
 font\_size (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 font\_size (rinoh.structure.TableOfContentsStyle attribute), 71  
 font\_size (rinoh.text.TextStyle attribute), 41  
 font\_slant (rinoh.image.CaptionStyle attribute), 30  
 font\_slant (rinoh.paragraph.ParagraphStyle attribute), 39  
 font\_slant (rinoh.reference.NoteMarkerStyle attribute), 50  
 font\_slant (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 font\_slant (rinoh.structure.HeadingStyle attribute), 58  
 font\_slant (rinoh.structure.ListItemLabelStyle attribute), 65  
 font\_slant (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 font\_slant (rinoh.structure.TableOfContentsStyle attribute), 71  
 font\_slant (rinoh.text.TextStyle attribute), 41  
 font\_variant (rinoh.image.CaptionStyle attribute), 30  
 font\_variant (rinoh.paragraph.ParagraphStyle attribute), 40  
 font\_variant (rinoh.reference.NoteMarkerStyle attribute), 51

font\_variant (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 font\_variant (rinoh.structure.HeadingStyle attribute), 58  
 font\_variant (rinoh.structure.ListItemLabelStyle attribute), 66  
 font\_variant (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 font\_variant (rinoh.structure.TableOfContentsStyle attribute), 71  
 font\_variant (rinoh.text.TextStyle attribute), 41  
 font\_weight (rinoh.image.CaptionStyle attribute), 30  
 font\_weight (rinoh.paragraph.ParagraphStyle attribute), 39  
 font\_weight (rinoh.reference.NoteMarkerStyle attribute), 50  
 font\_weight (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 font\_weight (rinoh.structure.HeadingStyle attribute), 58  
 font\_weight (rinoh.structure.ListItemLabelStyle attribute), 65  
 font\_weight (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 font\_weight (rinoh.structure.TableOfContentsStyle attribute), 70  
 font\_weight (rinoh.text.TextStyle attribute), 41  
 font\_width (rinoh.image.CaptionStyle attribute), 30  
 font\_width (rinoh.paragraph.ParagraphStyle attribute), 39  
 font\_width (rinoh.reference.NoteMarkerStyle attribute), 50  
 font\_width (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 font\_width (rinoh.structure.HeadingStyle attribute), 58  
 font\_width (rinoh.structure.ListItemLabelStyle attribute), 66  
 font\_width (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 font\_width (rinoh.structure.TableOfContentsStyle attribute), 71  
 font\_width (rinoh.text.TextStyle attribute), 41  
 fonts() (rinoh.font.Typeface method), 102  
 FontSlant (class in rinoh.font.style), 103  
 FontVariant (class in rinoh.font.style), 103  
 FontWeight (class in rinoh.font.style), 103  
 FontWidth (class in rinoh.font.style), 103  
 footer\_text (rinoh.template.PageTemplate attribute), 128  
 format\_number() (in module rinoh.number), 111  
 FR (in module rinoh.language), 106  
 front\_matter (rinoh.templates.article.Article attribute), 90  
 front\_matter (rinoh.templates.book.Book attribute), 92  
 front\_matter\_left\_page (rinoh.templates.book.Book attribute), 93  
 front\_matter\_page (rinoh.templates.article.Article attribute), 90  
 front\_matter\_page (rinoh.templates.book.Book attribute), 93  
 front\_matter\_right\_page (rinoh.templates.book.Book attribute), 93

## G

get\_font() (rinoh.font.Typeface method), 103  
 get\_glyph() (rinoh.font.Font method), 102  
 get\_kerning() (rinoh.font.Font method), 102  
 get\_ligature() (rinoh.font.Font method), 102  
 get\_position() (rinoh.paragraph.TabStop method), 115  
 get\_rowspanned\_columns() (rinoh.table.TableRow method), 122  
 get\_selector() (rinoh.style.StyleSheet method), 121  
 Glyph (class in rinoh.paragraph), 115  
 GlyphsSpan (class in rinoh.paragraph), 115  
 Gray (class in rinoh.color), 96  
 GroupedFlowables (class in rinoh.flowable), 100  
 GroupedFlowablesState (class in rinoh.flowable), 100  
 GroupedFlowablesStyle (class in rinoh.flowable), 22  
 GroupedLabeledFlowables (class in rinoh.flowable), 101  
 grow() (rinoh.dimension.Dimension method), 96

## H

has\_class (rinoh.style.Styled property), 121  
has\_classes (rinoh.style.Styled property), 121  
has\_id (rinoh.style.Styled property), 121  
header\_footer\_distance (rinoh.template.PageTemplate attribute), 127  
header\_text (rinoh.template.PageTemplate attribute), 127  
Heading (class in rinoh.structure), 117  
HeadingStyle (class in rinoh.structure), 54  
height (rinoh.image.ImageArgs attribute), 104  
height() (rinoh.text.StyledText method), 113  
HexColor (class in rinoh.color), 96  
hide (rinoh.draw.LineStyle attribute), 17  
hide (rinoh.draw.ShapeStyle attribute), 18  
hide (rinoh.flowable.FloatStyle attribute), 26  
hide (rinoh.flowable.FlowableStyle attribute), 20  
hide (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
hide (rinoh.flowable.LabeledFlowableStyle attribute), 22  
hide (rinoh.image.CaptionStyle attribute), 31  
hide (rinoh.image.FigureStyle attribute), 33  
hide (rinoh.index.IndexStyle attribute), 36  
hide (rinoh.inline.InlineFlowableStyle attribute), 44  
hide (rinoh.number.LabelStyle attribute), 110  
hide (rinoh.number.NumberStyle attribute), 111  
hide (rinoh.paragraph.ParagraphStyle attribute), 39  
hide (rinoh.reference.NoteMarkerStyle attribute), 52  
hide (rinoh.reference.ReferencingParagraphStyle attribute), 50  
hide (rinoh.structure.AdmonitionStyle attribute), 79  
hide (rinoh.structure.HeadingStyle attribute), 59  
hide (rinoh.structure.HorizontalRuleStyle attribute), 81  
hide (rinoh.structure.ListItemLabelStyle attribute), 67  
hide (rinoh.structure.ListStyle attribute), 62  
hide (rinoh.structure.SectionStyle attribute), 54  
hide (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
hide (rinoh.structure.TableOfContentsStyle attribute), 69  
hide (rinoh.style.Style attribute), 121  
hide (rinoh.table.TableCellBackgroundStyle attribute), 87  
hide (rinoh.table.TableCellBorderStyle attribute), 86  
hide (rinoh.table.TableCellStyle attribute), 86  
hide (rinoh.table.TableStyle attribute), 83  
hide (rinoh.text.TextStyle attribute), 42  
hint (rinoh.structure.AdmonitionTitles attribute), 119  
horizontal\_align (rinoh.flowable.FloatStyle attribute), 25  
horizontal\_align (rinoh.flowable.FlowableStyle attribute), 18  
horizontal\_align (rinoh.flowable.GroupedFlowablesStyle attribute), 23  
horizontal\_align (rinoh.flowable.LabeledFlowableStyle attribute), 20  
horizontal\_align (rinoh.image.CaptionStyle attribute), 28  
horizontal\_align (rinoh.image.FigureStyle attribute), 32  
horizontal\_align (rinoh.index.IndexStyle attribute), 34  
horizontal\_align (rinoh.inline.InlineFlowableStyle attribute), 43  
horizontal\_align (rinoh.paragraph.ParagraphStyle attribute), 37  
horizontal\_align (rinoh.reference.ReferencingParagraphStyle attribute), 46  
horizontal\_align (rinoh.structure.AdmonitionStyle attribute), 77  
horizontal\_align (rinoh.structure.HeadingStyle attribute), 56  
horizontal\_align (rinoh.structure.HorizontalRuleStyle attribute), 79  
horizontal\_align (rinoh.structure.ListItemLabelStyle attribute), 63  
horizontal\_align (rinoh.structure.ListStyle attribute), 60  
horizontal\_align (rinoh.structure.SectionStyle attribute), 53

