

---

# rinoh<sup>type</sup>

## Reference Guide

Release 0.5.6



# 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
17	Element Style Classes
<b>91</b>	<b>Chapter 4: Standard Document Templates</b>
91	Article
92	Book
<b>97</b>	<b>Chapter 5: API Documentation</b>
97	Attribute ( <code>rinoh.attribute</code> )
98	Color ( <code>rinoh.color</code> )
98	Dimension ( <code>rinoh.dimension</code> )
99	Document ( <code>rinoh.document</code> )
101	Drawing Primitives ( <code>rinoh.draw</code> )
101	Flowable ( <code>rinoh.flowable</code> )
104	Fonts and Typefaces ( <code>rinoh.font</code> )
106	Images and Figures ( <code>rinoh.image</code> )
108	Index ( <code>rinoh.index</code> )
108	Language ( <code>rinoh.language</code> )
110	Number ( <code>rinoh.number</code> )
112	Paper sizes ( <code>rinoh.paper</code> )
113	Paragraph ( <code>rinoh.paragraph</code> )
116	Cross-References and Fields ( <code>rinoh.reference</code> )
123	Structure ( <code>rinoh.structure</code> )
142	Strings ( <code>rinoh.strings</code> )
142	Style ( <code>rinoh.style</code> )
144	Tables ( <code>rinoh.table</code> )

148	Templates ( <code>rinoh.template</code> )
159	Python Module Index
161	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]
             [--versions] [--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

**--versions**  
list versions of rinohtype and related software (for inclusion in bug reports)

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

`rinoh.frontend.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
- **inline definition:** 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
- **graphviz code block:** ParagraphStyle
- **attribution:** ParagraphStyle
- **centered:** ParagraphStyle
- **line block:** ParagraphStyle
- **block quote:** GroupedFlowablesStyle
- **chapter:** SectionStyle
- **content chapter:** SectionStyle
- **heading level 1:** HeadingStyle
- **heading level 1 label:** TextStyle
- **unnumbered heading level 1:** HeadingStyle
- **heading level 2:** HeadingStyle
- **heading level 2 label:** TextStyle
- **unnumbered heading level 2:** HeadingStyle
- **heading level 3:** HeadingStyle
- **heading level 3 label:** TextStyle
- **unnumbered heading level 3:** HeadingStyle
- **heading level 4:** HeadingStyle
- **heading level 4 label:** TextStyle

- unnumbered heading level 4: HeadingStyle
- heading level 5: HeadingStyle
- heading level 5 label: TextStyle
- 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 signature space: TextStyle
- object signature name: TextStyle
- object signature operator: TextStyle
- object signature punctuation: TextStyle
- object signature keyword: 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 title paragraph: ParagraphStyle
- admonition inline title: TextStyle
- attention admonition: AdmonitionStyle
- attention admonition title: ParagraphStyle
- attention admonition title paragraph: ParagraphStyle
- attention admonition inline title: TextStyle
- caution admonition: AdmonitionStyle
- caution admonition title: ParagraphStyle
- caution admonition title paragraph: ParagraphStyle
- caution admonition inline title: TextStyle
- danger admonition: AdmonitionStyle
- danger admonition title: ParagraphStyle
- danger admonition title paragraph: ParagraphStyle
- danger admonition inline title: TextStyle
- error admonition: AdmonitionStyle
- error admonition title: ParagraphStyle
- error admonition title paragraph: ParagraphStyle
- error admonition inline title: TextStyle
- hint admonition: AdmonitionStyle
- hint admonition title: ParagraphStyle
- hint admonition title paragraph: ParagraphStyle
- hint admonition inline title: TextStyle
- important admonition: AdmonitionStyle
- important admonition title: ParagraphStyle
- important admonition title paragraph: ParagraphStyle
- important admonition inline title: TextStyle
- note admonition: AdmonitionStyle
- note admonition title: ParagraphStyle
- note admonition title paragraph: ParagraphStyle
- note admonition inline title: TextStyle
- tip admonition: AdmonitionStyle
- tip admonition title: ParagraphStyle
- tip admonition title paragraph: ParagraphStyle
- tip admonition inline title: TextStyle
- warning admonition: AdmonitionStyle
- warning admonition title: ParagraphStyle
- warning admonition title paragraph: ParagraphStyle
- warning admonition inline title: TextStyle
- seealso admonition: AdmonitionStyle
- seealso admonition title: ParagraphStyle
- seealso admonition title paragraph: 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
- **image alt:** 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
- **toc level 4:** TableOfContentsEntryStyle
- **toc level 5:** TableOfContentsEntryStyle
- **toc level 6:** 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

**same\_page**

Keep all flowables on a single page, if possible.

*Accepts* Bool: true or false

*Default:* false

**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 current page, to a dedicated page, or sideways to a dedicated page

*Accepts FloatLocation:* none, here, sideways

*Default: None*

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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 current page, to a dedicated page, or sideways to a dedicated page (inherited from `FloatStyle`)

*Accepts* `FloatLocation`: none, here, sideways

*Default*: None

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

*Accepts Bool:* true or false

*Default:* false

### 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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

*Accepts* Bool: true or false

*Default:* false

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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

**no\_break\_after**

Prevent a line break after these words

*Accepts* NoBreakAfter: language-default, or a space-separated list of words

*Default:* language-default

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts NoBreakAfter:* language-default, or a space-separated list of words

*Default:* language-default

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**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`

**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: (no value) (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

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)  
*Accepts* `Bool`: true or false  
*Default:* false

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts NoBreakAfter:* language-default, or a space-separated list of words

*Default:* language-default

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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*

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)  
*Accepts Bool: true or false*  
*Default: false*

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

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

*Default*: `false`

**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)

### **abstract\_location**

Where to place the abstract

*Accepts* `AbstractLocation`: title, body

*Default*: title

### **parts**

The parts making up this document (inherited from `DocumentTemplate`)

*Accepts* `PartsList`: a space-separated list of document part template names

*Default*: `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 in a section named 'STRINGS'

*Default*: none

### **contents**

base: None

Overrides these defaults:

- `page_number_format = number`

Type

`ArticleContentsPartTemplate`

**page**

base: `None`

Overrides these defaults:

- `page_size = $(paper_size)`
- `left_margin = 1.2cm`
- `right_margin = 1.2cm`
- `top_margin = 1.8cm`
- `bottom_margin = 1.6cm`
- `header_footer_distance = 2pt`

Type

`ArticleBodyPageTemplate`

**contents\_page**

base: `page`

Type

`ArticleBodyPageTemplate`

**Configuration**

alias of `ArticleConfiguration`

**ConfigurationFile**

alias of `ArticleConfigurationFile`

**class** `rinoh.templates.article.AbstractLocation`

Where to place the article's abstract

Accepts: `title`, `body`

## 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 in a section named 'STRINGS'

*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

BodyPageTemplate

### **cover\_page**

base: page

Type

BodyPageTemplate

### **title\_page**

base: page

Type

TitlePageTemplate

### **front\_matter\_page**

base: page

Overrides these defaults:

- header\_footer\_distance = 0
- header\_text = None

Type

BodyPageTemplate

### **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 0x7f9bf643bda0>]
- sideways = right

Type

BodyPageTemplate

### **front\_matter\_left\_page**

base: front\_matter\_page

Overrides these defaults:

- footer\_text = '{PAGE\_NUMBER}'

- sideways = left

Type

BodyPageTemplate

### **contents\_page**

base: page

Overrides these defaults:

- header\_footer\_distance = 0

Type

BodyPageTemplate

### **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 0x7f9bfa40ac30>, <rinoh.paragraph.Paragraph object at 0x7f9bfa40acc0>]
- sideways = right

Type

BodyPageTemplate

### **contents\_left\_page**

base: contents\_page

Overrides these defaults:

- header\_text = '{DOCUMENT\_TITLE}' ', ' '{DOCUMENT\_SUBTITLE}'
- footer\_text = '{PAGE\_NUMBER}' '\t' '\t' '{@chapter}' ' ' '{SECTION\_NUMBER(1)}' '. ' '{SECTION\_TITLE(1)}'
- sideways = left

Type

BodyPageTemplate

### **back\_matter\_page**

base: page

Overrides these defaults:

- columns = 2
- header\_footer\_distance = 0

Type

BodyPageTemplate

### **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 0x7f9bf643bda0>]`
- `sideways = right`

Type

BodyPageTemplate

#### **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)}'`
- `sideways = left`

Type

BodyPageTemplate

#### **Configuration**

alias of `BookConfiguration`

#### **ConfigurationFile**

alias of `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 (grewed) 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, '/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**

user-defined string variables and can override localized strings provided by *language*

**backend=None**

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', display\_sideways=None*)

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

**display\_sideways : Sideways**

display the page rotated

**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 **LineStyle**

**class** rinoh.draw.**Shape** ( *style=None, parent=None* )

Base class for closed shapes

**style\_class**

alias of **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 `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 `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.

**style\_class**

alias of `FloatStyle`

**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\_metrics** ( *char*, *variant* )

Return the glyph metrics 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 metrics

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**  
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*, *alt=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*, *alt=None*, *baseline=None*, *id=None*, *style=None*, *parent=None*, *source=None* )

**arguments**

alias of `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*, *alt=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 `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 `CaptionStyle`

**class** `rinoh.image.Figure` ( *flowables*, *align=None*, *width=None*, *id=None*, *style=None*, *parent=None*, *source=None* )

**style\_class**

alias of `FigureStyle`

```

class rinoh.image.ListOfFiguresSection
    list_class
        alias of 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 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, **localized_strings )
    Collects localized strings for a particular language

Parameters
    code : str
        short code identifying the language
    name : str
        native name of the language
    paragraph
        label for referencing paragraphs
    section
        label for referencing sections
    chapter
        label for top-level sections

```

**figure**  
caption label for figures

**table**  
caption label for tables

**listing**  
caption label for (source code) listings

**contents**  
title for the table of contents section

**list\_of\_figures**  
title for the list of figures section

**list\_of\_tables**  
title for the list of tables section

**index**  
title for the index section

**attention**  
title for attention admonitions

**caution**  
title for caution admonitions

**danger**  
title for danger admonitions

**error**  
title for error admonitions

**hint**  
title for hint admonitions

**important**  
title for important admonitions

**note**  
title for note admonitions

**tip** title for tip admonitions

**warning**  
title for warning admonitions

**seealso**  
title for see-also admonitions

## languages

Dictionary mapping codes to Languages

The following languages are supported:

```
rinoh.language.CS = Language('cs', 'Česky')
    Localized strings for Česky :figure: Obrázek :table: Tabulka :contents: Obsah :list_of_fig-
    ures: Seznam obrázků :list_of_tables: Seznam tabulek :chapter: Kapitola :index: Rejstřík
    :attention: Pozor! :caution: Pozor! :danger: !NEBEZPEČÍ! :error: Chyba :hint: Poznámka
    :important: Důležité :note: Poznámka :tip: Tip :warning: Varování :seealso: Viz také

rinoh.language.EN = Language('en', 'English')
    Localized strings for English :paragraph: Paragraph :section: Section :chapter: Chapter :fig-
    ure: Figure :table: Table :listing: Listing :contents: Table of Contents :list_of_figures: List of
    Figures :list_of_tables: List of Tables :index: Index :attention: Attention! :caution: Caution!
    :danger: !DANGER! :error: Error :hint: Hint :important: Important :note: Note :tip: Tip
    :warning: Warning :seealso: See also
```

```

rinoh.language.ES = Language('es', 'Spanish')
    Localized strings for Spanish :figure: Figura :table: Tabla :contents: Contenidos :list_of_fig-
    ures: Índice de figuras :list_of_tables: Índice de tablas :chapter: Capítulo :index: Índice
    :attention: ¡Atención! :caution: ¡Cuidado! :danger: ¡PELIGRO! :error: Error :hint: Pista
    :important: Importante :note: Nota :tip: Consejo :warning: Precaución :seealso: Vea también

rinoh.language.FR = Language('fr', 'Français')
    Localized strings for Français :figure: Figure :table: Tableau :contents: Table des Matières
    :list_of_figures: Liste des Figures :list_of_tables: Liste des Tableaux :chapter: Chapitre
    :index: Index :attention: Attention! :caution: Prudence! :danger: !DANGER! :error: Erreur
    :hint: Conseil :important: Important :note: Note :tip: Astuce :warning: Avertissement
    :seealso: Voir aussi

rinoh.language.HU = Language('hu', 'Hungarian')
    Localized strings for Hungarian :contents: Tartalomjegyzék :list_of_figures: Ábrák Listája
    :list_of_tables: Asztalok Listája :chapter: Fejezet :index: Index :attention: Figyelem! :caution:
    Vigyázat!! :danger: !VESZÉLY! :error: Hiba :hint: Tanács :important: Fontos :note: Megje-
    gyzés :tip: Tipp :warning: Figyelmeztetés :seealso: Lásd még

rinoh.language.IT = Language('it', 'Italiano')
    Localized strings for Italiano :figure: Figura :table: Tabelle :contents: Contenuti :list_of_fig-
    ures: Elenco delle Figure :list_of_tables: Elenco delle Tabelle :chapter: Capitolo :index:
    Indice :attention: Attenzione! :caution: Prudenza! :danger: !PERICOLO! :error: Errore :hint:
    Consiglio :important: Importante :note: Nota :tip: Suggerimento :warning: Avvertimento
    :seealso: Vedi anche

rinoh.language.NL = Language('nl', 'Nederlands')
    Localized strings for Nederlands :paragraph: Paragraaf :section: Sectie :chapter: Hoofdstuk
    :figure: Figuur :table: Tabel :listing: Lijst :contents: Inhoudsopgave :list_of_figures: Lijst van
    Figuren :list_of_tables: Lijst van Tabellen :index: Index :attention: Opgelet! :caution: Pas op!
    :danger: !GEVAAR! :error: Fout :hint: Hint :important: Belangrijk :note: Noot :tip: Tip :warn-
    ing: Waarschuwing :seealso: Zie ook

rinoh.language.PL = Language('pl', 'Polski')
    Localized strings for Polski :figure: Ilustracja :table: Tabela :contents: Spis Treści :list_of_fig-
    ures: Spis Ilustracji :list_of_tables: Spis Tabel :chapter: Rozdział :index: Skorowidz :atten-
    tion: 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 deter-

mined 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 `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, language, span\_index=0, group\_index=0, 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 `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=None**  
see `Styled`

**parent=None**  
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

**class** `rinoh.text.NoBreakAfter` ( *iterable=()*, */* )  
List of words after which no line break will be allowed

### 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 `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
            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 ( metrics, width, char )

class rinoh.paragraph.GlyphsSpan ( span, chars_to_glyphs, glyphs= [] )
```

