

The `telprint` package

Heiko Oberdiek
<heiko.oberdiek at gmail.com>

2008/08/11 v1.10

Abstract

Package `telprint` provides `\telprint` for formatting German phone numbers.

Contents

1	Documentation	2
1.1	Introduction	2
1.2	Short overview in English	2
1.2.1	Configuration	2
1.3	Documentation in German	2
2	Implementation	3
2.1	Reload check and package identification	3
2.2	Catcodes	4
2.3	Package macros	5
3	Test	7
3.1	Catcode checks for loading	7
4	Installation	9
4.1	Download	9
4.2	Bundle installation	9
4.3	Package installation	9
4.4	Refresh file name databases	10
4.5	Some details for the interested	10
5	History	10
	[1996/11/28 v1.0]	10
	[1997/09/16 v1.1]	11
	[1997/10/16 v1.2]	11
	[1997/12/09 v1.3]	11
	[2004/11/02 v1.4]	11
	[2005/09/30 v1.5]	11
	[2006/02/12 v1.6]	11
	[2006/08/26 v1.7]	11
	[2007/04/11 v1.8]	11
	[2007/09/09 v1.9]	11
	[2008/08/11 v1.10]	11
6	Index	12

1 Documentation

1.1 Introduction

This is a very old package that I have written to format phone numbers. It follows German conventions and the documentation is mainly in German.

1.2 Short overview in English

L^AT_EX:

```
\usepackage{telprint}
\telprint{123/456-789}
```

plain T_EX:

```
\input telprint.sty
\telprint{123/456-789}
```

`\telprint` `\telprint{...}` formats the explicitly given number. Digits, spaces and some special characters ('+', '/', '-', '(', ')', '~', ' ') are supported. Numbers are divided into groups of two digits from the right. Examples:

```
\telprint{0761/12345} ==> 07\,61/1\,23\,45
\telprint{01234/567-89} ==> 0\,12\,34/5\,67\leavevmode\hbox{-}89
\telprint{+49 (6221) 297} ==> +49~(62\,21)~2\,97
```

1.2.1 Configuration

The output of the symbols can be configured by `\telhyphen`, `\telslash`, `\telleftparen`, `\telrightparen`, `\telplus`, `\teltilde`. Example:

```
\telslash{\,/\/,}\ \telprint{12/34} ==> 12\,/\/,34
```

`\telspace` `\telspace` configures the space between digit groups.
`\telnumber` `\telnumber` only formats a number in digit groups; special characters are not recognized.

1.3 Documentation in German

`\telprint`

- `telprint#1`
Der eigentliche Anwenderbefehl zur formatierten Ausgabe von Telefonnummern. Diese dürfen dabei nur als Zahlen angegeben werden (, da sie tokenweise analysiert werden). Als Trenn- oder Sonderzeichen werden unterstützt: '+', '/', '-', '(', ')', '~', ' '. Einfache Leerzeichen werden erkannt und durch Tilden ersetzt, um Trennungen in der Telefonnummer zu verhindern. (Man beachte aus gleichem Grunde die `\hbox` bei '-'). Beispiele:

```
\telprint{0761/12345} ==> 07\,61/1\,23\,45
\telprint{01234/567-89} ==> 0\,12\,34/5\,67\leavevmode\hbox{-}89
\telprint{+49 (6221) 297} ==> +49~(62\,21)~2\,97
```

Der Rest enthält eher Technisches:

`\telspace`

- `\telspace#1`
Mit diesem Befehl wird der Abstand zwischen den Zifferngruppen angegeben (Default: \,). (Durch `\telspace{}` kann dieser zusätzliche Abstand abgestellt werden.)

