

# The listingsutf8 package

Heiko Oberdiek  
<heiko.oberdiek at gmail.com>

2007/11/11 v1.1

## Abstract

Package `listings` does not support files with multi-byte encodings such as UTF-8. In case of `\lstinputlisting` a simple workaround is possible if an one-byte encoding exists that the file can be converted to. Also  $\varepsilon$ -TeX and pdfTeX regardless of its mode are required.

## Contents

<b>1</b>	<b>Documentation</b>	<b>1</b>
1.1	User interface . . . . .	1
1.2	Future . . . . .	2
<b>2</b>	<b>Implementation</b>	<b>2</b>
2.1	Catcodes and identification . . . . .	2
2.2	Package options . . . . .	3
2.3	Check prerequisites . . . . .	3
2.4	Add support for UTF-8 . . . . .	4
2.4.1	Conversion . . . . .	4
2.4.2	Patch <code>\lst@InputListing</code> . . . . .	4
<b>3</b>	<b>Test</b>	<b>4</b>
3.1	Catcode checks for loading . . . . .	4
3.2	Test example for latin1 . . . . .	5
<b>4</b>	<b>Installation</b>	<b>6</b>
4.1	Download . . . . .	6
4.2	Bundle installation . . . . .	6
4.3	Package installation . . . . .	6
4.4	Refresh file name databases . . . . .	7
4.5	Some details for the interested . . . . .	7
<b>5</b>	<b>References</b>	<b>7</b>
<b>6</b>	<b>History</b>	<b>8</b>
	[2007/10/22 v1.0] . . . . .	8
	[2007/11/11 v1.1] . . . . .	8
<b>7</b>	<b>Index</b>	<b>8</b>

## 1 Documentation

### 1.1 User interface

Load this package after or instead of package `listings` [2]. The package does not define own options and passes given options to package `listings`.

The syntax of package `listings`' key `inputencoding` is extended:

`inputencoding=utf8/⟨one-byte-encoding⟩`

Example: `inputencoding=utf8/latin1`

That means the file is encoded in UTF-8 and can be converted to the given `⟨one-byte-encoding⟩`. The available encodings for `⟨one-byte-encoding⟩` are listed in section “1.2 Supported encodings” of package `stringenc`’s documentation [3]. Of course, the encoding must encode its characters with one byte exactly. This excludes the unicode encodings (`utf8`, `utf16`, ...).

Only `\lstinputlisting` is supported by the syntax extension of key `inputencoding`.

Internally package `listingsutf8` reads the file as binary file via primitives of pdfTeX (`\pdffiledump`). Then the file contents is converted as string using package `stringenc` and finally the string is read as virtual file by  $\varepsilon$ -TeX’s `\scantokens`.

## 1.2 Future

Workarounds are not provided for

- `\lstinline`
- Environment `lstlisting`.
- Environments defined by `\lstnewenvironment`.

Perhaps someone will find time to extend package `listings` with full native support for UTF-8. Then this package would become obsolete.

## 2 Implementation

1 `(*package)`

### 2.1 Catcodes and identification

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3 \catcode13=5 % ^^M
4 \endlinechar=13 %
5 \catcode123=1 % {
6 \catcode125=2 % }
7 \catcode64=11 % @
8 \def\x{\endgroup
9   \expandafter\edef\csname lstU@AtEnd\endcsname{%
10     \endlinechar=\the\endlinechar\relax
11     \catcode13=\the\catcode13\relax
12     \catcode32=\the\catcode32\relax
13     \catcode35=\the\catcode35\relax
14     \catcode61=\the\catcode61\relax
15     \catcode64=\the\catcode64\relax
16     \catcode123=\the\catcode123\relax
17     \catcode125=\the\catcode125\relax
18   }%
19 }%
20 \x\catcode61\catcode48\catcode32=10\relax%
21 \catcode13=5 % ^^M
22 \endlinechar=13 %
23 \catcode35=6 % #
24 \catcode64=11 % @
25 \catcode123=1 % {
26 \catcode125=2 % }
27 \def\TMP@EnsureCode#1#2{%
28   \edef\lstU@AtEnd{%
29     \lstU@AtEnd
30     \catcode#1=\the\catcode#1\relax
```

