The docext Package*

Mingyu Xia†

Released 2026-11-08 v0.2F

Abstract

The docext package is an extension for documenting the LaTeX source files. It is based on the doc package and works smoothly under the l3doc, ltxdoc, and article classes.

1 Loading & Configuring the Package

Write the line in the preamble

```
\usepackage[options]{docext}
```

while the package ⟨*options*⟩ accepts the following keys

`color` = ⟨*l3color*|*HTML*⟩: The color of keys (Default: `black`).

`font` = ⟨*font macros*⟩: Font of the keys.

`delimiter` = ⟨*string*⟩: Delimiter between the keys.

`ratio` = ⟨*fp num*⟩: Ratio of the boxes' widths in `codemo` environment (Default: `0.625`).

Apart from configuring the package with the package options, users can also use the `\docextset` function within the preamble and the context.

`\docextset` `\docextset {`⟨*options*⟩`}`

`New: 2026-06-07` The functions used in this package *after* `\docextset` will be influenced. The ⟨*options*⟩ accepts the same keys above.

*https://ctan.org/pkg/docext, https://github.com/myhsia/docext
†xiamingyu@westlake.edu.cn

2 User's Interfaces

\keys `\keys [⟨options⟩] {⟨comma list⟩}`

New: 2026-06-07
Updated: 2026-06-27

Typeset the keys with specific format. The ⟨*options*⟩ accepts the `delimiter`, `font`, `color` keys referred in section 1 and another `emph` key to assign # of the key will be emphasised.
The initial value of the `delimiter` key is `,\,\allowbreak` in this function.

```
\keys [font = \sl] {keya, keyb, keyc}\\
\keys [emph = 2, delimiter = \textbar]
      {true, false, not given}
```

keya, *keyb*, *keyc*
true|false|not given

\val `\val [⟨options⟩] {⟨comma list⟩}`

New: 2026-06-13

Typeset the value with specific format and surround with `\meta`. The ⟨*options*⟩ accepts the `delimiter`, `font`, `color` keys referred in section 1.
The initial value of the `delimiter` key is `\textbar` in this function.

\TF Shortcut for typeset Boolean-choice value.

New: 2026-06-07

```
\TF, \TTF, \TFF.
```

true|false, true|false, true|false.

\ttwd `\hspace \⟨fontsize⟩ttwd`

New: 2026-06-13

Output the dimension of the width of one typewriter letter. The ⟨*fontsize*⟩ prefix can be chosen as `tiny`, `scriptsize`, `footnotesize`, `small`, `normalsize`, `large`, `Large`, `LARGE`, `huge`, `Huge`.

`codemo`, `codemo*` (*env.*) Typeset the code on the left hand side and then execute the code in the right hand side, while the star version will put the code above the demo.

```
\begin{codemo(*)}
  Code needs to be typeset in verbatim and demoed
\end{codemo(*)}
```

`keyval` (*env.*) The effect is the same as the demos in the introductions of the functions above. Itemize-type environment for typesetting the keys.

```
\begin{keyval}
  \item [\keys{keya, keyb, keyc}]\meta{foo}
\end{keyval}
```

The effect is the same as the package options introduction of section 1 in this manual.

2.1 Tagged PDF

The `codemo` environment cooperates with the LaTeX tagging project. A document that declares

```
\DocumentMetadata{tagging=on}
```

before `\documentclass` gets the code panel tagged as code, the result panel as a text block, and the frames around them as artifacts. Under the l3doc class, load latex-lab-testphase-l3doc as well; that package supplies the tagging support of the class itself.

A The source code

Using codedoc as the namespace.

```
⟨@@=codedoc⟩
```

Start the optionlist package for l3docstrip.

```
⟨*package⟩
```

A.1 Variables

\l__codedoc_keys_typeset_tl
__codedoc_keys_typeset_box

Variables for storing the output of __codedoc_keys_aux:nnnnn.

```
\tl_new:N  \l__codedoc_keys_typeset_tl
\box_new:N \l__codedoc_keys_typeset_box
```

(*End of definition for* \l__codedoc_keys_typeset_tl *and* \l__codedoc_keys_typeset_box.)

