

The `getttitlestring` package

Heiko Oberdiek

<heiko.oberdiek at gmail.com>

2010/12/03 v1.4

Abstract

The \LaTeX package addresses packages that are dealing with references to titles (`\section`, `\caption`, ...). The package tries to remove `\label` and other commands from title strings.

Contents

1	Documentation	2
1.1	Macros	2
1.2	Options	2
2	Implementation	3
2.1	Options	4
2.2	<code>\GetTitleString</code>	5
2.2.1	Expand method	5
2.2.2	Non-expand method	6
3	Test	8
3.1	Catcode checks for loading	8
3.2	Test of non-expand method	9
4	Installation	10
4.1	Download	10
4.2	Bundle installation	10
4.3	Package installation	11
4.4	Refresh file name databases	11
4.5	Some details for the interested	11
5	Catalogue	12
6	References	12
7	History	13
	[2009/12/08 v1.0]	13
	[2009/12/12 v1.1]	13
	[2009/12/13 v1.2]	13
	[2009/12/18 v1.3]	13
	[2010/12/03 v1.4]	13
8	Index	13

1 Documentation

1.1 Macros

`\GetTitleStringSetup {⟨key value list⟩}`

The options are given as comma separated key value pairs. See section [1.2](#).

`\GetTitleString {⟨text⟩}`
`\GetTitleStringExpand {⟨text⟩}`
`\GetTitleStringNonExpand {⟨text⟩}`

Macro `\GetTitleString` tries to remove unwanted stuff from `⟨text⟩` the result is stored in Macro `\GetTitleStringResult`. Two methods are available:

`\GetTitleStringExpand`: The `⟨text⟩` is expanded in a context where the unwanted macros are redefined to remove themselves. This is the method used in packages `titleref` [\[2\]](#), `zref-titleref` [\[3\]](#) or class `memoir` [\[1\]](#). `\protect` is supported, but fragile material might break.

`\GetTitleStringNonExpand`: The `⟨text⟩` is not expanded. Thus the removal of unwanted material is more difficult. It is especially removed at the start of the `⟨text⟩` and spaces are removed from the end. Currently only `\label` is removed in the whole string, if it is not hidden inside curly braces or part of macro definitions. Thus the removal of unwanted stuff might not be complete, but fragile material will not break. (But the result string can break at a later time, of course).

Option `expand` controls which method is used by macro `\GetTitleString`.

`\GetTitleStringDisableCommands {⟨code⟩}`

The `⟨code⟩` is called right before the text is expanded in `\GetTitleStringExpand`. Additional definitions can be given for macros that should be removed. Keep in mind that expansion means that the definitions must work in expandable context. Macros like `\@ifstar` or `\@ifnextchar` or optional arguments will not work. The macro names in `⟨code⟩` may contain the at sign `@`, it has catcode 11 (letter).

1.2 Options

`expand`: Boolean option, takes values `true` or `false`. No value means `true`. The option specifies the method to remove unwanted stuff from the title string, see below.

Options can be set at the following places:

- `\usepackage`
- Configuration file `getttitlestring.cfg`.
- `\GetTitleStringSetup`

2 Implementation

```

1 \(*package)
Reload check, especially if the package is not used with LATEX.
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3 \catcode13=5 % ^~M
4 \endlinechar=13 %
5 \catcode35=6 % #
6 \catcode39=12 % '
7 \catcode44=12 % ,
8 \catcode45=12 % -
9 \catcode46=12 % .
10 \catcode58=12 % :
11 \catcode64=11 % @
12 \catcode123=1 % {
13 \catcode125=2 % }
14 \expandafter\let\expandafter\x\csname ver@getttitlestring.sty\endcsname
15 \ifx\x\relax % plain-TeX, first loading
16 \else
17 \def\empty{}%
18 \ifx\x\empty % LaTeX, first loading,
19 % variable is initialized, but \ProvidesPackage not yet seen
20 \else
21 \expandafter\ifx\csname PackageInfo\endcsname\relax
22 \def\x#1#2{%
23 \immediate\write-1{Package #1 Info: #2.}%
24 }%
25 \else
26 \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
27 \fi
28 \x{getttitlestring}{The package is already loaded}%
29 \aftergroup\endinput
30 \fi
31 \fi
32 \endgroup%