- \telhyphen
 - \telhyphen#1
Dieser Befehl gibt die Art des Bindestriches, wie er ausgegeben werden soll. In der Eingabe darf jedoch nur der einfache Bindestrich stehen: \telprint{123-45}, jedoch NIE \telprint{123--45}! Kopka-Bindestrich-Fans geben an: \telhyphen{\leavevmode\hbox{--}}
- \telslash
\telleftparen
\telrightparen
\telplus
\teltilde
\telnumber
 - \telslash#1, \telleftparen#1, \telrightparen#1, \telplus#1, \teltilde
Diese Befehle konfigurieren die Zeichen '/', '(,)', '+ und '~'. Sie funktionieren analog zu \telhyphen.
 - \telnumber#1
Richtung interner Befehl: Er dient dazu, eine Zifferngruppe in Zweiergruppen auszugeben. Die einzelnen Zahlen werden im Tokenregister \TELToks gespeichert. Abwechselnd werden dabei zwischen zwei Token (Zahlen) \TELx bzw. \TELY eingefuegt, abhängig von dem wechselnden Wert von \TELswitch. Zum Schluss kann dann einfach festgestellt werden ob die Nummer nun eine geradzahlige oder ungeradzahlige Zahl von Ziffern aufwies. Dem entsprechend wird \TELx mit dem Zusatzabstand belegt und \TELY leer definiert oder umgekehrt.)
 - \TEL... interne Befehle, Technisches:
 - \TELsplit dient zur Aufteilung einer zusammengesetzten Telefonnummer (Vorwahl, Hauptnummer, Nebenstelle). In dieser Implementation werden als Trennzeichen nur '/' und '-' erkannt. Die einzelnen Bestandteile wie Vorwahl werden dann dem Befehl \telnumber zur Formatierung uebergeben.
 - Die Erkennung von einfachen Leerzeichen ist um einiges schwieriger: Die Tokentrennung ueber Parameter #1#2 funktioniert nicht für einfache Leerzeichen, da TeX sie *niemals* als eigenständige Argumente behandelt! (The TeXbook, Chapter 20, p. 201)

(Anmerkung am Rande: Deshalb funktionieren die entsprechenden Tokenmakros auf S. 149 des Buches „Einführung in TeX“ von N. Schwarz (3. Aufl.) nicht, wenn im Tokenregister als erstes ein einfaches Leerzeichen steht!)

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\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@telprint.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{telprint}{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@telprint.sty\endcsname
67 \ProvidesPackage{telprint}%
68 [2008/08/11 v1.10 Formatting of German phone numbers (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 TELAtEnd\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\TELAtEnd{%
96     \TELAtEnd
97     \catcode#1=\the\catcode#1\relax
98   }%
99   \catcode#1=#2\relax
100 }
101 \TMP@EnsureCode{33}{12}% !
102 \TMP@EnsureCode{36}{3}% $
103 \TMP@EnsureCode{40}{12}% (
104 \TMP@EnsureCode{41}{12}% )
105 \TMP@EnsureCode{42}{12}% *
106 \TMP@EnsureCode{43}{12}% +
107 \TMP@EnsureCode{44}{12}% ,
108 \TMP@EnsureCode{45}{12}% -
109 \TMP@EnsureCode{46}{12}% .
110 \TMP@EnsureCode{47}{12}% /
111 \TMP@EnsureCode{91}{12}% [
112 \TMP@EnsureCode{93}{12}% ]
113 \TMP@EnsureCode{126}{13}% ~ (active)
114 \edef\TELAtEnd{\TELAtEnd\noexpand\endinput}

```

2.3 Package macros

```

115 \ifx\DeclareRobustCommand\UnDeFiNeD
116   \def\DeclareRobustCommand*#1[1]{\def#1##1}%
117   \def\TELreset{\let\DeclareRobustCommand=\UnDeFiNeD}%
118   \input infwarerr.sty\relax
119   \@PackageInfo{telprint}{%
120     Macros are not robust!%
121   }%
122 \else
123   \let\TELreset=\relax
124 \fi

\telspace

125 \DeclareRobustCommand*\telspace[1]{\def\TELspace{#1}}
126 \telspace{ }$ , ${}

\telhyphen

127 \DeclareRobustCommand*\telhyphen[1]{\def\TELhyphen{#1}}
128 \telhyphen{\leavevmode\hbox{-}}% \hbox zur Verhinderung der Trennung

\telslash

129 \DeclareRobustCommand*\telslash[1]{\def\TELslash{#1}}
130 \telslash{/}%

\telleftparen

```

```

131 \DeclareRobustCommand*\telleftparen}[1]{\def\TELleftparen{#1}}
132 \telleftparen{ }%

\telrightparen

133 \DeclareRobustCommand*\telrightparen}[1]{\def\TELrightparen{#1}}
134 \telrightparen{ }%