\g__codedoc_true_tl
\g__codedoc_false_tl
\g__codedoc_vbar_tl

Constant variables used in __codedoc_keys_TF_aux:n.

```
\tl_const:Nn \g__codedoc_true_tl  { true  }
\tl_const:Nn \g__codedoc_false_tl { false }
\tl_const:Nn \g__codedoc_vbar_tl
  { \sys_if_engine_opentype:TF { \textbar } { | } }
```

(*End of definition for* \g__codedoc_true_tl, \g__codedoc_false_tl, *and* \g__codedoc_vbar_tl.)

\l__codedoc_codemo_iow
\l__codedoc_codemo_ior
\l__codedoc_codemo_int
\l__codedoc_codemo_name_tl

Variables used in __codedoc_codemo_start: and __codedoc_codemo_end:Nn.

```
\iow_new:N   \l__codedoc_codemo_iow
\ior_new:N   \l__codedoc_codemo_ior
\int_new:N   \l__codedoc_codemo_int
\tl_const:Nn \l__codedoc_codemo_name_tl {\jobname.codemo.vrb}
```

(*End of definition for* \l__codedoc_codemo_iow *and others.*)

\l__codedoc_ttwd_box

Box for saving one typewriter letter temporally.

```
\box_new:N \l__codedoc_ttwd_box
```

(*End of definition for* \l__codedoc_ttwd_box.)

A.2 Messages

Raise when the width of the keys is wider than 1.2\marginparwidth.

```
\msg_new:nnn { docext } { Overfull ~ \string \keys }
  {
    Keys ~ wider ~ than ~ 1.2 \string \marginparwidth \iow_newline:
    \space\space Exceed ~ the ~ ability ~ of ~ scaling
  }
```

A.3 Options and configuration

__color_named_docolor_prop
\g__codedoc_keys_font_tl
\g__codedoc_keys_delimiter_tl
g__codedoc_codemo_ratio_fp

Key–value definitions of the package options, separate the keys into two groups. The first one, `@@ / pkgopt / keys`, is also used in `\keys`.

```
\keys_define:nn { __codedoc / pkgopt / keys }
  {
    color       .code:n    =
      {
        \color_if_exist:nTF {#1}
          { \color_set:nn  { docolor }          {#1} }
          { \color_set:nnn { docolor } { HTML } {#1} }
      },
      color     .initial:n = { black },
    font        .tl_set:N  = \g__codedoc_keys_font_tl,
    delimiter   .tl_set:N  = \g__codedoc_keys_delimiter_tl,
  }
\keys_define:nn { __codedoc / pkgopt / codemo }
  {
    ratio       .fp_set:N  = \g__codedoc_codemo_ratio_fp,
      ratio     .initial:n = { .625 },
  }
```

(*End of definition for* `\l__color_named_docolor_prop` *and others.*)

\l__codedoc_keys_emph_int

Key–value definitions special for the `\keys` macro.

```
\keys_define:nn { __codedoc / keys }
  { emph        .int_set:N = \l__codedoc_keys_emph_int }
```

(*End of definition for* `\l__codedoc_keys_emph_int`.)

Process the class options.

```
\ProcessKeyOptions [ __codedoc / pkgopt / keys,
                     __codedoc / pkgopt / codemo ]
```

\docextset

To control the variables.

```
\DeclareDocumentCommand \docextset { m }
  {
    \keys_set_known:nn { __codedoc / pkgopt / keys   } {#1}
    \keys_set_known:nn { __codedoc / pkgopt / codemo } {#1}
  }
```

(*End of definition for* `\docextset`. *This function is documented on page* 1.)