### 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.ReferenceStyle ( base=None, **attributes )
    Style class for Reference
    type
        How the reference should be displayed
```

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

**link**

Create a hyperlink to the reference target

*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

*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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)  
*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words  
*Default*: language-default

**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

**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

**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.ReferencingParagraph` ( *target\_id\_or\_flowable, id=None, style=None, parent=None* )

**style\_class**

alias of `ReferencingParagraphStyle`

**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 `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.NoteStyle ( base=None, **attributes )
```

Style class for Note

**location**

Where to place the note

*Accepts* NoteLocation: in-place, footer

*Default:* footer

**label\_min\_width**

Minimum label width (inherited from LabeledFlowableStyle)

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

*Default:* 12pt

**label\_max\_width**

Maximum label width (inherited from LabeledFlowableStyle)

*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 (inherited from LabeledFlowableStyle)

*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 (inherited from LabeledFlowableStyle)

*Accepts* Bool: true or false

*Default:* true

**wrap\_label**

Wrap the label at *label\_max\_width* (inherited from LabeledFlowableStyle)

*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*

```
class rinoh.reference.NoteMarkerBase ( custom_label=None, **kwargs )
```

**style\_class**

alias of `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

**content** : `StyledText`  
this heading's text

**style\_class**  
alias of `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.

### 5.15.2 Lists

**class** `rinoh.structure.List` ( *list\_items*, *start\_index=1*, *id=None*, *style=None*, *parent=None* )

**style\_class**  
alias of `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

#### **same\_page**

Keep all flowables on a single page, if possible. (inherited from GroupedFlowablesStyle)  
*Accepts* Bool: true or false  
*Default:* false

#### **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

```
class rinoh.structure.ListItem ( flowable, id=None, style=None, parent=None )
```

```
class rinoh.structure.ListItemLabel ( custom_label=None, align=None, width=None, id=None, style=None, parent=None, source=None )
```

**style\_class**

alias of `ListItemLabelStyle`

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

Style class for `ListItemLabel`

**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:* (no value) (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`

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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

```
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 `TableOfContentsStyle`

**flowables ( container )**

Generator yielding the `Flowables` to group

```
class rinoh.structure.TableOfContentsEntry ( flowable, id=None, style=None,
parent=None )
```

**style\_class**

alias of `TableOfContentsEntryStyle`

#### 5.15.4 List of ...

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

Style class for ListOf

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

*Accepts* `Bool`: true or false

*Default*: false

**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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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)

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

Style class for `ListOfEntry`

#### **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:* '{REFERENCE}' ': '{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

**no\_break\_after**

Prevent a line break after these words (inherited from `TextStyle`)

*Accepts* `NoBreakAfter`: language-default, or a space-separated list of words

*Default*: language-default

**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*

### **5.15.5 Admonitions**

```
class rinoh.structure.Admonition ( flowables, title=None, type=None, id=None,
style=None, parent=None )
```

#### **style\_class**

alias of `AdmonitionStyle`

#### **flowables ( container )**

Generator yielding the Flowables to group

### **5.15.6 Horizontal Rule**

```
class rinoh.structure.HorizontalRule ( align=None, width=None, id=None, style=None,
parent=None, source=None )
```

#### **style\_class**

alias of `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.Strings
```

Stores several `StringCollections`

```
class rinoh.strings.StringField ( key, style=None, parent=None, source=None,
user=False )
```

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.

**configuration\_class** = None

**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 mapping/iterable *E* and *F*.

If *E* is present and has a `.keys()` method, then does: for *k* in *E.keys()*: *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 `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 `TableWithCaptionStyle`

**class** `rinoh.table.TableWithCaptionStyle` ( *base=None*, *\*\*attributes* )

Style class for `TableWithCaption`

**float**

Float the flowable to the top or bottom of the current page, to a dedicated page, or sideways to a dedicated page (inherited from `FloatStyle`)

Accepts `FloatLocation`: none, here, sideways

Default: None

**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

**same\_page**

Keep all flowables on a single page, if possible. (inherited from `GroupedFlowablesStyle`)

*Accepts* Bool: true or false

*Default*: false

```
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 `TableCellStyle`

```
class rinoh.table.TableCellBorder ( rendered_cell, cell_height, position, style=None )
```

```
    style_class
```

alias of `TableCellBorderStyle`

```
class rinoh.table.TableCellBackground ( bottom_left, width, height, style=None,
parent=None )
```

```
    style_class
```

alias of `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 `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**=None

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 in a section named 'STRINGS'

*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 `DocumentTemplateConfiguration`

**ConfigurationFile**

alias of `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=None**

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 DocumentPartTemplate

**add\_page ( page )**

Append page (Page) to this DocumentPart.

**new\_page ( page\_number, chain, new\_chapter, sideways=False )**

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.BodyPageTemplateBase ( 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}'

**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

**sideways**

Rotate sideways pages to the left or right (inherited from `PageTemplateBase`)

*Accepts* `Sideways`: left, right

*Default*: right

