NAME

csvgrep − print selected fields from CSV files

SYNOPSIS

csvgrep [OPTION]... PATTERN [FILE]...

DESCRIPTION

Read CSV data from standard input or *FILE*s searching for *PATTERN* in the specified fields and print matching rows 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 search, 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.

-H, **--with-filename**
: prefix matches with the filename and a colon This option is implied when matching multiple files with the -c option

-h, **--no-filename**
: suppress filename prefixing

-l, **--files-with-matches**
: Suppress normal output and print only the names of files that would have produced output

-L, **--files-without-match**
: Suppress normal output and print only the names of files that would have resulted in no output

-P, **--perl-regexp**
: interpret *PATTERN* as a perl-compatible regular expression (requires pcre). See http://perldoc.perl.org/perlre.html for expression syntax

-i, **--ignore-case**
: perform case insensitive match

-v, **--invert-match**
: select all records that do not match pattern

-n, **--record-number**
: prefix matched records with record numbers

-F, **--fixed-strings**
: interpret pattern as a fixed literal string instead of a regular expression

-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, **--count**
print only a count of matching records

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

--print-header
print CSV header, this is the default when non-numeric field names are specified

--no-print-header
do not print a header

--help Display a help message and exit

--version
Print version information to stderr and exit

LIMITATIONS

Matching with POSIX regular expressions (-E) or fixed text (-F) will not match data after the first null character of a field, this limitation is not present when using Perl Compatible Regular Expressions (-P).

Using **-i** with **-F** may not produce the intended results with non-ASCII text.

AUTHOR

Written by Robert Gamble.

BUGS

Please send questions, comments, bugs, etc. to: rgamble99@gmail.com

COPYRIGHT

SEE ALSO

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