

NAME

tless – Quick Viewer for text files of tabular data

SYNOPSIS

tless [-f *inputfile*] [*options*] [-X “**Key Commands**”]

DESCRIPTION

Tless is a utility to quickly view tabular text data. It was inspired by ‘less’ and acts in a similar capacity for tables. To this end, it offers several essential features.

Columns: Tless is aware of the concept of a table “column” and operates in two dimensions instead of just one. Accordingly, it allows convenient movement and searches using both rows and columns.

Parsing: Rows are parsed on the fly in a customizable manner. The input may be delimited, fixed width, or csv.

Synchronized Scrolling: Rows and columns may be frozen and unfrozen dynamically. Tless performs synchronized scrolling with these (frozen columns only scroll vertically and frozen rows only scroll horizontally). This allows a user to scan through a large file of data, both horizontally and vertically, without losing sight of any header rows or index columns. Because the frozen rows and columns need not be consecutive, Tless offers more flexibility than a typical split-screen application.

Lightweight: Tless examines a file locally and uses a caching mechanism. Like ‘less’, it can be used to view very large files (in both directions) with minimal start-up time, performance loss, or memory foot-print. It uses curses and is a terminal application for both portability and speed.

Note that tless is a viewing utility. Although it provides basic search functionality, it is not a spreadsheet or database and allows no editing, calculation, or DB style queries. It can be thought of as a lightweight two dimensional data viewer.

SCREEN LAYOUT

The screen is broken into several portions, some of which may not be visible at any given time.

Row Index: The leftmost column displays the row number in the file. This may be turned on or off dynamically.

Frozen Columns: Next from the left are any frozen columns in the order in which they were frozen by the user. These only scroll vertically and may be added or removed dynamically.

Column Index: The topmost row displays the column numbers. This may be turned on or off dynamically.

Frozen Rows: Next down are any frozen rows in the order in which they were frozen by the user. These only scroll horizontally and may be added or removed dynamically.

Message Line: The bottom row is for commands and messages.

Active Region: The remainder of the columns and rows are “active” and scroll both horizontally and vertically. This is a window into the file.

Frozen Cells: Note that in the upper left region, if both rows and columns are frozen there will be an overlap which doesn’t scroll in either direction.

[illegible]

OPTIONS

Note that in some cases the order of options makes a difference. In particular, any default justification should be set before fixed columns are specified. Also, some options only are valid in the context of others. Sometimes invalid options are ignored, but if ambiguous or indicative of user error they may cause the program to exit.

General

```
-h  --help
    Display usage information.

-v  --version
    Display version number.

-f  --file filename
    Specify an input file. At this time, reading from stdin is not supported.

-z  --cache n
    Specify a cache size. The maximal cache size will be 3x this because front and end caches are also
    used. DEFAULT=10000.

-X  --exec ``cmds``
    Execute a series of key commands once initialized. The commands are space delimited and each has
    either the form 'c' or 'c:arg:arg:...' where c is the key for the command and any arguments are those
    that would be typed after them.
```

Parsing

Only one of `--delim`, `--sdelim`, `--csv`, or `--fixed` may be used.

WARNING: If escaped chars like `\t` appear in the argument to `—delim` or `—sdelim` then one must be wary of the shell. For example, in `bash`, one would write `—sdelim '$\t '` to delimit with tab and space.

- d `--delim string`
- s `--sdelim string`

The file will be parsed using the specified characters as delimiters. For `--delim`, Adjacent delimiters are merged while for `--sdelim` they are not. `DEFAULT=` tab and space.
- t Add a tab to the delimiter string. Some shell scripts make it a huge pain to pass a tab to the program. If that is the case, this may be used instead. It adds to the string so other characters may be included in the `-d` or `-s` options as well. To parse only with tabs, use `-s " -t` or something similar. Note that `-t` implies `-d` if no `-d` or `-s` option is explicitly specified.

