

# The etexcmds package

Heiko Oberdiek\*

<heiko.oberdiek at gmail.com>

2016/05/16 v1.6

## Abstract

New primitive commands are introduced in  $\varepsilon$ -TeX. Sometimes the names collide with existing macros. This package solves this name clashes by adding a prefix to  $\varepsilon$ -TeX's commands. For example,  $\varepsilon$ -TeX's `\unexpanded` is provided as `\etex@unexpanded`.

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	<code>\unexpanded</code> . . . . .	2
1.2	<code>\expanded</code> . . . . .	2
<b>2</b>	<b>Implementation</b>	<b>2</b>
2.1	Reload check and package identification . . . . .	2
2.2	Catcodes . . . . .	3
2.3	Provide <code>\newif</code> . . . . .	4
2.4	Load package <code>infwarerr</code> . . . . .	5
2.5	<code>\unexpanded</code> . . . . .	5
2.6	<code>\expanded</code> . . . . .	6
<b>3</b>	<b>Test</b>	<b>7</b>
3.1	Catcode checks for loading . . . . .	7
3.2	Macro tests . . . . .	9
<b>4</b>	<b>Installation</b>	<b>10</b>
4.1	Download . . . . .	10
4.2	Bundle installation . . . . .	10
4.3	Package installation . . . . .	10
4.4	Refresh file name databases . . . . .	11
4.5	Some details for the interested . . . . .	11
<b>5</b>	<b>Catalogue</b>	<b>11</b>
<b>6</b>	<b>History</b>	<b>12</b>
	[2007/05/06 v1.0] . . . . .	12
	[2007/09/09 v1.1] . . . . .	12
	[2007/12/12 v1.2] . . . . .	12
	[2010/01/28 v1.3] . . . . .	12
	[2011/01/30 v1.4] . . . . .	12

---

\*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

[2011/02/16 v1.5]	12
[2016/05/16 v1.6]	12
<b>7 Index</b>	<b>13</b>

# 1 Documentation

## 1.1 `\unexpanded`

`\etex@unexpanded`

New primitive commands are introduced in  $\varepsilon$ -TeX. Unhappily `\unexpanded` collides with a macro in ConTeXt with the same name. This also affects the L<sup>A</sup>TeX world. For example, package `m-ch-de` loads `base/syst-gen.tex` that redefines `\unexpanded`. Thus this package defines `\etex@unexpanded` to get rid of the name clash.

`\ifetex@unexpanded`

Package `etexcmds` can be loaded even if  $\varepsilon$ -TeX is not present or `\unexpanded` cannot be found. The switch `\ifetex@unexpanded` tells whether it is safe to use `\etex@unexpanded`. The switch is true (`\iftrue`) only if the primitive `\unexpanded` has been found and `\etex@unexpanded` is available.

## 1.2 `\expanded`

Probably `\expanded` will be added in pdfTeX 1.50 and LuaTeX. Again ConTeXt defines this as macro. Therefore version 1.2 of this packages also provides `\etex@expanded` and `\ifetex@unexpanded`.

# 2 Implementation

```
1 (*package)
```

## 2.1 Reload check and package identification

Reload check, especially if the package is not used with L<sup>A</sup>TeX.

```
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@etexcmds.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{etexcmds}{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@etexcmds.sty\endcsname
67 \ProvidesPackage{etexcmds}%
68 [2016/05/16 v1.6 Avoid name clashes with e-TeX commands (HO)]%

```

## 2.2 Catcodes

```

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 etexcmds@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\etexcmds@AtEnd{%
96 \etexcmds@AtEnd
97 \catcode#1=\the\catcode#1\relax
98 }%
99 \catcode#1=#2\relax
100 }
101 \TMP@EnsureCode{39}{12}% '
102 \TMP@EnsureCode{40}{12}% (
103 \TMP@EnsureCode{41}{12}% )
104 \TMP@EnsureCode{44}{12}% ,
105 \TMP@EnsureCode{45}{12}% -
106 \TMP@EnsureCode{46}{12}% .
107 \TMP@EnsureCode{47}{12}% /
108 \TMP@EnsureCode{60}{12}% <
109 \TMP@EnsureCode{91}{12}% [
110 \TMP@EnsureCode{93}{12}% ]
111 \edef\etexcmds@AtEnd{%
112 \etexcmds@AtEnd
113 \escapechar=\the\escapechar\relax
114 \noexpand\endinput
115 }
116 \escapechar=92 % backslash