A.4 Class and package loading / tweaks

Load the dependencies.

```
\RequirePackage { xcolor, enumitem, verbatim }
```

Overwrite `\prepare@family@series@update` again under pLaTeX and upLaTeX to resolve the error raised by `\settowidth \MacroIndent` in the doc package.

```
\bool_lazy_or:nnT { \sys_if_engine_ptex_p: } { \sys_if_engine_uptex_p: }
  {
    \def\prepare@family@series@update#1#2{%
      \def\@currentmetafamily{#1}%
      \if@forced@series
        \fontfamily#2%
      \else
        \expand@font@defaults
        \let\target@series@value\@empty
        \expandafter\edef\csname ??def@ult\endcsname{\f@family}%
        \let\@elt\update@series@target@value
            \@meta@family@list
            \@elt{??}%
        \let\@elt\relax
        \fontfamily#2%
        \ifx\target@series@value\@empty
        \else
          \ifx \f@series\target@series@value
          \else
            \maybe@load@fontshape
            \series@maybe@drop@one@m\target@series@value\f@series
          \fi
        \fi
      \fi
    }
  }
```

Patch the font of the `macrocode` environment.

```
\hook_gput_code:nnn { cmd / macro@font / after } { . } { \ttfamily }
```

Patch the macro `\meta` in the l3doc class and the doc package. Since the l3doc package has already loaded the doc package, a branch is required.

```
\cs_if_exist:cTF { ver@l3doc.cls }
```

Under the l3doc class, surround `\hbox:n` to the argument of the kernel function `\__codedoc_meta_original:n` of the `\meta` macro. Suppress the warning raised

by l3doc (checking l3syntax), and reset the counter: `section` at the beginning of the `documentation` environment.

```
{
  \cs_set_eq:NN \__codedoc_meta_patch:n \__codedoc_meta_original:n
  \cs_set_protected:Npn \__codedoc_meta_original:n #1
    { \__codedoc_meta_patch:n { \hbox:n {#1} } }
  \msg_redirect_module:nnn { l3doc } { warning } { none }
  \hook_gput_code:nnn { env / documentation / before } { . }
    { \setcounter { section } { 0 } }
}
```

Under other classes, load the doc package then take the same trick as in the l3doc class.

```
{
  \RequirePackage { doc }
  \cs_set_eq:NN \__codedoc_meta_patch:n \meta
  \DeclareDocumentCommand \meta { m }
    { \__codedoc_meta_patch:n { \hbox:n {#1} } }
}
```

A.5 User's interfaces

`\keys` LaTeX 2_ε user's interface for input keys quickly.

```
\DeclareDocumentCommand \keys { O {} m }
  {
    \group_begin:
      \keys_set_known:nn { __codedoc / pkgopt / keys }
        { delimiter = { ,\,\allowbreak }, #1 }
      \keys_set_known:nn { __codedoc / keys          } {#1}
      \tl_set:Nn \l__codedoc_keys_typeset_tl
        {
          \__codedoc_keys_aux:nnnnn {#2}
            { \l__codedoc_keys_emph_int             }
            { \textup \g__codedoc_keys_delimiter_tl }
            { \g__codedoc_keys_font_tl \ttfamily    }
            { docolor                         }
        }
      \hbox_set:Nn \l__codedoc_keys_typeset_box { \l__codedoc_keys_typeset_tl }
      \tl_if_eq:NnTF \@currenvir { keyval }
        {
          \dim_compare:nNnT
            { \box_wd:N \l__codedoc_keys_typeset_box } > { \marginparwidth }
            {
              \dim_compare:nNnT
```

```
                { \box_wd:N \l__codedoc_keys_typeset_box } > { 1.2 \marginparwidth }
                { \msg_warning:nn { docext } { Overfull ~ \string \keys }    }
              \box_scale:Nnn \l__codedoc_keys_typeset_box { 5 / 6 } { 1 }
            }
          \box_use_drop:N \l__codedoc_keys_typeset_box
          \g__codedoc_keys_font_tl \ttfamily \, = \,
        } { \l__codedoc_keys_typeset_tl }
    \group_end:
  }
```

(*End of definition for* `\keys`. *This function is documented on page* 2.)