\telplus

135 \DeclareRobustCommand*\telplus}[1]{\def\TELplus{#1}}
136 \telplus{+}%

\teltilde

137 \DeclareRobustCommand*\teltilde}[1]{\def\TELtilde{#1}}
138 \teltilde{~}%

\TELToks

139 \newtoks\TELToks

\TELnumber

140 \def\TELnumber#1#2\TELnumberEND{%
141   \begingroup
142   \def\0{#2}%
143   \expandafter\endgroup
144   \ifx\0\empty
145     \TELToks=\expandafter{\the\TELToks#1}%
146     \ifnum\TELswitch=0 %
147       \def\TELx{\TELSpace}\def\TELy{}%
148     \else
149       \def\TELx{}\def\TELy{\TELSpace}%
150     \fi
151     \the\TELToks
152   \else
153     \ifnum\TELswitch=0 %
154       \TELToks=\expandafter{\the\TELToks#1\TELx}%
155       \def\TELswitch{1}%
156     \else
157       \TELToks=\expandafter{\the\TELToks#1\TELy}%
158       \def\TELswitch{0}%
159     \fi
160     \TELnumber#2\TELnumberEND
161   \fi
162 }

\telnumber

163 \DeclareRobustCommand*\telnumber}[1]{%
164   \TELToks={}%
165   \def\TELswitch{0}%
166   \TELnumber#1{}\TELnumberEND
167 }

\TELSplit

168 \def\TELSplit{\futurelet\TELfuture\TELDosplit}

\TELDosplit

169 \def\TELDosplit#1#2\TELSplitEND
170 {%
171   \def\TELsp{ }%
172   \expandafter\ifx\TELsp\TELfuture
173     \let\TELfuture=\relax
174     \expandafter\telnumber\expandafter{\the\TELToks}~%
175     \telprint{#1#2}% Das Leerzeichen kann nicht #1 sein!

```

```

176 \else
177   \def\TELfirst{#1}%
178   \ifx\TELfirst\empty
179     \expandafter\telnumber\expandafter{\the\TELToks}%
180     \TELToks={}%
181   \else\if-\TELfirst
182     \expandafter\telnumber\expandafter{\the\TELToks}\TELhyphen
183     \telprint{#2}%
184   \else\if/\TELfirst
185     \expandafter\telnumber\expandafter{\the\TELToks}\TELSlash
186     \telprint{#2}%
187   \else\if(\TELfirst
188     \expandafter\telnumber\expandafter{\the\TELToks}\TELleftparen
189     \telprint{#2}%
190   \else\if)\TELfirst
191     \expandafter\telnumber\expandafter{\the\TELToks}\TELrightparen
192     \telprint{#2}%
193   \else\if+\TELfirst
194     \expandafter\telnumber\expandafter{\the\TELToks}\TELplus
195     \telprint{#2}%
196   \else\def\TELtemp{~}\ifx\TELtemp\TELfirst
197     \expandafter\telnumber\expandafter{\the\TELToks}\TELtilde
198     \telprint{#2}%
199   \else
200     \TELToks=\expandafter{\the\TELToks#1}%
201     \TELSplit#2{}\TELSplitEND
202     \fi\fi\fi\fi\fi\fi\fi
203 \fi
204 }

```

\telprint

```

205 \DeclareRobustCommand*\telprint}[1]{%
206   \TELToks={}%
207   \TELSplit#1{}\TELSplitEND
208 }

209 \TELreset\let\TELreset=\UnDeFiNeD

210 \TELAteEnd%
211 \</package>

```

3 Test

3.1 Catcode checks for loading

```

212 {*test1}

213 \catcode'\{=1 %
214 \catcode'\}=2 %
215 \catcode'\#=6 %
216 \catcode'\@=11 %
217 \expandafter\ifx\csname count@\endcsname\relax
218   \countdef\count@=255 %
219 \fi
220 \expandafter\ifx\csname @gobble\endcsname\relax
221   \long\def\@gobble#1{}%
222 \fi
223 \expandafter\ifx\csname @firstofone\endcsname\relax
224   \long\def\@firstofone#1{#1}%
225 \fi
226 \expandafter\ifx\csname loop\endcsname\relax
227   \expandafter\@firstofone
228 \else

```