```

Package identification:

```

33 \begingroup\catcode61\catcode48\catcode32=10\relax%
34 \catcode13=5 % ^~M
35 \endlinechar=13 %
36 \catcode35=6 % #
37 \catcode39=12 % '
38 \catcode40=12 % (
39 \catcode41=12 % )
40 \catcode44=12 % ,
41 \catcode45=12 % -
42 \catcode46=12 % .
43 \catcode47=12 % /
44 \catcode58=12 % :
45 \catcode64=11 % @
46 \catcode91=12 % [
47 \catcode93=12 % ]
48 \catcode123=1 % {
49 \catcode125=2 % }
50 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
51 \def\x#1#2#3[#4]{\endgroup
52 \immediate\write-1{Package: #3 #4}%
53 \xdef#1{#4}%
54 }%

```

```

55 \else
56   \def\x#1#2[#3]{\endgroup
57     #2[#3]}%
58   \ifx#1\@undefined
59     \xdef#1{#3}%
60   \fi
61   \ifx#1\relax
62     \xdef#1{#3}%
63   \fi
64 }%
65 \fi
66 \expandafter\x\csname ver@getttitlestring.sty\endcsname
67 \ProvidesPackage{getttitlestring}%
68 [2010/12/03 v1.4 Cleanup title references (H0)]%
69 \begingroup\catcode61\catcode48\catcode32=10\relax%
70 \catcode13=5 % ^~M
71 \endlinechar=13 %
72 \catcode123=1 % {
73 \catcode125=2 % }
74 \catcode64=11 % @
75 \def\x{\endgroup
76   \expandafter\edef\csname GTS@AtEnd\endcsname{%
77     \endlinechar=\the\endlinechar\relax
78     \catcode13=\the\catcode13\relax
79     \catcode32=\the\catcode32\relax
80     \catcode35=\the\catcode35\relax
81     \catcode61=\the\catcode61\relax
82     \catcode64=\the\catcode64\relax
83     \catcode123=\the\catcode123\relax
84     \catcode125=\the\catcode125\relax
85   }%
86 }%
87 \x\catcode61\catcode48\catcode32=10\relax%
88 \catcode13=5 % ^~M
89 \endlinechar=13 %
90 \catcode35=6 % #
91 \catcode64=11 % @
92 \catcode123=1 % {
93 \catcode125=2 % }
94 \def\TMP@EnsureCode#1#2{%
95   \edef\GTS@AtEnd{%
96     \GTS@AtEnd
97     \catcode#1=\the\catcode#1\relax
98   }%
99   \catcode#1=#2\relax
100 }
101 \TMP@EnsureCode{42}{12}% *
102 \TMP@EnsureCode{44}{12}% ,
103 \TMP@EnsureCode{45}{12}% -
104 \TMP@EnsureCode{46}{12}% .
105 \TMP@EnsureCode{47}{12}% /
106 \TMP@EnsureCode{91}{12}% [
107 \TMP@EnsureCode{93}{12}% ]
108 \edef\GTS@AtEnd{\GTS@AtEnd\noexpand\endinput}

```

2.1 Options

```

109 \RequirePackage{kvoptions}[2009/07/17]
110 \SetupKeyvalOptions{%

```

```

111 family=gettitlestring,%
112 prefix=GTS@%
113 }
114 \newcommand*{\GetTitleStringSetup}{%
115   \setkeys{gettitlestring}%
116 }
117 \DeclareBoolOption{expand}
118 \InputIfFileExists{gettitlestring.cfg}{}{}
119 \ProcessKeyvalOptions*\relax

```

2.2 \GetTitleString

\GetTitleString

```

120 \newcommand*{\GetTitleString}{%
121   \ifGTS@expand
122     \expandafter\GetTitleStringExpand
123   \else
124     \expandafter\GetTitleStringNonExpand
125   \fi
126 }