```

31 }%
32 \catcode#1=#2\relax
33 }
34 \TMP@EnsureCode{10}{12}% ^^J
35 \TMP@EnsureCode{33}{12}% !
36 \TMP@EnsureCode{36}{3}% $
37 \TMP@EnsureCode{38}{4}% &
38 \TMP@EnsureCode{39}{12}% '
39 \TMP@EnsureCode{40}{12}% (
40 \TMP@EnsureCode{41}{12}% )
41 \TMP@EnsureCode{42}{12}% *
42 \TMP@EnsureCode{43}{12}% +
43 \TMP@EnsureCode{44}{12}% ,
44 \TMP@EnsureCode{45}{12}% -
45 \TMP@EnsureCode{46}{12}% .
46 \TMP@EnsureCode{47}{12}% /
47 \TMP@EnsureCode{58}{12}% :
48 \TMP@EnsureCode{60}{12}% <
49 \TMP@EnsureCode{62}{12}% >
50 \TMP@EnsureCode{91}{12}% [
51 \TMP@EnsureCode{93}{12}% ]
52 \TMP@EnsureCode{94}{7}% ^ (superscript)
53 \TMP@EnsureCode{95}{8}% _ (subscript)
54 \TMP@EnsureCode{96}{12}% '
55 \TMP@EnsureCode{124}{12}% |
56 \TMP@EnsureCode{126}{13}% ~ (active)
57 \edef\lstU@AtEnd{\lstU@AtEnd\noexpand\endinput}

Package identification.
58 \NeedsTeXFormat{LaTeX2e}
59 \ProvidesPackage{listingsutf8}%
60 [2007/11/11 v1.1 Adding support for UTF-8 to listings (HO)]

```

## 2.2 Package options

Just pass options to package listings.

```

61 \DeclareOption*{%
62   \PassOptionsToPackage\CurrentOption{listings}%
63 }
64 \ProcessOptions*

```

Key inputencoding was introduced in version 2002/04/01 v1.0 of package listings.

```

65 \RequirePackage{listings}[2002/04/01]

```

Ensure that \inputencoding is provided.

```

66 \AtBeginDocument{%
67   \ifundefined{inputencoding}{%
68     \RequirePackage{inputenc}%
69   }{}%
70 }

```

## 2.3 Check prerequisites

```

71 \RequirePackage{pdftexcmds}[2007/11/11]
72 \def\lstU@temp#1#2{%
73   \begingroup\expandafter\expandafter\expandafter\endgroup
74   \expandafter\ifx\csname #1\endcsname\relax
75     \PackageWarningNoLine{listingsutf8}{%
76       Package loading is aborted because of missing %
77       \@backslashchar#1.\MessageBreak
78       #2%
79     }%
80     \expandafter\lstU@AtEnd
81   \fi

```

```

82 }
83 \lstU@temp{scantokens}{It is provided by e-TeX}%
84 \lstU@temp{pdf@unescapehex}{It is provided by pdfTeX >= 1.30}%
85 \lstU@temp{pdf@filedump}{It is provided by pdfTeX >= 1.30}%
86 \lstU@temp{pdf@filesize}{It is provided by pdfTeX >= 1.30}%
87 \RequirePackage[stringenc][2007/10/22]

```

## 2.4 Add support for UTF-8

```

\iflstU@utfviii
88 \newif\iflstU@utfviii

\lstU@inputenc

89 \def\lstU@inputenc#1{%
90   \expandafter\lstU@@inputenc#1utf8/utf8/\@nil
91 }

\lstU@@inputenc

92 \lst@Key{inputencoding}\relax{%
93   \def\lst@inputenc{#1}%
94   \lstU@inputenc{#1}%
95 }

```

### 2.4.1 Conversion

```

\lstU@input

96 \def\lstU@input#1{%
97   \iflstU@utfviii
98     \edef\lstU@text{%
99       \pdf@unescapehex{%
100         \pdf@filedump{0}{\pdf@filesize{#1}}{#1}%
101       }%
102     }%
103     \StringEncodingConvert\lstU@text\lstU@text{utf8}\lst@inputenc
104     \def\lstU@temp{%
105       \scantokens\expandafter{\lstU@text}%
106     }%
107   \else
108     \def\lstU@temp{%
109       \input{#1}%
110     }%
111   \fi
112   \lstU@temp
113 }

```

### 2.4.2 Patch \lst@InputListing

