NAME

csvcut – print selected fields from CSV files

SYNOPSIS

csvcut [OPTION]... [FILE]...

DESCRIPTION

Read CSV data from standard input or *FILE*s and print the selected fields to standard output.

-d, **--delimiter**=*DELIM*
> Use *DELIM* instead of the comma character as the delimiter character

-f, **--fields**=*FIELD-LIST*
> *FIELD-LIST* is a comma seperated list of fields or field ranges to select. Fields may be specified by number starting at 1 or by field name. A field name must contain at least one non-digit character. Ranges may also be used and consist of two field numbers or names seperated by a hyphen. For example, the field list **"1,3,5-City"** would select the first and third fields as well as all fields between the fifth field and the field named "City" inclusive. When using field names it is assumed that the first non-empty record contains a header with field names that match the names used in the field list, if any field names cannot be resolved from the first record an error will occur. If the same field name occurs multiple times in the header record, the first one seen is the one used.

NOTE: unlike the UNIX cut program, this program will output the list in the order requested and will also output the same field multiple times if requested. For example, **sort -f 3,1,1,4** would cause sort to print the first field, followed by the third field, followed but the fourth field whereas csvsort will output the third field once, the first field twice, and the fourth field one.

-r, **--reresolve-fields**
> re-resolve the field names specified for each file processed instead of using the positions resolved from the first file. By default, when processing multiple files, only the header from the first file is used to map the field names to field numbers. This option will cause csvcut to attempt to re-aquire this information for each file and will cause an error if any of the files fails to contain the neccessary information to do so.

-s, **--strict**
> enforce strict mode, mal-formed CSV files will cause an error

-c, **--complement**
> output all fields except those specified In this case the fields are output in the order in which they appear in the input ignoring those fields which are specified by the field list

-m, **--make-empty-fields**
> cause the creation of empty fields for those specified in the field specs but not in the data. By default records will only be printed if they exist in a record. For example, if a record contains only 2 fields the field list of 3-4 will not result in any fields being displayed. With this option set, empty fields will be printed, one for each field specified that does not exist in the input.

-q, **--quote**=*QUOTE*
> Use *QUOTE* instead of double quote as the quote character

--help Display a help message and exit

--version
> Print version information to stderr and exit

AUTHOR

Written by Robert Gamble.

BUGS

    Please send questions, comments, bugs, etc. to: rgamble@sourceforge.net

COPYRIGHT

    Copyright © 2007 Robert Gamble

    This is free software. You may redistribute copies of it under the terms of the GNU General Public License <http://www.gnu.org/licenses/gpl.html>. There is NO WARRANTY, to the extent permitted by law.

SEE ALSO

    csvcount(1), csvcheck(1), csvfix(1), csvcut(1), csvgrep(1), csvbreak(1)