```

## 2.3 Provide \newif

\etexcmds@newif

```

117 \def\etexcmds@newif#1{%
118 \expandafter\edef\csname etex@#1false\endcsname{%
119 \let
120 \expandafter\noexpand\csname ifetex@#1\endcsname
121 \noexpand\iffalse
122 }%
123 \expandafter\edef\csname etex@#1true\endcsname{%
124 \let
125 \expandafter\noexpand\csname ifetex@#1\endcsname
126 \noexpand\iftrue
127 }%

```

```

128 \csname etex@iffalse\endcsname
129 }

```

## 2.4 Load package infwarerr

```

130 \begingroup\expandafter\expandafter\expandafter\endgroup
131 \expandafter\ifx\csname RequirePackage\endcsname\relax
132 \def\TMP@RequirePackage#1[#2]{%
133 \begingroup\expandafter\expandafter\expandafter\endgroup
134 \expandafter\ifx\csname ver@#1.sty\endcsname\relax
135 \input #1.sty\relax
136 \fi
137 }%
138 \TMP@RequirePackage{infwarerr}[2007/09/09]%
139 \TMP@RequirePackage{ifluatex}[2010/03/01]%
140 \else
141 \RequirePackage{infwarerr}[2007/09/09]%
142 \RequirePackage{ifluatex}[2010/03/01]%
143 \fi

```

## 2.5 \unexpanded

\ifetex@unexpanded

```

144 \etexcmds@newif{unexpanded}

```

\etex@unexpanded

```

145 \begingroup
146 \edef\x{\string\unexpanded}%
147 \edef\y{\meaning\unexpanded}%
148 \ifx\x\y
149 \endgroup
150 \let\etex@unexpanded\unexpanded
151 \etex@unexpandedtrue
152 \else
153 \edef\y{\meaning\normalunexpanded}%
154 \ifx\x\y
155 \endgroup
156 \let\etex@unexpanded\normalunexpanded
157 \etex@unexpandedtrue
158 \else
159 \edef\y{\meaning\@@unexpanded}%
160 \ifx\x\y
161 \endgroup
162 \let\etex@unexpanded\@@unexpanded
163 \etex@unexpandedtrue
164 \else
165 \ifluatex
166 \ifnum\luatexversion<36 %
167 \else
168 \begingroup
169 \directlua{%
170 tex.enableprimitives('etex@',{'unexpanded'})}%
171 }%
172 \global\let\etex@unexpanded\etex@unexpanded
173 \endgroup
174 \fi
175 \fi
176 \edef\y{\meaning\etex@unexpanded}%

```

```

177     \ifx\x\y
178     \endgroup
179     \etex@unexpandedtrue
180   \else
181     \endgroup
182     \@PackageInfoNoLine{etexcmds}{%
183       Could not find \string\unexpanded.\MessageBreak
184       That can mean that you are not using e-TeX or%
185       \MessageBreak
186       that some package has redefined \string\unexpanded.%
187       \MessageBreak
188       In the latter case, load this package earlier%
189     }%
190     \etex@unexpandedfalse
191   \fi
192 \fi
193 \fi
194 \fi

```

## 2.6 \expanded

\ifetex@expanded

```

195 \etexcmds@newif{expanded}

```

\etex@expanded

```

196 \begingroup
197 \edef\x{\string\expanded}%
198 \edef\y{\meaning\expanded}%
199 \ifx\x\y
200   \endgroup
201   \let\etex@expanded\expanded
202   \etex@expandedtrue
203 \else
204   \edef\y{\meaning\normalexpanded}%
205   \ifx\x\y
206     \endgroup
207     \let\etex@expanded\normalexpanded
208     \etex@expandedtrue
209   \else
210     \edef\y{\meaning\@@expanded}%
211     \ifx\x\y
212       \endgroup
213       \let\etex@expanded\@@expanded
214       \etex@expandedtrue
215     \else
216       \ifluatex
217         \ifnum\luatexversion<36 %
218         \else
219           \begingroup
220             \directlua{%
221               tex.enableprimitives('etex@',{ 'expanded'})%
222             }%
223             \global\let\etex@expanded\etex@expanded
224           \endgroup
225         \fi
226       \fi
227       \edef\y{\meaning\etex@expanded}%
228       \ifx\x\y