```

114 \def\lstU@temp#1\def\lst@next#2#3\@nil{%
115   \def\lst@InputListing##1{%
116     #1%
117     \def\lst@next{\lstU@input{##1}}%
118     #3%
119   }%
120 }
121 \expandafter\lstU@temp\lst@InputListing{#1}\@nil
122 \lstU@AtEnd%
123 \</package>

```

## 3 Test

### 3.1 Catcode checks for loading

```

124 (*test1)

125 \NeedsTeXFormat{LaTeX2e}
126 \documentclass{minimal}
127 \makeatletter
128 \def\RestoreCatcodes{}
129 \count@=0 %
130 \loop
131   \edef\RestoreCatcodes{%
132     \RestoreCatcodes
133     \catcode\the\count@=\the\catcode\count@\relax
134   }%
135 \ifnum\count@<255 %
136   \advance\count@\@ne
137 \repeat
138
139 \def\RangeCatcodeInvalid#1#2{%
140   \count@=#1\relax
141   \loop
142     \catcode\count@=15 %
143   \ifnum\count@<#2\relax
144     \advance\count@\@ne
145   \repeat
146 }
147 \def\Test{%
148   \RangeCatcodeInvalid{0}{47}%
149   \RangeCatcodeInvalid{58}{64}%
150   \RangeCatcodeInvalid{91}{96}%
151   \RangeCatcodeInvalid{123}{127}%
152   \catcode'\@=12 %
153   \catcode'\=0 %
154   \catcode'\{=1 %
155   \catcode'\}=2 %
156   \catcode'\#=6 %
157   \catcode'\[=12 %
158   \catcode'\]=12 %
159   \catcode'\%=14 %
160   \catcode'\ =10 %
161   \catcode13=5 %
162   \RequirePackage{listingsutf8}[2007/11/11]\relax
163   \RestoreCatcodes
164 }
165 \Test
166 \csname @@end\endcsname
167 \end
168 </test1>

```

### 3.2 Test example for latin1

```

169 (*test2)

170 \NeedsTeXFormat{LaTeX2e}
171 \documentclass{minimal}
172 \usepackage{filecontents}
173 \def\do#1{%
174   \ifx#1\^%
175   \else
176     \noexpand\do\noexpand#1%
177   \fi
178 }
179 \expandafter\let\expandafter\dospecials\expandafter\empty
180 \expandafter\edef\expandafter\dospecials\expandafter{\dospecials}
181 \begin{filecontents*}{ExampleUTF8.java}
182 public class ExampleUTF8 {

```

```

183     public static String testString =
184         "Umlauts: " +
185         "^^c3^^84^^c3^^96^^c3^^9c^^c3^^a4^^c3^^b6^^c3^^bc^^c3^^9f";
186     public static void main(String[] args) {
187         System.out.println(testString);
188     }
189 }
190 \end{filecontents*}
191 \usepackage{listingsutf8}[2007/11/11]
192 \def\Text{%
193     Umlauts: %
194     ^^c3^^84^^c3^^96^^c3^^9c^^c3^^a4^^c3^^b6^^c3^^bc^^c3^^9f%
195 }
196 \begin{document}
197 \lstinputlisting[%
198     language=Java,%
199     inputencoding=utf8/latin1,%
200 ]{ExampleUTF8.java}
201 \end{document}
202 \end{test2}

```

## 4 Installation

### 4.1 Download

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

[CTAN:macros/latex/contrib/oberdiek/listingsutf8.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/listingsutf8.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 listingsutf8.dtx
```

---

<sup>1</sup><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):

```
listingsutf8.sty      → tex/latex/oberdiek/listingsutf8.sty
listingsutf8.pdf      → doc/latex/oberdiek/listingsutf8.pdf
test/listingsutf8-test1.tex → doc/latex/oberdiek/test/listingsutf8-test1.tex
test/listingsutf8-test2.tex → doc/latex/oberdiek/test/listingsutf8-test2.tex
test/listingsutf8-test3.tex → doc/latex/oberdiek/test/listingsutf8-test3.tex
test/listingsutf8-test4.tex → doc/latex/oberdiek/test/listingsutf8-test4.tex
test/listingsutf8-test5.tex → doc/latex/oberdiek/test/listingsutf8-test5.tex
listingsutf8.dtx      → source/latex/oberdiek/listingsutf8.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 listingsutf8.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{listingsutf8.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 listingsutf8.dtx
makeindex -s gind.ist listingsutf8.idx
pdflatex listingsutf8.dtx
makeindex -s gind.ist listingsutf8.idx
pdflatex listingsutf8.dtx
```

## 5 References

- [1] Alan Jeffrey, Frank Mittelbach, *inputenc.sty*, 2006/05/05 v1.1b. [CTAN:macros/latex/base/inputenc.dtx](#)
- [2] Carsten Heinz, Brooks Moses: *The listings package*; 2007/02/22; [CTAN:macros/latex/contrib/listings/](#).
- [3] Heiko Oberdiek: *The stringenc package*; 2007/10/22; [CTAN:macros/latex/contrib/oberdiek/stringenc.pdf](#).

