

# 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

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

# 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<sup>A</sup>T<sub>E</sub>X:

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

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

```
\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`
  - `\telslash#1, \telleftparen#1, \telrightparen#1, \telplus#1, \teltilde`  
Diese Befehle konfigurieren die Zeichen `'/'`, `'(, )'`, `'+'` und `'~'`. Sie funktionieren analog zu `\telhyphen`.
- `\telleftparen`
- `\telrightparen`
- `\telplus`
  - `\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. )
- `\teltilde`
- `\telnumber`
  - `\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 Tokenentrennung 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<sup>A</sup>T<sub>E</sub>X.

```
2 \begingroup
3 \catcode44 12 % ,
4 \catcode45 12 % -
5 \catcode46 12 % .
6 \catcode58 12 % :
7 \catcode64 11 % @
8 \catcode123 1 % {
9 \catcode125 2 % }
10 \expandafter\let\expandafter\x\csname ver@telprint.sty\endcsname
11 \ifx\x\relax % plain-TeX, first loading
12 \else
13 \def\empty{}%
14 \ifx\x\empty % LaTeX, first loading,
15 % variable is initialized, but \ProvidesPackage not yet seen
16 \else
17 \catcode35 6 % #
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{telprint}{The package is already loaded}%
26     \aftergroup\endinput
27     \fi
28     \fi
29 \endgroup
Package identification:
30 \begingroup
31 \catcode35 6 % #
32 \catcode40 12 % (
33 \catcode41 12 % )
34 \catcode44 12 % ,
35 \catcode45 12 % -
36 \catcode46 12 % .
37 \catcode47 12 % /
38 \catcode58 12 % :
39 \catcode64 11 % @
40 \catcode91 12 % [
41 \catcode93 12 % ]
42 \catcode123 1 % {
43 \catcode125 2 % }
44 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
45     \def\x#1#2#3[#4]{\endgroup
46         \immediate\write-1{Package: #3 #4}%
47         \xdef#1{#4}%
48     }%
49 \else
50     \def\x#1#2[#3]{\endgroup
51         #2[#3]}%
52     \ifx#1\undefined
53         \xdef#1{#3}%
54     \fi
55     \ifx#1\relax
56         \xdef#1{#3}%
57     \fi
58     }%
59 \fi
60 \expandafter\x\csname ver@telprint.sty\endcsname
61 \ProvidesPackage{telprint}%
62 [2008/08/11 v1.10 Formatting of German phone numbers (HO)]

```

## 2.2 Catcodes

```

63 \begingroup
64 \catcode123 1 % {
65 \catcode125 2 % }
66 \def\x{\endgroup
67     \expandafter\edef\csname TELAtEnd\endcsname{%
68         \catcode35 \the\catcode35\relax
69         \catcode64 \the\catcode64\relax
70         \catcode123 \the\catcode123\relax
71         \catcode125 \the\catcode125\relax
72     }%
73 }%
74 \x
75 \catcode35 6 % #
76 \catcode64 11 % @
77 \catcode123 1 % {
78 \catcode125 2 % }

```

```

79 \def\TMP@EnsureCode#1#2{%
80   \edef\TELAtEnd{%
81     \TELAtEnd
82     \catcode#1 \the\catcode#1\relax
83   }%
84   \catcode#1 #2\relax
85 }
86 \TMP@EnsureCode{33}{12}% !
87 \TMP@EnsureCode{36}{3}% $
88 \TMP@EnsureCode{40}{12}% (
89 \TMP@EnsureCode{41}{12}% )
90 \TMP@EnsureCode{42}{12}% *
91 \TMP@EnsureCode{43}{12}% +
92 \TMP@EnsureCode{44}{12}% ,
93 \TMP@EnsureCode{45}{12}% -
94 \TMP@EnsureCode{46}{12}% .
95 \TMP@EnsureCode{47}{12}% /
96 \TMP@EnsureCode{61}{12}% =
97 \TMP@EnsureCode{126}{13}% ~ (active)

```

## 2.3 Package macros

```

98 \ifx\DeclareRobustCommand\UnDeFiNeD
99   \def\DeclareRobustCommand#1[1]{\def#1##1}%
100  \def\TELreset{\let\DeclareRobustCommand=\UnDeFiNeD}%
101  \input infwarerr.sty\relax
102  \@PackageInfo{telprint}{%
103    Macros are not robust!%
104  }%
105 \else
106   \let\TELreset=\relax
107 \fi

```

\telspace