`\val` LaTeX 2_ε user's interface for input value quickly.

```
\DeclareDocumentCommand \val { O {} m }
  {
    \group_begin:
      \keys_set_known:nn { __codedoc / pkgopt / keys }
        { delimiter = { \g__codedoc_vbar_tl \allowbreak }, #1 }
      \meta
        {
          \__codedoc_keys_aux:nnnnn {#2}
            { 0                                }
            { \textup \g__codedoc_keys_delimiter_tl }
            { \g__codedoc_keys_font_tl \ttfamily    }
            { docolor                          }
        }
    \group_end:
  }
```

(*End of definition for* `\val`. *This function is documented on page* 2.)

`\__codedoc_keys_aux:nnnnn` `\__codedoc_keys_emph:n` Auxiliary commands of the `\keys` macro.

- comma list,
- font,
- # of the element to be emphasised,
- delimiter,
- color (works only for the key),

```
\cs_new_protected_nopar:Npn \__codedoc_keys_aux:nnnnn #1#2#3#4#5
  {
    \group_begin:
      \seq_set_from_clist:Nn \l_tmpa_seq {#1}
      \int_if_zero:nF {#2}
        {
          \seq_gset_item:Nnn \l_tmpa_seq {#2}
```

```
            { \__codedoc_keys_emph:n { \clist_item:nn {#1} {#2} } }
        }
      \seq_set_map:NNn \l_tmpb_seq \l_tmpa_seq
        {
          \color_group_begin:
            \exp_args:No \color_select:n {#5} ##1
          \color_group_end:
        }
      #4 \seq_use:Nn \l_tmpb_seq {#3}
    \group_end:
  }
\cs_new_protected_nopar:Npn \__codedoc_keys_emph:n #1
  {
    \group_begin:
      \upshape \bfseries \exp_args:No \color_select:n { red } #1
    \group_end:
  }
```

(*End of definition for* `\__codedoc_keys_aux:nnnnn` *and* `\__codedoc_keys_emph:n`.)

`\TF` LaTeX 2_ε user's interface for input Boolean-choice value quickly.

`\TTF`
`\TFF`

```
\DeclareDocumentCommand \TF  {} { \__codedoc_keys_TF_aux:n { 0 } }
\DeclareDocumentCommand \TTF {} { \__codedoc_keys_TF_aux:n { 1 } }
\DeclareDocumentCommand \TFF {} { \__codedoc_keys_TF_aux:n { 2 } }
```

(*End of definition for* `\TF`, `\TTF`, *and* `\TFF`. *These functions are documented on page* 2.)

`\__codedoc_keys_TF_aux:n` Auxiliary commands of the `\TF`, `\TTF`, `\TFF` macros.

```
\cs_new_protected_nopar:Npn \__codedoc_keys_TF_aux:n #1
  {
    \__codedoc_keys_aux:nnnnn
      { \g__codedoc_true_tl,       \g__codedoc_false_tl } {#1}
      { \textup \g__codedoc_vbar_tl } { \ttfamily } { docolor }
  }
```

(*End of definition for* `\__codedoc_keys_TF_aux:n`.)

`keyval` (*env.*) Itemize-type environment for typesetting the keys. Using the interfaces provided by the enumitem package.

The tagging emulation of enumitem does not compute the widest label the way enumitem does. The label width is therefore pinned to `\marginparwidth`, the width `\keys` already scales against, and the label hangs left by the same amount; both tagged and untagged documents then keep one = column.

```
\newlist{keyval}{itemize}{10}
\setlist[keyval]
```

```
  {
    leftmargin  = 0pt,
    labelsep    = 0pt,
    labelwidth  = \marginparwidth,
    labelindent = -\marginparwidth,
  }
```

`codemo, codemo*` (*env.*) Environment for typesetting the code within a `verbatim` and a `demo` environment at the same time.

```
\DeclareDocumentEnvironment { codemo  } { } { \__codedoc_codemo_start: }
  {
    \__codedoc_codemo_end:Nn \c_false_bool
      { \g__codedoc_codemo_ratio_fp }
  }
\DeclareDocumentEnvironment { codemo* } { } { \__codedoc_codemo_start: }
  {
    \__codedoc_codemo_end:Nn \c_true_bool
      { \g__codedoc_codemo_ratio_fp }
  }
```

