

The infwarerr package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/04/26 v1.0

Abstract

This package provides a complete set of macros for informations, warnings and error messages with support for plain-TeX.

Contents

1 Documentation	1
2 Implementation	2
2.1 Reload check and package identification	2
2.2 Catcodes	3
2.3 L ^A T _E X detection	3
2.4 Version for L ^A T _E X	3
2.5 Version for plain-TeX	4
3 Test	7
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 History	12
[2007/04/26 v1.0]	12
6 Index	12

1 Documentation

Package infwarerr defines the following set of commands, given as pseudo regular expression:

```
\@(Package|Class)(Info|InfoNoLine|Warning|WarningNoLine|Error)
```

The first argument is the package or class in question. The second argument contains the message text. `\MessageBreak` is supported. In case of error messages, the third argument contains the help text. Supported are L^AT_EX's message texts `\@ehc`:

```
Try typing <return> to proceed
If that doesn't work, type X <return> to quit.
```

and `\@ehd`:

You're in trouble here. Try typing <return> to proceed
 If that doesn't work, type X <return> to quit.

The package is intended for macro writers that want to write code that works for both L^AT_EX and plain-T_EX. If L^AT_EX is detected, then the macros of this package will only be wrappers for L^AT_EX's macros. Otherwise L^AT_EX's macros will be mimicked for use with plain-T_EX.

2 Implementation

1 (*package)

2.1 Reload check and package identification

Reload check, especially if the package is not used with L^AT_EX.

```

2 \begingroup
3 \catcode44 12 % ,
4 \catcode45 12 % -
5 \catcode46 12 % .
6 \catcode58 12 % :
7 \catcode64 11 % @
8 \expandafter\let\expandafter\x\csname ver@infwarerr.sty\endcsname
9 \ifcase 0%
10 \ifx\x\relax % plain
11 \else
12 \ifx\x\empty % LaTeX
13 \else
14 1%
15 \fi
16 \fi
17 \else
18 \expandafter\ifx\csname PackageInfo\endcsname\relax
19 \def\x#1#2{%
20 \immediate\write-1{Package #1 Info: #2.}%
21 }%
22 \else
23 \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
24 \fi
25 \x{infwarerr}{The package is already loaded}%
26 \endgroup
27 \expandafter\endinput
28 \fi
29 \endgroup

```

Package identification:

```

30 \begingroup
31 \catcode44 12 % ,
32 \catcode45 12 % -
33 \catcode46 12 % .
34 \catcode58 12 % :
35 \catcode64 11 % @
36 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
37 \def\x#1#2#3[#4]{\endgroup
38 \immediate\write-1{Package: #3 #4}%
39 \xdef#1{#4}%
40 }%
41 \else
42 \def\x#1#2[#3]{\endgroup
43 #2[#{#3}]%
44 \ifx#1\relax
45 \xdef#1{#3}%
46 \fi

```

```

47   }%
48   \fi
49 \expandafter\x\csname ver@infwarerr.sty\endcsname
50 \ProvidesPackage{infwarerr}%
51 [2007/04/26 v1.0 Providing info/warning/message (HO)]

```

2.2 Catcodes

```

52 \expandafter\edef\csname InfWarErr@End\endcsname{%
53   \catcode39 \the\catcode39\relax % '
54   \catcode40 \the\catcode40\relax % (
55   \catcode41 \the\catcode41\relax % )
56   \catcode44 \the\catcode44\relax % ,
57   \catcode45 \the\catcode45\relax % -
58   \catcode46 \the\catcode46\relax % .
59   \catcode58 \the\catcode58\relax % :
60   \catcode60 \the\catcode60\relax % <
61   \catcode61 \the\catcode61\relax % =
62   \catcode62 \the\catcode62\relax % >
63   \catcode64 \the\catcode64\relax % @
64   \catcode94 \the\catcode94\relax % ^
65   \noexpand\endinput
66 }
67 \catcode39 12 % '
68 \catcode40 12 % (
69 \catcode41 12 % )
70 \catcode44 12 % ,
71 \catcode45 12 % -
72 \catcode46 12 % .
73 \catcode58 12 % :
74 \catcode60 12 % <
75 \catcode61 12 % =
76 \catcode62 12 % >
77 \catcode64 11 % @
78 \catcode94 7 % ^