```
class rinoh.template.BodyPageTemplate ( base=None, **attributes )
```

**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

#### **header\_footer\_distance**

Distance of the header and footer to the content area (inherited from `BodyPageTemplateBase`)  
*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 (inherited from `BodyPageTemplateBase`)  
*Accepts* `Integer`: a natural number (positive integer)  
*Default*: 1

#### **column\_spacing**

The spacing between columns (inherited from `BodyPageTemplateBase`)  
*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 (inherited from `BodyPageTemplateBase`)  
*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 (inherited from `BodyPageTemplateBase`)  
*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}'

#### **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

**sideways**

Rotate sideways pages to the left or right (inherited from `PageTemplateBase`)

*Accepts Sideways:* left, right

*Default:* right

**class** `rinoh.template.TitlePageTemplate` (*base=None, \*\*attributes*)

**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

**sideways**

Rotate sideways pages to the left or right (inherited from `PageTemplateBase`)

*Accepts* `Sideways`: left, right

*Default*: right

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

**sideways**

Rotate sideways pages to the left or right

*Accepts* Sideways: left, right

*Default:* right



# Python Module Index

## r

### rhino

- rhino.attribute, 97
- rhino.color, 98
- rhino.dimension, 98
- rhino.document, 99
- rhino.draw, 101
- rhino.flowable, 101
- rhino.font, 104
- rhino.font.style, 106
- rhino.image, 106
- rhino.index, 108
- rhino.inline, 114
- rhino.language, 108
- rhino.number, 110
- rhino.paper, 112
- rhino.paragraph, 113
- rhino.reference, 116
- rhino.strings, 142
- rhino.structure, 123
- rhino.style, 142
- rhino.stylesheets, 11
- rhino.table, 144
- rhino.template, 148
- rhino.templates.article, 91
- rhino.templates.book, 92
- rhino.text, 113





# Index

## Symbols

- docs
  - rinoh command line option, 6
- format
  - rinoh command line option, 5
- help
  - rinoh command line option, 5
- install-resources
  - rinoh command line option, 5
- list-fonts
  - rinoh command line option, 6
- list-formats
  - rinoh command line option, 6
- list-options
  - rinoh command line option, 6
- list-stylesheets
  - rinoh command line option, 5
- list-templates
  - rinoh command line option, 5
- option
  - rinoh command line option, 5
- output
  - rinoh command line option, 5
- paper
  - rinoh command line option, 5
- stylesheet
  - rinoh command line option, 5
- template
  - rinoh command line option, 5
- version
  - rinoh command line option, 6
- versions
  - rinoh command line option, 6
- f
  - rinoh command line option, 5
- h
  - rinoh command line option, 5
- i
  - rinoh command line option, 5
- O
  - rinoh command line option, 5
- o

- rinoh command line option, 5
- P
  - rinoh command line option, 5
- s
  - rinoh command line option, 5
- t
  - rinoh command line option, 5

## A

- A0 (in module rinoh.paper), 112
- A1 (in module rinoh.paper), 112
- A10 (in module rinoh.paper), 112
- A2 (in module rinoh.paper), 112
- A3 (in module rinoh.paper), 112
- A4 (in module rinoh.paper), 112
- A5 (in module rinoh.paper), 112
- A6 (in module rinoh.paper), 112
- A7 (in module rinoh.paper), 112
- A8 (in module rinoh.paper), 112
- A9 (in module rinoh.paper), 112
- abstract\_location (rinoh.templates.article.Article attribute), 91
- AbstractLocation (class in rinoh.templates.article), 92
- AcceptNoneAttributeType (class in rinoh.attribute), 97
- add\_page() (rinoh.template.DocumentPart method), 149
- Admonition (class in rinoh.structure), 142
- AdmonitionStyle (class in rinoh.structure), 79
- advance() (rinoh.paragraph.DefaultSpacing method), 115
- advance() (rinoh.paragraph.FixedSpacing method), 115
- advance() (rinoh.paragraph.Leading method), 115
- advance() (rinoh.paragraph.LineSpacing method), 115
- advance() (rinoh.paragraph.ProportionalSpacing method), 115
- after (rinoh.image.CaptionStyle attribute), 32
- after (rinoh.inline.InlineFlowableStyle attribute), 46
- after (rinoh.paragraph.ParagraphStyle attribute), 42
- after (rinoh.reference.NoteMarkerStyle attribute), 53
- after (rinoh.reference.ReferenceStyle attribute), 118
- after (rinoh.reference.ReferencingParagraphStyle attribute), 51
- after (rinoh.structure.HeadingStyle attribute), 61
- after (rinoh.structure.ListItemLabelStyle attribute), 69, 131
- after (rinoh.structure.ListOfEntryStyle attribute), 141
- after (rinoh.structure.ListOfStyle attribute), 137
- after (rinoh.structure.TableOfContentsEntryStyle attribute), 79
- after (rinoh.structure.TableOfContentsStyle attribute), 74
- after (rinoh.text.TextStyle attribute), 43
- after\_break\_background (rinoh.template.BodyPageTemplate attribute), 155
- after\_break\_background (rinoh.template.BodyPageTemplateBase attribute), 153
- after\_break\_background (rinoh.template.PageTemplateBase attribute), 157
- after\_break\_background (rinoh.template.TitlePageTemplate attribute), 156
- align (rinoh.image.ImageArgs attribute), 107
- align\_baselines (rinoh.flowable.LabeledFlowableStyle attribute), 21
- align\_baselines (rinoh.reference.NoteStyle attribute), 120
- AnchorFlowable (class in rinoh.flowable), 102
- append() (rinoh.text.MixedStyledText method), 114
- arguments (rinoh.image.BackgroundImage attribute), 106
- arguments (rinoh.image.InlineImage attribute), 106
- Article (class in rinoh.templates.article), 91
- Attribute (class in rinoh.attribute), 97

AttributesDictionary (class in rinoh.attribute), 97

## B

back\_matter (rinoh.templates.book.Book attribute), 93  
back\_matter\_left\_page (rinoh.templates.book.Book attribute), 96  
back\_matter\_page (rinoh.templates.book.Book attribute), 95  
back\_matter\_right\_page (rinoh.templates.book.Book attribute), 95  
background (rinoh.template.BodyPageTemplate attribute), 155  
background (rinoh.template.BodyPageTemplateBase attribute), 153  
background (rinoh.template.PageTemplateBase attribute), 157  
background (rinoh.template.TitlePageTemplate attribute), 156  
background\_color (rinoh.flowable.FloatStyle attribute), 27  
background\_color (rinoh.flowable.FlowableStyle attribute), 20  
background\_color (rinoh.flowable.GroupedFlowablesStyle attribute), 25  
background\_color (rinoh.flowable.LabeledFlowableStyle attribute), 22  
background\_color (rinoh.image.CaptionStyle attribute), 30  
background\_color (rinoh.image.FigureStyle attribute), 34  
background\_color (rinoh.index.IndexStyle attribute), 37  
background\_color (rinoh.inline.InlineFlowableStyle attribute), 46  
background\_color (rinoh.paragraph.ParagraphStyle attribute), 40  
background\_color (rinoh.reference.NoteStyle attribute), 122  
background\_color (rinoh.reference.ReferencingParagraphStyle attribute), 49  
background\_color (rinoh.structure.AdmonitionStyle attribute), 81  
background\_color (rinoh.structure.HeadingStyle attribute), 59  
background\_color (rinoh.structure.HorizontalRuleStyle attribute), 83  
background\_color (rinoh.structure.ListItemLabelStyle attribute), 67, 129  
background\_color (rinoh.structure.ListOfEntryStyle attribute), 139  
background\_color (rinoh.structure.ListOfStyle attribute), 134  
background\_color (rinoh.structure.ListStyle attribute), 63, 125  
background\_color (rinoh.structure.SectionStyle attribute), 56  
background\_color (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
background\_color (rinoh.structure.TableOfContentsStyle attribute), 71  
background\_color (rinoh.table.TableCellStyle attribute), 88  
background\_color (rinoh.table.TableStyle attribute), 86  
background\_color (rinoh.table.TableWithCaptionStyle attribute), 146  
BackgroundImage (class in rinoh.image), 106  
baseline (rinoh.inline.InlineFlowableStyle attribute), 44  
before (rinoh.image.CaptionStyle attribute), 32  
before (rinoh.inline.InlineFlowableStyle attribute), 46  
before (rinoh.paragraph.ParagraphStyle attribute), 42  
before (rinoh.reference.NoteMarkerStyle attribute), 53  
before (rinoh.reference.ReferenceStyle attribute), 118  
before (rinoh.reference.ReferencingParagraphStyle attribute), 51  
before (rinoh.structure.HeadingStyle attribute), 61  
before (rinoh.structure.ListItemLabelStyle attribute), 68, 131  
before (rinoh.structure.ListOfEntryStyle attribute), 141  
before (rinoh.structure.ListOfStyle attribute), 136  
before (rinoh.structure.TableOfContentsEntryStyle attribute), 79  
before (rinoh.structure.TableOfContentsStyle attribute), 74  
before (rinoh.text.TextStyle attribute), 43  
BodyPageTemplate (class in rinoh.template), 153  
BodyPageTemplateBase (class in rinoh.template), 152  
Book (class in rinoh.templates.book), 92  
Bool (class in rinoh.attribute), 97  
border (rinoh.flowable.FloatStyle attribute), 26  
border (rinoh.flowable.FlowableStyle attribute), 19  
border (rinoh.flowable.GroupedFlowablesStyle attribute), 24

border (rinoh.flowable.LabeledFlowableStyle attribute), 22  
border (rinoh.image.CaptionStyle attribute), 29  
border (rinoh.image.FigureStyle attribute), 33  
border (rinoh.index.IndexStyle attribute), 36  
border (rinoh.inline.InlineFlowableStyle attribute), 45  
border (rinoh.paragraph.ParagraphStyle attribute), 39  
border (rinoh.reference.NoteStyle attribute), 121  
border (rinoh.reference.ReferencingParagraphStyle attribute), 48  
border (rinoh.structure.AdmonitionStyle attribute), 81  
border (rinoh.structure.HeadingStyle attribute), 58  
border (rinoh.structure.HorizontalRuleStyle attribute), 83  
border (rinoh.structure.ListItemLabelStyle attribute), 66, 128  
border (rinoh.structure.ListOfEntryStyle attribute), 139  
border (rinoh.structure.ListOfStyle attribute), 133  
border (rinoh.structure.ListStyle attribute), 63, 125  
border (rinoh.structure.SectionStyle attribute), 55  
border (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
border (rinoh.structure.TableOfContentsStyle attribute), 70  
border (rinoh.table.TableCellStyle attribute), 88  
border (rinoh.table.TableStyle attribute), 85  
border (rinoh.table.TableWithCaptionStyle attribute), 146  
border\_bottom (rinoh.flowable.FloatStyle attribute), 27  
border\_bottom (rinoh.flowable.FlowableStyle attribute), 20  
border\_bottom (rinoh.flowable.GroupedFlowablesStyle attribute), 25  
border\_bottom (rinoh.flowable.LabeledFlowableStyle attribute), 22  
border\_bottom (rinoh.image.CaptionStyle attribute), 30  
border\_bottom (rinoh.image.FigureStyle attribute), 34  
border\_bottom (rinoh.index.IndexStyle attribute), 37  
border\_bottom (rinoh.inline.InlineFlowableStyle attribute), 45  
border\_bottom (rinoh.paragraph.ParagraphStyle attribute), 39  
border\_bottom (rinoh.reference.NoteStyle attribute), 122  
border\_bottom (rinoh.reference.ReferencingParagraphStyle attribute), 49  
border\_bottom (rinoh.structure.AdmonitionStyle attribute), 81  
border\_bottom (rinoh.structure.HeadingStyle attribute), 59  
border\_bottom (rinoh.structure.HorizontalRuleStyle attribute), 83  
border\_bottom (rinoh.structure.ListItemLabelStyle attribute), 66, 129  
border\_bottom (rinoh.structure.ListOfEntryStyle attribute), 139  
border\_bottom (rinoh.structure.ListOfStyle attribute), 133  
border\_bottom (rinoh.structure.ListStyle attribute), 63, 125  
border\_bottom (rinoh.structure.SectionStyle attribute), 56  
border\_bottom (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
border\_bottom (rinoh.structure.TableOfContentsStyle attribute), 71  
border\_bottom (rinoh.table.TableCellStyle attribute), 88  
border\_bottom (rinoh.table.TableStyle attribute), 86  
border\_bottom (rinoh.table.TableWithCaptionStyle attribute), 146  
border\_left (rinoh.flowable.FloatStyle attribute), 26  
border\_left (rinoh.flowable.FlowableStyle attribute), 20  
border\_left (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
border\_left (rinoh.flowable.LabeledFlowableStyle attribute), 22  
border\_left (rinoh.image.CaptionStyle attribute), 29  
border\_left (rinoh.image.FigureStyle attribute), 33  
border\_left (rinoh.index.IndexStyle attribute), 36  
border\_left (rinoh.inline.InlineFlowableStyle attribute), 45  
border\_left (rinoh.paragraph.ParagraphStyle attribute), 39  
border\_left (rinoh.reference.NoteStyle attribute), 121  
border\_left (rinoh.reference.ReferencingParagraphStyle attribute), 48  
border\_left (rinoh.structure.AdmonitionStyle attribute), 81  
border\_left (rinoh.structure.HeadingStyle attribute), 58

border\_left (rinoh.structure.HorizontalRuleStyle attribute), 83  
border\_left (rinoh.structure.ListItemLabelStyle attribute), 66, 128  
border\_left (rinoh.structure.ListOfEntryStyle attribute), 139  
border\_left (rinoh.structure.ListOfStyle attribute), 133  
border\_left (rinoh.structure.ListStyle attribute), 63, 125  
border\_left (rinoh.structure.SectionStyle attribute), 55  
border\_left (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
border\_left (rinoh.structure.TableOfContentsStyle attribute), 71  
border\_left (rinoh.table.TableCellStyle attribute), 88  
border\_left (rinoh.table.TableStyle attribute), 85  
border\_left (rinoh.table.TableWithCaptionStyle attribute), 146  
border\_right (rinoh.flowable.FloatStyle attribute), 27  
border\_right (rinoh.flowable.FlowableStyle attribute), 20  
border\_right (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
border\_right (rinoh.flowable.LabeledFlowableStyle attribute), 22  
border\_right (rinoh.image.CaptionStyle attribute), 29  
border\_right (rinoh.image.FigureStyle attribute), 34  