```

\GetTitleStringExpand

```

127 \newcommand{\GetTitleStringExpand}[1]{%
128   \def\GetTitleStringResult{#1}%
129   \begingroup
130     \GTS@DisablePredefinedCmds
131     \GTS@DisableHook
132     \edef\x{\endgroup
133       \noexpand\def\noexpand\GetTitleStringResult{%
134         \GetTitleStringResult
135       }%
136     }%
137   \x
138 }

```

\GetTitleString

```

139 \newcommand{\GetTitleStringNonExpand}[1]{%
140   \def\GetTitleStringResult{#1}%
141   \global\let\GTS@GlobalString\GetTitleStringResult
142   \begingroup
143     \GTS@RemoveLeft
144     \GTS@RemoveRight
145   \endgroup
146   \let\GetTitleStringResult\GTS@GlobalString
147 }

```

2.2.1 Expand method

\GTS@DisablePredefinedCmds

```

148 \def\GTS@DisablePredefinedCmds{%
149   \let\label\@gobble
150   \let\zlabel\@gobble
151   \let\zref@label\@gobble
152   \let\zref@labelbylist\@gobbletwo
153   \let\zref@labelbyprops\@gobbletwo
154   \let\index\@gobble
155   \let\glossary\@gobble
156   \let\markboth\@gobbletwo

```

```

157 \let\@mkboth\@gobbletwo
158 \let\markright\@gobble
159 \let\phantomsection\@empty
160 \def\addcontentsline{\expandafter\@gobble\@gobbletwo}%
161 \let\raggedright\@empty
162 \let\raggedleft\@empty
163 \let\centering\@empty
164 \let\protect\@unexpandable@protect
165 \let\enit@format\@empty % package enumitem
166 }

```

\GTS@DisableHook

```

167 \providecommand*\GTS@DisableHook{}

```

\GetTitleStringDisableCommands

```

168 \def\GetTitleStringDisableCommands{%
169   \begingroup
170     \makeatletter
171     \GTS@DisableCommands
172 }

```

\GTS@DisableCommands

```

173 \long\def\GTS@DisableCommands#1{%
174   \toks0=\expandafter{\GTS@DisableHook}%
175   \toks2={#1}%
176   \xdef\GTS@GlobalString{\the\toks0 \the\toks2}%
177   \endgroup
178   \let\GTS@DisableHook\GTS@GlobalString
179 }

```

2.2.2 Non-expand method

```

180 \def\GTS@RemoveLeft{%
181   \toks@\expandafter\expandafter\expandafter{%
182     \expandafter\GTS@Car\GTS@GlobalString{}{}{}{}\GTS@Nil
183   }%
184   \edef\GTS@Token{\the\toks@}%
185   \GTS@PredefinedLeftCmds
186   \expandafter\futurelet\expandafter\GTS@Token
187   \expandafter\GTS@TestLeftSpace\GTS@GlobalString\GTS@Nil
188   \GTS@End
189 }
190 \def\GTS@End{}
191 \long\def\GTS@TestLeft#1#2{%
192   \def\GTS@temp{#1}%
193   \ifx\GTS@temp\GTS@Token
194     \toks@\expandafter\expandafter\expandafter{%
195       \expandafter#2\GTS@GlobalString\GTS@Nil
196     }%
197     \expandafter\GTS@TestLeftEnd
198   \fi
199 }
200 \long\def\GTS@TestLeftEnd#1\GTS@End{%
201   \xdef\GTS@GlobalString{\the\toks@}%
202   \GTS@RemoveLeft
203 }
204 \long\def\GTS@Car#1#2\GTS@Nil{#1}
205 \long\def\GTS@Cdr#1#2\GTS@Nil{#2}