```

```

229     \endgroup
230     \etex@expandedtrue
231   \else
232     \endgroup
233     \@PackageInfoNoLine{etexcmds}{%
234       Could not find \string\expanded.\MessageBreak
235       That can mean that you are not using pdfTeX 1.50 or%
236       \MessageBreak
237       that some package has redefined \string\expanded.%
238       \MessageBreak
239       In the latter case, load this package earlier%
240     }%
241     \etex@expandedfalse
242   \fi
243 \fi
244 \fi
245 \fi

246 \etexcmds@AtEnd%
247 \endpackage

```

### 3 Test

#### 3.1 Catcode checks for loading

```

248 \test1)

249 \catcode'\{=1 %
250 \catcode'\}=2 %
251 \catcode'\#=6 %
252 \catcode'\@=11 %
253 \expandafter\ifx\csname count@endcsname\relax
254   \countdef\count@=255 %
255 \fi
256 \expandafter\ifx\csname @gobble@endcsname\relax
257   \long\def\@gobble#1{%
258 \fi
259 \expandafter\ifx\csname @firstofone@endcsname\relax
260   \long\def\@firstofone#1{#1}%
261 \fi
262 \expandafter\ifx\csname loop@endcsname\relax
263   \expandafter\@firstofone
264 \else
265   \expandafter\@gobble
266 \fi
267 {%
268   \def\loop#1\repeat{%
269     \def\body{#1}%
270     \iterate
271   }%
272   \def\iterate{%
273     \body
274     \let\next\iterate
275   \else
276     \let\next\relax
277   \fi
278   \next
279 }%

```

```

280 \let\repeat=\fi
281 }%
282 \def\RestoreCatcodes{}
283 \count@=0 %
284 \loop
285 \edef\RestoreCatcodes{%
286 \RestoreCatcodes
287 \catcode\the\count@=\the\catcode\count@\relax
288 }%
289 \ifnum\count@<255 %
290 \advance\count@ 1 %
291 \repeat
292
293 \def\RangeCatcodeInvalid#1#2{%
294 \count@=#1\relax
295 \loop
296 \catcode\count@=15 %
297 \ifnum\count@<#2\relax
298 \advance\count@ 1 %
299 \repeat
300 }
301 \def\RangeCatcodeCheck#1#2#3{%
302 \count@=#1\relax
303 \loop
304 \ifnum#3=\catcode\count@
305 \else
306 \errmessage{%
307 Character \the\count@\space
308 with wrong catcode \the\catcode\count@\space
309 instead of \number#3%
310 }%
311 \fi
312 \ifnum\count@<#2\relax
313 \advance\count@ 1 %
314 \repeat
315 }
316 \def\space{ }
317 \expandafter\ifx\csname LoadCommand\endcsname\relax
318 \def\LoadCommand{\input etexcmds.sty\relax}%
319 \fi
320 \def\Test{%
321 \RangeCatcodeInvalid{0}{47}%
322 \RangeCatcodeInvalid{58}{64}%
323 \RangeCatcodeInvalid{91}{96}%
324 \RangeCatcodeInvalid{123}{255}%
325 \catcode'\@=12 %
326 \catcode'\=0 %
327 \catcode'\%=14 %
328 \LoadCommand
329 \RangeCatcodeCheck{0}{36}{15}%
330 \RangeCatcodeCheck{37}{37}{14}%
331 \RangeCatcodeCheck{38}{47}{15}%
332 \RangeCatcodeCheck{48}{57}{12}%
333 \RangeCatcodeCheck{58}{63}{15}%
334 \RangeCatcodeCheck{64}{64}{12}%
335 \RangeCatcodeCheck{65}{90}{11}%
336 \RangeCatcodeCheck{91}{91}{15}%
337 \RangeCatcodeCheck{92}{92}{0}%