```

108 \DeclareRobustCommand*\telspace[1]{\def\TElspace{#1}}
109 \telspace{ }$ , $ { }

```

\telhyphen

```

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

```

\telslash

```

112 \DeclareRobustCommand*\telslash[1]{\def\TElslash{#1}}
113 \telslash{/}%

```

\telleftparen

```

114 \DeclareRobustCommand*\telleftparen[1]{\def\TELleftparen{#1}}
115 \telleftparen{(}%

```

\telrightparen

```

116 \DeclareRobustCommand*\telrightparen[1]{\def\TELrightparen{#1}}
117 \telrightparen{)%

```

\telplus

```

118 \DeclareRobustCommand*\telplus[1]{\def\TELplus{#1}}
119 \telplus{+}%

```

\teltilde

```

120 \DeclareRobustCommand*\teltilde[1]{\def\TELtilde{#1}}
121 \teltilde{~}%

```

```

\TEltoks
122 \newtoks\TEltoks

\TELnumber
123 \def\TELnumber#1#2\TELnumberEND{%
124   \begingroup
125   \def\0{#2}%
126   \expandafter\endgroup
127   \ifx\0\empty
128     \TEltoks=\expandafter{\the\TEltoks#1}%
129     \ifnum\TElswitch=0 %
130       \def\TELx{\TElspace}\def\TEly{}%
131     \else
132       \def\TELx{}\def\TEly{\TElspace}%
133     \fi
134     \the\TEltoks
135   \else
136     \ifnum\TElswitch=0 %
137       \TEltoks=\expandafter{\the\TEltoks#1\TELx}%
138       \def\TElswitch{1}%
139     \else
140       \TEltoks=\expandafter{\the\TEltoks#1\TEly}%
141       \def\TElswitch{0}%
142     \fi
143   \TELnumber#2\TELnumberEND
144 \fi
145 }

\telnumber
146 \DeclareRobustCommand*\telnumber}[1]{%
147   \TEltoks={}%
148   \def\TElswitch{0}%
149   \TELnumber#1}\TELnumberEND
150 }

\TElsplit
151 \def\TElsplit{\futurelet\TElfuture\TEldosplit}

\TEldosplit
152 \def\TEldosplit#1#2\TElsplitEND
153 {%
154   \def\TElsp{ }%
155   \expandafter\ifx\TElsp\TElfuture
156     \let\TElfuture=\relax
157     \expandafter\telnumber\expandafter{\the\TEltoks}~%
158     \telprint{#1#2}% Das Leerzeichen kann nicht #1 sein!
159   \else
160     \def\TElfirst{#1}%
161     \ifx\TElfirst\empty
162       \expandafter\telnumber\expandafter{\the\TEltoks}%
163       \TEltoks={}%
164     \else\if-\TElfirst
165       \expandafter\telnumber\expandafter{\the\TEltoks}\TElhyphen
166       \telprint{#2}%
167     \else\if/\TElfirst
168       \expandafter\telnumber\expandafter{\the\TEltoks}\TElslash
169       \telprint{#2}%
170     \else\if(\TElfirst
171       \expandafter\telnumber\expandafter{\the\TEltoks}\TElleftparen
172       \telprint{#2}%
173     \else\if)\TElfirst

```

```

174     \expandafter\telnumber\expandafter{\the\TELToks}\TELrightparen
175     \telprint{#2}%
176   \else\if+\TELfirst
177     \expandafter\telnumber\expandafter{\the\TELToks}\TELplus
178     \telprint{#2}%
179   \else\def\TELtemp{~}\ifx\TELtemp\TELfirst
180     \expandafter\telnumber\expandafter{\the\TELToks}\TELtilde
181     \telprint{#2}%
182   \else
183     \TELToks=\expandafter{\the\TELToks#1}%
184     \TELSplit#2{}\TELSplitEND
185     \fi\fi\fi\fi\fi\fi\fi
186   \fi
187 }

```

`\telprint`

```

188 \DeclareRobustCommand*{\telprint}[1]{%
189   \TELToks={}%
190   \TELSplit#1{}\TELSplitEND
191 }

192 \TELreset\let\TELreset=\UnDeFiNeD
193 \TELAtEnd
194 </package>

```

## 3 Test

### 3.1 Catcode checks for loading

