

PTP Cheat Sheet

SYNOPSIS

Install with:

```
sudo apt-get install perl cpanminus perl-doc build-essential
# or
sudo yum install perl Perl-App-cpanminus perl-doc gcc make

sudo cpanm App::PTP -n -L /usr/local --man-pages --install-args \
'DESTINSTALLBIN=/usr/local/bin'
```

Run with:

```
ptp file1 file2 ... [--grep re] [--substitute re subst] ... [-o out]
```

INPUT FILES

Input files can appear anywhere on the command line and are processed in the order in which they are given.

filename (anywhere in the command line, not starting with a -)
- (reads from stdin)
-- *filename* ... (for any filename)

PIPELINE COMMANDS

Pipeline commands are applied, in order, to all the input files.

-g *regex* (**--grep**), **-s** *regex string* (**--substitute**)
-p *code* (**-perl**): read and write `$_` to modify the file
-n *code*: read from `$_`, write the return values
-f *code* (**--filter**): return *true* to keep the line
-e *code* (**--execute**): execute once per input file
-l *path* (**--load**): execute the given file, once per input file
-M *module*: load the given module
--sort, **--ns** (**--numeric-sort**), **--ls** (**--locale-sort**), **--ss** (**--semver-sort**), **--cs** *code* (**--custom-sort**)
-u (**--unique**), **--gu** (**--global-unique**)
--head [*n*], **--tail** [*n*], **--reverse** (**--tac**), **--shuffle**
--eat: discard the content of the file
--ml *code* (**--mark-line**): set the marker for the line with the return value
--clear-markers, **--set-all-markers**
--delete-marked, **--delete-before**, **--delete-after**, **--delete-at-offset** *offset*
--insert-before *string*, **--insert-after** *string*, **--insert-at-offset** *offset string*: insert interpolated text next to marked lines (offset 0 is just after)
--cut *N,N,...*: select fields according to **-F** and concatenate them with **-P**
--paste *file*: paste with **-P** line by line with the current content
--pivot: turn the file into a single line with **-P**
--anti-pivot: split all lines according to **-F**
--transpose: transpose lines and columns using **-F** and **-P**
--nl (**--number-lines**), **--pfn** (**--prefix-file-name**)
--fn (**--file-name**), **--lc**, **--line-count**: replace the content of the file
-m (**--merge**): merge all the files in a single one
--tee *filename*: duplicate the output
--shell *command*: sends the content as input to the command
--xargs *command*: execute the command (which can contain { }) for each line.
--push, **--pop**: push and pop the entire content of the current file.

PROGRAM BEHAVIOR

Global option for the program execution.

-o *output_file* (**--output**), **-a** *output_file* (**--append**), **-i** (**--in-place**): by default output to standard output
-R, **--recursive**, **--input-filter** *code*: expand directories, optionally filter input files
--input-encoding *encoding*, **--output-encoding** *encoding*: default is UTF-8
--input-separator *separator*, **--output-separator** *separator*: default is `\n`
--eol (**--preserve-input-separator**), **--fix-final-separator**
-0: set **--input-separator** to NUL and **--output-separator** to the empty string
--00: set **--output-separator** to NUL, useful with `xargs -0`
-h (**--help**), **--cheat**, **--helpshort**, **--version**: remember to have **perldoc** installed
-d (**--debug**), **--abort**
--preserve-perl-env: keep environment across files
--safe [*n*]: default is 0, strictest is 2

PIPELINE MODES

Options for the pipeline commands coming after them. Most modes have a reverse mode to return to the default.

-I (**--case-insensitive**), **-S** (**--case-sensitive**): mode for any *regex* argument, **-S** is the default
-Q (**--quote-regex**), **-E** (**--end-quote-regex**): disable interpolation in any *regex*, *string*, *filename* or *command* argument, **-E** is the default
-V (**--inverse-match**), **-N** (**--normal-match**): inverse behavior of **--grep** and **--filter**
-L (**--local-match**), **-G** (**--global-match**): apply **--substitute** once per line or as much as possible (this is the default)
-C *code* (**--comparator**): for **--sort**, default is `$a cmp $b`
-F *regex* (**--input-field-spec**): how to split fields, default is `\s*,\s*/t`
-P *string* (**--output-field-spec**): how to paste fields, default is a tab
--default, **--bytes**, **--csv**, **--tsv**, **--none**: set the **-F** and **-P** flags
--sq *string* (**--single-quote-replacement**), **--dq** *string* (**--double-quote-replacement**), **--ds** *string* (**--dollar-sigil-replacement**): replace the given character or string by ' ', ", or \$ in all *code* arguments
--re engine, **--regex-engine**: use the specified regex engine (e.g. *RE2*, *PCRE*, *TRE*, *GNU*, etc.) if installed
-X (**--fatal-error**), **--ignore-error**: die on error in **--perl**, **-n** and **--filter**

PERL ENVIRONMENT

Variables and functions available to *code* arguments as well as *regex*, *string*, *filename*, and *command* ones (unless **-Q** has been passed).

Dumper, **fileparse**, **basename**, **dirname**, **copy**, **move**, **make_path**, **remove_tree**, **canonpath**, **any**, **all**, **none**, **min**, **max**, **sum**, **pi**, **acos**, **expand**, **unexpand**, **\$tabstop**, **wrap**, **fill**, **\$columns**, ...

\$_: current line content (*RW*)

\$f, **\$F**: current file name, current absolute file name (*RO*)

\$n: current line number (same as standard **\$.**) (*RO*)

\$N: number of lines in the current file (*RO*)

\$m: marker of the current line (*RW*)

@m: markers for all the lines, current line is at index 0 (*RW*)

\$!: 1-based index of the file being processed (*RW*)

ss *start*[, *len*][, *\$var*]: like `substr` but returns '' instead of undef.

pf *format*[, *args...*]: like `$_ = sprintf format, arg...`

spf *format*[, *args...*]: like `sprintf`

AUTHOR AND LICENCE

Copyright 2019-2024 Mathias Kende (<mailto:mathias@cpan.org>).

This program is distributed under the MIT (X11) License: <http://www.opensource.org/licenses/mit-license.php>

See more in the full documentation at <https://metacpan.org/pod/ptp>.