

NAME

table – format simple tables

SYNOPSIS

table [**-R**] [**-acdhp**] [**-l** *length*] [**-w** *width*] [**-t** *separator*] [**-i** *indent*] [**-H** *num_headerlines*]
 [**-O** *formatline*] [**-I** *insertline*] [**-A** *appendline*] *file* ...

DESCRIPTION

table is intended for formatting simple *.csv* and similarly structured data into a human-readable tables suitable for displaying on alphanumeric terminals. It is designed to overcome some of the limitations of **column -t** while less complex than full processing by **troff** and **tbl**. At the same time, the *roff* mode of **table** allows to format a subset of **tbl** tables in a somewhat simplified manner. The output can be suitable for paginating via **more** or **less**, and can utilize ANSI escape sequences for highlighting and coloring the output.

OPTIONS

The command line is parsed according to the usual UNIX convention. Whitespace is *required* between a command-line option and its argument.

-a Renders font and color changes using ANSI escape sequences. To display them properly in e. g. **less**, an appropriate option (i. e. **-R** for **less**) is required. The default is to do no font and color rendering.

-A *append_line*

Appends the *append_line* after the data contained in *files*. This option can be used multiple times to insert multiple lines, and is ignored in *roff* mode.

-c Center the table to the terminal width. In *roff* mode, this option is ignored, use the **center** global option within the table specification instead.

-d Renders font changes using the backspace overstriking recognized by pagers like **more** or **less**. The default is to do no font rendering.

-O *format_line*

Adds one format specification line, with format specifiers separated by whitespace, or by the , (comma) character, or by the character defined by the **-t** option (to avoid the need to quote the argument within the shell). There can be multiple **-O** options, each specifying the format for one data input line, the last one being used for the rest of the table. Only a subset of **tbl** key characters is supported:

l Left-justify the field content within the column.

c Center the field content within the column.

r Right-justify the field content within the column.

n Numerically justify the field content within the column: The field content is expected to be a decimal number, and the dots (or commas) within the number are aligned vertically. More than one decimal separator is not supported.

The column specifiers are fairly limited, only the following being recognized by **table**:

B Renders the field in bold font.

I Renders the field in italic font, if the output device supports it, or underlined, if the output device supports underlining instead of italicizing text.

U Renders the field underlined. This is **table** extension.

O Renders the field striked over, if the output device supports it. This is **table** extension.

@N

Renders the field text using ANSI color *N*. This is **table** extension.

@NM

Renders the field text using ANSI color *N*, and the field background using ANSI color *M*. This is **table** extension.

This option is ignored in *roff* mode.

-g Guess the format from the table data. Note: This option is not yet implemented.

-h Print short synopsis message.

-H num_headerlines

Specifies the number of data lines considered to be a table header. If pagination is used, these lines are printed at the top of each page.

This option is ignored in *roff* mode. In *roff* mode, use the **.TS H** and **.TH** macros/directives instead.

-i indent

Sets the output table indent to *indent* characters. Mutually exclusive with **-c**. This option is ignored in *roff* mode.

-I insert_line

Inserts (prepends) the *insert_line* before the data contained in *files*. This option can be used multiple times to insert multiple lines, and is ignored in *roff* mode.

-l length

Sets the output page length in lines. Note that when feeding the output into e. g. *more(1)* or *less(1)* one line is spent by the pager itself, therefore the page length must be set to one line less than the actual terminal window length. The default is the *LINES* environment variable, or 24, if not set.

-p Paginate. Table header is printed on top of each page, and empty lines are supplied on last page to fill the page. Be careful when piping the output to a pager like **more** or **less**, as they use the last line of the terminal for displaying a message, and therefore the length length supplied via **-l** or *LINES* needs to be one line less than the actual terminal size.

-R *Roff* mode. See **tbl(1)** for reference on Troff tables, but beware of the limitations:

- Only *tab(c)*, *center*, and *box* global options are recognized
- Only **l**, **c**, **r**, and **n** format keys are supported
- Only very limited format specifiers are recognized
- ANSI color support using the **@** key is a **table** extension.

Also note that **-A**, **-c**, **-O**, **-H**, **-i**, **-I**, and **-t** options are ignored in *roff* mode.

-t separator

Sets the input field separator to *separator* similarly to the **tab(separator)** global option in **tbl**. This option is ignored in *roff* mode, use the **tab(separator)** instead. The default is to use the horizontal tab (ASCII HT, 9 dec, 011 oct, 09 hex). Note that using a character that has special meaning in regular expressions can cause weird effect and will probably cause **table** to crash.

-w width

Sets the output page width in characters. The default is the *COLUMNS* environment variable, or 80, if not set.

ENVIRONMENT

LINES The number of output page lines, unless **-l** is specified.

COLUMNS

The number of output page lines, unless **-w** is specified.

EXAMPLES

The table in csv format contains four columns, the first one being left-justified, the other three being numbers. The headings are supplied using the **-I** option, are set in bold, the headings for the numeric columns being centered. There is a horizontal line between the headings and the body of the table, and at the end of the table. This output is suitable for displaying on a terminal:

```
table -t , -a -O lB,cB,cB,cB -O l,n,n,n \
      -I Item,Height,Width,Length -I _ -A _ \
      file.csv
```

The same table being paginated and fed to a printer (the page dimensions being 65 lines and 80 characters). Note that no bold rendering is done here:

```
table -t , -O lB,cB,cB,cB -O l,n,n,n -l 65 -w 80 -P \
      -I Item,Height,Width,Length -I _ -A _ \
      file.csv | lpr -P myprinter
```

The table intended for the **tbl** preprocessor of **troff** is paginated and fed into the **less** pager. All the formatting specifications, including the headers, are contained within the file. Please note that when paged into e.g. **less**, the terminal length needs to be compensated to paginate properly:

```
eval $(resize)
table -Racp -w $COLUMNS -l $(expr $LINES - 1) file.troff \
less -R
```

NOTES

Keep in mind that at the current stage of development, there is almost no diagnostics and error recovery, and that **table** may silently fail, crash, or produce incorrect output when invalid input or options are supplied.

VERSION

This is **table** version 0.1.

AUTHORS

table is being written (and maintained) by Martin Cernohorsky. At this stage, **table** is to be considered to be a work in progress. Please note that option names and functionality may change in future versions of **table**.

SEE ALSO

column(1), **tbl**(1), **roff**(7)