

# iexec: L<sup>A</sup>T<sub>E</sub>X Package for Inputable Shell Executions\*

Yegor Bugayenko  
yegor256@gmail.com

2026-07-10, 0.16.0

**NB!** This package does not operate on Windows!

## 1 Introduction

This package permits the execution of shell commands directly from within the document and the inclusion of their output in the resulting text:

Today is **10-Jul-2026!**

```
1 \documentclass{article}
2 \usepackage{iexec}
3 \usepackage[paperwidth=3in]{geometry}
4 \pagestyle{empty}
5 \begin{document}
6 Today is \textbf{%
7   \iexec{date +%e-%b-%Y}}\unskip!
8 \end{document}
```

**\iexec** The sole command supplied by this package is `\iexec [options] {cmd}`. Its only mandatory argument, *cmd*, denotes the command to be executed through the terminal shell (`bash`, or whichever shell is configured as the default in the host console).

Compilation must be performed with `pdflatex` (or simply `latex`) under the `--shell-escape` flag, so that [shellesc](#) may execute the supplied shell command.

It is worth noting that L<sup>A</sup>T<sub>E</sub>X invariably employs the “`/bin/sh`” shell. This behaviour cannot be altered, as [explained here](#).

## 2 Options

**quiet** Should the output be required to remain invisible, `\phantom{\iexec{...}}` may be employed. Alternatively, the “`quiet`” option is available:

```
I just want to delete some file:
\iexec[quiet]{rm -f foo.txt}
```

Under this option, whatever the shell command produces is excluded from the document.

**stdout** The output of the executed code is written to the file specified as an optional

---

\*The sources are in GitHub at [yegor256/iexec](#)

argument of `\iexec` (the default being “`iexec.tmp`”):

```
Today is \iexec[stdout=date.txt]{date +%e-%b-%Y | tr -d '\n'}.
```

The trailing portion of the command above removes all line terminators.

**stderr** The error output of the executed code is written to the file specified as an optional argument of `\iexec` (by default the error output is redirected into “`stdout`”):

```
Today is \iexec[stderr=my.txt]{broken-command}.
```

**exit** The exit code of the command is written to a file. The name of this file may be altered by means of the “`exit`” option:

```
Today is \iexec[exit=code.txt]{./broken-command.sh}.
```

**trace** The file thus produced is deleted immediately after its use. To prevent this deletion, the “`trace`” package option may be employed: all files will then remain in the directory where they were created. Tracing may also be enabled globally, for the entire document, by means of the “`trace`” option of the package:

```
\documentclass{article}
\usepackage[trace]{iexec}
\begin{document}
This file is preserved: \iexec[stdout=me.txt]{whoami}.
\end{document}
```

**append** The “`stdout`” thus produced is appended to the file specified:

```
\documentclass{article}
\usepackage[trace]{iexec}
\begin{document}
\iexec[append,stdout=foo.txt,quiet]{echo 'Hello, '}
\iexec[append,stdout=foo.txt,quiet]{echo 'Jeffrey!'}
\input{foo.txt}
\end{document}
```

**unskip** To remove the trailing space following the content, the `unskip` package option may be employed, which appends an `\unskip` command to every `\iexec`:

```
\documentclass{article}
\usepackage[unskip]{iexec}
\begin{document}
Today is \iexec{date +%Y}%!
\end{document}
```

**log** The “`stdout`” thus produced is printed to the TeX log:

```
\iexec[log]{echo 'Hello, \LaTeX!'}.
```

**null** The “`stdout`” of the command is redirected to “`/dev/null`” (or to “`NUL`” on Windows):

```
\iexec[null]{rm some-file.txt}
```

**ignore** By default, an error is reported whenever the exit code differs from zero. This behaviour may be suppressed by means of the “`ignore`” option:

```
\iexec[ignore]{broken-command}
```

maybe In the absence of the `--shell-escape` flag, the `\iexec` command would normally produce a compilation failure. Such failure may be averted by means of the `maybe` option, whereby the execution of `\iexec` is silently skipped whenever `--shell-escape` is not in force:

```
\iexec[maybe]{echo 'Hello, world!'}
```

### 3 Implementation

First, we include the [shellesc](#) package, which we use in order to execute shell commands:

```
1 \RequirePackage{shellesc}
```

Then, we include the [pdftexcmds](#) package, which provides `\pdf@filesize`, working uniformly across `pdftex`, `luatex`, and `xetex` (the `\pdf@filesize` primitive exists in `pdftex` only):

```
2 \RequirePackage{pdftexcmds}
```