horizontal\_align (rinoh.structure.TableOfContentsEntryStyle attribute), 73  
 horizontal\_align (rinoh.structure.TableOfContentsStyle attribute), 67  
 horizontal\_align (rinoh.table.TableCellStyle attribute), 84  
 horizontal\_align (rinoh.table.TableStyle attribute), 82  
 HorizontalAlignment (class in rinoh.flowable), 101  
 HorizontalRule (class in rinoh.structure), 120  
 HorizontalRuleStyle (class in rinoh.structure), 79  
 hyphen\_chars (rinoh.image.CaptionStyle attribute), 31  
 hyphen\_chars (rinoh.paragraph.ParagraphStyle attribute), 40  
 hyphen\_chars (rinoh.reference.NoteMarkerStyle attribute), 51  
 hyphen\_chars (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 hyphen\_chars (rinoh.structure.HeadingStyle attribute), 59  
 hyphen\_chars (rinoh.structure.ListItemLabelStyle attribute), 66  
 hyphen\_chars (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 hyphen\_chars (rinoh.structure.TableOfContentsStyle attribute), 71  
 hyphen\_chars (rinoh.text.TextStyle attribute), 42  
 hyphen\_lang (rinoh.image.CaptionStyle attribute), 31  
 hyphen\_lang (rinoh.paragraph.ParagraphStyle attribute), 40  
 hyphen\_lang (rinoh.reference.NoteMarkerStyle attribute), 51  
 hyphen\_lang (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 hyphen\_lang (rinoh.structure.HeadingStyle attribute), 59  
 hyphen\_lang (rinoh.structure.ListItemLabelStyle attribute), 66  
 hyphen\_lang (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 hyphen\_lang (rinoh.structure.TableOfContentsStyle attribute), 72  
 hyphen\_lang (rinoh.text.TextStyle attribute), 42  
 hyphenate (rinoh.image.CaptionStyle attribute), 31  
 hyphenate (rinoh.paragraph.ParagraphStyle attribute), 40  
 hyphenate (rinoh.reference.NoteMarkerStyle attribute), 51  
 hyphenate (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 hyphenate (rinoh.structure.HeadingStyle attribute), 59  
 hyphenate (rinoh.structure.ListItemLabelStyle attribute), 66  
 hyphenate (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 hyphenate (rinoh.structure.TableOfContentsStyle attribute), 71  
 hyphenate (rinoh.text.TextStyle attribute), 42  
 HyphenatorStore (class in rinoh.paragraph), 115

## I

Image (class in rinoh.image), 103  
 ImageArgs (class in rinoh.image), 104  
 important (rinoh.structure.AdmonitionTitles attribute), 119  
 INCH (in module rinoh.dimension), 97  
 indent\_first (rinoh.image.CaptionStyle attribute), 27  
 indent\_first (rinoh.paragraph.ParagraphStyle attribute), 36  
 indent\_first (rinoh.reference.ReferencingParagraphStyle attribute), 45  
 indent\_first (rinoh.structure.HeadingStyle attribute), 55  
 indent\_first (rinoh.structure.ListItemLabelStyle attribute), 62  
 indent\_first (rinoh.structure.TableOfContentsEntryStyle attribute), 72  
 indent\_first (rinoh.structure.TableOfContentsStyle attribute), 69  
 Index (class in rinoh.index), 105  
 index (rinoh.structure.SectionTitles attribute), 118  
 IndexLabel (class in rinoh.index), 105  
 IndexSection (class in rinoh.index), 105  
 IndexStyle (class in rinoh.index), 34  
 IndexTarget (class in rinoh.index), 105  
 IndexTerm (class in rinoh.index), 105  
 initials (rinoh.index.IndexStyle attribute), 34  
 inline\_title (rinoh.structure.AdmonitionStyle attribute), 77



InlineFlowable (class in rinoh.inline), 113  
InlineFlowableStyle (class in rinoh.inline), 42  
InlineImage (class in rinoh.image), 103  
InlineIndexTarget (class in rinoh.index), 105  
input  
    rinoh command line option, 5  
Integer (class in rinoh.attribute), 95  
is\_script() (rinoh.text.StyledText method), 113  
IT (in module rinoh.language), 107  
items (rinoh.text.MixedStyledText property), 113  
items (rinoh.text.StyledText property), 113

## J

JUNIOR\_LEGAL (in module rinoh.paper), 112

## K

keep\_with\_next (rinoh.flowable.FloatStyle attribute), 26  
keep\_with\_next (rinoh.flowable.FlowableStyle attribute), 19  
keep\_with\_next (rinoh.flowable.GroupedFlowablesStyle attribute), 23  
keep\_with\_next (rinoh.flowable.LabeledFlowableStyle attribute), 21  
keep\_with\_next (rinoh.image.CaptionStyle attribute), 28  
keep\_with\_next (rinoh.image.FigureStyle attribute), 32  
keep\_with\_next (rinoh.index.IndexStyle attribute), 35  
keep\_with\_next (rinoh.inline.InlineFlowableStyle attribute), 44  
keep\_with\_next (rinoh.paragraph.ParagraphStyle attribute), 38  
keep\_with\_next (rinoh.reference.ReferencingParagraphStyle attribute), 47  
keep\_with\_next (rinoh.structure.AdmonitionStyle attribute), 78  
keep\_with\_next (rinoh.structure.HeadingStyle attribute), 55  
keep\_with\_next (rinoh.structure.HorizontalRuleStyle attribute), 80  
keep\_with\_next (rinoh.structure.ListItemLabelStyle attribute), 64  
keep\_with\_next (rinoh.structure.ListStyle attribute), 61  
keep\_with\_next (rinoh.structure.SectionStyle attribute), 53  
keep\_with\_next (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
keep\_with\_next (rinoh.structure.TableOfContentsStyle attribute), 68  
keep\_with\_next (rinoh.table.TableCellStyle attribute), 85  
keep\_with\_next (rinoh.table.TableStyle attribute), 83  
kerning (rinoh.image.CaptionStyle attribute), 30  
kerning (rinoh.paragraph.ParagraphStyle attribute), 40  
kerning (rinoh.reference.NoteMarkerStyle attribute), 51  
kerning (rinoh.reference.ReferencingParagraphStyle attribute), 49  
kerning (rinoh.structure.HeadingStyle attribute), 58  
kerning (rinoh.structure.ListItemLabelStyle attribute), 66  
kerning (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
kerning (rinoh.structure.TableOfContentsStyle attribute), 71  
kerning (rinoh.text.TextStyle attribute), 42