```

2.3 L^AT_EX detection

```

79 \begingroup
80   \let\x=Y%
81   \def\check#1{%
82     \@check{Generic}{#1}%
83     \@check{Package}{#1}%
84     \@check{Class}{#1}%
85   }%
86   \def\@check#1#2{%
87     \expandafter\ifx\csname #1#2\endcsname\relax
88       \let\x=N%
89     \fi
90   }%
91   \check{Info}%
92   \check{Warning}%
93   \check{Error}%
94   \@check{Package}{WarningNoLine}%
95   \@check{Class}{WarningNoLine}%
96   \@check{}{@ehc}%
97   \@check{}{@ehd}%
98 \expandafter\endgroup
99 \ifx\x Y%

```

2.4 Version for L^AT_EX

```

\@PackageInfo
100 \let\@PackageInfo\PackageInfo

```

```

\@PackageInfoNoLine
101 \def\@PackageInfoNoLine#1#2{%
102   \PackageInfo{#1}{#2\@gobble}%
103 }%

\@ClassInfo
104 \let\@ClassInfo\ClassInfo

\@ClassInfoNoLine
105 \def\@ClassInfoNoLine#1#2{%
106   \ClassInfo{#1}{#2\@gobble}%
107 }%

\@PackageWarning
108 \let\@PackageWarning\PackageWarning

\@PackageWarningNoLine
109 \let\@PackageWarningNoLine\PackageWarningNoLine

\@ClassWarning
110 \let\@ClassWarning\ClassWarning

\@ClassWarningNoLine
111 \let\@ClassWarningNoLine\ClassWarningNoLine

\@PackageError
112 \let\@PackageError\PackageError

\@ClassError
113 \let\@ClassError\ClassError

114 \csname fi\endcsname
115 \InfWarErr@End
116 \fi

```

2.5 Version for plain-TeX

```

\InfWarErr@online
117 \begingroup\expandafter\endgroup
118 \ifcase\expandafter\ifx\csname inputlineno\endcsname\relax
119   0 %
120 \else
121   \ifnum\inputlineno<0 %
122     0 %
123 \else
124   1 %
125 \fi
126 \fi
127 \def\InfWarErr@online{}%
128 \else
129 \def\InfWarErr@online{ on input line \the\inputlineno}%
130 \fi

\InfWarErr@protected
131 \begingroup\expandafter\expandafter\expandafter\endgroup
132 \expandafter\ifx\csname protected\endcsname\relax
133 \def\InfWarErr@protected{}%
134 \else
135 \let\InfWarErr@protected\protected
136 \fi

```

```

\InfWarErr@unused
137 \chardef\InfWarErr@unused=16 %

\InfWarErr@ehc
138 \def\InfWarErr@ehc{%
139   Try typing \space <return> \space to proceed.\MessageBreak
140   If that doesn't work, type \space X <return> \space to quit.%
141 }

\InfWarErr@ehd
142 \def\InfWarErr@ehd{%
143   You're in trouble here. \space\@ehc
144 }

\InfWarErr@GenericInfo
145 \InfWarErr@protected\def\InfWarErr@GenericInfo#1#2{%
146   \begingroup
147     \let\on@line\InfWarErr@online
148     \long\def\@gobble##1{%
149       \def\@spaces{\space\space\space\space}%
150       \def\MessageBreak{^^J#1}%
151       \newlinechar=10 %
152       \immediate\write-1{#2\on@line.}%
153     \endgroup
154 }

\InfWarErr@GenericWarning
155 \InfWarErr@protected\def\InfWarErr@GenericWarning#1#2{%
156   \begingroup
157     \let\on@line\InfWarErr@online
158     \long\def\@gobble##1{%
159       \def\@spaces{\space\space\space\space}%
160       \def\MessageBreak{^^J#1}%
161       \newlinechar=10 %
162       \immediate\write\InfWarErr@unused{^^J#2\on@line.^^J}%
163     \endgroup
164 }

\InfWarErr@GenericError Old TeX versions below 3.141 where ^^J is not working as newline in error message
and help texts are not supported.
165 \InfWarErr@protected\def\InfWarErr@GenericError#1#2#3#4{%
166   \begingroup
167     \immediate\write\InfWarErr@unused{%
168       \newlinechar=10 %
169       \def\@spaces{\space\space\space\space}%
170       \let\@ehc\InfWarErr@ehc
171       \let\@ehd\InfWarErr@ehd
172       \let\on@line\InfWarErr@online
173       \def\MessageBreak{^^J}%
174       \edef\InfWarErr@help{\noexpand\errhelp{#4}}%
175       \InfWarErr@help
176       \def\MessageBreak{^^J#1}%
177       \errmessage{%
178         #2.^^J^^J%
179         #3^^J%
180         Type \space H <return> \space for immediate help%
181       }%
182     \endgroup
183 }