```

229 \expandafter\@gobble
230 \fi
231 {%
232 \def\loop#1\repeat{%
233 \def\body{#1}%
234 \iterate
235 }%
236 \def\iterate{%
237 \body
238 \let\next\iterate
239 \else
240 \let\next\relax
241 \fi
242 \next
243 }%
244 \let\repeat=\fi
245 }%
246 \def\RestoreCatcodes{}
247 \count@=0 %
248 \loop
249 \edef\RestoreCatcodes{%
250 \RestoreCatcodes
251 \catcode\the\count@=\the\catcode\count@\relax
252 }%
253 \ifnum\count@<255 %
254 \advance\count@ 1 %
255 \repeat
256
257 \def\RangeCatcodeInvalid#1#2{%
258 \count@=#1\relax
259 \loop
260 \catcode\count@=15 %
261 \ifnum\count@<#2\relax
262 \advance\count@ 1 %
263 \repeat
264 }
265 \def\RangeCatcodeCheck#1#2#3{%
266 \count@=#1\relax
267 \loop
268 \ifnum#3=\catcode\count@
269 \else
270 \errmessage{%
271 Character \the\count@\space
272 with wrong catcode \the\catcode\count@\space
273 instead of \number#3%
274 }%
275 \fi
276 \ifnum\count@<#2\relax
277 \advance\count@ 1 %
278 \repeat
279 }
280 \def\space{ }
281 \expandafter\ifx\csname LoadCommand\endcsname\relax
282 \def\LoadCommand{\input telprint.sty\relax}%
283 \fi
284 \def\Test{%
285 \RangeCatcodeInvalid{0}{47}%
286 \RangeCatcodeInvalid{58}{64}%
287 \RangeCatcodeInvalid{91}{96}%
288 \RangeCatcodeInvalid{123}{255}%
289 \catcode'\@=12 %
290 \catcode'\=0 %

```



```

291 \catcode'\%=14 %
292 \LoadCommand
293 \RangeCatcodeCheck{0}{36}{15}%
294 \RangeCatcodeCheck{37}{37}{14}%
295 \RangeCatcodeCheck{38}{47}{15}%
296 \RangeCatcodeCheck{48}{57}{12}%
297 \RangeCatcodeCheck{58}{63}{15}%
298 \RangeCatcodeCheck{64}{64}{12}%
299 \RangeCatcodeCheck{65}{90}{11}%
300 \RangeCatcodeCheck{91}{91}{15}%
301 \RangeCatcodeCheck{92}{92}{0}%
302 \RangeCatcodeCheck{93}{96}{15}%
303 \RangeCatcodeCheck{97}{122}{11}%
304 \RangeCatcodeCheck{123}{255}{15}%
305 \RestoreCatcodes
306 }
307 \Test
308 \csname @@end\endcsname
309 \end
310 </test1>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

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

[CTAN:macros/latex/contrib/oberdiek/telprint.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
```

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 telprint.dtx
```

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

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

```
telprint.sty      → tex/generic/oberdiek/telprint.sty
telprint.pdf     → doc/latex/oberdiek/telprint.pdf
test/telprint-test1.tex → doc/latex/oberdiek/test/telprint-test1.tex
telprint.dtx     → source/latex/oberdiek/telprint.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 `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 telprint.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{telprint.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 telprint.dtx
makeindex -s gind.ist telprint.idx
pdflatex telprint.dtx
makeindex -s gind.ist telprint.idx
pdflatex telprint.dtx
```

5 History

[1996/11/28 v1.0]

- Erste lauffähige Version.
- Nur `'-` und `'/'` als zulässige Sonderzeichen.

[1997/09/16 v1.1]

- Dokumentation und Kommentare (Posting in de.comp.text.tex).
- Erweiterung um Sonderzeichen '(,)', '+', '~' und ' '.
- Trennungsverhinderung am 'hyphen'.

[1997/10/16 v1.2]

- Schutz vor wiederholtem Einlesen.
- Unter L^AT_EX 2_ε Nutzung des \DeclareRobustCommand-Features.

[1997/12/09 v1.3]

- Temporäre Variable eingespart.
- Posted in newsgroup [de.comp.text.tex](#):
“Re: Generisches Markup für Telefonnummern?”²

[2004/11/02 v1.4]

- Fehler in der Dokumentation korrigiert.

[2005/09/30 v1.5]

- Konfigurierbare Symbole: '/', '(,)', '+ und '~'.