`\__codedoc_codemo_start:`
`\__codedoc_codemo_end:Nn` Auxiliary commands of the `codemo` environment.

```
\cs_new_protected:Npn \__codedoc_codemo_start:
  {
```

The I/O streams part[1].

```
    \group_begin:
      \@bsphack
      \iow_open:Nn \l__codedoc_codemo_iow { \l__codedoc_codemo_name_tl }
      \let \do \char_set_catcode_other:N
      \dospecials \char_set_catcode_active:N \^^M
      \cs_set:Npn \verbatim@processline
        {
          \exp_args:NNo \iow_now:Nn
            \l__codedoc_codemo_iow { \int_use:N \verbatim@line }
        }
      \verbatim@start
  }
\cs_new_protected:Npn \__codedoc_codemo_end:Nn #1#2
  {
      \iow_close:N \l__codedoc_codemo_iow
      \@esphack
    \group_end:
    \char_set_catcode_comment:N \%
```

[1] By modifying @AlphaZTX's `phy2docdef.tex` magically using LaTeX3.

```
    \ior_open:Nn \l__codedoc_codemo_ior { \l__codedoc_codemo_name_tl }
    \int_zero:N \l__codedoc_codemo_int
    \ior_map_inline:Nn \l__codedoc_codemo_ior
      { \int_incr:N \l__codedoc_codemo_int }
```

The typeset part. Both panel widths are arithmetic expressions, so `\dim_eval:n` turns each into a single dimension: `\setlength` understands the subtraction only with calc loaded, which the l3doc class does but article does not. The result panel is a `\parbox`, which takes its width from its content, so the body starts with `\mode_leave_vertical:`: a body that is a single vertical box, as a display-like environment produces, would otherwise leave the frame as wide as that box.

```
    \par \smallskip
    \group_begin:
      \dim_set:Nn \parindent { \c_zero_skip }
      \fbox
        {
          \bool_if:NT #1 { \null \skip_horizontal:n { 2\ttwd } }
          \parbox [ c ] [ \l__codedoc_codemo_int \baselineskip + .48ex ]
            {
              \dim_eval:n
                {
                  \fp_eval:n { \bool_if:NTF #1 { 1 } {#2} * .96 }
                  \linewidth \bool_if:NT #1 { -.8pt }
                }
            } { \small \verbatiminput \l__codedoc_codemo_name_tl }
        }
      \bool_if:NTF #1
        { \par \skip_vertical:n { -1.5pt } } { \hfill }
      \fbox
        {
          \null \skip_horizontal:n { 2\ttwd }
          \parbox [ c ] [ \l__codedoc_codemo_int \baselineskip + .48ex ]
            {
              \dim_eval:n
                {
                  \fp_eval:n { \bool_if:NTF #1 { 1 } { (1 - #2) } * .96 }
                  \linewidth \bool_if:NTF #1 { -.8pt } { - 2\ttwd }
                }
            } { \mode_leave_vertical: \file_input:n  { \l__codedoc_codemo_name_tl } }
        }
      \par \smallskip
    \group_end:
    \char_set_catcode_ignore:N \%
  }
```

(*End of definition for* `\__codedoc_codemo_start:` *and* `\__codedoc_codemo_end:Nn`.)

`\ttwd` Dimensions of the width of one typewriter letter under different fontsizes.

```
\clist_map_inline:nn
  {
    tiny,  scriptsize, footnotesize, small, \@empty,
    large, Large,      LARGE,        huge,  Huge
  }
  {
    \dim_new:c { #1 ttwd }
    \cs_new_protected_nopar:cpn { update#1ttwd }
      {
        \hbox_set:Nn \l__codedoc_ttwd_box { \use:c {#1} \ttfamily x }
        \dim_gset:cn { #1 ttwd } { \box_wd:N \l__codedoc_ttwd_box }
      }
    \hook_gput_code:nnn { begindocument / before } { . }
      { \use:c { update#1ttwd } }
  }
```

(*End of definition for* `\ttwd`. *This function is documented on page* 2.)