```



```

338 \RangeCatcodeCheck{93}{96}{15}%
339 \RangeCatcodeCheck{97}{122}{11}%
340 \RangeCatcodeCheck{123}{255}{15}%
341 \RestoreCatcodes
342 }
343 \Test
344 \csname @@end\endcsname
345 \end
346 </test1>

```

## 3.2 Macro tests

```

347 <*test2>
348 \immediate\write16{etexcmds-test2.tex: test file for plainTeX}
349 \input etexcmds.sty\relax
350 \catcode'\@=11 %
351 \edef\x{\string\unexpanded}
352 \edef\y{\meaning\etex@unexpanded}
353 \ifx\x\y
354 \else
355 \@@PackageError{etexcmds-test2}{Test failed}\@ehc
356 \fi
357 \end
358 </test2>

359 <*test3>
360 \NeedsTeXFormat{LaTeX2e}
361 \ProvidesFile{etexcmds-test3.tex}[2016/05/16 v1.6 Test file for LaTeX]
362 \RequirePackage{etexcmds}
363 \makeatletter
364 \edef\x{\string\unexpanded}
365 \edef\y{\meaning\etex@unexpanded}
366 \ifx\x\y
367 \else
368 \@@PackageError{etexcmds-test3}{Test failed}\@ehc
369 \fi
370 \stop
371 </test3>

372 <*test4>
373 \NeedsTeXFormat{LaTeX2e}
374 \ProvidesFile{etexcmds-test4.tex}[2016/05/16 v1.6 Test file for LaTeX]
375 \documentclass{article}
376 \usepackage{m-pictex}
377 \def\normalwritestatus#1#2{%
378 \typeout{EMERGENCY HACK \string\normalwritestatus}%
379 \typeout{#1: #2}%
380 }
381 \usepackage{m-ch-de}
382 \usepackage{etexcmds}
383 \makeatletter
384 \ifetex@unexpanded
385 \edef\x{\string\unexpanded}%
386 \edef\y{\meaning\etex@unexpanded}%
387 \ifx\x\y
388 \else
389 \@@PackageWarningNoLine{etexcmds-test4}{Test failed}%
390 \fi
391 \else
392 \@@PackageWarningNoLine{etexcmds-test4}{%

```

```

393     Test failed because of ConTeXt%
394 }%
395 \fi
396 \stop
397 </test4>

```

## 4 Installation

### 4.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

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

[CTAN:macros/latex/contrib/oberdiek/etexcmds.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 T<sub>E</sub>X 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
```

**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 T<sub>E</sub>X:

```
tex etexcmds.dtx
```

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

```

etexcmds.sty      → tex/generic/oberdiek/etexcmds.sty
etexcmds.pdf      → doc/latex/oberdiek/etexcmds.pdf
test/etexcmds-test1.tex → doc/latex/oberdiek/test/etexcmds-test1.tex
test/etexcmds-test2.tex → doc/latex/oberdiek/test/etexcmds-test2.tex
test/etexcmds-test3.tex → doc/latex/oberdiek/test/etexcmds-test3.tex
test/etexcmds-test4.tex → doc/latex/oberdiek/test/etexcmds-test4.tex
etexcmds.dtx      → source/latex/oberdiek/etexcmds.dtx

```

---

<sup>1</sup><http://ctan.org/pkg/etexcmds>

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  $\text{\TeX}$  distribution (`te\TeX`, `mik\TeX`, ...) relies on file name databases, you must refresh these. For example, `te\TeX` users run `texhash` or `mktextlsr`.

## 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 etexcmds.pdf unpack_files output .
```

**Unpacking with  $\text{\LaTeX}$ .** The `.dtx` chooses its action depending on the format:

**plain  $\text{\TeX}$ :** Run `docstrip` and extract the files.

**$\text{\LaTeX}$ :** Generate the documentation.

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

```
latex \let\install=y\input{etexcmds.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 etexcmds.dtx
makeindex -s gind.ist etexcmds.idx
pdflatex etexcmds.dtx
makeindex -s gind.ist etexcmds.idx
pdflatex etexcmds.dtx
```