Then, we parse package options, with the help of [pgfopts](#):

```
3 \RequirePackage{pgfopts}
4 \pgfkeys{
5   /iexec/.cd,
6   trace/.store in=\iexec@trace,
7 }
8 \ProcessPgfPackageOptions{/iexec}
```

Then, we prepare to parse the options of the `\iexec` command, with the help of [pgfkeys](#):

```
9 \RequirePackage{pgfkeys}
10 \makeatletter\pgfkeys{
11   /iexec/.is family,
12   /iexec,
13   exit/.store in = \iexec@exit,
14   exit/.default = iexec.ret,
15   stdout/.store in = \iexec@stdout,
16   stdout/.default = iexec.tmp,
17   stderr/.store in = \iexec@stderr,
18   trace/.store in = \iexec@traceit,
19   append/.store in = \iexec@append,
20   log/.store in = \iexec@log,
21   null/.store in = \iexec@null,
22   unskip/.store in = \iexec@unskip,
23   quiet/.store in = \iexec@quiet,
24   ignore/.store in = \iexec@ignore,
25   maybe/.store in = \iexec@maybe,
26   stdout,exit
27 }\makeatother
```

`\iexec@typeout` Then, we define an internal command `\iexec@typeout` for printing the content of a file, as suggested [here](#):

```
28 \RequirePackage{expl3}
29 \makeatletter\ExplSyntaxOn
30 \NewDocumentCommand{\iexec@typeout}{m}{
```

```

31 \iexec_typeout_file:n { #1 }}
32 \ior_new:N \g_iexec_typeout_ior
33 \cs_new_protected:Nn \iexec_typeout_file:n
34 {
35 \ior_open:Nn \g_iexec_typeout_ior { #1 }
36 \ior_str_map_inline:Nn \g_iexec_typeout_ior
37 {\iow_term:n { ##1 }}
38 \ior_close:N \g_iexec_typeout_ior
39 }
40 \ExplSyntaxOff\makeatother

```

Then, we detect the operating system, because a few shell fragments differ between Windows and Unix: the null device (`/dev/null` versus `NUL`), the file removal command (`rm` versus `del`), and the way the exit code of the command is captured.

```

41 \makeatletter
42 \newif\ifiexec@windows
43 \ExplSyntaxOn
44 \sys_if_platform_windows:TF{\iexec@windowstrue}{\iexec@windowsfalse}
45 \ExplSyntaxOff
46 \ifiexec@windows
47 \def\iexec@devnull{NUL}
48 \def\iexec@rm{del}
49 \else
50 \def\iexec@devnull{/dev/null}
51 \def\iexec@rm{rm}
52 \fi
53 \makeatother

```

`\iexec` Then, we define `\iexec` command. It is implemented with the help of `\ShellEscape` from `shellesc` package:

```

54 \makeatletter
55 \newread\iexec@exitfile
56 \newcommand\iexec[2][]{%
57 \begingroup%
58 \pgfqkeys{/iexec}{#1}%

```

First, we verify that `latex` is running with the `--shell-escape` option, since without it nothing will work; so, it's better to throw an error earlier than later:

```

59 \ifnum\ShellEscapeStatus=1%
60 \begingroup%

```

Then, we start the log from a clean line:

```

61 \ifdefined\iexec@log%
62 \message{^^J}%
63 \fi%

```

Then, we define a few special chars in order to escape them in the shell (the full list of them is in [macros2e](#)):

```

64 \let%\@percentchar%
65 \let\\ \@backslashchar%
66 \let\{\@charlb%
67 \let\}\@charrb%

```

Then, we execute it and save exit code into a file (where we also add % in order to trim the content to exactly one number, as suggested [here](#)):

```

68      \def\iexec@cmd{(#2)
69      \ifdefined\iexec@append>\fi>
70      \ifdefined\iexec@null\iexec@devnull\else\iexec@stdout\fi
71      \space\ifdefined\iexec@stderr2>\iexec@stderr\else2>&1\fi%
72      \ifiexec@windows%
73      \space&&\space(echo 0)>\iexec@exit\space||\space(echo 1)>\iexec@exit%
74      \else%
75      ;\space/bin/echo -n \string$?\% >\iexec@exit%
76      \fi}%
77      \ShellEscape{\iexec@cmd}%

```

Then, a message is printed to T<sub>E</sub>X log:

```

78      \ifdefined\iexec@log%
79      \message{iexec: [\iexec@cmd]^^J}%
80      \fi%
81      \endgroup%