End the optionlist package for l3docstrip.

```
⟨/package⟩
```

Restore the namespace.

```
⟨@@=⟩
```

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

Symbols

\% 203, 239
\, ... 92, 114
\^ 190

A

\allowbreak 92, 122

B

\baselineskip 214, 228
\bfseries 154
bool commands:
\bool_if:NTF 213, 218, 219, 223, 232, 233
\bool_lazy_or:nnTF 46
\c_false_bool 176
\c_true_bool 181
box commands:
\box_new:N 4, 13
\box_scale:Nnn 111
\box_use_drop:N 113
\box_wd:N 106, 109, 251

C

char commands:
\char_set_catcode_active:N . . 190
\char_set_catcode_comment:N . 203
\char_set_catcode_ignore:N . . 239
\char_set_catcode_other:N . . . 189
clist commands:
\clist_item:nn 140
\clist_map_inline:nn 241
codedoc internal commands:
__codedoc_codemo_end:Nn . . . .
. . . . . 4, 176, 181, 184, 198
\l__codedoc_codemo_int . . . . .
. . . . . 9, 205, 207, 214, 228
\l__codedoc_codemo_ior 9, 204, 206
\l__codedoc_codemo_iow . . . . .
. . . . . 9, 188, 194, 200
\l__codedoc_codemo_name_tl . .
. . . . . 9, 188, 204, 221, 235
\g__codedoc_codemo_ratio_fp .
. . . . . 19, 177, 182
__codedoc_codemo_start: . . . .
. . . . . 4, 174, 179, 184, 184
\g__codedoc_false_tl 5, 163
__codedoc_keys_aux:nnnnn . . .
. . . . . 4, 96, 125, 133, 133, 162
\g__codedoc_keys_delimiter_tl
. . . . . 19, 98, 127
__codedoc_keys_emph:n 133, 140, 151
\l__codedoc_keys_emph_int . 36, 97
\g__codedoc_keys_font_tl . . . .
. . . . . 19, 99, 114, 128
__codedoc_keys_TF_aux:n . . . .
. . . . . 4, 157, 158, 159, 160, 160
\l__codedoc_keys_typeset_box
. . . . . 3, 102, 106, 109, 111, 113
\l__codedoc_keys_typeset_tl .
. . . . . 3, 94, 102, 115
__codedoc_meta_original:n . .
. . . . . 6, 75, 76
__codedoc_meta_patch:n . . . .
. . . . . 75, 77, 84, 86
\g__codedoc_true_tl 5, 163
\l__codedoc_ttwd_box . . 13, 250, 251
\g__codedoc_vbar_tl . . . 5, 122, 164
codemo, codemo* (env.) 2, 10
color commands:
\color_group_begin: 144
\color_group_end: 146
\color_if_exist:nTF 23
\color_select:n 145, 154
\color_set:nn 24
\color_set:nnn 25
color internal commands:
\l__color_named_docolor_prop . 19

cs commands:
\cs_if_exist:NTF 73
\cs_new_protected:Npn . . . 184, 198
\cs_new_protected_nopar:Npn 133, 151, 160, 248
\cs_set:Npn 191
\cs_set_eq:NN 75, 84
\cs_set_protected:Npn 76
\csname 55

D

\DeclareDocumentCommand 40, 85, 88, 118, 157, 158, 159
\DeclareDocumentEnvironment . 174, 179
\def 48, 49
dim commands:
\dim_compare:nNnTF 105, 108
\dim_eval:n *11*, 216, 230
\dim_gset:Nn 251
\dim_new:N 247
\dim_set:Nn 210
\do 189
\docextset *1*, 40
\dospecials 190