```

```

206 \long\def\GTS@CdrTwo#1#2#3\GTS@Nil{#3}
207 \long\def\GTS@CdrThree#1#2#3#4\GTS@Nil{#4}
208 \long\def\GTS@CdrFour#1#2#3#4#5\GTS@Nil{#5}
209 \long\def\GTS@TestLeftSpace#1\GTS@Nil{%
210   \ifx\GTS@Token\@sptoken
211     \toks@\expandafter{%
212       \romannumeral-0\GTS@GlobalString
213     }%
214     \expandafter\GTS@TestLeftEnd
215   \fi
216 }

\GTS@PredefinedLeftCmds

217 \def\GTS@PredefinedLeftCmds{%
218   \GTS@TestLeft\Hy@phantomsection\GTS@Cdr
219   \GTS@TestLeft\Hy@SectionAnchor\GTS@Cdr
220   \GTS@TestLeft\Hy@SectionAnchorHref\GTS@CdrTwo
221   \GTS@TestLeft\label\GTS@CdrTwo
222   \GTS@TestLeft\zlabel\GTS@CdrTwo
223   \GTS@TestLeft\index\GTS@CdrTwo
224   \GTS@TestLeft\glossary\GTS@CdrTwo
225   \GTS@TestLeft\markboth\GTS@CdrThree
226   \GTS@TestLeft\@mkboth\GTS@CdrThree
227   \GTS@TestLeft\addcontentsline\GTS@CdrFour
228   \GTS@TestLeft\enit@format\GTS@Cdr % package enumitem
229 }

230 \def\GTS@RemoveRight{%
231   \toks@{}%
232   \expandafter\GTS@TestRightLabel\GTS@GlobalString
233     \label{}\GTS@Nil\@nil
234   \GTS@RemoveRightSpace
235 }
236 \begingroup
237   \def\GTS@temp#1{\endgroup
238     \def\GTS@RemoveRightSpace{%
239       \expandafter\GTS@TestRightSpace\GTS@GlobalString
240         \GTS@Nil#1\GTS@Nil\@nil
241     }%
242   }%
243 \GTS@temp{ }
244 \def\GTS@TestRightSpace#1 \GTS@Nil#2\@nil{%
245   \ifx\relax#2\relax
246   \else
247     \gdef\GTS@GlobalString{#1}%
248     \expandafter\GTS@RemoveRightSpace
249   \fi
250 }
251 \def\GTS@TestRightLabel#1\label#2#3\GTS@Nil#4\@nil{%
252   \def\GTS@temp{#3}%
253   \ifx\GTS@temp\@empty
254     \expandafter\gdef\expandafter\GTS@GlobalString\expandafter{%
255       \the\toks@
256       #1%
257     }%
258     \expandafter\@gobble
259   \else
260     \expandafter\@firstofone
261   \fi

```

```

262   {%
263     \toks@\expandafter{\the\toks@#1}%
264     \GTS@TestRightLabel#3\GTS@Nil\@nil
265   }%
266 }

267 \GTS@AtEnd%
268 \endpackage

```

3 Test

3.1 Catcode checks for loading

```

269 \test1

270 \catcode'\{=1 %
271 \catcode'\}=2 %
272 \catcode'\#=6 %
273 \catcode'\@=11 %
274 \expandafter\ifx\csname count\endcsname\relax
275   \countdef\count@=255 %
276 \fi
277 \expandafter\ifx\csname @gobble\endcsname\relax
278   \long\def\@gobble#1{%
279 \fi
280 \expandafter\ifx\csname @firstofone\endcsname\relax
281   \long\def\@firstofone#1{#1}%
282 \fi
283 \expandafter\ifx\csname loop\endcsname\relax
284   \expandafter\@firstofone
285 \else
286   \expandafter\@gobble
287 \fi
288 {%
289   \def\loop#1\repeat{%
290     \def\body{#1}%
291     \iterate
292   }%
293   \def\iterate{%
294     \body
295     \let\next\iterate
296   \else
297     \let\next\relax
298   \fi
299   \next
300 }%
301 \let\repeat=\fi
302 }%
303 \def\RestoreCatcodes{
304 \count@=0 %
305 \loop
306   \edef\RestoreCatcodes{%
307     \RestoreCatcodes
308     \catcode\the\count@=\the\catcode\count@\relax
309   }%
310 \ifnum\count@<255 %
311   \advance\count@ 1 %
312 \repeat
313