border\_right (rinoh.index.IndexStyle attribute), 36  
border\_right (rinoh.inline.InlineFlowableStyle attribute), 45  
border\_right (rinoh.paragraph.ParagraphStyle attribute), 39  
border\_right (rinoh.reference.NoteStyle attribute), 121  
border\_right (rinoh.reference.ReferencingParagraphStyle attribute), 48  
border\_right (rinoh.structure.AdmonitionStyle attribute), 81  
border\_right (rinoh.structure.HeadingStyle attribute), 58  
border\_right (rinoh.structure.HorizontalRuleStyle attribute), 83  
border\_right (rinoh.structure.ListItemLabelStyle attribute), 66, 128  
border\_right (rinoh.structure.ListOfEntryStyle attribute), 139  
border\_right (rinoh.structure.ListOfStyle attribute), 133  
border\_right (rinoh.structure.ListStyle attribute), 63, 125  
border\_right (rinoh.structure.SectionStyle attribute), 55  
border\_right (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
border\_right (rinoh.structure.TableOfContentsStyle attribute), 71  
border\_right (rinoh.table.TableCellStyle attribute), 88  
border\_right (rinoh.table.TableStyle attribute), 85  
border\_right (rinoh.table.TableWithCaptionStyle attribute), 146  
border\_top (rinoh.flowable.FloatStyle attribute), 27  
border\_top (rinoh.flowable.FlowableStyle attribute), 20  
border\_top (rinoh.flowable.GroupedFlowablesStyle attribute), 25  
border\_top (rinoh.flowable.LabeledFlowableStyle attribute), 22  
border\_top (rinoh.image.CaptionStyle attribute), 30  
border\_top (rinoh.image.FigureStyle attribute), 34  
border\_top (rinoh.index.IndexStyle attribute), 36  
border\_top (rinoh.inline.InlineFlowableStyle attribute), 45  
border\_top (rinoh.paragraph.ParagraphStyle attribute), 39  
border\_top (rinoh.reference.NoteStyle attribute), 122  
border\_top (rinoh.reference.ReferencingParagraphStyle attribute), 49  
border\_top (rinoh.structure.AdmonitionStyle attribute), 81  
border\_top (rinoh.structure.HeadingStyle attribute), 58  
border\_top (rinoh.structure.HorizontalRuleStyle attribute), 83  
border\_top (rinoh.structure.ListItemLabelStyle attribute), 66, 129  
border\_top (rinoh.structure.ListOfEntryStyle attribute), 139  
border\_top (rinoh.structure.ListOfStyle attribute), 133  
border\_top (rinoh.structure.ListStyle attribute), 63, 125  
border\_top (rinoh.structure.SectionStyle attribute), 56  
border\_top (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
border\_top (rinoh.structure.TableOfContentsStyle attribute), 71  
border\_top (rinoh.table.TableCellStyle attribute), 88  
border\_top (rinoh.table.TableStyle attribute), 85

- border\_top (rinoh.table.TableWithCaptionStyle attribute), 146
- bottom\_margin (rinoh.template.BodyPageTemplate attribute), 155
- bottom\_margin (rinoh.template.BodyPageTemplateBase attribute), 153
- bottom\_margin (rinoh.template.PageTemplateBase attribute), 157
- bottom\_margin (rinoh.template.TitlePageTemplate attribute), 156
- build\_document() (rinoh.flowable.StaticGroupedFlowables method), 103
- bullet (rinoh.structure.ListStyle attribute), 61, 123

## C

Caption (class in rinoh.image), 107  
 CaptionStyle (class in rinoh.image), 27  
 chapter\_footer\_text (rinoh.template.BodyPageTemplate attribute), 153  
 chapter\_header\_text (rinoh.template.BodyPageTemplate attribute), 153  
 chapter\_title\_flowables (rinoh.template.BodyPageTemplate attribute), 154  
 chapter\_title\_height (rinoh.template.BodyPageTemplate attribute), 154  
 CM (in module rinoh.dimension), 99  
 Color (class in rinoh.color), 98  
 column\_spacing (rinoh.template.BodyPageTemplate attribute), 154  
 column\_spacing (rinoh.template.BodyPageTemplateBase attribute), 152  
 column\_widths (rinoh.table.TableStyle attribute), 84  
 columns (rinoh.template.BodyPageTemplate attribute), 154  
 columns (rinoh.template.BodyPageTemplateBase attribute), 152  
 Configuration (rinoh.template.DocumentTemplate attribute), 148  
 Configuration (rinoh.templates.article.Article attribute), 92  
 Configuration (rinoh.templates.book.Book attribute), 96  
 configuration\_class (rinoh.style.Styled attribute), 143  
 configuration\_class (rinoh.template.DocumentPart attribute), 149  
 ConfigurationFile (rinoh.template.DocumentTemplate attribute), 148  
 ConfigurationFile (rinoh.templates.article.Article attribute), 92  
 ConfigurationFile (rinoh.templates.book.Book attribute), 96  
 contents (rinoh.templates.article.Article attribute), 91  
 contents (rinoh.templates.book.Book attribute), 93  
 contents\_left\_page (rinoh.templates.book.Book attribute), 95  
 contents\_page (rinoh.templates.article.Article attribute), 92  
 contents\_page (rinoh.templates.book.Book attribute), 95  
 contents\_right\_page (rinoh.templates.book.Book attribute), 95  
 ContentsPartTemplate (class in rinoh.template), 151  
 copy() (rinoh.text.MixedStyledText method), 114  
 cover (rinoh.templates.book.Book attribute), 93  
 cover\_page (rinoh.templates.book.Book attribute), 94  
 CS (in module rinoh.language), 109  
 custom\_label (rinoh.image.CaptionStyle attribute), 30  
 custom\_label (rinoh.number.LabelStyle attribute), 111  
 custom\_label (rinoh.number.NumberStyle attribute), 111  
 custom\_label (rinoh.paragraph.ParagraphStyle attribute), 40  
 custom\_label (rinoh.reference.NoteMarkerStyle attribute), 53  
 custom\_label (rinoh.reference.ReferenceStyle attribute), 119  
 custom\_label (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 custom\_label (rinoh.structure.HeadingStyle attribute), 59  
 custom\_label (rinoh.structure.ListItemLabelStyle attribute), 67, 129  
 custom\_label (rinoh.structure.ListOfEntryStyle attribute), 140  
 custom\_label (rinoh.structure.ListOfStyle attribute), 135  
 custom\_label (rinoh.structure.ListStyle attribute), 64, 126  
 custom\_label (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
 custom\_label (rinoh.structure.TableOfContentsStyle attribute), 72

## D

DEFAULT (in module rinoh.paragraph), 116  
DefaultSpacing (class in rinoh.paragraph), 115  
DefinitionList (class in rinoh.structure), 131  
depth (rinoh.structure.TableOfContentsStyle attribute), 69  
Dimension (class in rinoh.dimension), 98  
doc\_format() (rinoh.color.Color class method), 98  
Document (class in rinoh.document), 99  
document() (rinoh.template.TemplateConfiguration method), 149  
DocumentPart (class in rinoh.template), 149  
DocumentPartTemplate (class in rinoh.template), 150  
DocumentTemplate (class in rinoh.template), 148  
DocumentTree (class in rinoh.document), 99  
DOUBLE (in module rinoh.paragraph), 116  
dpi (rinoh.image.ImageArgs attribute), 107  
drop\_if\_empty (rinoh.template.ContentsPartTemplate attribute), 151  
drop\_if\_empty (rinoh.template.DocumentPartTemplate attribute), 150  
drop\_if\_empty (rinoh.template.FixedDocumentPartTemplate attribute), 152  
drop\_if\_empty (rinoh.template.TitlePartTemplate attribute), 150  
DummyFlowable (class in rinoh.flowable), 102

## E

EN (in module rinoh.language), 109  
encoding (rinoh.font.Font attribute), 104  
end\_at\_page (rinoh.template.ContentsPartTemplate attribute), 151  
end\_at\_page (rinoh.template.DocumentPartTemplate attribute), 150  
end\_at\_page (rinoh.template.FixedDocumentPartTemplate attribute), 152  
end\_at\_page (rinoh.template.TitlePartTemplate attribute), 151  
ES (in module rinoh.language), 110  
extra (rinoh.template.TitlePageTemplate attribute), 155

## F

Field (class in rinoh.reference), 122  
Figure (class in rinoh.image), 107  
FigureStyle (class in rinoh.image), 32  
file\_or\_filename (rinoh.image.ImageArgs attribute), 107  
fill\_color (rinoh.draw.ShapeStyle attribute), 18  
fill\_color (rinoh.table.TableCellBackgroundStyle attribute), 89  
find\_source() (rinoh.attribute.RuleSet method), 97  
FixedDocumentPartTemplate (class in rinoh.template), 151  
FixedSpacing (class in rinoh.paragraph), 115  
Float (class in rinoh.flowable), 103  
float (rinoh.flowable.FloatStyle attribute), 25  
float (rinoh.image.FigureStyle attribute), 32  
float (rinoh.table.TableWithCaptionStyle attribute), 144  
FloatStyle (class in rinoh.flowable), 25  
flow() (rinoh.flowable.Float method), 104  
flow() (rinoh.index.IndexTarget method), 108  
flow() (rinoh.reference.Note method), 119  
flow() (rinoh.structure.Heading method), 123  
Flowable (class in rinoh.flowable), 101  
flowable\_spacing (rinoh.flowable.GroupedFlowablesStyle attribute), 23  
flowable\_spacing (rinoh.image.FigureStyle attribute), 34  
flowable\_spacing (rinoh.index.IndexStyle attribute), 35  
flowable\_spacing (rinoh.structure.AdmonitionStyle attribute), 79  
flowable\_spacing (rinoh.structure.ListOfStyle attribute), 132

flowable\_spacing (rinoh.structure.ListStyle attribute), 62, 124  
 flowable\_spacing (rinoh.structure.SectionStyle attribute), 54  
 flowable\_spacing (rinoh.structure.TableOfContentsStyle attribute), 69  
 flowable\_spacing (rinoh.table.TableCellStyle attribute), 86  
 flowable\_spacing (rinoh.table.TableWithCaptionStyle attribute), 147  
 flowables (rinoh.template.FixedDocumentPartTemplate attribute), 151  
 flowables() (rinoh.flowable.GroupedFlowables method), 103  
 flowables() (rinoh.flowable.StaticGroupedFlowables method), 103  
 flowables() (rinoh.index.Index method), 108  
 flowables() (rinoh.structure.Admonition method), 142  
 flowables() (rinoh.structure.TableOfContents method), 131  
 FlowableState (class in rinoh.flowable), 101  
 FlowableStyle (class in rinoh.flowable), 18  
 FlowableWidth (class in rinoh.flowable), 104  
 Font (class in rinoh.font), 104  
 font\_color (rinoh.image.CaptionStyle attribute), 31  
 font\_color (rinoh.paragraph.ParagraphStyle attribute), 41  
 font\_color (rinoh.reference.NoteMarkerStyle attribute), 52  
 font\_color (rinoh.reference.ReferenceStyle attribute), 117  
 font\_color (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 font\_color (rinoh.structure.HeadingStyle attribute), 60  
 font\_color (rinoh.structure.ListItemLabelStyle attribute), 68, 130  
 font\_color (rinoh.structure.ListOfEntryStyle attribute), 140  
 font\_color (rinoh.structure.ListOfStyle attribute), 136  
 font\_color (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 font\_color (rinoh.structure.TableOfContentsStyle attribute), 73  
 font\_color (rinoh.text.TextStyle attribute), 43  
 font\_size (rinoh.image.CaptionStyle attribute), 31  
 font\_size (rinoh.paragraph.ParagraphStyle attribute), 41  
 font\_size (rinoh.reference.NoteMarkerStyle attribute), 52  
 font\_size (rinoh.reference.ReferenceStyle attribute), 117  
 font\_size (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 font\_size (rinoh.structure.HeadingStyle attribute), 60  
 font\_size (rinoh.structure.ListItemLabelStyle attribute), 68, 130  
 font\_size (rinoh.structure.ListOfEntryStyle attribute), 140  
 font\_size (rinoh.structure.ListOfStyle attribute), 136  
 font\_size (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 font\_size (rinoh.structure.TableOfContentsStyle attribute), 73  
 font\_size (rinoh.text.TextStyle attribute), 42  
 font\_slant (rinoh.image.CaptionStyle attribute), 31  
 font\_slant (rinoh.paragraph.ParagraphStyle attribute), 40  
 font\_slant (rinoh.reference.NoteMarkerStyle attribute), 52  
 font\_slant (rinoh.reference.ReferenceStyle attribute), 117  
 font\_slant (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 font\_slant (rinoh.structure.HeadingStyle attribute), 59  
 font\_slant (rinoh.structure.ListItemLabelStyle attribute), 67, 130  
 font\_slant (rinoh.structure.ListOfEntryStyle attribute), 140  
 font\_slant (rinoh.structure.ListOfStyle attribute), 135  
 font\_slant (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 font\_slant (rinoh.structure.TableOfContentsStyle attribute), 73  
 font\_slant (rinoh.text.TextStyle attribute), 42  
 font\_variant (rinoh.image.CaptionStyle attribute), 31  
 font\_variant (rinoh.paragraph.ParagraphStyle attribute), 41  
 font\_variant (rinoh.reference.NoteMarkerStyle attribute), 52  
 font\_variant (rinoh.reference.ReferenceStyle attribute), 117  
 font\_variant (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 font\_variant (rinoh.structure.HeadingStyle attribute), 60  
 font\_variant (rinoh.structure.ListItemLabelStyle attribute), 68, 130



font\_variant (rinoh.structure.ListOfEntryStyle attribute), 141  
 font\_variant (rinoh.structure.ListOfStyle attribute), 136  
 font\_variant (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 font\_variant (rinoh.structure.TableOfContentsStyle attribute), 73  
 font\_variant (rinoh.text.TextStyle attribute), 43  
 font\_weight (rinoh.image.CaptionStyle attribute), 30  
 font\_weight (rinoh.paragraph.ParagraphStyle attribute), 40  
 font\_weight (rinoh.reference.NoteMarkerStyle attribute), 52  
 font\_weight (rinoh.reference.ReferenceStyle attribute), 117  
 font\_weight (rinoh.reference.ReferencingParagraphStyle attribute), 49  
 font\_weight (rinoh.structure.HeadingStyle attribute), 59  
 font\_weight (rinoh.structure.ListItemLabelStyle attribute), 67, 129  
 font\_weight (rinoh.structure.ListOfEntryStyle attribute), 140  
 font\_weight (rinoh.structure.ListOfStyle attribute), 135  
 font\_weight (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
 font\_weight (rinoh.structure.TableOfContentsStyle attribute), 73  
 font\_weight (rinoh.text.TextStyle attribute), 42  
 font\_width (rinoh.image.CaptionStyle attribute), 31  
 font\_width (rinoh.paragraph.ParagraphStyle attribute), 41  
 font\_width (rinoh.reference.NoteMarkerStyle attribute), 52  
 font\_width (rinoh.reference.ReferenceStyle attribute), 117  
