

**NAME**

csv – facilities for creating Comma-Separated-Values (CSV) data with gawk.

**SYNOPSIS**

```
@include "csv"
result = csvcompose(afield, option...)
result = csvformat(record, option...)
result = csvquote(field, option...)
```

**DESCRIPTION**

The `csv.awk` library provides control variables and functions for composing CSV data records and fields:

**CSVFS**

The expected field separator in the clean text record to be formatted. Default the null character `'\0'`.

**CSVCOMMA**

The resulting CSV field delimiter. Default comma `','`.

**CSVQUOTE**

The resulting CSV quoting character. Default double quote `'\"'`.

**csvcompose(afield [, comma [, quote]])**

Returns a CSV formatted string by composing the values in the *afield* array. The arguments are as follows:

**afield** An array of field values, indexed from 1 to *N*.

**comma**

Optional. The resulting CSV field delimiter. Default CSVCOMMA.

**quote**

Optional. The resulting CSV quoting character. Default CSVQUOTE.

**csvformat(record, [fs [, comma [, quote]])**

Returns a CSV formatted string by composing the fields in the *record* string. The arguments are as follows:

**record** A string record with fields delimited by *fs*.

**fs**

Optional. The actual field separator in *record*. Default CSVFS.

**comma**

Optional. The desired CSV field delimiter. Default CSVCOMMA.

**quote**

Optional. The desired CSV quoting character. Default CSVQUOTE.

**csvquote(field [, comma [, quote]])**

Returns a CSV formatted string by escaping the required characters in the *field* string. The arguments are as follows:

**field** A single field clean text string.

**comma**

Optional. The desired CSV field delimiter. Default CSVCOMMA.

**quote**

Optional. The desired CSV quoting character. Default CSVQUOTE.

**EXAMPLES**

Explicit CSV composition:

```
f[1] = "007"
f[2] = "Bond, James"
f[3] = "United Kingdom"
result = csvcompose(f) # -> '007,"Bond, James",United Kingdom'
result = csvcompose(f, ";") # -> '007;Bond, James;United Kingdom'

record = "007/Bond, James/United Kingdom"
```

```
result = csvformat(record, "/" ) # -> '007,"Bond, James",United Kingdom'  
result = csvformat(record, "/", ";") # -> '007;Bond, James;United Kingdom'
```

**NOTES**

The *csv* library automatically loads the CSV extension.

**BUGS****AUTHOR**

Manuel Collado, [m-collado@users.sourceforge.net](mailto:m-collado@users.sourceforge.net).

**SEE ALSO**

*CSV Processing With gawk*, *csvparse(3am)*, *csvmode(3am)*.

**COPYING PERMISSIONS**

Copyright (C) 2018, Free Software Foundation, Inc.

Permission is granted to make and distribute verbatim copies of this manual page provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual page under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual page into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Foundation.