[2006/02/12 v1.6]

- LPPL 1.3.
- Kurze Übersicht in Englisch.
- CTAN.

[2006/08/26 v1.7]

- New DTX framework.

[2007/04/11 v1.8]

- Line ends sanitized.

[2007/09/09 v1.9]

- Catcode section added.
- Missing docstrip tag added.

[2008/08/11 v1.10]

- Code is not changed.
- URLs updated.

²Url: <http://groups.google.com/group/de.comp.text.tex/msg/86b3a86140007309>

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; plain numbers refer to the code lines where the entry is used.

Symbols		I	
<code>\#</code>	215	<code>\if</code>	181, 184, 187, 190, 193
<code>\%</code>	291	<code>\ifnum</code>	146, 153, 253, 261, 268, 276
<code>\,</code>	126	<code>\ifx</code>	15,
<code>\@</code>	216, 289		18, 21, 50, 58, 61, 115, 144, 172,
<code>\@PackageInfo</code>	119		178, 196, 217, 220, 223, 226, 281
<code>\@firstofone</code>	224, 227	<code>\immediate</code>	23, 52
<code>\@gobble</code>	221, 229	<code>\input</code>	118, 282
<code>\@undefined</code>	58	<code>\iterate</code>	234, 236, 238
<code>\</code>	290	L	
<code>\{</code>	213	<code>\leavevmode</code>	128
<code>\}</code>	214	<code>\LoadCommand</code>	282, 292
Numbers		<code>\loop</code>	232, 248, 259, 267
<code>\0</code>	142, 144	N	
A		<code>\newtoks</code>	139
<code>\advance</code>	254, 262, 277	<code>\next</code>	238, 240, 242
<code>\aftergroup</code>	29	<code>\number</code>	273
B		P	
<code>\body</code>	233, 237	<code>\PackageInfo</code>	26
C		<code>\ProvidesPackage</code>	19, 67
<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, 213, 214, 215, 216, 251, 260, 268, 272, 289, 290, 291	R	
<code>\count@</code>	218, 247, 251, 253, 254, 258, 260, 261, 262, 266, 268, 271, 272, 276, 277	<code>\RangeCatcodeCheck</code>	265, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304
<code>\countdef</code>	218	<code>\RangeCatcodeInvalid</code>	257, 285, 286, 287, 288
<code>\csname</code>	14, 21, 50, 66, 76, 217, 220, 223, 226, 281, 308	<code>\repeat</code>	232, 244, 255, 263, 278
D		<code>\RestoreCatcodes</code> ..	246, 249, 250, 305
<code>\DeclareRobustCommand</code>	115, 116, 117, 125, 127, 129, 131, 133, 135, 137, 163, 205	S	
E		<code>\space</code>	271, 272, 280
<code>\empty</code>	17, 18, 144, 178	T	
<code>\end</code>	309	<code>\TELEnd</code>	95, 96, 114, 210
<code>\endcsname</code>	14, 21, 50, 66, 76, 217, 220, 223, 226, 281, 308	<code>\TELEndsplit</code>	168, <u>169</u>
<code>\endinput</code>	29, 114	<code>\TELfirst</code>	177, 178, 181, 184, 187, 190, 193, 196
<code>\endlinechar</code>	4, 35, 71, 77, 89	<code>\TELfutur</code>	168, 172, 173
<code>\errmessage</code>	270	<code>\TELhyphen</code>	127, 182
F		<code>\telhyphen</code>	3, <u>127</u>
<code>\futurelet</code>	168	<code>\TELleftparen</code>	131, 188
H		<code>\telleftparen</code>	3, <u>131</u>
<code>\hbox</code>	128	<code>\TELnumber</code>	<u>140</u> , 166
		<code>\telnumber</code>	2, 3, <u>163</u> , 174, 179, 182, 185, 188, 191, 194, 197
		<code>\TELnumberEND</code>	140, 160, 166
		<code>\TELplus</code>	135, 194
		<code>\telplus</code>	3, <u>135</u>
		<code>\telprint</code>	2, 2, 175, 183, 186, 189, 192, 195, 198, <u>205</u>
		<code>\TELreset</code>	117, 123, 209
		<code>\TELrightparen</code>	133, 191
		<code>\telrightparen</code>	3, <u>133</u>