 font\_width (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 font\_width (rinoh.structure.HeadingStyle attribute), 60  
 font\_width (rinoh.structure.ListItemLabelStyle attribute), 67, 130  
 font\_width (rinoh.structure.ListOfEntryStyle attribute), 140  
 font\_width (rinoh.structure.ListOfStyle attribute), 135  
 font\_width (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 font\_width (rinoh.structure.TableOfContentsStyle attribute), 73  
 font\_width (rinoh.text.TextStyle attribute), 42  
 fonts() (rinoh.font.Typeface method), 105  
 FontSlant (class in rinoh.font.style), 106  
 FontVariant (class in rinoh.font.style), 106  
 FontWeight (class in rinoh.font.style), 106  
 FontWidth (class in rinoh.font.style), 106  
 footer\_text (rinoh.template.BodyPageTemplate attribute), 154  
 footer\_text (rinoh.template.BodyPageTemplateBase attribute), 152  
 format\_number() (in module rinoh.number), 112  
 FR (in module rinoh.language), 110  
 front\_matter (rinoh.templates.book.Book attribute), 93  
 front\_matter\_left\_page (rinoh.templates.book.Book attribute), 94  
 front\_matter\_page (rinoh.templates.book.Book attribute), 94  
 front\_matter\_right\_page (rinoh.templates.book.Book attribute), 94

## G

get\_font() (rinoh.font.Typeface method), 105  
 get\_glyph\_metrics() (rinoh.font.Font method), 104  
 get\_kerning() (rinoh.font.Font method), 105  
 get\_ligature() (rinoh.font.Font method), 105  
 get\_position() (rinoh.paragraph.TabStop method), 116  
 get\_rowspanned\_columns() (rinoh.table.TableRow method), 147  
 get\_selector() (rinoh.style.StyleSheet method), 143  
 Glyph (class in rinoh.paragraph), 116  
 GlyphsSpan (class in rinoh.paragraph), 116  
 Gray (class in rinoh.color), 98  
 GroupedFlowables (class in rinoh.flowable), 103  
 GroupedFlowablesState (class in rinoh.flowable), 103  
 GroupedFlowablesStyle (class in rinoh.flowable), 23

GroupedLabeledFlowables (class in rinoh.flowable), 103  
grow() (rinoh.dimension.Dimension method), 99

## H

has\_class (rinoh.style.Styled property), 143  
has\_classes (rinoh.style.Styled property), 143  
has\_id (rinoh.style.Styled property), 143  
header\_footer\_distance (rinoh.template.BodyPageTemplate attribute), 154  
header\_footer\_distance (rinoh.template.BodyPageTemplateBase attribute), 152  
header\_text (rinoh.template.BodyPageTemplate attribute), 154  
header\_text (rinoh.template.BodyPageTemplateBase attribute), 152  
Heading (class in rinoh.structure), 123  
HeadingStyle (class in rinoh.structure), 56  
height (rinoh.image.ImageArgs attribute), 107  
height() (rinoh.text.StyledText method), 113  
HexColor (class in rinoh.color), 98  
hide (rinoh.draw.LineStyle attribute), 18  
hide (rinoh.draw.ShapeStyle attribute), 18  
hide (rinoh.flowable.FloatStyle attribute), 27  
hide (rinoh.flowable.FlowableStyle attribute), 20  
hide (rinoh.flowable.GroupedFlowablesStyle attribute), 25  
hide (rinoh.flowable.LabeledFlowableStyle attribute), 23  
hide (rinoh.image.CaptionStyle attribute), 32  
hide (rinoh.image.FigureStyle attribute), 34  
hide (rinoh.index.IndexStyle attribute), 37  
hide (rinoh.inline.InlineFlowableStyle attribute), 46  
hide (rinoh.number.LabelStyle attribute), 111  
hide (rinoh.number.NumberStyle attribute), 112  
hide (rinoh.paragraph.ParagraphStyle attribute), 40  
hide (rinoh.reference.NoteMarkerStyle attribute), 54  
hide (rinoh.reference.NoteStyle attribute), 122  
hide (rinoh.reference.ReferenceStyle attribute), 118  
hide (rinoh.reference.ReferencingParagraphStyle attribute), 51  
hide (rinoh.structure.AdmonitionStyle attribute), 81  
hide (rinoh.structure.HeadingStyle attribute), 61  
hide (rinoh.structure.HorizontalRuleStyle attribute), 83  
hide (rinoh.structure.ListItemLabelStyle attribute), 69, 131  
hide (rinoh.structure.ListOfEntryStyle attribute), 142  
hide (rinoh.structure.ListOfStyle attribute), 134  
hide (rinoh.structure.ListStyle attribute), 63, 126  
hide (rinoh.structure.SectionStyle attribute), 56  
hide (rinoh.structure.TableOfContentsEntryStyle attribute), 79  
hide (rinoh.structure.TableOfContentsStyle attribute), 71  
hide (rinoh.style.Style attribute), 143  
hide (rinoh.table.TableCellBackgroundStyle attribute), 89  
hide (rinoh.table.TableCellBorderStyle attribute), 89  
hide (rinoh.table.TableCellStyle attribute), 88  
hide (rinoh.table.TableStyle attribute), 86  
hide (rinoh.table.TableWithCaptionStyle attribute), 146  
hide (rinoh.text.TextStyle attribute), 44  
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), 21  
horizontal\_align (rinoh.image.CaptionStyle attribute), 28  
horizontal\_align (rinoh.image.FigureStyle attribute), 32  
horizontal\_align (rinoh.index.IndexStyle attribute), 35

horizontal\_align (rinoh.inline.InlineFlowableStyle attribute), 44  
 horizontal\_align (rinoh.paragraph.ParagraphStyle attribute), 38  
 horizontal\_align (rinoh.reference.NoteStyle attribute), 120  
 horizontal\_align (rinoh.reference.ReferencingParagraphStyle attribute), 47  
 horizontal\_align (rinoh.structure.AdmonitionStyle attribute), 80  
 horizontal\_align (rinoh.structure.HeadingStyle attribute), 57  
 horizontal\_align (rinoh.structure.HorizontalRuleStyle attribute), 82  
 horizontal\_align (rinoh.structure.ListItemLabelStyle attribute), 65, 127  
 horizontal\_align (rinoh.structure.ListOfEntryStyle attribute), 138  
 horizontal\_align (rinoh.structure.ListOfStyle attribute), 132  
 horizontal\_align (rinoh.structure.ListStyle attribute), 62, 124  
 horizontal\_align (rinoh.structure.SectionStyle attribute), 54  
 horizontal\_align (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 horizontal\_align (rinoh.structure.TableOfContentsStyle attribute), 69  
 horizontal\_align (rinoh.table.TableCellStyle attribute), 87  
 horizontal\_align (rinoh.table.TableStyle attribute), 84  
 horizontal\_align (rinoh.table.TableWithCaptionStyle attribute), 145  
 HorizontalAlignment (class in rinoh.flowable), 104  
 HorizontalRule (class in rinoh.structure), 142  
 HorizontalRuleStyle (class in rinoh.structure), 82  
 HU (in module rinoh.language), 110  
 hyphen\_chars (rinoh.image.CaptionStyle attribute), 31  
 hyphen\_chars (rinoh.paragraph.ParagraphStyle attribute), 41  
 hyphen\_chars (rinoh.reference.NoteMarkerStyle attribute), 53  
 hyphen\_chars (rinoh.reference.ReferenceStyle attribute), 118  
 hyphen\_chars (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 hyphen\_chars (rinoh.structure.HeadingStyle attribute), 60  
 hyphen\_chars (rinoh.structure.ListItemLabelStyle attribute), 68, 130  
 hyphen\_chars (rinoh.structure.ListOfEntryStyle attribute), 141  
 hyphen\_chars (rinoh.structure.ListOfStyle attribute), 136  
 hyphen\_chars (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 hyphen\_chars (rinoh.structure.TableOfContentsStyle attribute), 74  
 hyphen\_chars (rinoh.text.TextStyle attribute), 43  
 hyphen\_lang (rinoh.image.CaptionStyle attribute), 32  
 hyphen\_lang (rinoh.paragraph.ParagraphStyle attribute), 41  
 hyphen\_lang (rinoh.reference.NoteMarkerStyle attribute), 53  
 hyphen\_lang (rinoh.reference.ReferenceStyle attribute), 118  
 hyphen\_lang (rinoh.reference.ReferencingParagraphStyle attribute), 51  
 hyphen\_lang (rinoh.structure.HeadingStyle attribute), 60  
 hyphen\_lang (rinoh.structure.ListItemLabelStyle attribute), 68, 131  
 hyphen\_lang (rinoh.structure.ListOfEntryStyle attribute), 141  
 hyphen\_lang (rinoh.structure.ListOfStyle attribute), 136  
 hyphen\_lang (rinoh.structure.TableOfContentsEntryStyle attribute), 79  
 hyphen\_lang (rinoh.structure.TableOfContentsStyle attribute), 74  
 hyphen\_lang (rinoh.text.TextStyle attribute), 43  
 hyphenate (rinoh.image.CaptionStyle attribute), 31  
 hyphenate (rinoh.paragraph.ParagraphStyle attribute), 41  
 hyphenate (rinoh.reference.NoteMarkerStyle attribute), 52  
 hyphenate (rinoh.reference.ReferenceStyle attribute), 118  
 hyphenate (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 hyphenate (rinoh.structure.HeadingStyle attribute), 60  
 hyphenate (rinoh.structure.ListItemLabelStyle attribute), 68, 130  
 hyphenate (rinoh.structure.ListOfEntryStyle attribute), 141  
 hyphenate (rinoh.structure.ListOfStyle attribute), 136  
 hyphenate (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 hyphenate (rinoh.structure.TableOfContentsStyle attribute), 73  
 hyphenate (rinoh.text.TextStyle attribute), 43  
 HyphenatorStore (class in rinoh.paragraph), 116

## I

Image (class in rinoh.image), 106  
ImageArgs (class in rinoh.image), 106  
INCH (in module rinoh.dimension), 99  
indent\_first (rinoh.image.CaptionStyle attribute), 28  
indent\_first (rinoh.paragraph.ParagraphStyle attribute), 37  
indent\_first (rinoh.reference.ReferencingParagraphStyle attribute), 46  
indent\_first (rinoh.structure.HeadingStyle attribute), 56  
indent\_first (rinoh.structure.ListItemLabelStyle attribute), 64, 127  
indent\_first (rinoh.structure.ListOfEntryStyle attribute), 137  
indent\_first (rinoh.structure.ListOfStyle attribute), 134  
indent\_first (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
indent\_first (rinoh.structure.TableOfContentsStyle attribute), 71  
Index (class in rinoh.index), 108  
IndexLabel (class in rinoh.index), 108  
IndexSection (class in rinoh.index), 108  
IndexStyle (class in rinoh.index), 35  
IndexTarget (class in rinoh.index), 108  
IndexTerm (class in rinoh.index), 108  
initials (rinoh.index.IndexStyle attribute), 35  
inline\_title (rinoh.structure.AdmonitionStyle attribute), 79  
InlineFlowable (class in rinoh.inline), 114  
InlineFlowableStyle (class in rinoh.inline), 44  
InlineImage (class in rinoh.image), 106  
InlineIndexTarget (class in rinoh.index), 108  
input  
    rinoh command line option, 5  
Integer (class in rinoh.attribute), 97  
is\_script() (rinoh.text.StyledText method), 113  
IT (in module rinoh.language), 110  
items (rinoh.text.MixedStyledText property), 114  
items (rinoh.text.StyledText property), 114

## 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), 24  
keep\_with\_next (rinoh.flowable.LabeledFlowableStyle attribute), 22  
keep\_with\_next (rinoh.image.CaptionStyle attribute), 29  
keep\_with\_next (rinoh.image.FigureStyle attribute), 33  
keep\_with\_next (rinoh.index.IndexStyle attribute), 36  
keep\_with\_next (rinoh.inline.InlineFlowableStyle attribute), 45  
keep\_with\_next (rinoh.paragraph.ParagraphStyle attribute), 39  
keep\_with\_next (rinoh.reference.NoteStyle attribute), 121  
keep\_with\_next (rinoh.reference.ReferencingParagraphStyle attribute), 48  
keep\_with\_next (rinoh.structure.AdmonitionStyle attribute), 81  
keep\_with\_next (rinoh.structure.HeadingStyle attribute), 56  
keep\_with\_next (rinoh.structure.HorizontalRuleStyle attribute), 83  
keep\_with\_next (rinoh.structure.ListItemLabelStyle attribute), 66, 128  
keep\_with\_next (rinoh.structure.ListOfEntryStyle attribute), 139  
keep\_with\_next (rinoh.structure.ListOfStyle attribute), 133  
keep\_with\_next (rinoh.structure.ListStyle attribute), 63, 125  
keep\_with\_next (rinoh.structure.SectionStyle attribute), 55

- keep\_with\_next (rinoh.structure.TableOfContentsEntryStyle attribute), 76
- keep\_with\_next (rinoh.structure.TableOfContentsStyle attribute), 70
- keep\_with\_next (rinoh.table.TableCellStyle attribute), 88
- keep\_with\_next (rinoh.table.TableStyle attribute), 85
- keep\_with\_next (rinoh.table.TableWithCaptionStyle attribute), 145
- kerning (rinoh.image.CaptionStyle attribute), 31
- kerning (rinoh.paragraph.ParagraphStyle attribute), 41
- kerning (rinoh.reference.NoteMarkerStyle attribute), 52
- kerning (rinoh.reference.ReferenceStyle attribute), 118
- kerning (rinoh.reference.ReferencingParagraphStyle attribute), 50
- kerning (rinoh.structure.HeadingStyle attribute), 60
- kerning (rinoh.structure.ListItemLabelStyle attribute), 68, 130
- kerning (rinoh.structure.ListOfEntryStyle attribute), 141
- kerning (rinoh.structure.ListOfStyle attribute), 136
- kerning (rinoh.structure.TableOfContentsEntryStyle attribute), 78
- kerning (rinoh.structure.TableOfContentsStyle attribute), 73
- kerning (rinoh.text.TextStyle attribute), 43

## L

- Label (class in rinoh.number), 110
- label\_max\_width (rinoh.flowable.LabeledFlowableStyle attribute), 20
- label\_max\_width (rinoh.reference.NoteStyle attribute), 120
- label\_min\_width (rinoh.flowable.LabeledFlowableStyle attribute), 20
- label\_min\_width (rinoh.reference.NoteStyle attribute), 120
- label\_prefix (rinoh.image.CaptionStyle attribute), 30
- label\_prefix (rinoh.number.LabelStyle attribute), 111
- label\_prefix (rinoh.number.NumberStyle attribute), 111
- label\_prefix (rinoh.paragraph.ParagraphStyle attribute), 40
- label\_prefix (rinoh.reference.NoteMarkerStyle attribute), 53
- label\_prefix (rinoh.reference.ReferenceStyle attribute), 119
- label\_prefix (rinoh.reference.ReferencingParagraphStyle attribute), 49
- label\_prefix (rinoh.structure.HeadingStyle attribute), 59
- label\_prefix (rinoh.structure.ListItemLabelStyle attribute), 67, 129
- label\_prefix (rinoh.structure.ListOfEntryStyle attribute), 140
- label\_prefix (rinoh.structure.ListOfStyle attribute), 135
- label\_prefix (rinoh.structure.ListStyle attribute), 64, 126
- label\_prefix (rinoh.structure.TableOfContentsEntryStyle attribute), 77
- label\_prefix (rinoh.structure.TableOfContentsStyle attribute), 72
- label\_spacing (rinoh.flowable.LabeledFlowableStyle attribute), 21
- label\_spacing (rinoh.reference.NoteStyle attribute), 120
- label\_suffix (rinoh.image.CaptionStyle attribute), 30
- label\_suffix (rinoh.number.LabelStyle attribute), 111