```



```

314 \def\RangeCatcodeInvalid#1#2{%
315   \count@=#1\relax
316   \loop
317     \catcode\count@=15 %
318   \ifnum\count@<#2\relax
319     \advance\count@ 1 %
320   \repeat
321 }
322 \def\RangeCatcodeCheck#1#2#3{%
323   \count@=#1\relax
324   \loop
325     \ifnum#3=\catcode\count@
326   \else
327     \errmessage{%
328       Character \the\count@\space
329       with wrong catcode \the\catcode\count@\space
330       instead of \number#3%
331     }%
332   \fi
333   \ifnum\count@<#2\relax
334     \advance\count@ 1 %
335   \repeat
336 }
337 \def\space{ }
338 \expandafter\ifx\csname LoadCommand\endcsname\relax
339   \def\LoadCommand{\input gettitlestring.sty\relax}%
340 \fi
341 \def\Test{%
342   \RangeCatcodeInvalid{0}{47}%
343   \RangeCatcodeInvalid{58}{64}%
344   \RangeCatcodeInvalid{91}{96}%
345   \RangeCatcodeInvalid{123}{255}%
346   \catcode'\@=12 %
347   \catcode'\=0 %
348   \catcode'\%=14 %
349   \LoadCommand
350   \RangeCatcodeCheck{0}{36}{15}%
351   \RangeCatcodeCheck{37}{37}{14}%
352   \RangeCatcodeCheck{38}{47}{15}%
353   \RangeCatcodeCheck{48}{57}{12}%
354   \RangeCatcodeCheck{58}{63}{15}%
355   \RangeCatcodeCheck{64}{64}{12}%
356   \RangeCatcodeCheck{65}{90}{11}%
357   \RangeCatcodeCheck{91}{91}{15}%
358   \RangeCatcodeCheck{92}{92}{0}%
359   \RangeCatcodeCheck{93}{96}{15}%
360   \RangeCatcodeCheck{97}{122}{11}%
361   \RangeCatcodeCheck{123}{255}{15}%
362   \RestoreCatcodes
363 }
364 \Test
365 \csname @@end\endcsname
366 \end
367 </test1>

```

3.2 Test of non-expand method

```

368 (*test2)
369 \NeedsTeXFormat{LaTeX2e}

```

```

370 \documentclass{minimal}
371 \usepackage{getttitlestring}[2010/12/03]
372 \usepackage{qstest}
373 \IncludeTests{*}
374 \LogTests{log}{*}{*}
375 \begin{document}
376 \begin{qstest}{non-expand}{non-expand}
377   \def\test#1#2{%
378     \sbox0{%
379       \GetTitleString{#1}%
380       \Expect{#2}*{\GetTitleStringResult}%
381     }%
382     \Expect{0.0pt}*{\the\wd0}%
383   }%
384   \test{}{}%
385   \test{ }{}%
386   \test{ x }{x}%
387   \test{ x y }{x y}%
388   \test{ \relax }{\relax}%
389   \test{\label{f}a}{a}%
390   \test{ \label{f}a }{a}%
391   \test{\label{f} a}{a}%
392   \test{ \label{f} a }{a}%
393   \test{a\label{f}}{a}%
394   \test{a\label{f} }{a}%
395   \test{a \label{f}}{a}%
396   \test{a \label{f} }{a}%
397   \test{a\label{f}b\label{g}}{ab}%
398   \test{a \label{f}b \label{g} }{a b}%
399   \test{a\label{f} b \label{g} }{a b}%
400 \end{qstest}
401 \end{document}
402 \end{test2}

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/getttitlestring.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/getttitlestring.pdf](#) Documentation.

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for \TeX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

¹[ftp://ftp.ctan.org/tex-archive/](http://ftp.ctan.org/tex-archive/)

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain \TeX :

```
tex gettitlestring.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
gettitlestring.sty      → tex/generic/oberdiek/gettitlestring.sty
gettitlestring.pdf      → doc/latex/oberdiek/gettitlestring.pdf
test/gettitlestring-test1.tex → doc/latex/oberdiek/test/gettitlestring-test1.tex
test/gettitlestring-test2.tex → doc/latex/oberdiek/test/gettitlestring-test2.tex
gettitlestring.dtx      → source/latex/oberdiek/gettitlestring.dtx
```

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

4.4 Refresh file name databases

If your \TeX distribution (`te \TeX` , `mik \TeX` , ...) relies on file name databases, you must refresh these. For example, `te \TeX` users run `texhash` or `mktexlsr`.

4.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk gettitlestring.pdf unpack_files output .
```

Unpacking with \LaTeX . The `.dtx` chooses its action depending on the format:

plain \TeX : Run `docstrip` and extract the files.