```

195 <*test1>

196 \catcode'\{=1 %
197 \catcode'\}=2 %
198 \catcode'\#=6 %
199 \catcode'\@=11 %
200 \expandafter\ifx\csname count@\endcsname\relax
201   \countdef\count@=255 %
202 \fi
203 \expandafter\ifx\csname @gobble\endcsname\relax
204   \long\def@gobble#1{%
205 \fi
206 \expandafter\ifx\csname @firstofone\endcsname\relax
207   \long\def@firstofone#1{#1}%
208 \fi
209 \expandafter\ifx\csname loop\endcsname\relax
210   \expandafter@firstofone
211 \else
212   \expandafter@gobble
213 \fi
214 {%
215   \def\loop#1\repeat{%
216     \def\body{#1}%
217     \iterate
218   }%
219   \def\iterate{%
220     \body
221     \let\next\iterate
222   \else
223     \let\next\relax
224   \fi
225   \next
226 }%

```

```

227 \let\repeat=\fi
228 }%
229 \def\RestoreCatcodes{}
230 \count@=0 %
231 \loop
232 \edef\RestoreCatcodes{%
233 \RestoreCatcodes
234 \catcode\the\count@=\the\catcode\count@\relax
235 }%
236 \ifnum\count@<255 %
237 \advance\count@ 1 %
238 \repeat
239
240 \def\RangeCatcodeInvalid#1#2{%
241 \count@=#1\relax
242 \loop
243 \catcode\count@=15 %
244 \ifnum\count@<#2\relax
245 \advance\count@ 1 %
246 \repeat
247 }
248 \expandafter\ifx\csname LoadCommand\endcsname\relax
249 \def\LoadCommand{\input telprint.sty\relax}%
250 \fi
251 \def\Test{%
252 \RangeCatcodeInvalid{0}{47}%
253 \RangeCatcodeInvalid{58}{64}%
254 \RangeCatcodeInvalid{91}{96}%
255 \RangeCatcodeInvalid{123}{255}%
256 \catcode'\@=12 %
257 \catcode'\=0 %
258 \catcode'\{=1 %
259 \catcode'\}=2 %
260 \catcode'\#=6 %
261 \catcode'\ [=12 %
262 \catcode'\]=12 %
263 \catcode'\%=14 %
264 \catcode'\ =10 %
265 \catcode13=5 %
266 \LoadCommand
267 \RestoreCatcodes
268 }
269 \Test
270 \csname @@end\endcsname
271 \end
272 </test1>

```

## 4 Installation

### 4.1 Download

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

[CTAN:macros/latex/contrib/oberdiek/telprint.dtx](http://ftp.ctan.org/macros/latex/contrib/oberdiek/telprint.dtx) The source file.

[CTAN:macros/latex/contrib/oberdiek/telprint.pdf](http://ftp.ctan.org/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.

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



`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 telprint.dtx
```

**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 T<sub>E</sub>X distribution (teT<sub>E</sub>X, miK<sub>T</sub>E<sub>X</sub>, ...) relies on file name databases, you must refresh these. For example, teT<sub>E</sub>X 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 L<sup>A</sup>T<sub>E</sub>X.** The `.dtx` chooses its action depending on the format:

**plain T<sub>E</sub>X:** Run `docstrip` and extract the files.

**L<sup>A</sup>T<sub>E</sub>X:** Generate the documentation.

If you insist on using L<sup>A</sup>T<sub>E</sub>X for `docstrip` (really, `docstrip` does not need L<sup>A</sup>T<sub>E</sub>X), 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 pdfL<sup>A</sup>T<sub>E</sub>X:

```
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<sup>A</sup>T<sub>E</sub>X<sub>2 $\epsilon$</sub>  Nutzung des `\DeclareRobustCommand-Features`.

### [1997/12/09 v1.3]

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

### [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.

---

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

[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.

## 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.

<b>Symbols</b>	
<code>\#</code> .....	198, 260
<code>\%</code> .....	263
<code>\,</code> .....	109
<code>\@</code> .....	199, 256
<code>\@PackageInfo</code> .....	102
<code>\@firstofone</code> .....	207, 210
<code>\@gobble</code> .....	204, 212
<code>\@undefined</code> .....	52
<code>\[</code> .....	261
<code>\]</code> .....	257
<code>\{</code> .....	196, 258
<code>\}</code> .....	197, 259
<code>\]</code> .....	262
<b>Numbers</b>	
<code>\0</code> .....	125, 127
<code>\_</code> .....	264
<b>A</b>	
<code>\advance</code> .....	237, 245
<code>\aftergroup</code> .....	26
<b>B</b>	
<code>\body</code> .....	216, 220
<b>C</b>	
<code>\catcode</code> <i>3, 4, 5, 6, 7, 8, 9, 17, 31, 32,</i>	
<i>33, 34, 35, 36, 37, 38, 39, 40, 41,</i>	
<i>42, 43, 64, 65, 68, 69, 70, 71, 75,</i>	
<i>76, 77, 78, 82, 84, 196, 197, 198,</i>	
<i>199, 234, 243, 256, 257, 258,</i>	
<i>259, 260, 261, 262, 263, 264, 265</i>	
<code>\count@</code> .....	201, 230, 234, 236, 237, 241, 243, 244, 245
<code>\countdef</code> .....	201
<code>\csname</code> .....	10, 18, 44, 60, 67, 200, 203, 206, 209, 248, 270
<b>D</b>	
<code>\DeclareRobustCommand</code> .....	98, 99, 100, 108, 110, 112, 114, 116, 118, 120, 146, 188
<b>E</b>	
<code>\empty</code> .....	13, 14, 127, 161
<code>\end</code> .....	271
<code>\endcsname</code> .....	10, 18, 44, 60, 67, 200, 203, 206, 209, 248, 270
<code>\endinput</code> .....	26
<b>F</b>	
<code>\futurelet</code> .....	151
<b>H</b>	
<code>\hbox</code> .....	111
<b>I</b>	
<code>\if</code> .....	164, 167, 170, 173, 176
<code>\ifnum</code> .....	129, 136, 236, 244
<code>\ifx</code> .....	11, 14, 18, 44, 52, 55, 98, 127, 155, 161, 179, 200, 203, 206, 209, 248
<code>\immediate</code> .....	20, 46
<code>\input</code> .....	101, 249
<code>\iterate</code> .....	217, 219, 221
<b>L</b>	
<code>\leavevmode</code> .....	111

\LoadCommand	249, 266	\TELreset	100, 106, 192
\loop	215, 231, 242	\TELrightparen	116, 174
<b>N</b>			
\newtoks	122	\telrightparen	3, <u>116</u>
\next	221, 223, 225	\TELslash	112, 168
<b>P</b>			
\PackageInfo	23	\telslash	3, <u>112</u>
\ProvidesPackage	15, 61	\TELsp	154, 155
<b>R</b>			
\RangeCatcodeInvalid	240, 252, 253, 254, 255	\TELspace	108, 130, 132
\repeat	215, 227, 238, 246	\telspace	2, 2, <u>108</u>
\RestoreCatcodes	229, 232, 233, 267	\TELsplit	<u>151</u> , 184, 190
<b>T</b>			
\TELAtEnd	80, 81, 193	\TELsplitEND	152, 184, 190
\TELdosplit	151, <u>152</u>	\TELswitch	129, 136, 138, 141, 148
\TELfirst	160, 161, 164, 167, 170, 173, 176, 179	\TELtemp	179
\TELfuture	151, 155, 156	\TELtilde	120, 180
\TELhyphen	110, 165	\teltilde	3, <u>120</u>
\telhyphen	3, <u>110</u>	\TELToks	<u>122</u> , 128, 134, 137, 140, 147, 157, 162, 163, 165, 168, 171, 174, 177, 180, 183, 189
\TELleftparen	114, 171	\TELx	130, 132, 137
\telleftparen	3, <u>114</u>	\TELy	130, 132, 140
\TELnumber	<u>123</u> , 149	\Test	251, 269
\telnumber	2, 3, <u>146</u> , 157, 162, 165, 168, 171, 174, 177, 180	\the	68, 69, 70, 71, 82, 128, 134, 137, 140, 157, 162, 165, 168, 171, 174, 177, 180, 183, 234
\TELnumberEND	123, 143, 149	\TMP@EnsureCode	79, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97
\TELplus	118, 177	<b>U</b>	
\telplus	3, <u>118</u>	\UnDeFiNeD	98, 100, 192
\telprint	2, 2, 158, 166, 169, 172, 175, 178, 181, <u>188</u>	<b>W</b>	
		\write	20, 46
		<b>X</b>	
		\x	10, 11, 14, 19, 23, 25, 45, 50, 60, 66, 74