- label\_suffix (rinoh.number.NumberStyle attribute), 111
- label\_suffix (rinoh.paragraph.ParagraphStyle attribute), 40
- label\_suffix (rinoh.reference.NoteMarkerStyle attribute), 53
- label\_suffix (rinoh.reference.ReferenceStyle attribute), 119
- label\_suffix (rinoh.reference.ReferencingParagraphStyle attribute), 49
- label\_suffix (rinoh.structure.HeadingStyle attribute), 59
- label\_suffix (rinoh.structure.ListItemLabelStyle attribute), 67, 129
- label\_suffix (rinoh.structure.ListOfEntryStyle attribute), 140
- label\_suffix (rinoh.structure.ListOfStyle attribute), 135
- label\_suffix (rinoh.structure.ListStyle attribute), 64, 126
- label\_suffix (rinoh.structure.TableOfContentsEntryStyle attribute), 77
- label\_suffix (rinoh.structure.TableOfContentsStyle attribute), 72
- LabeledFlowable (class in rinoh.flowable), 102
- LabeledFlowableState (class in rinoh.flowable), 102
- LabeledFlowableStyle (class in rinoh.flowable), 20

LabelStyle (class in rinoh.number), 111  
 Language (class in rinoh.language), 108  
 language (rinoh.template.DocumentTemplate attribute), 148  
 language (rinoh.templates.article.Article attribute), 91  
 language (rinoh.templates.book.Book attribute), 93  
 languages (rinoh.language.Language attribute), 109  
 Leading (class in rinoh.paragraph), 115  
 LEDGER (in module rinoh.paper), 112  
 left\_margin (rinoh.template.BodyPageTemplate attribute), 155  
 left\_margin (rinoh.template.BodyPageTemplateBase attribute), 153  
 left\_margin (rinoh.template.PageTemplateBase attribute), 157  
 left\_margin (rinoh.template.TitlePageTemplate attribute), 156  
 LEGAL (in module rinoh.paper), 112  
 LETTER (in module rinoh.paper), 112  
 ligatures (rinoh.image.CaptionStyle attribute), 31  
 ligatures (rinoh.paragraph.ParagraphStyle attribute), 41  
 ligatures (rinoh.reference.NoteMarkerStyle attribute), 52  
 ligatures (rinoh.reference.ReferenceStyle attribute), 118  
 ligatures (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 ligatures (rinoh.structure.HeadingStyle attribute), 60  
 ligatures (rinoh.structure.ListItemLabelStyle attribute), 68, 130  
 ligatures (rinoh.structure.ListOfEntryStyle attribute), 141  
 ligatures (rinoh.structure.ListOfStyle attribute), 136  
 ligatures (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 ligatures (rinoh.structure.TableOfContentsStyle attribute), 73  
 ligatures (rinoh.text.TextStyle attribute), 43  
 limit\_width (rinoh.image.ImageArgs attribute), 106  
 Line (class in rinoh.draw), 101  
 line\_spacing (rinoh.image.CaptionStyle attribute), 28  
 line\_spacing (rinoh.paragraph.ParagraphStyle attribute), 37  
 line\_spacing (rinoh.reference.ReferencingParagraphStyle attribute), 47  
 line\_spacing (rinoh.structure.HeadingStyle attribute), 57  
 line\_spacing (rinoh.structure.ListItemLabelStyle attribute), 64, 127  
 line\_spacing (rinoh.structure.ListOfEntryStyle attribute), 137  
 line\_spacing (rinoh.structure.ListOfStyle attribute), 134  
 line\_spacing (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 line\_spacing (rinoh.structure.TableOfContentsStyle attribute), 71  
 LineSpacing (class in rinoh.paragraph), 115  
 LineStyle (class in rinoh.draw), 18  
 link (rinoh.reference.NoteMarkerStyle attribute), 51  
 link (rinoh.reference.ReferenceStyle attribute), 117  
 List (class in rinoh.structure), 123  
 list\_class (rinoh.image.ListOfFiguresSection attribute), 108  
 list\_class (rinoh.table.ListOfTablesSection attribute), 147  
 ListItem (class in rinoh.structure), 126  
 ListItemLabel (class in rinoh.structure), 126  
 ListItemLabelStyle (class in rinoh.structure), 64, 126  
 ListOfEntryStyle (class in rinoh.structure), 137  
 ListOfFigures (class in rinoh.image), 108  
 ListOfFiguresSection (class in rinoh.image), 108  
 ListOfStyle (class in rinoh.structure), 132  
 ListOfTables (class in rinoh.table), 147  
 ListOfTablesSection (class in rinoh.table), 147  
 ListStyle (class in rinoh.structure), 61, 123  
 Locale (class in rinoh.text), 114  
 location (rinoh.attribute.RuleSetFile property), 97  
 location (rinoh.reference.NoteStyle attribute), 120

## M

- margin\_left (rinoh.flowable.FloatStyle attribute), 26
- margin\_left (rinoh.flowable.FlowableStyle attribute), 19
- margin\_left (rinoh.flowable.GroupedFlowablesStyle attribute), 24
- margin\_left (rinoh.flowable.LabeledFlowableStyle attribute), 21
- margin\_left (rinoh.image.CaptionStyle attribute), 29
- margin\_left (rinoh.image.FigureStyle attribute), 33
- margin\_left (rinoh.index.IndexStyle attribute), 35
- margin\_left (rinoh.inline.InlineFlowableStyle attribute), 44
- margin\_left (rinoh.paragraph.ParagraphStyle attribute), 38
- margin\_left (rinoh.reference.NoteStyle attribute), 121
- margin\_left (rinoh.reference.ReferencingParagraphStyle attribute), 48
- margin\_left (rinoh.structure.AdmonitionStyle attribute), 80
- margin\_left (rinoh.structure.HeadingStyle attribute), 57
- margin\_left (rinoh.structure.HorizontalRuleStyle attribute), 82
- margin\_left (rinoh.structure.ListItemLabelStyle attribute), 65, 128
- margin\_left (rinoh.structure.ListOfEntryStyle attribute), 138
- margin\_left (rinoh.structure.ListOfStyle attribute), 132
- margin\_left (rinoh.structure.ListStyle attribute), 62, 124
- margin\_left (rinoh.structure.SectionStyle attribute), 55
- margin\_left (rinoh.structure.TableOfContentsEntryStyle attribute), 76
- margin\_left (rinoh.structure.TableOfContentsStyle attribute), 70
- margin\_left (rinoh.table.TableCellStyle attribute), 87
- margin\_left (rinoh.table.TableStyle attribute), 84
- margin\_left (rinoh.table.TableWithCaptionStyle attribute), 145
- margin\_right (rinoh.flowable.FloatStyle attribute), 26
- margin\_right (rinoh.flowable.FlowableStyle attribute), 19
- margin\_right (rinoh.flowable.GroupedFlowablesStyle attribute), 24
- margin\_right (rinoh.flowable.LabeledFlowableStyle attribute), 21
- margin\_right (rinoh.image.CaptionStyle attribute), 29
- margin\_right (rinoh.image.FigureStyle attribute), 33
- margin\_right (rinoh.index.IndexStyle attribute), 36
- margin\_right (rinoh.inline.InlineFlowableStyle attribute), 44
- margin\_right (rinoh.paragraph.ParagraphStyle attribute), 38
- margin\_right (rinoh.reference.NoteStyle attribute), 121
- margin\_right (rinoh.reference.ReferencingParagraphStyle attribute), 48
- margin\_right (rinoh.structure.AdmonitionStyle attribute), 80
- margin\_right (rinoh.structure.HeadingStyle attribute), 58
- margin\_right (rinoh.structure.HorizontalRuleStyle attribute), 82
- margin\_right (rinoh.structure.ListItemLabelStyle attribute), 65, 128
- margin\_right (rinoh.structure.ListOfEntryStyle attribute), 138
- margin\_right (rinoh.structure.ListOfStyle attribute), 132
- margin\_right (rinoh.structure.ListStyle attribute), 62, 124
- margin\_right (rinoh.structure.SectionStyle attribute), 55
- margin\_right (rinoh.structure.TableOfContentsEntryStyle attribute), 76
- margin\_right (rinoh.structure.TableOfContentsStyle attribute), 70
- margin\_right (rinoh.table.TableCellStyle attribute), 87
- margin\_right (rinoh.table.TableStyle attribute), 85
- margin\_right (rinoh.table.TableWithCaptionStyle attribute), 145
- matcher (in module rinoh.stylesheets), 11
- MixedStyledText (class in rinoh.text), 114
- MM (in module rinoh.dimension), 99
- module
  - rinoh.attribute, 97
  - rinoh.color, 98
  - rinoh.dimension, 98
  - rinoh.document, 99

- rinoh.draw, 101
- rinoh.flowable, 101
- rinoh.font, 104
- rinoh.font.style, 106
- rinoh.image, 106
- rinoh.index, 108
- rinoh.inline, 114
- rinoh.language, 108
- rinoh.number, 110
- rinoh.paper, 112
- rinoh.paragraph, 113
- rinoh.reference, 116
- rinoh.strings, 142
- rinoh.structure, 123
- rinoh.style, 142
- rinoh.stylesheets, 11
- rinoh.table, 144
- rinoh.template, 148
- rinoh.templates.article, 91
- rinoh.templates.book, 92
- rinoh.text, 113

## N

- new\_page() (rinoh.template.DocumentPart method), 149
- NL (in module rinoh.language), 110
- no\_break\_after (rinoh.image.CaptionStyle attribute), 32
- no\_break\_after (rinoh.paragraph.ParagraphStyle attribute), 42
- no\_break\_after (rinoh.reference.NoteMarkerStyle attribute), 53
- no\_break\_after (rinoh.reference.ReferenceStyle attribute), 118
- no\_break\_after (rinoh.reference.ReferencingParagraphStyle attribute), 51
- no\_break\_after (rinoh.structure.HeadingStyle attribute), 61
- no\_break\_after (rinoh.structure.ListItemLabelStyle attribute), 68, 131
- no\_break\_after (rinoh.structure.ListOfEntryStyle attribute), 141
- no\_break\_after (rinoh.structure.ListOfStyle attribute), 136
- no\_break\_after (rinoh.structure.TableOfContentsEntryStyle attribute), 79
- no\_break\_after (rinoh.structure.TableOfContentsStyle attribute), 74
- no\_break\_after (rinoh.text.TextStyle attribute), 43
- NoBreakAfter (class in rinoh.text), 114
- Note (class in rinoh.reference), 119
- NoteMarkerBase (class in rinoh.reference), 122
- NoteMarkerByID (class in rinoh.reference), 122
- NoteMarkerStyle (class in rinoh.reference), 51
- NoteMarkerWithNote (class in rinoh.reference), 122
- NoteStyle (class in rinoh.reference), 120
- number\_format (rinoh.image.CaptionStyle attribute), 27
- number\_format (rinoh.number.NumberStyle attribute), 111
- number\_format (rinoh.paragraph.ParagraphStyle attribute), 38
- number\_format (rinoh.reference.NoteMarkerStyle attribute), 53
- number\_format (rinoh.reference.ReferenceStyle attribute), 118
- number\_format (rinoh.reference.ReferencingParagraphStyle attribute), 47
- number\_format (rinoh.structure.HeadingStyle attribute), 57
- number\_format (rinoh.structure.ListItemLabelStyle attribute), 65, 127
- number\_format (rinoh.structure.ListOfEntryStyle attribute), 137
- number\_format (rinoh.structure.ListOfStyle attribute), 134
- number\_format (rinoh.structure.ListStyle attribute), 64, 126
- number\_format (rinoh.structure.TableOfContentsEntryStyle attribute), 75
- number\_format (rinoh.structure.TableOfContentsStyle attribute), 72



- number\_separator (rinoh.image.CaptionStyle attribute), 28
- number\_separator (rinoh.paragraph.ParagraphStyle attribute), 38
- number\_separator (rinoh.reference.ReferencingParagraphStyle attribute), 47
- number\_separator (rinoh.structure.HeadingStyle attribute), 57
- number\_separator (rinoh.structure.ListItemLabelStyle attribute), 64, 126
- number\_separator (rinoh.structure.ListOfEntryStyle attribute), 138
- number\_separator (rinoh.structure.ListOfStyle attribute), 134
- number\_separator (rinoh.structure.TableOfContentsEntryStyle attribute), 75
- number\_separator (rinoh.structure.TableOfContentsStyle attribute), 72
- NumberFormat (class in rinoh.number), 110
- numbering\_level (rinoh.image.CaptionStyle attribute), 27
- numbering\_level (rinoh.paragraph.ParagraphStyle attribute), 38
- numbering\_level (rinoh.reference.ReferencingParagraphStyle attribute), 47
- numbering\_level (rinoh.structure.HeadingStyle attribute), 56
- numbering\_level (rinoh.structure.ListItemLabelStyle attribute), 65, 127
- numbering\_level (rinoh.structure.ListOfEntryStyle attribute), 138
- numbering\_level (rinoh.structure.ListOfStyle attribute), 135
- numbering\_level (rinoh.structure.TableOfContentsEntryStyle attribute), 75
- numbering\_level (rinoh.structure.TableOfContentsStyle attribute), 72
- NumberStyle (class in rinoh.number), 111

## O

- OptionSet (class in rinoh.attribute), 97
- ordered (rinoh.structure.ListStyle attribute), 61, 123
- OverrideDefault (class in rinoh.attribute), 97

## P

- padding (rinoh.flowable.FloatStyle attribute), 26
- padding (rinoh.flowable.FlowableStyle attribute), 19
- padding (rinoh.flowable.GroupedFlowablesStyle attribute), 24
- padding (rinoh.flowable.LabeledFlowableStyle attribute), 21
- padding (rinoh.image.CaptionStyle attribute), 29
- padding (rinoh.image.FigureStyle attribute), 33
- padding (rinoh.index.IndexStyle attribute), 36
- padding (rinoh.inline.InlineFlowableStyle attribute), 44
- padding (rinoh.paragraph.ParagraphStyle attribute), 38
- padding (rinoh.reference.NoteStyle attribute), 121
- padding (rinoh.reference.ReferencingParagraphStyle attribute), 48
- padding (rinoh.structure.AdmonitionStyle attribute), 80
- padding (rinoh.structure.HeadingStyle attribute), 58
- padding (rinoh.structure.HorizontalRuleStyle attribute), 82
- padding (rinoh.structure.ListItemLabelStyle attribute), 65, 128
- padding (rinoh.structure.ListOfEntryStyle attribute), 138
- padding (rinoh.structure.ListOfStyle attribute), 133
- padding (rinoh.structure.ListStyle attribute), 62, 124
- padding (rinoh.structure.SectionStyle attribute), 55
- padding (rinoh.structure.TableOfContentsEntryStyle attribute), 76
- padding (rinoh.structure.TableOfContentsStyle attribute), 70
- padding (rinoh.table.TableCellStyle attribute), 87
- padding (rinoh.table.TableStyle attribute), 85
- padding (rinoh.table.TableWithCaptionStyle attribute), 145
- padding\_bottom (rinoh.flowable.FloatStyle attribute), 26
- padding\_bottom (rinoh.flowable.FlowableStyle attribute), 19
- padding\_bottom (rinoh.flowable.GroupedFlowablesStyle attribute), 24
- padding\_bottom (rinoh.flowable.LabeledFlowableStyle attribute), 22
- padding\_bottom (rinoh.image.CaptionStyle attribute), 29
- padding\_bottom (rinoh.image.FigureStyle attribute), 33

padding\_bottom (rinoh.index.IndexStyle attribute), 36

padding\_bottom (rinoh.inline.InlineFlowableStyle attribute), 45

padding\_bottom (rinoh.paragraph.ParagraphStyle attribute), 39

padding\_bottom (rinoh.reference.NoteStyle attribute), 121

padding\_bottom (rinoh.reference.ReferencingParagraphStyle attribute), 48

padding\_bottom (rinoh.structure.AdmonitionStyle attribute), 80