```

\@PackageInfo

```
184 \def\@PackageInfo#1#2{%
185   \InfWarErr@GenericInfo{%
186     (#1) \@spaces\@spaces\@spaces
187   }{%
188     Package #1 Info: #2%
189   }%
190 }
```

\@PackageInfoNoLine

```
191 \def\@PackageInfoNoLine#1#2{%
192   \@PackageInfo{#1}{#2\@gobble}%
193 }
```

\@ClassInfo

```
194 \def\@ClassInfo#1#2{%
195   \InfWarErr@GenericInfo{%
196     (#1) \space\space\@spaces\@spaces
197   }{%
198     Class #1 Info: #2%
199   }%
200 }
```

\@ClassInfoNoLine

```
201 \def\@ClassInfoNoLine#1#2{%
202   \@ClassInfo{#1}{#2\@gobble}%
203 }
```

\@PackageWarning

```
204 \def\@PackageWarning#1#2{%
205   \InfWarErr@GenericWarning{%
206     (#1)\@spaces\@spaces\@spaces\@spaces
207   }{%
208     Package #1 Warning: #2%
209   }%
210 }
```

\@PackageWarningNoLine

```
211 \def\@PackageWarningNoLine#1#2{%
212   \@PackageWarning{#1}{#2\@gobble}%
213 }
```

\@ClassWarning

```
214 \def\@ClassWarning#1#2{%
215   \InfWarErr@GenericWarning{%
216     (#1) \space\@spaces\@spaces\@spaces
217   }{%
218     Class #1 Warning: #2%
219   }%
220 }
```

\@ClassWarningNoLine

```
221 \def\@ClassWarningNoLine#1#2{%
222   \@ClassWarning{#1}{#2\@gobble}%
223 }
```

\@PackageError

```
224 \def\@PackageError#1#2#3{%
225   \InfWarErr@GenericError{%
226     (#1)\@spaces\@spaces\@spaces\@spaces
```

```

227 }{%
228   Package #1 Error: #2%
229 }{%
230   See the #1 package documentation for explanation.%
231 }{#3}%
232 }

```

\@ClassError

```

233 \def\@ClassError#1#2#3{%
234   \InfWarErr@GenericError{%
235     (#1) \space\@spaces\@spaces\@spaces
236   }{%
237     Class #1 Error: #2%
238   }{%
239     See the #1 class documentation for explanation.%
240   }{#3}%
241 }

242 \InfWarErr@End
243 \</package>

```

3 Test

```

244 <*test1>
245 \let\PackageInfo\relax
246 \input infwarerr.sty\relax
247 \let\OrgWrite\write
248 \chardef\WriteUnused=16 %
249 \def\msg#{\immediate\OrgWrite\WriteUnused}
250 \msg{File: infwarerr-test1.tex 2007/04/26 v1.0 Test file for plain-TeX}
251 \catcode64=11 %
252 \</test1>

253 <*test2>
254 \NeedsTeXFormat{LaTeX2e}
255 \ProvidesFile{infwarerr-test2.tex}[2007/04/26 v1.0 Test file for LaTeX]
256 \RequirePackage{infwarerr}
257 \makeatletter
258 \let\OrgWrite\write
259 \chardef\WriteUnused=\@unused
260 \def\msg#{\immediate\OrgWrite\WriteUnused}
261 \</test2>

262 <*test1 | test2>
263 \def\pkgname{PACKAGE}
264 \def\clsname{CLASS}
265 \def\msgtext{MESSAGE\MessageBreak SECOND LINE}
266 \def\hlptext{HELP\MessageBreak SECOND LINE}
267
268 \let\OrgWrite\write
269 \let\OrgErrorMessage\errmessage
270
271 \def\write#1#{%
272   \@write{#1}%
273 }
274 \def\@write#1#2{%
275   \OrgWrite#1{#2}%
276   \xdef\WriteNum{\number#1}%
277   \xdef\WriteText{#2}%
278 }
279 \def\TestFailed#1#2{%
280   \OrgErrorMessage{Test (\string#1) failed: #2}%
281 }