\LaTeX : Generate the documentation.

If you insist on using \LaTeX for `docstrip` (really, `docstrip` does not need \LaTeX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{gettitlestring.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfL^AT_EX:

```
pdflatex gettitlestring.dtx
makeindex -s gind.ist gettitlestring.idx
pdflatex gettitlestring.dtx
makeindex -s gind.ist gettitlestring.idx
pdflatex gettitlestring.dtx
```

5 Catalogue

The following XML file can be used as source for the [T_EX Catalogue](#). The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `gettitlestring.xml`.

```
403 (*catalogue)
404 <?xml version='1.0' encoding='us-ascii'?>
405 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
406 <entry datestamp='$Date$' modifier='$Author$' id='gettitlestring'>
407   <name>gettitlestring</name>
408   <caption>Clean up title references.</caption>
409   <authorref id='auth:oberdiek'>/>
410   <copyright owner='Heiko Oberdiek' year='2009,2010'>/>
411   <license type='lppl1.3'>/>
412   <version number='1.4'>/>
413   <description>
414     Cleans up the title string (removing <tt>\label</tt> commands) for
415     packages (such as <xref refid='nameref'>nameref</xref>) that
416     typeset such strings.
417   <p/>
418   The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
419 </description>
420 <documentation details='Package documentation'
421   href='ctan:/macros/latex/contrib/oberdiek/gettitlestring.pdf'>/>
422 <ctan file='true' path='/macros/latex/contrib/oberdiek/gettitlestring.dtx'>/>
423 <miktex location='oberdiek'>/>
424 <texlive location='oberdiek'>/>
425 <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'>/>
426 </entry>
427 </catalogue>
```

6 References

- [1] Peter Wilson, Lars Madsen: *The Memoir Class*; 2009/11/17 v1.61803398c; [CTAN:macros/latex/contrib/memoir/](#)
- [2] Donald Arsenaun: *Titleref.sty*; 2001/04/05 ver 3.1; [CTAN:macros/latex/contrib/misc/titleref.sty](#)
- [3] Heiko Oberdiek: *The zref package*; 2009/12/08 v2.7; [CTAN:macros/latex/contrib/oberdiek/zref.pdf](#)

7 History

[2009/12/08 v1.0]

- The first version.

[2009/12/12 v1.1]

- Short info shortened.

[2009/12/13 v1.2]

- Forgotten third argument for `\InputIfFileExists` added.

[2009/12/18 v1.3]

- `\Hy@SectionAnchorHref` added for filtering (hyperref 2009/12/18 v6.79w).

[2010/12/03 v1.4]

- Support of package `enumitem`: removing `\enit@format` from title string (problem report by GL).

8 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

Symbols		C	
<code>\#</code>	272	<code>\catcode</code>	2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 69, 70, 72, 73, 74, 78, 79, 80, 81, 82, 83, 84, 87, 88, 90, 91, 92, 93, 97, 99, 270, 271, 272, 273, 308, 317, 325, 329, 346, 347, 348
<code>\%</code>	348	<code>\centering</code>	163
<code>\@</code>	273, 346	<code>\count@</code>	275, 304, 308, 310, 311, 315, 317, 318, 319, 323, 325, 328, 329, 333, 334
<code>\@empty</code> ...	159, 161, 162, 163, 165, 253	<code>\countdef</code>	275
<code>\@firstofone</code>	260, 281, 284	<code>\csname</code>	14, 21, 50, 66, 76, 274, 277, 280, 283, 338, 365
<code>\@gobble</code>	149, 150, 151, 154, 155, 158, 160, 258, 278, 286	D	
<code>\@gobbletwo</code> ...	152, 153, 156, 157, 160	<code>\DeclareBoolOption</code>	117
<code>\@mkboth</code>	157, 226	<code>\documentclass</code>	370
<code>\@nil</code>	233, 240, 244, 251, 264	E	
<code>\@sptoken</code>	210	<code>\empty</code>	17, 18
<code>\@undefined</code>	58	<code>\end</code>	366, 400, 401
<code>\@unexpandable@protect</code>	164	<code>\endcsname</code>	14, 21, 50, 66, 76, 274, 277, 280, 283, 338, 365
<code>\@</code>	347	<code>\endinput</code>	29, 108
<code>\{</code>	270	<code>\endlinechar</code>	4, 35, 71, 77, 89
<code>\}</code>	271		
A			
<code>\addcontentsline</code>	160, 227		
<code>\advance</code>	311, 319, 334		
<code>\aftergroup</code>	29		
B			
<code>\begin</code>	375, 376		
<code>\body</code>	290, 294		