## L

Label (class in rinoh.number), 110  
label\_max\_width (rinoh.flowable.LabeledFlowableStyle attribute), 20  
label\_min\_width (rinoh.flowable.LabeledFlowableStyle attribute), 20  
label\_prefix (rinoh.image.CaptionStyle attribute), 29  
label\_prefix (rinoh.number.LabelStyle attribute), 110  
label\_prefix (rinoh.number.NumberStyle attribute), 111  
label\_prefix (rinoh.paragraph.ParagraphStyle attribute), 39  
label\_prefix (rinoh.reference.NoteMarkerStyle attribute), 52  
label\_prefix (rinoh.reference.ReferencingParagraphStyle attribute), 48  
label\_prefix (rinoh.structure.HeadingStyle attribute), 57

label\_prefix (rinoh.structure.ListItemLabelStyle attribute), 65  
 label\_prefix (rinoh.structure.ListStyle attribute), 62  
 label\_prefix (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 label\_prefix (rinoh.structure.TableOfContentsStyle attribute), 70  
 label\_spacing (rinoh.flowable.LabeledFlowableStyle attribute), 20  
 label\_suffix (rinoh.image.CaptionStyle attribute), 29  
 label\_suffix (rinoh.number.LabelStyle attribute), 110  
 label\_suffix (rinoh.number.NumberStyle attribute), 111  
 label\_suffix (rinoh.paragraph.ParagraphStyle attribute), 39  
 label\_suffix (rinoh.reference.NoteMarkerStyle attribute), 52  
 label\_suffix (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 label\_suffix (rinoh.structure.HeadingStyle attribute), 57  
 label\_suffix (rinoh.structure.ListItemLabelStyle attribute), 65  
 label\_suffix (rinoh.structure.ListStyle attribute), 62  
 label\_suffix (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 label\_suffix (rinoh.structure.TableOfContentsStyle attribute), 70  
 LabeledFlowable (class in rinoh.flowable), 100  
 LabeledFlowableState (class in rinoh.flowable), 100  
 LabeledFlowableStyle (class in rinoh.flowable), 20  
 LabelStyle (class in rinoh.number), 110  
 Language (class in rinoh.language), 105  
 language (rinoh.template.DocumentTemplate attribute), 123  
 language (rinoh.templates.article.Article attribute), 89  
 language (rinoh.templates.book.Book attribute), 91  
 languages (rinoh.language.Language attribute), 105  
 Leading (class in rinoh.paragraph), 114  
 LEDGER (in module rinoh.paper), 112  
 left\_margin (rinoh.template.PageTemplate attribute), 128  
 left\_margin (rinoh.template.PageTemplateBase attribute), 130  
 left\_margin (rinoh.template.TitlePageTemplate attribute), 129  
 LEGAL (in module rinoh.paper), 112  
 LETTER (in module rinoh.paper), 112  
 ligatures (rinoh.image.CaptionStyle attribute), 31  
 ligatures (rinoh.paragraph.ParagraphStyle attribute), 40  
 ligatures (rinoh.reference.NoteMarkerStyle attribute), 51  
 ligatures (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 ligatures (rinoh.structure.HeadingStyle attribute), 58  
 ligatures (rinoh.structure.ListItemLabelStyle attribute), 66  
 ligatures (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
 ligatures (rinoh.structure.TableOfContentsStyle attribute), 71  
 ligatures (rinoh.text.TextStyle attribute), 42  
 limit\_width (rinoh.image.ImageArgs attribute), 104  
 Line (class in rinoh.draw), 98  
 line\_spacing (rinoh.image.CaptionStyle attribute), 27  
 line\_spacing (rinoh.paragraph.ParagraphStyle attribute), 36  
 line\_spacing (rinoh.reference.ReferencingParagraphStyle attribute), 45  
 line\_spacing (rinoh.structure.HeadingStyle attribute), 55  
 line\_spacing (rinoh.structure.ListItemLabelStyle attribute), 62  
 line\_spacing (rinoh.structure.TableOfContentsEntryStyle attribute), 72  
 line\_spacing (rinoh.structure.TableOfContentsStyle attribute), 69  
 LineSpacing (class in rinoh.paragraph), 114  
 LineStyle (class in rinoh.draw), 17  
 link (rinoh.reference.NoteMarkerStyle attribute), 50  
 List (class in rinoh.structure), 118  
 list\_class (rinoh.image.ListOfFiguresSection attribute), 105  
 list\_class (rinoh.table.ListOfTablesSection attribute), 123  
 list\_of\_figures (rinoh.structure.SectionTitles attribute), 117  
 list\_of\_tables (rinoh.structure.SectionTitles attribute), 117

- ListItemLabelStyle (class in rinoh.structure), 62
- ListOfFigures (class in rinoh.image), 105
- ListOfFiguresSection (class in rinoh.image), 105
- ListOfTables (class in rinoh.table), 123
- ListOfTablesSection (class in rinoh.table), 123
- ListStyle (class in rinoh.structure), 59
- Locale (class in rinoh.text), 113
- location (rinoh.attribute.RuleSetFile property), 95

## M

- margin\_left (rinoh.flowable.FloatStyle attribute), 25
- margin\_left (rinoh.flowable.FlowableStyle attribute), 18
- margin\_left (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- margin\_left (rinoh.flowable.LabeledFlowableStyle attribute), 21
- margin\_left (rinoh.image.CaptionStyle attribute), 28
- margin\_left (rinoh.image.FigureStyle attribute), 32
- margin\_left (rinoh.index.IndexStyle attribute), 34
- margin\_left (rinoh.inline.InlineFlowableStyle attribute), 43
- margin\_left (rinoh.paragraph.ParagraphStyle attribute), 37
- margin\_left (rinoh.reference.ReferencingParagraphStyle attribute), 46
- margin\_left (rinoh.structure.AdmonitionStyle attribute), 78
- margin\_left (rinoh.structure.HeadingStyle attribute), 56
- margin\_left (rinoh.structure.HorizontalRuleStyle attribute), 80
- margin\_left (rinoh.structure.ListItemLabelStyle attribute), 63
- margin\_left (rinoh.structure.ListStyle attribute), 60
- margin\_left (rinoh.structure.SectionStyle attribute), 53
- margin\_left (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- margin\_left (rinoh.structure.TableOfContentsStyle attribute), 68
- margin\_left (rinoh.table.TableCellStyle attribute), 84
- margin\_left (rinoh.table.TableStyle attribute), 82
- margin\_right (rinoh.flowable.FloatStyle attribute), 25
- margin\_right (rinoh.flowable.FlowableStyle attribute), 18
- margin\_right (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- margin\_right (rinoh.flowable.LabeledFlowableStyle attribute), 21
- margin\_right (rinoh.image.CaptionStyle attribute), 28
- margin\_right (rinoh.image.FigureStyle attribute), 32
- margin\_right (rinoh.index.IndexStyle attribute), 34
- margin\_right (rinoh.inline.InlineFlowableStyle attribute), 43
- margin\_right (rinoh.paragraph.ParagraphStyle attribute), 37
- margin\_right (rinoh.reference.ReferencingParagraphStyle attribute), 46
- margin\_right (rinoh.structure.AdmonitionStyle attribute), 78
- margin\_right (rinoh.structure.HeadingStyle attribute), 56
- margin\_right (rinoh.structure.HorizontalRuleStyle attribute), 80
- margin\_right (rinoh.structure.ListItemLabelStyle attribute), 63
- margin\_right (rinoh.structure.ListStyle attribute), 60
- margin\_right (rinoh.structure.SectionStyle attribute), 53
- margin\_right (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- margin\_right (rinoh.structure.TableOfContentsStyle attribute), 68
- margin\_right (rinoh.table.TableCellStyle attribute), 84
- margin\_right (rinoh.table.TableStyle attribute), 82
- matcher (in module rinoh.stylesheets), 11
- MixedStyledText (class in rinoh.text), 113
- MM (in module rinoh.dimension), 97
- module
  - rinoh.attribute, 95
  - rinoh.color, 96
  - rinoh.dimension, 96