padding\_bottom (rinoh.structure.HeadingStyle attribute), 58

padding\_bottom (rinoh.structure.HorizontalRuleStyle attribute), 83

padding\_bottom (rinoh.structure.ListItemLabelStyle attribute), 66, 128

padding\_bottom (rinoh.structure.ListOfEntryStyle attribute), 139

padding\_bottom (rinoh.structure.ListOfStyle attribute), 133

padding\_bottom (rinoh.structure.ListStyle attribute), 63, 125

padding\_bottom (rinoh.structure.SectionStyle attribute), 55

padding\_bottom (rinoh.structure.TableOfContentsEntryStyle attribute), 76

padding\_bottom (rinoh.structure.TableOfContentsStyle attribute), 70

padding\_bottom (rinoh.table.TableCellStyle attribute), 87

padding\_bottom (rinoh.table.TableStyle attribute), 85

padding\_bottom (rinoh.table.TableWithCaptionStyle attribute), 145

padding\_left (rinoh.flowable.FloatStyle attribute), 26

padding\_left (rinoh.flowable.FlowableStyle attribute), 19

padding\_left (rinoh.flowable.GroupedFlowablesStyle attribute), 24

padding\_left (rinoh.flowable.LabeledFlowableStyle attribute), 21

padding\_left (rinoh.image.CaptionStyle attribute), 29

padding\_left (rinoh.image.FigureStyle attribute), 33

padding\_left (rinoh.index.IndexStyle attribute), 36

padding\_left (rinoh.inline.InlineFlowableStyle attribute), 45

padding\_left (rinoh.paragraph.ParagraphStyle attribute), 39

padding\_left (rinoh.reference.NoteStyle attribute), 121

padding\_left (rinoh.reference.ReferencingParagraphStyle attribute), 48

padding\_left (rinoh.structure.AdmonitionStyle attribute), 80

padding\_left (rinoh.structure.HeadingStyle attribute), 58

padding\_left (rinoh.structure.HorizontalRuleStyle attribute), 82

padding\_left (rinoh.structure.ListItemLabelStyle attribute), 66, 128

padding\_left (rinoh.structure.ListOfEntryStyle attribute), 138

padding\_left (rinoh.structure.ListOfStyle attribute), 133

padding\_left (rinoh.structure.ListStyle attribute), 62, 124

padding\_left (rinoh.structure.SectionStyle attribute), 55

padding\_left (rinoh.structure.TableOfContentsEntryStyle attribute), 76

padding\_left (rinoh.structure.TableOfContentsStyle attribute), 70

padding\_left (rinoh.table.TableCellStyle attribute), 87

padding\_left (rinoh.table.TableStyle attribute), 85

padding\_left (rinoh.table.TableWithCaptionStyle attribute), 145

padding\_right (rinoh.flowable.FloatStyle attribute), 26

padding\_right (rinoh.flowable.FlowableStyle attribute), 19

padding\_right (rinoh.flowable.GroupedFlowablesStyle attribute), 24

padding\_right (rinoh.flowable.LabeledFlowableStyle attribute), 22

padding\_right (rinoh.image.CaptionStyle attribute), 29

padding\_right (rinoh.image.FigureStyle attribute), 33

padding\_right (rinoh.index.IndexStyle attribute), 36

padding\_right (rinoh.inline.InlineFlowableStyle attribute), 45

padding\_right (rinoh.paragraph.ParagraphStyle attribute), 39

padding\_right (rinoh.reference.NoteStyle attribute), 121

padding\_right (rinoh.reference.ReferencingParagraphStyle attribute), 48

padding\_right (rinoh.structure.AdmonitionStyle attribute), 80

padding\_right (rinoh.structure.HeadingStyle attribute), 58

padding\_right (rinoh.structure.HorizontalRuleStyle attribute), 82

padding\_right (rinoh.structure.ListItemLabelStyle attribute), 66, 128

padding\_right (rinoh.structure.ListOfEntryStyle attribute), 138

padding\_right (rinoh.structure.ListOfStyle attribute), 133  
padding\_right (rinoh.structure.ListStyle attribute), 62, 125  
padding\_right (rinoh.structure.SectionStyle attribute), 55  
padding\_right (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
padding\_right (rinoh.structure.TableOfContentsStyle attribute), 70  
padding\_right (rinoh.table.TableCellStyle attribute), 87  
padding\_right (rinoh.table.TableStyle attribute), 85  
padding\_right (rinoh.table.TableWithCaptionStyle attribute), 145  
padding\_top (rinoh.flowable.FloatStyle attribute), 26  
padding\_top (rinoh.flowable.FlowableStyle attribute), 19  
padding\_top (rinoh.flowable.GroupedFlowablesStyle attribute), 24  
padding\_top (rinoh.flowable.LabeledFlowableStyle attribute), 22  
padding\_top (rinoh.image.CaptionStyle attribute), 29  
padding\_top (rinoh.image.FigureStyle attribute), 33  
padding\_top (rinoh.index.IndexStyle attribute), 36  
padding\_top (rinoh.inline.InlineFlowableStyle attribute), 45  
padding\_top (rinoh.paragraph.ParagraphStyle attribute), 39  
padding\_top (rinoh.reference.NoteStyle attribute), 121  
padding\_top (rinoh.reference.ReferencingParagraphStyle attribute), 48  
padding\_top (rinoh.structure.AdmonitionStyle attribute), 80  
padding\_top (rinoh.structure.HeadingStyle attribute), 58  
padding\_top (rinoh.structure.HorizontalRuleStyle attribute), 82  
padding\_top (rinoh.structure.ListItemLabelStyle attribute), 66, 128  
padding\_top (rinoh.structure.ListOfEntryStyle attribute), 139  
padding\_top (rinoh.structure.ListOfStyle attribute), 133  
padding\_top (rinoh.structure.ListStyle attribute), 62, 125  
padding\_top (rinoh.structure.SectionStyle attribute), 55  
padding\_top (rinoh.structure.TableOfContentsEntryStyle attribute), 76  
padding\_top (rinoh.structure.TableOfContentsStyle attribute), 70  
padding\_top (rinoh.table.TableCellStyle attribute), 87  
padding\_top (rinoh.table.TableStyle attribute), 85  
padding\_top (rinoh.table.TableWithCaptionStyle attribute), 145  
Page (class in rinoh.document), 100  
page (rinoh.document.Page property), 100  
page (rinoh.templates.article.Article attribute), 92  
page (rinoh.templates.book.Book attribute), 94  
page\_break (rinoh.flowable.FloatStyle attribute), 27  
page\_break (rinoh.flowable.FlowableStyle attribute), 20  
page\_break (rinoh.flowable.GroupedFlowablesStyle attribute), 25  
page\_break (rinoh.flowable.LabeledFlowableStyle attribute), 23  
page\_break (rinoh.image.CaptionStyle attribute), 30  
page\_break (rinoh.image.FigureStyle attribute), 34  
page\_break (rinoh.index.IndexStyle attribute), 37  
page\_break (rinoh.inline.InlineFlowableStyle attribute), 46  
page\_break (rinoh.paragraph.ParagraphStyle attribute), 40  
page\_break (rinoh.reference.NoteStyle attribute), 122  
page\_break (rinoh.reference.ReferencingParagraphStyle attribute), 49  
page\_break (rinoh.structure.AdmonitionStyle attribute), 81  
page\_break (rinoh.structure.HeadingStyle attribute), 59  
page\_break (rinoh.structure.HorizontalRuleStyle attribute), 83  
page\_break (rinoh.structure.ListItemLabelStyle attribute), 67, 129  
page\_break (rinoh.structure.ListOfEntryStyle attribute), 139  
page\_break (rinoh.structure.ListOfStyle attribute), 134  
page\_break (rinoh.structure.ListStyle attribute), 63, 126  
page\_break (rinoh.structure.SectionStyle attribute), 56  
page\_break (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
page\_break (rinoh.structure.TableOfContentsStyle attribute), 71  
page\_break (rinoh.table.TableCellStyle attribute), 88

page\_break (rinoh.table.TableStyle attribute), 86  
 page\_break (rinoh.table.TableWithCaptionStyle attribute), 146  
 page\_number\_format (rinoh.template.ContentsPartTemplate attribute), 151  
 page\_number\_format (rinoh.template.DocumentPartTemplate attribute), 150  
 page\_number\_format (rinoh.template.FixedDocumentPartTemplate attribute), 151  
 page\_number\_format (rinoh.template.TitlePartTemplate attribute), 150  
 page\_number\_prefix (rinoh.template.ContentsPartTemplate attribute), 151  
 page\_number\_prefix (rinoh.template.DocumentPartTemplate attribute), 150  
 page\_number\_prefix (rinoh.template.FixedDocumentPartTemplate attribute), 151  
 page\_number\_prefix (rinoh.template.TitlePartTemplate attribute), 150  
 page\_orientation (rinoh.template.BodyPageTemplate attribute), 154  
 page\_orientation (rinoh.template.BodyPageTemplateBase attribute), 153  
 page\_orientation (rinoh.template.PageTemplateBase attribute), 157  
 page\_orientation (rinoh.template.TitlePageTemplate attribute), 156  
 page\_size (rinoh.template.BodyPageTemplate attribute), 154  
 page\_size (rinoh.template.BodyPageTemplateBase attribute), 152  
 page\_size (rinoh.template.PageTemplateBase attribute), 156  
 page\_size (rinoh.template.TitlePageTemplate attribute), 155  
 PageBreak (class in rinoh.flowable), 102  
 PageNumberFormat (class in rinoh.document), 100  
 PageOrientation (class in rinoh.document), 100  
 PageTemplateBase (class in rinoh.template), 156  
 PageType (class in rinoh.document), 100  
 Paper (class in rinoh.paper), 112  
 Paragraph (class in rinoh.paragraph), 113  
 ParagraphBase (class in rinoh.paragraph), 113  
 ParagraphState (class in rinoh.paragraph), 113  
 ParagraphStyle (class in rinoh.paragraph), 37  
 parse\_string() (rinoh.color.Color class method), 98  
 ParseError, 97  
 parts (rinoh.template.DocumentTemplate attribute), 148  
 parts (rinoh.templates.article.Article attribute), 91  
 parts (rinoh.templates.book.Book attribute), 92  
 PartsList (class in rinoh.template), 149  
 PERCENT (in module rinoh.dimension), 99  
 PICA (in module rinoh.dimension), 99  
 PL (in module rinoh.language), 110  
 place() (rinoh.document.Page method), 100  
 Polygon (class in rinoh.draw), 101  
 position (rinoh.image.CaptionStyle attribute), 31  
 position (rinoh.paragraph.ParagraphStyle attribute), 41  
 position (rinoh.reference.NoteMarkerStyle attribute), 52  
 position (rinoh.reference.ReferenceStyle attribute), 117  
 position (rinoh.reference.ReferencingParagraphStyle attribute), 50  
 position (rinoh.structure.HeadingStyle attribute), 60  
 position (rinoh.structure.ListItemLabelStyle attribute), 68, 130  
 position (rinoh.structure.ListOfEntryStyle attribute), 141  
 position (rinoh.structure.ListOfStyle attribute), 136  
 position (rinoh.structure.TableOfContentsEntryStyle attribute), 78  
 position (rinoh.structure.TableOfContentsStyle attribute), 73  
 position (rinoh.text.TextStyle attribute), 43  
 prepare() (rinoh.flowable.LabeledFlowable method), 102  
 prepare() (rinoh.flowable.StaticGroupedFlowables method), 103  
 prepare() (rinoh.paragraph.ParagraphBase method), 113  
 prepare() (rinoh.reference.NoteMarkerBase method), 122  
 prepare() (rinoh.reference.NoteMarkerWithNote method), 122  
 prepare() (rinoh.structure.Heading method), 123  
 prepare() (rinoh.table.Table method), 144

prepare() (rinoh.table.TableRow method), 147  
prepare() (rinoh.table.TableSection method), 147  
prepare() (rinoh.text.MixedStyledText method), 114  
ProportionalSpacing (class in rinoh.paragraph), 115  
PT (in module rinoh.dimension), 99

## Q

QUARTERS (in module rinoh.dimension), 99  
quiet (rinoh.reference.NoteMarkerStyle attribute), 51  
quiet (rinoh.reference.ReferenceStyle attribute), 117

## R

Rectangle (class in rinoh.draw), 101  
Reference (class in rinoh.reference), 116  
ReferenceField (class in rinoh.reference), 119  
ReferenceStyle (class in rinoh.reference), 116  
ReferenceText (class in rinoh.reference), 119  
ReferenceType (class in rinoh.reference), 119  
ReferencingParagraph (class in rinoh.reference), 119  
ReferencingParagraphStyle (class in rinoh.reference), 46  
render() (rinoh.document.Document method), 100  
render() (rinoh.document.Page method), 100  
render() (rinoh.flowable.GroupedFlowables method), 103  
render() (rinoh.flowable.GroupedLabeledFlowables method), 103  
render() (rinoh.flowable.LabeledFlowable method), 102  
render() (rinoh.paragraph.ParagraphBase method), 113  
render() (rinoh.structure.HorizontalRule method), 142  
render() (rinoh.table.Table method), 144  
repeat\_head (rinoh.table.TableStyle attribute), 84  
rgba (rinoh.color.Color property), 98  
right\_margin (rinoh.template.BodyPageTemplate attribute), 155  
right\_margin (rinoh.template.BodyPageTemplateBase attribute), 153  
right\_margin (rinoh.template.PageTemplateBase attribute), 157  
right\_margin (rinoh.template.TitlePageTemplate attribute), 156  
rinoh command line option  
    --docs, 6  
    --format, 5  
    --help, 5  
    --install-resources, 5  
    --list-fonts, 6  
    --list-formats, 6  
    --list-options, 6  
    --list-stylesheets, 5  
    --list-templates, 5  
    --option, 5  
    --output, 5  
    --paper, 5  
    --stylesheet, 5  
    --template, 5  
    --version, 6  
    --versions, 6  
    -f, 5  
    -h, 5  
    -i, 5  
    -O, 5  
    -o, 5  
    -p, 5

- s, 5
- t, 5
- input, 5
- rinoh.attribute
  - module, 97
- rinoh.color
  - module, 98
- rinoh.dimension
  - module, 98
- rinoh.document
  - module, 99
- rinoh.draw
  - module, 101
- rinoh.flowable
  - module, 101
- rinoh.font
  - module, 104
- rinoh.font.style
  - module, 106
- rinoh.image
  - module, 106
- rinoh.index
  - module, 108
- rinoh.inline
  - module, 114
- rinoh.language
  - module, 108
- rinoh.number
  - module, 110
- rinoh.paper
  - module, 112
- rinoh.paragraph
  - module, 113
- rinoh.reference
  - module, 116
- rinoh.strings
  - module, 142
- rinoh.structure
  - module, 123
- rinoh.style
  - module, 142
- rinoh.stylesheets
  - module, 11
- rinoh.table
  - module, 144
- rinoh.template
  - module, 148
- rinoh.templates.article
  - module, 91
- rinoh.templates.book
  - module, 92
- rinoh.text
  - module, 113
- 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), 97
- rotate (rinoh.image.ImageArgs attribute), 107
- RuleSet (class in rinoh.attribute), 97
- RuleSetFile (class in rinoh.attribute), 97

## S

- same\_page (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- same\_page (rinoh.image.FigureStyle attribute), 34
- same\_page (rinoh.index.IndexStyle attribute), 35
- same\_page (rinoh.structure.AdmonitionStyle attribute), 80
- same\_page (rinoh.structure.ListOfStyle attribute), 132