## 6 History

[2007/10/22 v1.0]

- First version.

[2007/11/11 v1.1]

- Use of package pdftexcmds.

## 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>	156	<code>\endlinechar</code>	4, 10, 22
<code>\%</code>	159	<b>I</b>	
<code>\@</code>	152	<code>\iflstU@utfviii</code>	88, 97
<code>\@backslashchar</code>	77	<code>\ifnum</code>	135, 143
<code>\@ifundefined</code>	67	<code>\ifx</code>	74, 174
<code>\@ne</code>	136, 144	<code>\input</code>	109
<code>\@nil</code>	90, 114, 121	<b>L</b>	
<code>\[</code>	157	<code>\loop</code>	130, 141
<code>\]</code>	153	<code>\lst@inputenc</code>	93, 103
<code>\{</code>	154	<code>\lst@InputListing</code>	115, 121
<code>\}</code>	155	<code>\lst@Key</code>	92
<code>\]</code>	158	<code>\lst@next</code>	114, 117
<code>\^</code>	174	<code>\lstinputlisting</code>	197
		<code>\lstU@inputenc</code>	90, <u>92</u>
<code>\_</code>	160	<code>\lstU@AtEnd</code>	28, 29, 57, 80, 122
<b>A</b>		<code>\lstU@input</code>	<u>96</u> , 117
<code>\advance</code>	136, 144	<code>\lstU@inputenc</code>	<u>89</u> , 94
<code>\AtBeginDocument</code>	66	<code>\lstU@temp</code>	72, 83,
<b>B</b>			84, 85, 86, 104, 108, 112, 114, 121
<code>\begin</code>	181, 196	<code>\lstU@text</code>	98, 103, 105
<b>C</b>		<b>M</b>	
<code>\catcode</code>	2, 3, 5, 6, 7, 11, 12, 13, 14, 15, 16, 17, 20, 21, 23, 24, 25, 26, 30, 32, 133, 142, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161	<code>\makeatletter</code>	127
<code>\count@</code>	129, 133, 135, 136, 140, 142, 143, 144	<code>\MessageBreak</code>	77
<code>\csname</code>	9, 74, 166	<b>N</b>	
<code>\CurrentOption</code>	62	<code>\NeedsTeXFormat</code>	58, 125, 170
<b>D</b>		<code>\newif</code>	88
<code>\DeclareOption</code>	61	<b>P</b>	
<code>\do</code>	173, 176	<code>\PackageWarningNoLine</code>	75
<code>\documentclass</code>	126, 171	<code>\PassOptionsToPackage</code>	62
<code>\dospecials</code>	179, 180	<code>\pdf@filedump</code>	100
<b>E</b>		<code>\pdf@filesize</code>	100
<code>\empty</code>	179	<code>\pdf@unescapehex</code>	99
<code>\end</code>	167, 190, 201	<code>\ProcessOptions</code>	64
<code>\endcsname</code>	9, 74, 166	<code>\ProvidesPackage</code>	59
<code>\endinput</code>	57	<b>R</b>	
		<code>\RangeCatcodeInvalid</code>	139, 148, 149, 150, 151
		<code>\repeat</code>	137, 145
		<code>\RequirePackage</code>	65, 68, 71, 87, 162
		<code>\RestoreCatcodes</code>	128, 131, 132, 163



S		\TMP@EnsureCode	
\scantokens .....	105	27, 34, 35, 36, 37,	
\StringEncodingConvert .....	103	38, 39, 40, 41, 42, 43, 44, 45, 46,	
		47, 48, 49, 50, 51, 52, 53, 54, 55, 56	
T		U	
\Test .....	147, 165	\usepackage .....	172, 191
\Text .....	192	X	
\the	10, 11, 12, 13, 14, 15, 16, 17, 30, 133	\x .....	8, 20