```

```

282 \def\TestWrite#1#2#3{%
283   \ifnum\WriteNum=#2\relax
284   \else
285     \TestFailed#1{write stream number: \WriteNum <> #2}%
286   \fi
287 \begingroup
288   \let~\space
289   \edef\TestString{#3}%
290   \ifx\WriteText\TestString
291   \else
292     \msg{}%
293     \msg{[\WriteText]}%
294     \msg{<>}%
295     \msg{[#3]}%
296     \TestFailed#1{write text}%
297   \fi
298 \endgroup
299 }
300 \newtoks\ErrHelpToks
301 \def\errmessage#1{%
302   \global\ErrHelpToks=\expandafter\the\errhelp}%
303   \msg{***** error message *****}%
304   \msg{! #1}%
305   \msg{***** help message *****}%
306   \msg{\the\errhelp}%
307   \msg{*****}%
308   \xdef\ErrorMessage{! #1}%
309 }
310 \def\TestInfo{%
311   \@PackageInfo\pkgname\msgtext
312   \TestWrite\@PackageInfo{-1}{%
313     Package PACKAGE Info: MESSAGE^^J%
314     (PACKAGE)~~~~~SECOND LINE on input line \the\inputlineno.%
315   }%
316   \@PackageInfoNoLine\pkgname\msgtext
317   \TestWrite\@PackageInfoNoLine{-1}{%
318     Package PACKAGE Info: MESSAGE^^J%
319     (PACKAGE)~~~~~SECOND LINE.%
320   }%
321   \@ClassInfo\clsname\msgtext
322   \TestWrite\@ClassInfo{-1}{%
323     Class CLASS Info: MESSAGE^^J%
324     (CLASS)~~~~~SECOND LINE on input line \the\inputlineno.%
325   }%
326   \@ClassInfoNoLine\clsname\msgtext
327   \TestWrite\@ClassInfoNoLine{-1}{%
328     Class CLASS Info: MESSAGE^^J%
329     (CLASS)~~~~~SECOND LINE.%
330   }%
331 }
332 \def\TestWarning{%
333   \@PackageWarning\pkgname\msgtext
334   \TestWrite\@PackageWarning\WriteUnused{%
335     ^^J%
336     Package PACKAGE Warning: MESSAGE^^J%
337     (PACKAGE)~~~~~SECOND LINE on input line \the\inputlineno.%
338     ^^J%
339   }%
340   \@PackageWarningNoLine\pkgname\msgtext
341   \TestWrite\@PackageWarningNoLine\WriteUnused{%
342     ^^J%
343     Package PACKAGE Warning: MESSAGE^^J%