\enit@format	165, 228	\input	339
\errmessage	327	\InputIfFileExists	118
\Expect	380, 382	\iterate	291, 293, 295
F		L	
\futurelet	186	\label	149, 221, 233, 251, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 414
G		\LoadCommand	339, 349
\gdef	247, 254	\LogTests	374
\GetTitleString	2, 120, 139, 379	\loop	289, 305, 316, 324
\GetTitleStringDisableCommands	2, 168	M	
\GetTitleStringExpand	122, 127	\makeatletter	170
\GetTitleStringNonExpand	124, 139	\markboth	156, 225
\GetTitleStringResult	128, 133, 134, 140, 141, 146, 380	\markright	158
\GetTitleStringSetup	2, 114	N	
\glossary	155, 224	\NeedsTeXFormat	369
\GTS@AtEnd	95, 96, 108, 267	\newcommand	114, 120, 127, 139
\GTS@Car	182, 204	\next	295, 297, 299
\GTS@Cdr	205, 218, 219, 228	\number	330
\GTS@CdrFour	208, 227	P	
\GTS@CdrThree	207, 225, 226	\PackageInfo	26
\GTS@CdrTwo	206, 220, 221, 222, 223, 224	\phantomsection	159
\GTS@DisableCommands	171, 173	\ProcessKeyvalOptions	119
\GTS@DisableHook	131, 167, 174, 178	\protect	164
\GTS@DisablePredefinedCmds	130, 148	\providecommand	167
\GTS@End	188, 190, 200	\ProvidesPackage	19, 67
\GTS@GlobalString	141, 146, 176, 178, 182, 187, 195, 201, 212, 232, 239, 247, 254	R	
\GTS@Nil	182, 187, 195, 204, 205, 206, 207, 208, 209, 233, 240, 244, 251, 264	\raggedleft	162
\GTS@PredefinedLeftCmds	185, 217	\raggedright	161
\GTS@RemoveLeft	143, 180, 202	\RangeCatcodeCheck	322, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361
\GTS@RemoveRight	144, 230	\RangeCatcodeInvalid	314, 342, 343, 344, 345
\GTS@RemoveRightSpace	234, 238, 248	\repeat	289, 301, 312, 320, 335
\GTS@temp	192, 193, 237, 243, 252, 253	\RequirePackage	109
\GTS@TestLeft	191, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228	\RestoreCatcodes	303, 306, 307, 362
\GTS@TestLeftEnd	197, 200, 214	\romannumeral	212
\GTS@TestLeftSpace	187, 209	S	
\GTS@TestRightLabel	232, 251, 264	\sbox	378
\GTS@TestRightSpace	239, 244	\setkeys	115
\GTS@Token	184, 186, 193, 210	\SetupKeyvalOptions	110
H		\space	328, 329, 337
\Hy@phantomsection	218	T	
\Hy@SectionAnchor	219	\Test	341, 364
\Hy@SectionAnchorHref	220	\test	377, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399
I		\the	77, 78, 79, 80, 81, 82, 83, 84, 97, 176, 184, 201, 255, 263, 308, 328, 329, 382
\ifGTS@expand	121	\TMP@EnsureCode	94, 101, 102, 103, 104, 105, 106, 107
\ifnum	310, 318, 325, 333	\toks	174, 175, 176
\ifx	15, 18, 21, 50, 58, 61, 193, 210, 245, 253, 274, 277, 280, 283, 338		
\immediate	23, 52		
\IncludeTests	373		
\index	154, 223		