- c --csv
The file will be parsed according to the csv specification. NOTE: At present, this is a simplified parser. It does not allow for arbitrary csv files. It allows "" or non-"" around field, and allows ,s inside of "" but nothing more complex than this.
- x --fixed wid1:wid2:wid3:...
The file will be parsed using the specified fixed column widths.
- l --ignoreleading on/off
Ignore leading delimiters. This eliminates misalignments due to some rows starting on a delimiter and some not. DEFAULT=on.
- y --doseol
Assume DOS-style end of line ('\r').
- e --eol char
Assume this specific char marks the end of line. If 0 then either \$\n\$ or \$\r\$ will match. Note that null always matches and that adjacent eols yield separate lines and are not merged. DEFAULT= 0 ('\n' or '\r').
- M --maxcols n
Specify a maximum number of columns to parse and display. Any additional columns are ignored.
- Q --removequotes
Strip outer double-quotes (") if they are present for each field. The quotes are only stripped from a field if they appear as a pair.
- k --skip n
Skip the first n rows of the file for all purposes.
- m --allowexpansion
Allow the file to expand while in use. The reason this must be specified is that a file without a terminal eol would lose the last line if we allow expansion (because we must hold off processing any line without an eol until it is complete).

Display

- a --aligntofirst
Use the specified delimiters to set the column widths to those in the first row (or first non-skipped row). Only for use with --delim and --sdelim. Note that this may lead to visually unappealing results if the first row is a header with headings that are smaller than the data entries below them (but this may always be corrected using the key commands "+", "-", and "W").
- w --defwidth n
Set the column width for all columns to n. DEFAULT=12.
- G --grid [c]
Display separators between columns. If 'c' is specified, use that character as the separator. Otherwise, "|" is the default. Note that the grid may be turned on and off dynamically as well (see key command "x").
- S --seps [c]
Display separators between the frozen rows/cols and the active portion of the screen. If 'c' is specified, use that character as the separator. Otherwise, the default is "#". The separators may be toggled on and off dynamically as well (see key command "y").
- R --rowindex on/off/n
Activate or deactivate the leftmost column, which is a row index. If the argument is a number instead of on/off, use that as the width. Otherwise, the default width is 6 characters (followed by a space). This may be toggled on and off dynamically using the "P" key command. DEFAULT= 6 (on).

- C --colindex on/off
Activate or deactivate the topmost row, which is a column index. This may be toggled on and off dynamically using the “V” key command. DEFAULT=on.
- j --justify right/left
Set the default justification for all columns. This may be changed dynamically using the “j” and “J” key commands. DEFAULT=right.
- F --filler char
Specify a filler character for any empty space on the screen. DEFAULT= “ ” (white space).

KEYBOARD COMMANDS

The following keys perform actions while running. Those with [foo] after them indicate a required argument (as opposed to the option section where it indicated an optional argument). This is input on the command line (the bottom line of the terminal) followed by enter. Note that the number of columns may grow if the document width changes). The movement commands only affect the active region.

Searches may either be ordinary (simple text matching) or regexps. The type is autodetected depending on whether the string is surround by a // pair. Note that for searches, the matching text may not be visible on the screen if the cell is too narrow to display it. Using the 'v' command should allow full visibility. Regardless, the matching cell row and col are displayed in the message bar. Also note that regexps are compared per cell. That is, the cell itself is viewed as 'line' rather than the row as a whole. Typically, the active window aligns to the next search result in the upper left corner. However, it may not do so (or appear to do so) if the result appears in the frozen part of the screen. Hidden columns are not searched.