```

```

344 (PACKAGE)~~~~~SECOND LINE.%
345 ^^J%
346 }%
347 \@ClassWarning\clsname\msgtext
348 \TestWrite\@ClassWarning\WriteUnused{%
349 ^^J%
350 Class CLASS Warning: MESSAGE^^J%
351 (CLASS)~~~~~SECOND LINE on input line \the\inputlineno.%
352 ^^J%
353 }%
354 \@ClassWarningNoLine\clsname\msgtext
355 \TestWrite\@ClassWarningNoLine\WriteUnused{%
356 ^^J%
357 Class CLASS Warning: MESSAGE^^J%
358 (CLASS)~~~~~SECOND LINE.%
359 ^^J%
360 }%
361 }
362 \def\TestError{%
363 (test2) \let\write\OrgWrite
364 \@PackageError\pkgname\msgtext\hlptext
365 \@TestError\@PackageError{%
366 ! Package PACKAGE Error: MESSAGE^^J%
367 (PACKAGE)~~~~~SECOND LINE.^^J%
368 ^^J%
369 See the PACKAGE package documentation for explanation.^^J%
370 Type~~H <return>~~for immediate help%
371 }{%
372 HELP^^J%
373 SECOND LINE%
374 }%
375 \@ClassError\clsname\msgtext\hlptext
376 \@TestError\@ClassError{%
377 ! Class CLASS Error: MESSAGE^^J%
378 (CLASS)~~~~~SECOND LINE.^^J%
379 ^^J%
380 See the CLASS class documentation for explanation.^^J%
381 Type~~H <return>~~for immediate help%
382 }{%
383 HELP^^J%
384 SECOND LINE%
385 }%
386 \@PackageError\pkgname\msgtext\@ehc
387 \@TestError\@PackageError{%
388 ! Package PACKAGE Error: MESSAGE^^J%
389 (PACKAGE)~~~~~SECOND LINE.^^J%
390 ^^J%
391 See the PACKAGE package documentation for explanation.^^J%
392 Type~~H <return>~~for immediate help%
393 }{%
394 Try typing~~<return>~~to proceed.^^J%
395 If that doesn't work, type~~X <return>~~to quit.%
396 }%
397 \@PackageError\pkgname\msgtext\@ehd
398 \@TestError\@PackageError{%
399 ! Package PACKAGE Error: MESSAGE^^J%
400 (PACKAGE)~~~~~SECOND LINE.^^J%
401 ^^J%
402 See the PACKAGE package documentation for explanation.^^J%
403 Type~~H <return>~~for immediate help%
404 }{%
405 You're in trouble here.~~Try typing~~<return>~~to proceed.^^J%

```

```

406     If that doesn't work, type~~X <return>~~to quit.%
407 }%
408 }
409 \def\@TestError#1#2#3{%
410   \begingroup
411     \let~\space
412     \edef\x{#2}%
413     \test2) \@onelevel@sanitize\x
414     \test2) \@onelevel@sanitize\ErrorMessage
415     \ifx\x\ErrorMessage
416       \else
417         \msg{}%
418         \msg{[\ErrorMessage]}%
419         \msg{<>}%
420         \msg{[\x]}%
421         \TestFailed#1{errmessage text}%
422       \fi
423       \edef\x{\the\ErrHelpToks}%
424       \edef\y{#3}%
425       \ifx\x\y
426         \else
427           \msg{}%
428           \msg{[\the\ErrHelpToks]}%
429           \msg{<>}%
430           \msg{[\y]}%
431           \TestFailed#1{errhelp text}%
432         \fi
433       \endgroup
434 }
435 \TestInfo
436 \TestWarning
437 \TestError
438 </test1 | test2>
439 \test1\end
440 \test2\@@end

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

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

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

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<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 infwarerr.dtx
```