# 5 Catalogue

The following XML file can be used as source for the  [\$\text{\TeX}\$  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 `etexcmds.xml`.

```
398 (*catalogue)
399 <?xml version='1.0' encoding='us-ascii'?>
400 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
401 <entry datestamp='$Date$' modifier='$Author$' id='etexcmds'>
402   <name>etexcmds</name>
403   <caption>Avoid name clashes with e-TeX commands.</caption>
404   <authorref id='auth:oberdiek'/>
```

```

405 <copyright owner='Heiko Oberdiek' year='2007,2010,2011' />
406 <license type='lpl1.3' />
407 <version number='1.6' />
408 <description>
409   New primitive commands are introduced in e-TeX; sometimes the
410   names collide with existing macros. This package solves the
411   name clashes by adding a prefix to e-TeX's commands. For
412   example, e-TeX's \unexpanded is provided as
413   \etex@unexpanded.
414
415   <p/>
416   The package is part of the <xref refid='oberdiek'>oberdiek</xref>
417   bundle.
418 </description>
419 <documentation details='Package documentation'
420   href='ctan:/macros/latex/contrib/oberdiek/etexcmds.pdf' />
421 <ctan file='true' path='/macros/latex/contrib/oberdiek/etexcmds.dtx' />
422 <miktex location='oberdiek' />
423 <texlive location='oberdiek' />
424 <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip' />
425 </entry>
426 </catalogue>

```

## 6 History

[2007/05/06 v1.0]

- First version.

[2007/09/09 v1.1]

- Documentation for `\ifetex@unexpanded` added.
- Catcode section rewritten.

[2007/12/12 v1.2]

- `\etex@expanded` added.

[2010/01/28 v1.3]

- Compatibility to `iniTEX` added.

[2011/01/30 v1.4]

- Already loaded package files are not input in plain `TEX`.

[2011/02/16 v1.5]

- Using Lua`TEX`'s `tex.enableprimitives` if available.

[2016/05/16 v1.6]

- Documentation updates.