- same\_page (rinoh.structure.ListStyle attribute), 62, 124
- same\_page (rinoh.structure.SectionStyle attribute), 54
- same\_page (rinoh.structure.TableOfContentsStyle attribute), 69
- same\_page (rinoh.table.TableCellStyle attribute), 87
- same\_page (rinoh.table.TableWithCaptionStyle attribute), 147
- Scale (class in rinoh.image), 107
- scale (rinoh.image.ImageArgs attribute), 107
- script\_level() (rinoh.text.StyledText method), 113
- Section (class in rinoh.structure), 123
- SectionStyle (class in rinoh.structure), 54
- SetMetadataFlowable (class in rinoh.flowable), 102
- Shape (class in rinoh.draw), 101
- ShapeStyle (class in rinoh.draw), 18
- show\_in\_toc (rinoh.structure.SectionStyle attribute), 54
- sideways (rinoh.template.BodyPageTemplate attribute), 155
- sideways (rinoh.template.BodyPageTemplateBase attribute), 153
- sideways (rinoh.template.PageTemplateBase attribute), 157
- sideways (rinoh.template.TitlePageTemplate attribute), 156
- SINGLE (in module rinoh.paragraph), 116
- SingleStyledText (class in rinoh.text), 114
- space\_above (rinoh.flowable.FloatStyle attribute), 25
- space\_above (rinoh.flowable.FlowableStyle attribute), 19
- space\_above (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- space\_above (rinoh.flowable.LabeledFlowableStyle attribute), 21
- space\_above (rinoh.image.CaptionStyle attribute), 28
- space\_above (rinoh.image.FigureStyle attribute), 32
- space\_above (rinoh.index.IndexStyle attribute), 35
- space\_above (rinoh.inline.InlineFlowableStyle attribute), 44
- space\_above (rinoh.paragraph.ParagraphStyle attribute), 38
- space\_above (rinoh.reference.NoteStyle attribute), 120
- space\_above (rinoh.reference.ReferencingParagraphStyle attribute), 47
- space\_above (rinoh.structure.AdmonitionStyle attribute), 80
- space\_above (rinoh.structure.HeadingStyle attribute), 57
- space\_above (rinoh.structure.HorizontalRuleStyle attribute), 82
- space\_above (rinoh.structure.ListItemLabelStyle attribute), 65, 127

- space\_above (rinoh.structure.ListOfEntryStyle attribute), 138
- space\_above (rinoh.structure.ListOfStyle attribute), 132
- space\_above (rinoh.structure.ListStyle attribute), 62, 124
- space\_above (rinoh.structure.SectionStyle attribute), 54
- space\_above (rinoh.structure.TableOfContentsEntryStyle attribute), 75
- space\_above (rinoh.structure.TableOfContentsStyle attribute), 70
- space\_above (rinoh.table.TableCellStyle attribute), 87
- space\_above (rinoh.table.TableStyle attribute), 84
- space\_above (rinoh.table.TableWithCaptionStyle attribute), 145
- space\_below (rinoh.flowable.FloatStyle attribute), 26
- space\_below (rinoh.flowable.FlowableStyle attribute), 19
- space\_below (rinoh.flowable.GroupedFlowablesStyle attribute), 23
- space\_below (rinoh.flowable.LabeledFlowableStyle attribute), 21
- space\_below (rinoh.image.CaptionStyle attribute), 28
- space\_below (rinoh.image.FigureStyle attribute), 33
- space\_below (rinoh.index.IndexStyle attribute), 35
- space\_below (rinoh.inline.InlineFlowableStyle attribute), 44
- space\_below (rinoh.paragraph.ParagraphStyle attribute), 38
- space\_below (rinoh.reference.NoteStyle attribute), 120
- space\_below (rinoh.reference.ReferencingParagraphStyle attribute), 47
- space\_below (rinoh.structure.AdmonitionStyle attribute), 80
- space\_below (rinoh.structure.HeadingStyle attribute), 57
- space\_below (rinoh.structure.HorizontalRuleStyle attribute), 82
- space\_below (rinoh.structure.ListItemLabelStyle attribute), 65, 127
- space\_below (rinoh.structure.ListOfEntryStyle attribute), 138
- space\_below (rinoh.structure.ListOfStyle attribute), 132
- space\_below (rinoh.structure.ListStyle attribute), 62, 124
- space\_below (rinoh.structure.SectionStyle attribute), 54
- space\_below (rinoh.structure.TableOfContentsEntryStyle attribute), 75
- space\_below (rinoh.structure.TableOfContentsStyle attribute), 70
- space\_below (rinoh.table.TableCellStyle attribute), 87
- space\_below (rinoh.table.TableStyle attribute), 84
- space\_below (rinoh.table.TableWithCaptionStyle attribute), 145
- 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), 84
- STANDARD (in module rinoh.paragraph), 116
- StaticGroupedFlowables (class in rinoh.flowable), 103
- StringField (class in rinoh.strings), 142
- Strings (class in rinoh.strings), 142
- strings (rinoh.template.DocumentTemplate attribute), 148
- strings (rinoh.templates.article.Article attribute), 91
- strings (rinoh.templates.book.Book attribute), 93
- Stroke (class in rinoh.draw), 101
- stroke (rinoh.draw.LineStyle attribute), 18
- stroke (rinoh.draw.ShapeStyle attribute), 18
- stroke (rinoh.structure.HorizontalRuleStyle attribute), 84
- stroke (rinoh.table.TableCellBackgroundStyle attribute), 89



stroke (rinoh.table.TableCellBorderStyle attribute), 89  
 Style (class in rinoh.style), 143  
 style\_class (rinoh.draw.Line attribute), 101  
 style\_class (rinoh.draw.Shape attribute), 101  
 style\_class (rinoh.flowable.Float attribute), 104  
 style\_class (rinoh.flowable.GroupedFlowables attribute), 103  
 style\_class (rinoh.flowable.LabeledFlowable attribute), 102  
 style\_class (rinoh.image.Caption attribute), 107  
 style\_class (rinoh.image.Figure attribute), 107  
 style\_class (rinoh.index.Index attribute), 108  
 style\_class (rinoh.inline.InlineFlowable attribute), 114  
 style\_class (rinoh.paragraph.ParagraphBase attribute), 113  
 style\_class (rinoh.reference.Note attribute), 119  
 style\_class (rinoh.reference.NoteMarkerBase attribute), 122  
 style\_class (rinoh.reference.ReferencingParagraph attribute), 119  
 style\_class (rinoh.structure.Admonition attribute), 142  
 style\_class (rinoh.structure.Heading attribute), 123  
 style\_class (rinoh.structure.HorizontalRule attribute), 142  
 style\_class (rinoh.structure.List attribute), 123  
 style\_class (rinoh.structure.ListItemLabel attribute), 126  
 style\_class (rinoh.structure.TableOfContents attribute), 131  
 style\_class (rinoh.structure.TableOfContentsEntry attribute), 131  
 style\_class (rinoh.style.Styled attribute), 142  
 style\_class (rinoh.table.Table attribute), 144  
 style\_class (rinoh.table.TableCell attribute), 147  
 style\_class (rinoh.table.TableCellBackground attribute), 147  
 style\_class (rinoh.table.TableCellBorder attribute), 147  
 style\_class (rinoh.table.TableWithCaption attribute), 144  
 style\_class (rinoh.text.StyledText attribute), 114  
 Styled (class in rinoh.style), 142  
 StyledMatcher (class in rinoh.style), 144  
 StyledText (class in rinoh.text), 113  
 StyleSheet (class in rinoh.style), 143  
 stylesheet (rinoh.template.DocumentTemplate attribute), 148  
 stylesheet (rinoh.templates.article.Article attribute), 91  
 stylesheet (rinoh.templates.book.Book attribute), 92  
 StyleSheetFile (class in rinoh.style), 144

## T

tab\_stops (rinoh.image.CaptionStyle attribute), 28  
 tab\_stops (rinoh.paragraph.ParagraphStyle attribute), 38  
 tab\_stops (rinoh.reference.ReferencingParagraphStyle attribute), 47  
 tab\_stops (rinoh.structure.HeadingStyle attribute), 57  
 tab\_stops (rinoh.structure.ListItemLabelStyle attribute), 65, 127  
 tab\_stops (rinoh.structure.ListOfEntryStyle attribute), 137  
 tab\_stops (rinoh.structure.ListOfStyle attribute), 134  
 tab\_stops (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 tab\_stops (rinoh.structure.TableOfContentsStyle attribute), 72  
 TabAlign (class in rinoh.paragraph), 116  
 Table (class in rinoh.table), 144  
 TableBody (class in rinoh.table), 147  
 TableCell (class in rinoh.table), 147  
 TableCellBackground (class in rinoh.table), 147  
 TableCellBackgroundStyle (class in rinoh.table), 89  
 TableCellBorder (class in rinoh.table), 147  
 TableCellBorderStyle (class in rinoh.table), 89  
 TableCellStyle (class in rinoh.table), 86

TableHead (class in rinoh.table), 147  
 TableOfContents (class in rinoh.structure), 131  
 TableOfContentsEntry (class in rinoh.structure), 131  
 TableOfContentsEntryStyle (class in rinoh.structure), 74  
 TableOfContentsSection (class in rinoh.structure), 131  
 TableOfContentsStyle (class in rinoh.structure), 69  
 TableRow (class in rinoh.table), 147  
 TableSection (class in rinoh.table), 147  
 TableStyle (class in rinoh.table), 84  
 TableWithCaption (class in rinoh.table), 144  
 TableWithCaptionStyle (class in rinoh.table), 144  
 TABLOID (in module rinoh.paper), 112  
 TabStop (class in rinoh.paragraph), 116  
 TabStopList (class in rinoh.paragraph), 116  
 target\_has\_class() (rinoh.reference.ReferencingParagraph method), 119  
 target\_has\_classes() (rinoh.reference.ReferencingParagraph method), 119  
 target\_is\_of\_type() (rinoh.reference.ReferencingParagraph method), 119  
 target\_style() (rinoh.reference.ReferencingParagraph method), 119  
 template (rinoh.template.TemplateConfiguration attribute), 149  
 TemplateConfiguration (class in rinoh.template), 148  
 text (rinoh.reference.ReferencingParagraphStyle attribute), 46  
 text (rinoh.structure.ListOfEntryStyle attribute), 137  
 text (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
 text\_align (rinoh.image.CaptionStyle attribute), 28  
 text\_align (rinoh.paragraph.ParagraphStyle attribute), 37  
 text\_align (rinoh.reference.ReferencingParagraphStyle attribute), 47  
 text\_align (rinoh.structure.HeadingStyle attribute), 57  
 text\_align (rinoh.structure.ListItemLabelStyle attribute), 65, 127  
 text\_align (rinoh.structure.ListOfEntryStyle attribute), 137  
 text\_align (rinoh.structure.ListOfStyle attribute), 134  
 text\_align (rinoh.structure.TableOfContentsEntryStyle attribute), 75  
 text\_align (rinoh.structure.TableOfContentsStyle attribute), 72  
 TextAlign (class in rinoh.paragraph), 114  
 TextPosition (class in rinoh.font.style), 106  
 TextStyle (class in rinoh.text), 42  
 title (rinoh.flowable.GroupedFlowablesStyle attribute), 23  
 title (rinoh.image.FigureStyle attribute), 34  
 title (rinoh.index.IndexStyle attribute), 35  
 title (rinoh.structure.AdmonitionStyle attribute), 79  
 title (rinoh.structure.ListOfStyle attribute), 132  
 title (rinoh.structure.ListStyle attribute), 61, 124  
 title (rinoh.structure.SectionStyle attribute), 54  
 title (rinoh.structure.TableOfContentsStyle attribute), 69  
 title (rinoh.table.TableCellStyle attribute), 86  
 title (rinoh.table.TableWithCaptionStyle attribute), 146  
 title (rinoh.templates.book.Book attribute), 93  
 title\_page (rinoh.templates.book.Book attribute), 94  
 TitlePageTemplate (class in rinoh.template), 155  
 TitlePartTemplate (class in rinoh.template), 150  
 to\_string() (rinoh.index.InlineIndexTarget method), 108  
 to\_string() (rinoh.text.StyledText method), 113  
 top\_margin (rinoh.template.BodyPageTemplate attribute), 155  
 top\_margin (rinoh.template.BodyPageTemplateBase attribute), 153  
 top\_margin (rinoh.template.PageTemplateBase attribute), 157  
 top\_margin (rinoh.template.TitlePageTemplate attribute), 156  
 type (rinoh.reference.NoteMarkerStyle attribute), 51  
 type (rinoh.reference.ReferenceStyle attribute), 116  
 Typeface (class in rinoh.font), 105

typeface (rinoh.image.CaptionStyle attribute), 30  
typeface (rinoh.paragraph.ParagraphStyle attribute), 40  
typeface (rinoh.reference.NoteMarkerStyle attribute), 51  
typeface (rinoh.reference.ReferenceStyle attribute), 117  
typeface (rinoh.reference.ReferencingParagraphStyle attribute), 49  
typeface (rinoh.structure.HeadingStyle attribute), 59  
typeface (rinoh.structure.ListItemLabelStyle attribute), 67, 129  
typeface (rinoh.structure.ListOfEntryStyle attribute), 140  
typeface (rinoh.structure.ListOfStyle attribute), 135  
typeface (rinoh.structure.TableOfContentsEntryStyle attribute), 77  
typeface (rinoh.structure.TableOfContentsStyle attribute), 72  
typeface (rinoh.text.TextStyle attribute), 42

## U

update() (rinoh.style.StyledMatcher method), 144

## V

vertical\_align (rinoh.table.TableCellStyle attribute), 86  
VerticalAlign (class in rinoh.table), 147

## W

WarnFlowable (class in rinoh.flowable), 102  
width (rinoh.flowable.FloatStyle attribute), 25  
width (rinoh.flowable.FlowableStyle attribute), 18  
width (rinoh.flowable.GroupedFlowablesStyle attribute), 23  
width (rinoh.flowable.LabeledFlowableStyle attribute), 21  
width (rinoh.image.CaptionStyle attribute), 28  
width (rinoh.image.FigureStyle attribute), 32  
width (rinoh.image.ImageArgs attribute), 107  
width (rinoh.index.IndexStyle attribute), 35  
width (rinoh.inline.InlineFlowableStyle attribute), 44  
width (rinoh.paragraph.ParagraphStyle attribute), 37  
width (rinoh.reference.NoteStyle attribute), 120  
width (rinoh.reference.ReferencingParagraphStyle attribute), 46  
width (rinoh.structure.AdmonitionStyle attribute), 79  
width (rinoh.structure.HeadingStyle attribute), 56  
width (rinoh.structure.HorizontalRuleStyle attribute), 82  
width (rinoh.structure.ListItemLabelStyle attribute), 64, 127  
width (rinoh.structure.ListOfEntryStyle attribute), 137  
width (rinoh.structure.ListOfStyle attribute), 132  
width (rinoh.structure.ListStyle attribute), 61, 123  
width (rinoh.structure.SectionStyle attribute), 54  
width (rinoh.structure.TableOfContentsEntryStyle attribute), 74  
width (rinoh.structure.TableOfContentsStyle attribute), 69  
width (rinoh.table.TableCellStyle attribute), 86  
width (rinoh.table.TableStyle attribute), 84  
width (rinoh.table.TableWithCaptionStyle attribute), 144  
wrap\_label (rinoh.flowable.LabeledFlowableStyle attribute), 21  
wrap\_label (rinoh.reference.NoteStyle attribute), 120

## Y

y\_offset() (rinoh.text.StyledText method), 114