- h Print usage information and exit.
- l,r Move left or right one column.
- a,e Move to the far left or right.
- <, > Move to the left or right one screen.
- y,k Move up one line.
- j Move down one line.
- b,u Move up one screen.
- f,d, spc Move down one screen.
- g,G Move to the top or bottom of the document (note that G may take a while for a large file).
- C [n] Toggle whether column n is frozen. Frozen columns appear on the left in the order in which they were frozen.
- C [.] Unfreeze all columns.
- C [n] Toggle whether row n is frozen. Row n must be visible (or in the cache) in order to freeze it (but any frozen row may be unfrozen). Frozen rows appear on the top in the order in which they were frozen.
- R [.] Unfreeze all rows.
- s or / [foo] Row-wise search. Places top-left corner at search result and displays the result on the message line. The search is forward and down from the upper left active cell (or the previous result). If foo is of the form /bar/ then it is interpreted as a regexp. Otherwise it is treated like an ordinary string to match.
- S or ? [foo] Column-wise search. Places top-left corner at search result and displays the result on the message line. The search is down and forward from the upper left active cell (or the previous result). If foo is of the form /bar/ then it is interpreted as a regexp. Otherwise it is treated like an ordinary string to match.
- n,p Next, previous search result for last search. “p” may also be used to search backwards.
- v [row:col] Display the contents of the specified cell (on screen) in the message area. Useful when a cell is too small to fully display its contents.
- H [n] Toggle hidden status of column n. Hidden columns are not shown or searched and may or may not be frozen.

H [.]	Unhide all columns.
+ [n]	Expand width of column n by 1.
- [n]	Shrink width of column n by 1. The width cannot drop below 1 (to hide the column, use 'H').
+ [.]	Expand width of all columns by 1.
- [.]	Shrink width of all columns by 1. The width of any column cannot drop below 1.
W [n:w]	Specify width of column n as w.
W [.:w]	Specify width of all columns as w.
m [c]	Mark position and label it with letter c.
' [c]	Return to the position labelled with letter c.
^ [n]	Justify column n to left.
\$ [n]	Justify column n to right.
^ [.]	Justify all columns to left.
\$ [.]	Justify all columns to right.
x	Toggle grid.
z	Toggle frozen section separators.
P	Toggle row index column.
V	Toggle column index row.
L [row:col]	Move the upper left corner of the active region to the specified row and column. If they are frozen or hidden, find a nearby substitute.
q	Quit.
Y	Resize the screen to fit the terminal. This should be done automatically when the terminal is resized, but a command is provided in case a manual repaint is needed.
F	Clear EOF info so that new lines added afterward can be read.

Note that the G and S commands necessarily involve a seek to the end of the file and will be expensive for large files. Also, the commands s, n, p, and L may prove expensive depending on the details of the file and request.

Key Summary

Action	Cols	Rows
Move	a < l r > e	g b u y k j f d spc G
(Un)Freeze	C [n .]	R [n .]
(Un)Hide	H [n .]	
Search	S ? [str /str/]	s / [str /str/]
Col width:	+/-[n .]	W[n:m] W[.:m]
Cell just (lft rgt):	^ \$ [n .]	
Mark set recover:	m ' [c]	
Cell:	L[row:col] (goto) v[row:col] (disp)	
General:	q(quit) h(hlp) Y(draw) F(rsteof)	
Disp:	x(grid) z(seps) P(rowidx) V(colidx)	

EXAMPLES

Parse csv file: **tless -c -f foo.csv**

Parse tab-delimited: **cat foo.txt | tless -s "\\t" -a**

Parse white-space separated columns: **cat foo.txt | tless -d " " -a**

Parse left-justified fixed width columns: **tless -f foo.txt -j left -x:10:20:30:40:50:65**

Parse ':' separated columns that are all the same width: **tless -f foo.txt -d ":" -w 15**

A simple setup **tless -f foo.txt -X "x y C:0 R:0 L:24:10"**

EXIT STATUS

tless returns a zero exit status. It restores the terminal on exit.

AUTHOR

Ken Halpern

LICENSE

tless may be used under the terms of the following BSD license.

Copyright (c) 2011, Ken Halpern

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the author nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

SEE ALSO

less(1), *troff*(1)