- rinoh.document, 97
- rinoh.draw, 98
- rinoh.flowable, 99
- rinoh.font, 101
- rinoh.font.style, 103
- rinoh.image, 103
- rinoh.index, 105
- rinoh.inline, 113
- rinoh.language, 105
- rinoh.number, 110
- rinoh.paper, 111
- rinoh.paragraph, 112
- rinoh.reference, 115
- rinoh.strings, 120
- rinoh.structure, 117
- rinoh.style, 120
- rinoh.stylesheets, 11
- rinoh.table, 122
- rinoh.template, 123
- rinoh.templates.article, 89
- rinoh.templates.book, 91
- rinoh.text, 112

## N

- new\_page() (rinoh.template.DocumentPart method), 125
- NL (in module rinoh.language), 108
- Note (class in rinoh.reference), 116
- note (rinoh.structure.AdmonitionTitles attribute), 119
- NoteMarkerBase (class in rinoh.reference), 116
- NoteMarkerByID (class in rinoh.reference), 116
- NoteMarkerStyle (class in rinoh.reference), 50
- NoteMarkerWithNote (class in rinoh.reference), 116
- number\_format (rinoh.image.CaptionStyle attribute), 27
- number\_format (rinoh.number.NumberStyle attribute), 110
- number\_format (rinoh.paragraph.ParagraphStyle attribute), 37
- number\_format (rinoh.reference.NoteMarkerStyle attribute), 52
- number\_format (rinoh.reference.ReferencingParagraphStyle attribute), 46
- number\_format (rinoh.structure.HeadingStyle attribute), 55
- number\_format (rinoh.structure.ListItemLabelStyle attribute), 63
- number\_format (rinoh.structure.ListStyle attribute), 62
- number\_format (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- number\_format (rinoh.structure.TableOfContentsStyle attribute), 70
- number\_separator (rinoh.image.CaptionStyle attribute), 27
- number\_separator (rinoh.paragraph.ParagraphStyle attribute), 37
- number\_separator (rinoh.reference.ReferencingParagraphStyle attribute), 46
- number\_separator (rinoh.structure.HeadingStyle attribute), 55
- number\_separator (rinoh.structure.ListItemLabelStyle attribute), 63
- number\_separator (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- number\_separator (rinoh.structure.TableOfContentsStyle attribute), 70
- NumberFormat (class in rinoh.number), 110
- numbering\_level (rinoh.image.CaptionStyle attribute), 27
- numbering\_level (rinoh.paragraph.ParagraphStyle attribute), 37
- numbering\_level (rinoh.reference.ReferencingParagraphStyle attribute), 46
- numbering\_level (rinoh.structure.HeadingStyle attribute), 55
- numbering\_level (rinoh.structure.ListItemLabelStyle attribute), 63
- numbering\_level (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- numbering\_level (rinoh.structure.TableOfContentsStyle attribute), 70