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

```
infwarerr.sty      → tex/generic/oberdiek/infwarerr.sty
infwarerr.pdf      → doc/latex/oberdiek/infwarerr.pdf
infwarerr-test1.tex → doc/latex/oberdiek/infwarerr-test1.tex
infwarerr-test2.tex → doc/latex/oberdiek/infwarerr-test2.tex
infwarerr.dtx      → source/latex/oberdiek/infwarerr.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 , mi \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 infwarerr.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{infwarerr.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 pdf \LaTeX :

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

5 History

[2007/04/26 v1.0]

- First version.

6 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; numbers in *roman* refer to the code lines where the entry is used.

Symbols	
\@@end	440
\@ClassError	<u>113</u> , <u>233</u> , 375, 376
\@ClassInfo	<u>104</u> , <u>194</u> , 202, 321, 322
\@ClassInfoNoLine	<u>105</u> , <u>201</u> , 326, 327
\@ClassWarning	<u>110</u> , <u>214</u> , 222, 347, 348
\@ClassWarningNoLine	<u>111</u> , <u>221</u> , 354, 355
\@PackageError	<u>112</u> , 224, 364, 365, 386, 387, 397, 398
\@PackageInfo	<u>100</u> , <u>184</u> , 192, 311, 312
\@PackageInfoNoLine	<u>101</u> , <u>191</u> , 316, 317
\@PackageWarning	<u>108</u> , <u>204</u> , 212, 333, 334
\@PackageWarningNoLine <u>109</u> , <u>211</u> , 340, 341
\@TestError	365, 376, 387, 398, 409
\@check	82, 83, 84, 86, 94, 95, 96, 97
\@ehc	143, 170, 386
\@ehd	171, 397
\@gobble	102, 106, 148, 158, 192, 202, 212, 222
\@onelevel@sanitize	413, 414
\@spaces	149, 159, 169, 186, 196, 206, 216, 226, 235
\@unused	259
\@write	272, 274
C	
\catcode	3, 4, 5, 6, 7, 31, 32, 33, 34, 35, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 251
\chardef	137, 248, 259
\check	81, 91, 92, 93
\ClassError	113
\ClassInfo	104, 106
\ClassWarning	110
\ClassWarningNoLine	111
\clsname	264, 321, 326, 347, 354, 375
\csname	8, 18, 36, 49, 52, 87, 114, 118, 132
E	
\empty	12
\end	439
\endcsname	8, 18, 36, 49, 52, 87, 114, 118, 132
\endinput	27, 65
\errhelp	174, 302, 306
\ErrHelpToks	300, 302, 423, 428
\ErrMsgage	308, 414, 415, 418
\errmessage	177, 269, 301
H	
\hlptext	266, 364, 375
I	
\ifcase	9, 118
\ifnum	121, 283
\ifx	10, 12, 18, 36, 44, 87, 99, 118, 132, 290, 415, 425
\immediate	20, 38, 152, 162, 167, 249, 260
\InfWarErr@ehc	<u>138</u> , 170
\InfWarErr@ehd	<u>142</u> , 171
\InfWarErr@End	115, 242
\InfWarErr@GenericError	<u>165</u> , 225, 234
\InfWarErr@GenericInfo	<u>145</u> , 185, 195
\InfWarErr@GenericWarning <u>155</u> , 205, 215
\InfWarErr@help	174, 175
\InfWarErr@online	<u>117</u> , 147, 157, 172
\InfWarErr@protected	<u>131</u> , 145, 155, 165
\InfWarErr@unused	137, 162, 167
\input	246
\inputlineno	121, 129, 314, 324, 337, 351
M	
\makeatletter	257
\MessageBreak 139, 150, 160, 173, 176, 265, 266
\msg	249, 250, 260, 292, 293, 294, 295, 303, 304, 305, 306, 307, 417, 418, 419, 420, 427, 428, 429, 430
\msgtext	265, 311, 316, 321, 326, 333, 340, 347, 354, 364, 375, 386, 397
N	
\NeedsTeXFormat	254

<code>\newlinechar</code>	151, 161, 168		
<code>\newtoks</code>	300		
<code>\number</code>	276		
O			
<code>\on@line</code>	147, 152, 157, 162, 172		
<code>\OrgErrMessage</code>	269, 280		
<code>\OrgWrite</code>			
	247, 249, 258, 260, 268, 275, 363		
P			
<code>\PackageError</code>	112		
<code>\PackageInfo</code>	23, 100, 102, 245		
<code>\PackageWarning</code>	108		
<code>\PackageWarningNoLine</code>	109		
<code>\pkgname</code>	263,		
	311, 316, 333, 340, 364, 386, 397		
<code>\protected</code>	135		
<code>\ProvidesFile</code>	255		
<code>\ProvidesPackage</code>	50		
R			
<code>\RequirePackage</code>	256		
S			
<code>\space</code>	139, 140, 143, 149, 159,		
	169, 180, 196, 216, 235, 288, 411		
T			
<code>\TestError</code>	362, 437		
<code>\TestFailed</code>	279, 285, 296, 421, 431		
<code>\TestInfo</code>	310, 435		
<code>\TestString</code>	289, 290		
<code>\TestWarning</code>	332, 436		
<code>\TestWrite</code>	282, 312,		
	317, 322, 327, 334, 341, 348, 355		
<code>\the</code>	53, 54, 55, 56, 57, 58,		
	59, 60, 61, 62, 63, 64, 129, 302,		
	306, 314, 324, 337, 351, 423, 428		
W			
<code>\write</code>	20, 38, 152,		
	162, 167, 247, 258, 268, 271, 363		
<code>\WriteNum</code>	276, 283, 285		
<code>\WriteText</code>	277, 290, 293		
<code>\WriteUnused</code>	248,		
	249, 259, 260, 334, 341, 348, 355		
X			
<code>\x</code>	8, 10, 12, 19, 23, 25, 37, 42, 49, 80,		
	88, 99, 412, 413, 415, 420, 423, 425		
Y			
<code>\y</code>	424, 425, 430		