

# The csvsimple package

Manual for version 2.2.1 (2022/01/07)

Thomas F. Sturm<sup>1</sup>

<https://www.ctan.org/pkg/csvsimple>

<https://github.com/T-F-S/csvsimple>

## Abstract

`csvsimple` provides a simple L<sup>A</sup>T<sub>E</sub>X interface for the processing of files with comma separated values (CSV). `csvsimple` relies heavily on a key value syntax which results in an easy way of usage. Filtering and table generation is especially supported. Since the package is considered as a lightweight tool, there is no support for data sorting or data base storage.

## 1 Package Options

`csvsimple` is a stub which merely selects to load exclusively one of the following packages:

- «The `csvsimple-13` package»:  
This is the pure L<sup>A</sup>T<sub>E</sub>X3 version of `csvsimple`. It is considered to be the *current* version. New documents are encouraged to use this package.

`csvsimple-13` is loaded with *one* of the following alternatives inside the preamble:

```
\usepackage[13]{csvsimple}
% or alternatively (not simultaneously!)
\usepackage{csvsimple-13}
```

- «The `csvsimple-legacy` package»:  
This is the L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> version of `csvsimple`. It is considered to be the *superseded* version identical to version 1.22 of `csvsimple`. Documents based on that former version do *not have to be changed* and stay compilable in future.

`csvsimple-legacy` is loaded with *one* of the following alternatives inside the preamble:

```
\usepackage{csvsimple}
% or alternatively (not simultaneously!)
\usepackage[legacy]{csvsimple}
% or alternatively (not simultaneously!)
\usepackage{csvsimple-legacy}
```

---

<sup>1</sup>Prof. Dr. Dr. Thomas F. Sturm, Institut für Mathematik und Informatik, Universität der Bundeswehr München, D-85577 Neubiberg, Germany; email: [thomas.sturm@unibw.de](mailto:thomas.sturm@unibw.de)

## 2 Differences between csvsimple-13 and csvsimple-legacy

This section is intended for users who know `csvsimple` before version 2.00.

`csvsimple-13` is a *nearly* drop-in replacement for `csvsimple-legacy`. Although old documents have no *need* to be changed, adopting the new L<sup>A</sup>T<sub>E</sub>X3 version for existing documents should impose not too much effort. Actually, it depends on how intense `pgfkeys` specific styles were used.

That brings us to the differences between the two packages and a more precise understanding what *nearly* drop-in replacement means. The following enumeration does not list new features of `csvsimple-13` (if any), but takes an upgrade point of view.

- Any patches or additions using undocumented internals of `csvsimple-legacy` will stop to function, because `csvsimple-13` has a completely implementation.
- `csvsimple-13` is programmed in `expl3` code using the L<sup>A</sup>T<sub>E</sub>X3 interfaces. No additional packages are loaded or needed with exception of several options which allow to access methods from `ifthen`, `etoolbox`, `longtable`, etc. On the other hand, `csvsimple-legacy` is programmed in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> with dirty tricks from here and there.
- The most significant change of the user interface is that the key value engine of `csvsimple-legacy` is `pgfkeys` (root `/csv/`) while `csvsimple-13` uses `l3keys` (root `/csvsim/`). Names and usage of the keys are *unchanged*. But, if you made own `pgfkeys` *styles* using the `pgfkeys` style handler, these *styles* have to be adapted to `.meta` keys of `l3keys`. The good news is that styles made with `\csvstyle` become `.meta` keys automatically.
- The macro `\csvheadset` is removed. It is not supportable by the new implementation. I never used it and I forgot why I ever wrote it – I hope the same is true for you. If not, `csvsimple-legacy` can be used for documents which needs it.
- Option `/csv/filter` is removed. Instead, `/csvsim/filter ifthen` can be used (also true with `/csv/filter ifthen` for the old version).
- The deprecated options `/csv/nofilter` and `/csv/nohead` are removed. They were not documented any more since years. Obviously, use `/csvsim/no filter` and `/csvsim/no head` instead.
- Compilation problems are to be expected, if an `S` column of the `siunitx` package is used as first or last column. Documents neglecting this rule successfully for `csvsimple-legacy`, may fail to compile with `csvsimple-13`.
- The L<sup>A</sup>T<sub>E</sub>X counters `csvinputline` and `csvrow` are replaced by L<sup>A</sup>T<sub>E</sub>X3 integers `g_csvsim_inputline_int` and `g_csvsim_row_int`, but accessors `\thecsvinputline` and `\thecsvrow` are still valid.
- The packages `pgfrcs`, `pgfkeys`, `ifthen`, `etoolbox`, and `shellesc` are not included anymore (include manually, if needed).
- `\csviffirstrow` and `\csvifoddrow` are deprecated and replaced by `\ifcsvfirstrow` `\ifcsvoddrow` which are more consistent in nomenclature.
- For `csvsimple-13`, data lines are allowed to begin with an backslash.
- Assigned macros like `\myname` for e.g. the third column contain not `\csvcoliii` anymore, but are equal to the content of `\csvcoliii` now.
- Character code changes with `/csvsim/respect percent` etc. and the tabulator as separator should work for `csvsimple-13` as expected in every situation (not always worked

for `csvsimple-legacy`).

- A drawback of `csvsimple-13` against `csvsimple-legacy` is a higher compilation time. This may vary by used compiler. An example document of 5061 pages using a CSV file with 166 992 lines took about 28 seconds with `csvsimple-legacy` and about 51 seconds with `csvsimple-13` on my machine (just a singular observation, no scientific analysis at all).