## 7 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	
<code>\#</code> .....	251
<code>\%</code> .....	327
<code>\@</code> .....	252, 325, 350
<code>\@expanded</code> .....	210, 213
<code>\@unexpanded</code> .....	159, 162
<code>\@PackageError</code> .....	355, 368
<code>\@PackageInfoNoLine</code> .....	182, 233
<code>\@PackageWarningNoLine</code> ....	389, 392
<code>\@ehc</code> .....	355, 368
<code>\@firstofone</code> .....	260, 263
<code>\@gobble</code> .....	257, 265
<code>\@undefined</code> .....	58
<code>\%</code> .....	326
<code>\{</code> .....	249
<code>\}</code> .....	250
A	
<code>\advance</code> .....	290, 298, 313
<code>\aftergroup</code> .....	29
B	
<code>\body</code> .....	269, 273
C	
<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, 249, 250, 251, 252, 287, 296, 304, 308, 325, 326, 327, 350
<code>\count@</code> .....	254, 283, 287, 289, 290, 294, 296, 297, 298, 302, 304, 307, 308, 312, 313
<code>\countdef</code> .....	254
<code>\csname</code> .....	14, 21, 50, 66, 76, 118, 120, 123, 125, 128, 131, 134, 253, 256, 259, 262, 317, 344
D	
<code>\directlua</code> .....	169, 220
<code>\documentclass</code> .....	375
E	
<code>\empty</code> .....	17, 18
<code>\end</code> .....	345, 357
<code>\endcsname</code> .....	14, 21, 50, 66, 76, 118, 120, 123, 125, 128, 131, 134, 253, 256, 259, 262, 317, 344
<code>\endinput</code> .....	29, 114
<code>\endlinechar</code> .....	4, 35, 71, 77, 89
<code>\errmessage</code> .....	306
<code>\escapechar</code> .....	113, 116
<code>\etex@expanded</code> .....	196
<code>\etex@expandedfalse</code> .....	241
<code>\etex@expandedtrue</code> ..	202, 208, 214, 230
<code>\etex@unexpanded</code> .....	2, 145, 352, 365, 386, 413
<code>\etex@unexpandedfalse</code> .....	190
<code>\etex@unexpandedtrue</code> ..	151, 157, 163, 179
<code>\etexcmds@AtEnd</code> ..	95, 96, 111, 112, 246
<code>\etexcmds@newif</code> .....	117, 144, 195
<code>\expanded</code> ....	197, 198, 201, 234, 237
I	
<code>\ifetex@expanded</code> .....	195
<code>\ifetex@unexpanded</code> .....	2, 144, 384
<code>\iffalse</code> .....	121
<code>\ifluatex</code> .....	165, 216
<code>\ifnum</code> ....	166, 217, 289, 297, 304, 312
<code>\iftrue</code> .....	126
<code>\ifx</code> .....	15, 18, 21, 50, 58, 61, 131, 134, 148, 154, 160, 177, 199, 205, 211, 228, 253, 256, 259, 262, 317, 353, 366, 387
<code>\immediate</code> .....	23, 52, 348
<code>\input</code> .....	135, 318, 349
<code>\iterate</code> .....	270, 272, 274
L	
<code>\LoadCommand</code> .....	318, 328
<code>\loop</code> .....	268, 284, 295, 303
<code>\luatexversion</code> .....	166, 217
M	
<code>\makeatletter</code> .....	363, 383
<code>\meaning</code> .....	147, 153, 159, 176, 198, 204, 210, 227, 352, 365, 386
<code>\MessageBreak</code> .....	183, 185, 187, 234, 236, 238
N	
<code>\NeedsTeXFormat</code> .....	360, 373
<code>\next</code> .....	274, 276, 278
<code>\normalexpanded</code> .....	204, 207
<code>\normalunexpanded</code> .....	153, 156
<code>\normalwritestatus</code> .....	377, 378
<code>\number</code> .....	309
P	
<code>\PackageInfo</code> .....	26
<code>\ProvidesFile</code> .....	361, 374
<code>\ProvidesPackage</code> .....	19, 67

<b>R</b>		<code>\typeout</code> . . . . . 378, 379
<code>\RangeCatcodeCheck</code> . . . . .		
. 301, 329, 330, 331, 332, 333,		
334, 335, 336, 337, 338, 339, 340		
<code>\RangeCatcodeInvalid</code> . . . . .		
. . . . . 293, 321, 322, 323, 324		
<code>\repeat</code> . . . . . 268, 280, 291, 299, 314		
<code>\RequirePackage</code> . . . . . 141, 142, 362		
<code>\RestoreCatcodes</code> . . 282, 285, 286, 341		
<b>S</b>		
<code>\space</code> . . . . . 307, 308, 316		
<code>\stop</code> . . . . . 370, 396		
<b>T</b>		
<code>\Test</code> . . . . . 320, 343		
<code>\the</code> . . . . . 77, 78, 79, 80,		
81, 82, 83, 84, 97, 113, 287, 307, 308		
<code>\TMP@EnsureCode</code> . 94, 101, 102, 103,		
104, 105, 106, 107, 108, 109, 110		
<code>\TMP@RequirePackage</code> . . . 132, 138, 139		
<b>U</b>		
<code>\unexpanded</code> . . . . . 146, 147,		
150, 183, 186, 351, 364, 385, 412		
<code>\usepackage</code> . . . . . 376, 381, 382		
<b>W</b>		
<code>\write</code> . . . . . 23, 52, 348		
<b>X</b>		
<code>\x</code> . . . . . 14, 15, 18, 22, 26, 28,		
51, 56, 66, 75, 87, 146, 148, 154,		
160, 177, 197, 199, 205, 211,		
228, 351, 353, 364, 366, 385, 387		
<b>Y</b>		
<code>\y</code> . . . . . 147, 148, 153,		
154, 159, 160, 176, 177, 198,		
199, 204, 205, 210, 211, 227,		
228, 352, 353, 365, 366, 386, 387		