NumberStyle (class in `rinoh.number`), 110

## O

OptionSet (class in `rinoh.attribute`), 95

ordered (`rinoh.structure.ListStyle` attribute), 59

OverrideDefault (class in `rinoh.attribute`), 95

## P

padding (`rinoh.flowable.FloatStyle` attribute), 25

padding (`rinoh.flowable.FlowableStyle` attribute), 18

padding (`rinoh.flowable.GroupedFlowablesStyle` attribute), 23

padding (`rinoh.flowable.LabeledFlowableStyle` attribute), 21

padding (`rinoh.image.CaptionStyle` attribute), 28

padding (`rinoh.image.FigureStyle` attribute), 32

padding (`rinoh.index.IndexStyle` attribute), 35

padding (`rinoh.inline.InlineFlowableStyle` attribute), 43

padding (`rinoh.paragraph.ParagraphStyle` attribute), 37

padding (`rinoh.reference.ReferencingParagraphStyle` attribute), 46

padding (`rinoh.structure.AdmonitionStyle` attribute), 78

padding (`rinoh.structure.HeadingStyle` attribute), 56

padding (`rinoh.structure.HorizontalRuleStyle` attribute), 80

padding (`rinoh.structure.ListItemLabelStyle` attribute), 64

padding (`rinoh.structure.ListStyle` attribute), 60

padding (`rinoh.structure.SectionStyle` attribute), 53

padding (`rinoh.structure.TableOfContentsEntryStyle` attribute), 74

padding (`rinoh.structure.TableOfContentsStyle` attribute), 68

padding (`rinoh.table.TableCellStyle` attribute), 85

padding (`rinoh.table.TableStyle` attribute), 82

padding\_bottom (`rinoh.flowable.FloatStyle` attribute), 25

padding\_bottom (`rinoh.flowable.FlowableStyle` attribute), 19

padding\_bottom (`rinoh.flowable.GroupedFlowablesStyle` attribute), 23

padding\_bottom (`rinoh.flowable.LabeledFlowableStyle` attribute), 21

padding\_bottom (`rinoh.image.CaptionStyle` attribute), 28

padding\_bottom (`rinoh.image.FigureStyle` attribute), 32

padding\_bottom (`rinoh.index.IndexStyle` attribute), 35

padding\_bottom (`rinoh.inline.InlineFlowableStyle` attribute), 43

padding\_bottom (`rinoh.paragraph.ParagraphStyle` attribute), 38

padding\_bottom (`rinoh.reference.ReferencingParagraphStyle` attribute), 47

padding\_bottom (`rinoh.structure.AdmonitionStyle` attribute), 78

padding\_bottom (`rinoh.structure.HeadingStyle` attribute), 56

padding\_bottom (`rinoh.structure.HorizontalRuleStyle` attribute), 80

padding\_bottom (`rinoh.structure.ListItemLabelStyle` attribute), 64

padding\_bottom (`rinoh.structure.ListStyle` attribute), 61

padding\_bottom (`rinoh.structure.SectionStyle` attribute), 53

padding\_bottom (`rinoh.structure.TableOfContentsEntryStyle` attribute), 74

padding\_bottom (`rinoh.structure.TableOfContentsStyle` attribute), 68

padding\_bottom (`rinoh.table.TableCellStyle` attribute), 85

padding\_bottom (`rinoh.table.TableStyle` attribute), 83

padding\_left (`rinoh.flowable.FloatStyle` attribute), 25

padding\_left (`rinoh.flowable.FlowableStyle` attribute), 18

padding\_left (`rinoh.flowable.GroupedFlowablesStyle` attribute), 23

padding\_left (`rinoh.flowable.LabeledFlowableStyle` attribute), 21

padding\_left (`rinoh.image.CaptionStyle` attribute), 28

padding\_left (`rinoh.image.FigureStyle` attribute), 32

padding\_left (`rinoh.index.IndexStyle` attribute), 35

padding\_left (`rinoh.inline.InlineFlowableStyle` attribute), 43

padding\_left (`rinoh.paragraph.ParagraphStyle` attribute), 37

padding\_left (rinoh.reference.ReferencingParagraphStyle attribute), 46

padding\_left (rinoh.structure.AdmonitionStyle attribute), 78

padding\_left (rinoh.structure.HeadingStyle attribute), 56

padding\_left (rinoh.structure.HorizontalRuleStyle attribute), 80

padding\_left (rinoh.structure.ListItemLabelStyle attribute), 64

padding\_left (rinoh.structure.ListStyle attribute), 60

padding\_left (rinoh.structure.SectionStyle attribute), 53

padding\_left (rinoh.structure.TableOfContentsEntryStyle attribute), 74

padding\_left (rinoh.structure.TableOfContentsStyle attribute), 68

padding\_left (rinoh.table.TableCellStyle attribute), 85

padding\_left (rinoh.table.TableStyle attribute), 82

padding\_right (rinoh.flowable.FloatStyle attribute), 25

padding\_right (rinoh.flowable.FlowableStyle attribute), 18

padding\_right (rinoh.flowable.GroupedFlowablesStyle attribute), 23

padding\_right (rinoh.flowable.LabeledFlowableStyle attribute), 21

padding\_right (rinoh.image.CaptionStyle attribute), 28

padding\_right (rinoh.image.FigureStyle attribute), 32

padding\_right (rinoh.index.IndexStyle attribute), 35

padding\_right (rinoh.inline.InlineFlowableStyle attribute), 43

padding\_right (rinoh.paragraph.ParagraphStyle attribute), 38

padding\_right (rinoh.reference.ReferencingParagraphStyle attribute), 47

padding\_right (rinoh.structure.AdmonitionStyle attribute), 78

padding\_right (rinoh.structure.HeadingStyle attribute), 56

padding\_right (rinoh.structure.HorizontalRuleStyle attribute), 80

padding\_right (rinoh.structure.ListItemLabelStyle attribute), 64

padding\_right (rinoh.structure.ListStyle attribute), 60

padding\_right (rinoh.structure.SectionStyle attribute), 53

padding\_right (rinoh.structure.TableOfContentsEntryStyle attribute), 74

padding\_right (rinoh.structure.TableOfContentsStyle attribute), 68

padding\_right (rinoh.table.TableCellStyle attribute), 85

padding\_right (rinoh.table.TableStyle attribute), 82

padding\_top (rinoh.flowable.FloatStyle attribute), 25

padding\_top (rinoh.flowable.FlowableStyle attribute), 19

padding\_top (rinoh.flowable.GroupedFlowablesStyle attribute), 23

padding\_top (rinoh.flowable.LabeledFlowableStyle attribute), 21

padding\_top (rinoh.image.CaptionStyle attribute), 28

padding\_top (rinoh.image.FigureStyle attribute), 32

padding\_top (rinoh.index.IndexStyle attribute), 35

padding\_top (rinoh.inline.InlineFlowableStyle attribute), 43

padding\_top (rinoh.paragraph.ParagraphStyle attribute), 38

padding\_top (rinoh.reference.ReferencingParagraphStyle attribute), 47

padding\_top (rinoh.structure.AdmonitionStyle attribute), 78

padding\_top (rinoh.structure.HeadingStyle attribute), 56

padding\_top (rinoh.structure.HorizontalRuleStyle attribute), 80

padding\_top (rinoh.structure.ListItemLabelStyle attribute), 64

padding\_top (rinoh.structure.ListStyle attribute), 61

padding\_top (rinoh.structure.SectionStyle attribute), 53

padding\_top (rinoh.structure.TableOfContentsEntryStyle attribute), 74

padding\_top (rinoh.structure.TableOfContentsStyle attribute), 68

padding\_top (rinoh.table.TableCellStyle attribute), 85

padding\_top (rinoh.table.TableStyle attribute), 82

Page (class in rinoh.document), 98

page (rinoh.document.Page property), 98

page (rinoh.templates.article.Article attribute), 90

page (rinoh.templates.book.Book attribute), 92

page\_break (rinoh.flowable.FloatStyle attribute), 26

page\_break (rinoh.flowable.FlowableStyle attribute), 19

page\_break (rinoh.flowable.GroupedFlowablesStyle attribute), 24

page\_break (rinoh.flowable.LabeledFlowableStyle attribute), 22  
 page\_break (rinoh.image.CaptionStyle attribute), 29  
 page\_break (rinoh.image.FigureStyle attribute), 33  
 page\_break (rinoh.index.IndexStyle attribute), 36  
 page\_break (rinoh.inline.InlineFlowableStyle attribute), 44  