```

Then, we read back the exit code, from the file:

```

82      \immediate\openin\iexec@exitfile=\iexec@exit%
83      \read\iexec@exitfile to \iexec@code%
84      \immediate\closein\iexec@exitfile%

```

Then, if required, we print the content of the stdout file to T<sub>E</sub>X log:

```

85      \ifdefined\iexec@null\else%
86      \IfFileExists
87      {\iexec@stdout}
88      {}
89      {\PackageError{iexec}{The "\iexec@stdout" file is absent
90      after processing, looks like some internal error}{}}%
91      \ifdefined\iexec@log%
92      \message{iexec: This is the content of '\iexec@stdout'%
93      \space(\pdf@filesize{\iexec@stdout} bytes):^^J}%
94      \IfFileExists
95      {\iexec@stdout}
96      {\iexec@typeout{\iexec@stdout}}
97      {\PackageError{iexec}{The "\iexec@stdout" file is absent
98      after processing, looks like some internal error}{}}%
99      \message{<EOF>^^J}%
100     \else%
101     \ifnum\iexec@code=0\else%
102     \ifdefined\iexec@ignore\else%
103     \message{iexec: See the content of '\iexec@stdout'%
104     \space(\pdf@filesize{\iexec@stdout} bytes)
105     after failure:^^J}%
106     \iexec@typeout{\iexec@stdout}%
107     \message{<EOF>^^J}%
108     \fi%
109     \fi%
110     \fi%
111     \fi%

```

Then, we check whether it's zero or not (if not zero, we either print a message or fail the build, depending on the presence of ignore option):

```

112 \ifnum\iexec@code=0\else%
113 \ifdefined\iexec@ignore%
114 \ifdefined\iexec@log%
115 \message{iexec: Execution failure ignored,
116 the exit code was \iexec@code^^J}%
117 \fi%
118 \else%
119 \PackageError{iexec}{Execution failure,
120 the exit code was \iexec@code}{}%
121 \fi%
122 \fi%

```

Then, we include the produced output into the current document:

```

123 \ifdefined\iexec@null\else%
124 \ifdefined\iexec@quiet%
125 \ifdefined\iexec@log%
126 \message{iexec: Due to 'quiet' option we didn't read
127 the content of '\iexec@stdout'
128 (\pdf@filesize{\iexec@stdout} bytes)^^J}%
129 \fi%
130 \else%
131 \ifdefined\iexec@log%
132 \message{iexec: We are going to include the content of
133 '\iexec@stdout' (\pdf@filesize{\iexec@stdout} bytes)...^^J}%
134 \fi%
135 \input{\iexec@stdout}%
136 \ifdefined\iexec@unskip\unskip\fi%
137 \message{iexec: The content of '\iexec@stdout'
138 was included into the document^^J}%
139 \fi\fi%

```

Then, we delete the file or leave it untouched:

```

140 \ifdefined\iexec@null\else%
141 \ifdefined\iexec@trace%
142 \ifdefined\iexec@log%
143 \message{iexec: Due to package option 'trace',
144 the files '\iexec@stdout' and '\iexec@exit'
145 were not deleted^^J}%
146 \fi%
147 \else%
148 \ifdefined\iexec@traceit%
149 \ifdefined\iexec@log%
150 \message{iexec: Due to 'trace' package option,
151 the files '\iexec@stdout' and '\iexec@exit'
152 were not deleted^^J}%
153 \fi%
154 \else%
155 \ShellEscape{\iexec@rm\space\iexec@stdout}%
156 \ifdefined\iexec@log%
157 \message{iexec: The file '\iexec@stdout' was deleted^^J}%
158 \fi%
159 \ShellEscape{\iexec@rm\space\iexec@exit}%
160 \ifdefined\iexec@log%
161 \message{iexec: The file '\iexec@exit' was deleted^^J}%
162 \fi%

```

```

163      \fi%
164      \fi\fi%

```

Finally, we ignore the whole story if the maybe option is provided and the `--shell-escape` is not set:

```

165      \else%
166      \ifdefined\iexec@maybe%
167      \message{iexec: The execution skipped because --shell-escape
168      is not set and 'maybe' option is provided^^J}%
169      \else%
170      \PackageError{iexec}{You must run TeX processor with
171      --shell-escape option}{}%
172      \fi%
173      \fi%
174      \endgroup%
175 }\makeatother

176 \endinput