E

\edef 55
\else 52, 62, 64
\endcsname 55
environments:
codemo, codemo* *2*, *10*
keyval *2*, *9*
exp commands:
\exp_args:NNo 193
\exp_args:No 145, 154
\expandafter 55

F

\fbox 211, 225
\fi 67, 68, 69
file commands:
\file_input:n 235
\fontfamily 51, 60
fp commands:
\fp_eval:n 218, 232

G

group commands:
\group_begin: 90, 120, 135, 153, 186, 209
\group_end: 116, 131, 149, 155, 202, 238

H

hbox commands:
\hbox:n *6*, 77, 86
\hbox_set:Nn 102, 250
\hfill 224
hook commands:
\hook_gput_code:nnn . . . 72, 79, 253

I

\ifx 61, 63
int commands:
\int_if_zero:nTF 137
\int_incr:N 207
\int_new:N 11
\int_use:N 194
\int_zero:N 205
ior commands:
\ior_map_inline:Nn 206
\ior_new:N 10
\ior_open:Nn 204
iow commands:
\iow_close:N 200
\iow_new:N 9
\iow_newline: 16
\iow_now:Nn 193
\iow_open:Nn 188

J

\jobname 12

K

\keys *2*, *9*, 14, 88
keys commands:
\keys_define:nn 19, 31, 36
\keys_set_known:nn 42, 43, 91, 93, 121
keyval (env.) *2*, *9*

L

\let 54, 56, 59, 189
\linewidth 219, 233

M

\marginparwidth . . 16, 106, 109, 171, 172
\meta 84, 85, 123
mode commands:
\mode_leave_vertical: *11*, 235
msg commands:
\msg_new:nnn 14
\msg_redirect_module:nnn 78
\msg_warning:nn 110

N

\newlist 166
\null 213, 227

P

\par 208, 224, 237
\parbox 214, 228
\parindent 210
\ProcessKeyOptions 38

R

\relax 59
\RequirePackage 45, 83

S

seq commands:
\seq_gset_item:Nnn 139
\seq_set_from_clist:Nn 136
\seq_set_map:NNn 142
\seq_use:Nn 148
\l_tmpa_seq 136, 139, 142
\l_tmpb_seq 142, 148
\setcounter 80
\setlist 167
skip commands:
\skip_horizontal:n 213, 227
\skip_vertical:n 224
\c_zero_skip 210
\small 221
\smallskip 208, 237
\space 17
\string 14, 16, 110
sys commands:
\sys_if_engine_opentype:TF 8
\sys_if_engine_ptex_p: 46
\sys_if_engine_uptex_p: 46

T

TEX and LATEX 2ε commands:
\@bsphack 187
\@currentmetafamily 49
\@currenvir 103
\@elt 56, 58, 59
\@empty 54, 61, 243
\@esphack 201
\@meta@family@list 57
\⟨*fontsize*⟩ttwd *2*
\docextset *1*
\documentclass *3*
\expand@font@defaults 53
\f@family 55
\f@series 63, 66
\hspace *2*
\if@forced@series 50
\keys *2*, *5*, *8*
\marginparwidth *4*, *9*
\maybe@load@fontshape 65
\meta *2*, *6*
\parbox *11*
\prepare@family@series@update *6*, 48
\series@maybe@drop@one@m 66
\setlength *11*
\target@series@value . 54, 61, 63, 66
\textbar *2*
\TF *9*
\TFF *9*
\TTF *9*
\update@series@target@value . . 56
\val *2*
\verbatim@line 194
\verbatim@processline 191
\verbatim@start 196
\textbar 8
\textup 98, 127, 164
\TF *2*, 157
\TFF 157
tl commands:
\tl_const:Nn 5, 6, 7, 12
\tl_if_eq:NnTF 103
\tl_new:N 3
\tl_set:Nn 94

\TTF 157
\ttfamily 72, 99, 114, 128, 164, 250
\ttwd 2, 213, 227, 233, 241

U

\upshape 154
use commands:
\use:N 250, 254

V

\val 2, 118
\verbatiminput 221