 page\_break (rinoh.paragraph.ParagraphStyle attribute), 39  
 page\_break (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 page\_break (rinoh.structure.AdmonitionStyle attribute), 79  
 page\_break (rinoh.structure.HeadingStyle attribute), 57  
 page\_break (rinoh.structure.HorizontalRuleStyle attribute), 81  
 page\_break (rinoh.structure.ListItemLabelStyle attribute), 65  
 page\_break (rinoh.structure.ListStyle attribute), 61  
 page\_break (rinoh.structure.SectionStyle attribute), 54  
 page\_break (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 page\_break (rinoh.structure.TableOfContentsStyle attribute), 69  
 page\_break (rinoh.table.TableCellStyle attribute), 86  
 page\_break (rinoh.table.TableStyle attribute), 83  
 page\_number\_format (rinoh.template.ContentsPartTemplate attribute), 126  
 page\_number\_format (rinoh.template.DocumentPartTemplate attribute), 125  
 page\_number\_format (rinoh.template.FixedDocumentPartTemplate attribute), 127  
 page\_number\_format (rinoh.template.TitlePartTemplate attribute), 126  
 page\_number\_prefix (rinoh.template.ContentsPartTemplate attribute), 126  
 page\_number\_prefix (rinoh.template.DocumentPartTemplate attribute), 125  
 page\_number\_prefix (rinoh.template.FixedDocumentPartTemplate attribute), 127  
 page\_number\_prefix (rinoh.template.TitlePartTemplate attribute), 126  
 page\_orientation (rinoh.template.PageTemplate attribute), 128  
 page\_orientation (rinoh.template.PageTemplateBase attribute), 130  
 page\_orientation (rinoh.template.TitlePageTemplate attribute), 129  
 page\_size (rinoh.template.PageTemplate attribute), 128  
 page\_size (rinoh.template.PageTemplateBase attribute), 130  
 page\_size (rinoh.template.TitlePageTemplate attribute), 129  
 PageBreak (class in rinoh.flowable), 99  
 PageNumberFormat (class in rinoh.document), 98  
 PageOrientation (class in rinoh.document), 98  
 PageTemplate (class in rinoh.template), 127  
 PageTemplateBase (class in rinoh.template), 130  
 PageType (class in rinoh.document), 98  
 Paper (class in rinoh.paper), 111  
 Paragraph (class in rinoh.paragraph), 112  
 ParagraphBase (class in rinoh.paragraph), 112  
 ParagraphState (class in rinoh.paragraph), 112  
 ParagraphStyle (class in rinoh.paragraph), 36  
 parse\_string() (rinoh.color.Color class method), 96  
 ParseError, 95  
 parts (rinoh.template.DocumentTemplate attribute), 124  
 parts (rinoh.templates.article.Article attribute), 89  
 parts (rinoh.templates.book.Book attribute), 91  
 PartsList (class in rinoh.template), 124  
 PERCENT (in module rinoh.dimension), 97  
 PICA (in module rinoh.dimension), 97  
 PL (in module rinoh.language), 109  
 place() (rinoh.document.Page method), 98  
 Polygon (class in rinoh.draw), 99  
 position (rinoh.image.CaptionStyle attribute), 30  
 position (rinoh.paragraph.ParagraphStyle attribute), 40  
 position (rinoh.reference.NoteMarkerStyle attribute), 51  
 position (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 position (rinoh.structure.HeadingStyle attribute), 58

- position (rinoh.structure.ListItemLabelStyle attribute), 66
- position (rinoh.structure.TableOfContentsEntryStyle attribute), 76
- position (rinoh.structure.TableOfContentsStyle attribute), 71
- position (rinoh.text.TextStyle attribute), 41
- prepare() (rinoh.flowable.LabeledFlowable method), 100
- prepare() (rinoh.flowable.StaticGroupedFlowables method), 101
- prepare() (rinoh.paragraph.ParagraphBase method), 112
- prepare() (rinoh.reference.NoteMarkerBase method), 116
- prepare() (rinoh.reference.NoteMarkerWithNote method), 116
- prepare() (rinoh.structure.Heading method), 117
- prepare() (rinoh.table.Table method), 122
- prepare() (rinoh.table.TableRow method), 122
- prepare() (rinoh.table.TableSection method), 122
- prepare() (rinoh.text.MixedStyledText method), 113
- ProportionalSpacing (class in rinoh.paragraph), 114
- PT (in module rinoh.dimension), 97

## Q

- QUARTERS (in module rinoh.dimension), 97
- quiet (rinoh.reference.NoteMarkerStyle attribute), 50

## R

- Rectangle (class in rinoh.draw), 99
- Reference (class in rinoh.reference), 115
- ReferenceField (class in rinoh.reference), 115
- ReferenceText (class in rinoh.reference), 116
- ReferenceType (class in rinoh.reference), 116
- ReferencingParagraph (class in rinoh.reference), 116
- ReferencingParagraphStyle (class in rinoh.reference), 45
- render() (rinoh.document.Document method), 97
- render() (rinoh.document.Page method), 98
- render() (rinoh.flowable.GroupedFlowables method), 100
- render() (rinoh.flowable.GroupedLabeledFlowables method), 101
- render() (rinoh.flowable.LabeledFlowable method), 100
- render() (rinoh.paragraph.ParagraphBase method), 112
- render() (rinoh.structure.HorizontalRule method), 120
- render() (rinoh.table.Table method), 122
- repeat\_head (rinoh.table.TableStyle attribute), 82
- rgba (rinoh.color.Color property), 96
- right\_margin (rinoh.template.PageTemplate attribute), 128
- right\_margin (rinoh.template.PageTemplateBase attribute), 130
- right\_margin (rinoh.template.TitlePageTemplate attribute), 130
- rinoh command line option
  - docs, 6
  - format <format>, 5
  - help, 5
  - install-resources, 5
  - list-fonts <filename>, 6
  - list-formats, 6
  - list-options <frontend>, 6
  - list-stylesheets, 5
  - list-templates, 5
  - option <option=value>, 5
  - output <filename or directory>, 5
  - paper <paper>, 5
  - stylesheet <name or filename>, 5
  - template <name or filename>, 5



- version, 6
- f <format>, 5
- h, 5
- i, 5
- O <filename or directory>, 5
- o <option=value>, 5
- p <paper>, 5
- s <name or filename>, 5
- t <name or filename>, 5
- input, 5
- rinoh.attribute
  - module, 95
- rinoh.color
  - module, 96
- rinoh.dimension
  - module, 96
- rinoh.document
  - module, 97
- rinoh.draw
  - module, 98
- rinoh.flowable
  - module, 99
- rinoh.font
  - module, 101
- rinoh.font.style
  - module, 103
- rinoh.image
  - module, 103
- rinoh.index
  - module, 105
- rinoh.inline
  - module, 113
- rinoh.language
  - module, 105
- rinoh.number
  - module, 110
- rinoh.paper
  - module, 111
- rinoh.paragraph
  - module, 112
- rinoh.reference
  - module, 115
- rinoh.strings
  - module, 120
- rinoh.structure
  - module, 117
- rinoh.style
  - module, 120
- rinoh.stylesheets
  - module, 11
- rinoh.table
  - module, 122
- rinoh.template
  - module, 123
- rinoh.templates.article
  - module, 89
- rinoh.templates.book
  - module, 91

- rinoh.text
  - module, 112
- rinoh\_documents
  - Sphinx builder configuration value, 7
- rinoh\_domain\_indices
  - Sphinx builder configuration value, 9
- rinoh\_logo
  - Sphinx builder configuration value, 9
- rinoh\_metadata
  - Sphinx builder configuration value, 9
- rinoh\_paper\_size
  - Sphinx builder configuration value, 8
- rinoh\_stylesheet
  - Sphinx builder configuration value, 8
- rinoh\_targets
  - Sphinx builder configuration value, 8
- rinoh\_template
  - Sphinx builder configuration value, 8
- root (rinoh.attribute.RuleSetFile property), 95
- rotate (rinoh.image.ImageArgs attribute), 104
- RuleSet (class in rinoh.attribute), 95
- RuleSetFile (class in rinoh.attribute), 95