```

## Change History

|        |                                                                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0.10.0 | <p><code>\iexec</code>: The ability to track exit code was added. Now, the code is saved into “<code>iexec.ret</code>” file, which is then read and checked for zero value. . . . . 5</p> <p>The file “<code>iexec.ret</code>” is reused for all scripts. . . . . 4</p> <p>The option “<code>ignore</code>” suppresses the checking of “<code>iexec.ret</code>” value. . . . . 4</p> | <p><code>\iexec</code>: The <code>maybe</code> option introduced, allowing the user to skip the entire execution of the <code>\iexec</code> command, when <code>--shell-escape</code> option is off. . . 7</p>                                                                                                                                                                                                                                                                                                                                     |
| 0.11.0 | <p><code>\iexec</code>: The file with exit code now contains just numbers, without end of line. . . . . 5</p> <p>The option “<code>exit</code>” allows to change the name of the file with exit code. . . . . 4</p>                                                                                                                                                                  | <p>0.16.0</p> <p>General: The package now works on Windows too, not only on Unix, because the shell fragments are chosen depending on the operating system. . . . . 4</p>                                                                                                                                                                                                                                                                                                                                                                          |
| 0.11.1 | <p><code>\iexec</code>: When exit code is printed to the file, we add percentchar at the end of line in order to avoid extra space when reading it back. . . . . 5</p>                                                                                                                                                                                                               | <p>0.16.1</p> <p>General: The file size is now printed to the log under <code>luatex</code> and <code>xetex</code> too, not only <code>pdfTeX</code>, because we use <code>\pdf@filesize</code> from <code>pdfTeXcmds</code> instead of the <code>\pdffilesize</code> primitive. . . . . 3</p>                                                                                                                                                                                                                                                     |
| 0.11.2 | <p><code>\iexec</code>: If execution fails, we print the content of “<code>stdout</code>” anyway, even if the “<code>log</code>” is not turned on. . . . . 5</p>                                                                                                                                                                                                                     | <p>0.7.0</p> <p><code>\iexec</code>: The option “<code>append</code>” was introduced — if it’s turned on, <code>stdout</code> will be appended to the file, instead of rewriting it (this is how it was before). . . . . 4</p> <p>The option “<code>log</code>” was introduced, to turn on log/debug messages in TeX log (they were all visible always, which was sometimes annoying). Also, this option enables printing of the entire content of <code>stdout</code> to the log too (this may be pretty convenient for debugging). . . . . 4</p> |
| 0.11.3 | <p><code>\iexec</code>: Bug fixed, because of which we had an extra leading space. . . . . 5</p>                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 0.11.4 | <p><code>\iexec</code>: In this version we escape dollar sign with <code>\string</code> command. . . . . 5</p>                                                                                                                                                                                                                                                                       | <p>0.8.0</p> <p><code>\iexec</code>: The option “<code>null</code>” was introduced, allowing redirection of <code>stdout</code> to “<code>/dev/null</code>”. Doesn’t work on Windows, though. . . . . 5</p>                                                                                                                                                                                                                                                                                                                                        |
| 0.12.0 | <p><code>\iexec</code>: The option “<code>unskip</code>” adds <code>\unskip</code> after each <code>\iexec</code>, in order to strip the trailing end-of-line space. . . . . 4</p>                                                                                                                                                                                                   | <p>0.9.0</p> <p><code>\iexec</code>: The option “<code>stderr</code>” was introduced, allowing redirection of <code>stderr</code> to a file. Without this option specified, <code>stderr</code> will go to <code>stdout</code>. . . . . 5</p>                                                                                                                                                                                                                                                                                                      |
| 0.14.0 | <p>General: The <code>xkeyval</code> package is not used anymore. Instead, we use <code>pgfopts</code> in order to parse package options. . . . . 3</p>                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |



## 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.

|                                          |                                           |                                        |
|------------------------------------------|-------------------------------------------|----------------------------------------|
| <b>Symbols</b>                           | <b>\iexec@maybe</b> . . . . . 25, 166     | <b>\message</b> . . . . . 62,          |
| <b>\%</b> . . . . . 64, 75               | <b>\iexec@null</b> . . . . .              | 79, 92, 99, 103,                       |
| <b>\@backslashchar</b> . . . . . 65      | 21, 70, 85, 123, 140                      | 107, 115, 126,                         |
| <b>\@charlb</b> . . . . . 66             | <b>\iexec@quiet</b> . . . . . 23, 124     | 132, 137, 143,                         |
| <b>\@charrb</b> . . . . . 67             | <b>\iexec@rm</b> 48, 51, 155, 159         | 150, 157, 161, 167                     |
| <b>\@percentchar</b> . . . . . 64        | <b>\iexec@stderr</b> . . . . . 17, 71     |                                        |
| <b>\@</b> . . . . . 65                   | <b>\iexec@stdout</b> 15, 70,              | <b>N</b>                               |
| <b>\{</b> . . . . . 66                   | 87, 89, 92, 93, 95,                       | <b>\NewDocumentCommand</b> 30          |
| <b>\}</b> . . . . . 67                   | 96, 97, 103, 104,                         | <b>\newif</b> . . . . . 42             |
|                                          | 106, 127, 128,                            | <b>\newread</b> . . . . . 55           |
| <b>C</b>                                 | 133, 135, 137,                            |                                        |
| <b>\closein</b> . . . . . 84             | 144, 151, 155, 157                        | <b>O</b>                               |
| <b>\cs</b> . . . . . 33                  | <b>\iexec@trace</b> . . . . . 6, 141      | <b>\openin</b> . . . . . 82            |
|                                          | <b>\iexec@traceit</b> . . . . . 18, 148   |                                        |
| <b>D</b>                                 | <b>\iexec@typeout</b> 28, 96, 106         | <b>P</b>                               |
| <b>\def</b> . . . . . 47, 48, 50, 51, 68 | <b>\iexec@unskip</b> . . . . . 22, 136    | <b>\PackageError</b> . . . . .         |
|                                          | <b>\iexec@windowsfalse</b> 44             | . . . . . 89, 97, 119, 170             |
| <b>E</b>                                 | <b>\iexec&gt;windowstrue</b> . . . . . 44 | <b>\pdf@filesize</b> . . . . .         |
| <b>\endinput</b> . . . . . 176           | <b>\ifdefined</b> . . . . . 61,           | . . . . . 93, 104, 128, 133            |
| <b>\ExplSyntaxOff</b> . . . . . 40, 45   | 69, 70, 71, 78, 85,                       | <b>\pgfkeys</b> . . . . . 4, 10        |
| <b>\ExplSyntaxOn</b> . . . . . 29, 43    | 91, 102, 113, 114,                        | <b>\pgfqkeys</b> . . . . . 58          |
|                                          | 123, 124, 125,                            | <b>\ProcessPgfPackageOptions</b>       |
| <b>G</b>                                 | 131, 136, 140,                            | . . . . . 8                            |
| <b>\g</b> . . . . . 32, 35, 36, 38       | 141, 142, 148,                            |                                        |
|                                          | 149, 156, 160, 166                        | <b>R</b>                               |
| <b>I</b>                                 | <b>\IfFileExists</b> . . . . . 86, 94     | <b>\read</b> . . . . . 83              |
| <b>\iexec</b> . . . . . 31, 33, 54       | <b>\ifiexec@windows</b> . . . . .         | <b>\RequirePackage</b> . . . . .       |
| <b>\iexec@append</b> . . . . . 19, 69    | . . . . . 42, 46, 72                      | . . . . . 1, 2, 3, 9, 28               |
| <b>\iexec@cmd</b> . . . . . 68, 77, 79   | <b>\ifnum</b> . . . . . 59, 101, 112      |                                        |
| <b>\iexec@code</b> . . . . . 83,         | <b>\immediate</b> . . . . . 82, 84        | <b>S</b>                               |
| 101, 112, 116, 120                       | <b>\input</b> . . . . . 135               | <b>\ShellEscape</b> 77, 155, 159       |
| <b>\iexec@devnull</b> 47, 50, 70         | <b>\ior</b> . . . . . 32, 35, 36, 38      | <b>\ShellEscapeStatus</b> . . . . . 59 |
| <b>\iexec@exit</b> . . . . .             | <b>\iow</b> . . . . . 37                  | <b>\space</b> . . . . . 71, 73,        |
| . . . . . 13, 73, 75, 82,                |                                           | 75, 93, 104, 155, 159                  |
| 144, 151, 159, 161                       | <b>M</b>                                  | <b>\string</b> . . . . . 75            |
| <b>\iexec@exitfile</b> . . . . .         | <b>\makeatletter</b> . . . . .            | <b>\sys</b> . . . . . 44               |
| . . . . . 55, 82, 83, 84                 | . . . . . 10, 29, 41, 54                  |                                        |
| <b>\iexec@ignore</b> 24, 102, 113        | <b>\makeatother</b> . . . . .             | <b>U</b>                               |
| <b>\iexec@log</b> 20, 61, 78,            | . . . . . 27, 40, 53, 175                 | <b>\unskip</b> . . . . . 136           |
| 91, 114, 125, 131,                       |                                           |                                        |
| 142, 149, 156, 160                       |                                           |                                        |