## S

- Scale (class in rinoh.image), 104
- scale (rinoh.image.ImageArgs attribute), 104
- script\_level() (rinoh.text.StyledText method), 113
- Section (class in rinoh.structure), 117
- SectionStyle (class in rinoh.structure), 52
- SectionTitles (class in rinoh.structure), 117
- seealso (rinoh.structure.AdmonitionTitles attribute), 119
- SetMetadataFlowable (class in rinoh.flowable), 99
- Shape (class in rinoh.draw), 99
- ShapeStyle (class in rinoh.draw), 17
- show\_author (rinoh.template.TitlePageTemplate attribute), 129
- show\_date (rinoh.template.TitlePageTemplate attribute), 129
- show\_in\_toc (rinoh.structure.SectionStyle attribute), 52
- SINGLE (in module rinoh.paragraph), 115
- SingleStyledText (class in rinoh.text), 113
- space\_above (rinoh.flowable.FloatStyle attribute), 25
- space\_above (rinoh.flowable.FlowableStyle attribute), 18
- space\_above (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- space\_above (rinoh.flowable.LabeledFlowableStyle attribute), 20
- space\_above (rinoh.image.CaptionStyle attribute), 28
- space\_above (rinoh.image.FigureStyle attribute), 32
- space\_above (rinoh.index.IndexStyle attribute), 34
- space\_above (rinoh.inline.InlineFlowableStyle attribute), 43
- space\_above (rinoh.paragraph.ParagraphStyle attribute), 37
- space\_above (rinoh.reference.ReferencingParagraphStyle attribute), 46
- space\_above (rinoh.structure.AdmonitionStyle attribute), 77
- space\_above (rinoh.structure.HeadingStyle attribute), 56
- space\_above (rinoh.structure.HorizontalRuleStyle attribute), 79
- space\_above (rinoh.structure.ListItemLabelStyle attribute), 63
- space\_above (rinoh.structure.ListStyle attribute), 60
- space\_above (rinoh.structure.SectionStyle attribute), 53
- space\_above (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- space\_above (rinoh.structure.TableOfContentsStyle attribute), 67

- space\_above (rinoh.table.TableCellStyle attribute), 84
- space\_above (rinoh.table.TableStyle attribute), 82
- space\_below (rinoh.flowable.FloatStyle attribute), 25
- space\_below (rinoh.flowable.FlowableStyle attribute), 18
- space\_below (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- space\_below (rinoh.flowable.LabeledFlowableStyle attribute), 20
- space\_below (rinoh.image.CaptionStyle attribute), 28
- space\_below (rinoh.image.FigureStyle attribute), 32
- space\_below (rinoh.index.IndexStyle attribute), 34
- space\_below (rinoh.inline.InlineFlowableStyle attribute), 43
- space\_below (rinoh.paragraph.ParagraphStyle attribute), 37
- space\_below (rinoh.reference.ReferencingParagraphStyle attribute), 46
- space\_below (rinoh.structure.AdmonitionStyle attribute), 77
- space\_below (rinoh.structure.HeadingStyle attribute), 56
- space\_below (rinoh.structure.HorizontalRuleStyle attribute), 79
- space\_below (rinoh.structure.ListItemLabelStyle attribute), 63
- space\_below (rinoh.structure.ListStyle attribute), 60
- space\_below (rinoh.structure.SectionStyle attribute), 53
- space\_below (rinoh.structure.TableOfContentsEntryStyle attribute), 73
- space\_below (rinoh.structure.TableOfContentsStyle attribute), 68
- space\_below (rinoh.table.TableCellStyle attribute), 84
- space\_below (rinoh.table.TableStyle attribute), 82
- sphinx (in module rinoh.stylesheets), 11
- Sphinx builder configuration value
  - rinoh\_documents, 7
  - rinoh\_domain\_indices, 9
  - rinoh\_logo, 9
  - rinoh\_metadata, 9
  - rinoh\_paper\_size, 8
  - rinoh\_stylesheet, 8
  - rinoh\_targets, 8
  - rinoh\_template, 8
- sphinx\_article (in module rinoh.stylesheets), 11
- sphinx\_base14 (in module rinoh.stylesheets), 11
- split\_minimum\_rows (rinoh.table.TableStyle attribute), 81
- STANDARD (in module rinoh.paragraph), 115
- StaticGroupedFlowables (class in rinoh.flowable), 100
- String (class in rinoh.strings), 120
- StringCollection (class in rinoh.strings), 120
- StringField (class in rinoh.strings), 120
- Strings (class in rinoh.strings), 120
- strings (rinoh.template.DocumentTemplate attribute), 123
- strings (rinoh.templates.article.Article attribute), 89
- strings (rinoh.templates.book.Book attribute), 91
- Stroke (class in rinoh.draw), 98
- stroke (rinoh.draw.LineStyle attribute), 17
- stroke (rinoh.draw.ShapeStyle attribute), 17
- stroke (rinoh.structure.HorizontalRuleStyle attribute), 81
- stroke (rinoh.table.TableCellBackgroundStyle attribute), 86
- stroke (rinoh.table.TableCellBorderStyle attribute), 86
- Style (class in rinoh.style), 121
- style\_class (rinoh.draw.Line attribute), 98
- style\_class (rinoh.draw.Shape attribute), 99
- style\_class (rinoh.flowable.GroupedFlowables attribute), 100
- style\_class (rinoh.flowable.LabeledFlowable attribute), 100
- style\_class (rinoh.image.Caption attribute), 104
- style\_class (rinoh.image.Figure attribute), 105
- style\_class (rinoh.index.Index attribute), 105

style\_class (rinoh.inline.InlineFlowable attribute), 113  
 style\_class (rinoh.paragraph.ParagraphBase attribute), 112  
 style\_class (rinoh.reference.Note attribute), 116  
 style\_class (rinoh.reference.NoteMarkerBase attribute), 116  
 style\_class (rinoh.reference.ReferencingParagraph attribute), 116  
 style\_class (rinoh.structure.Admonition attribute), 118  
 style\_class (rinoh.structure.Heading attribute), 117  
 style\_class (rinoh.structure.HorizontalRule attribute), 120  
 style\_class (rinoh.structure.List attribute), 118  
 style\_class (rinoh.structure.TableOfContents attribute), 118  
 style\_class (rinoh.structure.TableOfContentsEntry attribute), 118  
 style\_class (rinoh.style.Styled attribute), 121  
 style\_class (rinoh.table.Table attribute), 122  
 style\_class (rinoh.table.TableCell attribute), 123  
 style\_class (rinoh.table.TableCellBackground attribute), 123  
 style\_class (rinoh.table.TableCellBorder attribute), 123  
 style\_class (rinoh.table.TableWithCaption attribute), 122  
 style\_class (rinoh.text.StyledText attribute), 113  
 Styled (class in rinoh.style), 120  
 StyledMatcher (class in rinoh.style), 122  
 StyledText (class in rinoh.text), 112  
 StyleSheet (class in rinoh.style), 121  
 stylesheet (rinoh.template.DocumentTemplate attribute), 123  
 stylesheet (rinoh.templates.article.Article attribute), 89  
 stylesheet (rinoh.templates.book.Book attribute), 91  
 StyleSheetFile (class in rinoh.style), 121

## T

tab\_stops (rinoh.image.CaptionStyle attribute), 27  
 tab\_stops (rinoh.paragraph.ParagraphStyle attribute), 36  
 tab\_stops (rinoh.reference.ReferencingParagraphStyle attribute), 45  
 tab\_stops (rinoh.structure.HeadingStyle attribute), 55  
 tab\_stops (rinoh.structure.ListItemLabelStyle attribute), 63  
 tab\_stops (rinoh.structure.TableOfContentsEntryStyle attribute), 73  
 tab\_stops (rinoh.structure.TableOfContentsStyle attribute), 70  
 TabAlign (class in rinoh.paragraph), 115  
 Table (class in rinoh.table), 122  
 table\_of\_contents (rinoh.templates.article.Article attribute), 89  
 TableBody (class in rinoh.table), 122  
 TableCell (class in rinoh.table), 123  
 TableCellBackground (class in rinoh.table), 123  
 TableCellBackgroundStyle (class in rinoh.table), 86  
 TableCellBorder (class in rinoh.table), 123  
 TableCellBorderStyle (class in rinoh.table), 86  
 TableCellStyle (class in rinoh.table), 84  
 TableHead (class in rinoh.table), 122  
 TableOfContents (class in rinoh.structure), 118  
 TableOfContentsEntry (class in rinoh.structure), 118  
 TableOfContentsEntryStyle (class in rinoh.structure), 72  
 TableOfContentsSection (class in rinoh.structure), 118  
 TableOfContentsStyle (class in rinoh.structure), 67  
 TableRow (class in rinoh.table), 122  
 TableSection (class in rinoh.table), 122  
 TableStyle (class in rinoh.table), 81  
 TableWithCaption (class in rinoh.table), 122  
 TABLOID (in module rinoh.paper), 112  
 TabStop (class in rinoh.paragraph), 115

TabStopList (class in rinoh.paragraph), 115  
 target\_has\_class() (rinoh.reference.ReferencingParagraph method), 116  
 target\_has\_classes() (rinoh.reference.ReferencingParagraph method), 116  
 target\_is\_of\_type() (rinoh.reference.ReferencingParagraph method), 116  
 target\_style() (rinoh.reference.ReferencingParagraph method), 116  
 template (rinoh.template.TemplateConfiguration attribute), 124  
 TemplateConfiguration (class in rinoh.template), 124  
 text (rinoh.reference.ReferencingParagraphStyle attribute), 45  
 text (rinoh.structure.TableOfContentsEntryStyle attribute), 72  
 text\_align (rinoh.image.CaptionStyle attribute), 27  
 text\_align (rinoh.paragraph.ParagraphStyle attribute), 36  
 text\_align (rinoh.reference.ReferencingParagraphStyle attribute), 45  
 text\_align (rinoh.structure.HeadingStyle attribute), 55  
 text\_align (rinoh.structure.ListItemLabelStyle attribute), 63  
 text\_align (rinoh.structure.TableOfContentsEntryStyle attribute), 72  
 text\_align (rinoh.structure.TableOfContentsStyle attribute), 69  
 TextAlign (class in rinoh.paragraph), 114  
 TextPosition (class in rinoh.font.style), 103  
 TextStyle (class in rinoh.text), 41  
 tip (rinoh.structure.AdmonitionTitles attribute), 119  
 title (rinoh.flowable.GroupedFlowablesStyle attribute), 22  
 title (rinoh.image.FigureStyle attribute), 33  
 title (rinoh.index.IndexStyle attribute), 34  
 title (rinoh.structure.AdmonitionStyle attribute), 77  
 title (rinoh.structure.ListStyle attribute), 60  
 title (rinoh.structure.SectionStyle attribute), 52  
 title (rinoh.structure.TableOfContentsStyle attribute), 67  
 title (rinoh.table.TableCellStyle attribute), 84  
 title (rinoh.templates.article.Article attribute), 90  
 title (rinoh.templates.book.Book attribute), 91  
 title\_page (rinoh.templates.article.Article attribute), 90  
 title\_page (rinoh.templates.book.Book attribute), 92  
 TitlePageTemplate (class in rinoh.template), 129  
 TitlePartTemplate (class in rinoh.template), 125  
 to\_string() (rinoh.index.InlineIndexTarget method), 105  
 to\_string() (rinoh.text.StyledText method), 112  
 top\_margin (rinoh.template.PageTemplate attribute), 128  
 top\_margin (rinoh.template.PageTemplateBase attribute), 131  
 top\_margin (rinoh.template.TitlePageTemplate attribute), 130  
 type (rinoh.reference.NoteMarkerStyle attribute), 50  
 Typeface (class in rinoh.font), 102  
 typeface (rinoh.image.CaptionStyle attribute), 30  
 typeface (rinoh.paragraph.ParagraphStyle attribute), 39  
 typeface (rinoh.reference.NoteMarkerStyle attribute), 50  
 typeface (rinoh.reference.ReferencingParagraphStyle attribute), 48  
 typeface (rinoh.structure.HeadingStyle attribute), 58  
 typeface (rinoh.structure.ListItemLabelStyle attribute), 65  
 typeface (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 typeface (rinoh.structure.TableOfContentsStyle attribute), 70  
 typeface (rinoh.text.TextStyle attribute), 41

## U

update() (rinoh.style.StyledMatcher method), 122  
 UserStrings (class in rinoh.strings), 120

## V

vertical\_align (rinoh.table.TableCellStyle attribute), 84

VerticalAlign (class in rinoh.table), 123

## W

WarnFlowable (class in rinoh.flowable), 99  
warning (rinoh.structure.AdmonitionTitles attribute), 119  
width (rinoh.flowable.FloatStyle attribute), 25  
width (rinoh.flowable.FlowableStyle attribute), 18  
width (rinoh.flowable.GroupedFlowablesStyle attribute), 22  
width (rinoh.flowable.LabeledFlowableStyle attribute), 20  
width (rinoh.image.CaptionStyle attribute), 27  
width (rinoh.image.FigureStyle attribute), 31  
width (rinoh.image.ImageArgs attribute), 104  
width (rinoh.index.IndexStyle attribute), 34  
width (rinoh.inline.InlineFlowableStyle attribute), 43  
width (rinoh.paragraph.ParagraphStyle attribute), 36  
width (rinoh.reference.ReferencingParagraphStyle attribute), 45  
width (rinoh.structure.AdmonitionStyle attribute), 77  
width (rinoh.structure.HeadingStyle attribute), 55  
width (rinoh.structure.HorizontalRuleStyle attribute), 79  
width (rinoh.structure.ListItemLabelStyle attribute), 62  
width (rinoh.structure.ListStyle attribute), 59  
width (rinoh.structure.SectionStyle attribute), 52  
width (rinoh.structure.TableOfContentsEntryStyle attribute), 72  
width (rinoh.structure.TableOfContentsStyle attribute), 67  
width (rinoh.table.TableCellStyle attribute), 84  
width (rinoh.table.TableStyle attribute), 82  
wrap\_label (rinoh.flowable.LabeledFlowableStyle attribute), 20

## Y

y\_offset() (rinoh.text.StyledText method), 113