

VIM-PLUG-IN  
**perl-support.vim**  
 VERSION 5.4  
**HOT KEYS**

Key mappings for Vim and gVim

<http://www.vim.org> — Wolfgang Mehner, wolfgang-mehner@web.de

(i) insert mode, (n) normal mode, (v) visual mode

<i>Perl</i>		
<b>\ft</b>	file tests	(n,i)
<i>Help</i>		
<b>\he</b>	English dictionary	(n,i)
<b>\h</b>	read perldoc for word under cursor	(n,i)
<b>\hp</b>	help (plug-in)	(n,i)
<i>Comments</i>		
<b>[n]\cl</b>	end-of-line comment	(n, v, i)
<b>[n]\cj</b>	adjust end-of-line comments	(n, v, i)
<b>\cs</b>	set end-of-line comment col.	(n)
<b>[n]\cc</b>	code ↔ comment	(n, v)
<b>\cb</b>	code block → comment	(n, v)
<b>\cub</b>	uncomment code block	(n)
<b>\cfr</b>	frame comment	(n, i)
<b>\cfu</b>	function description	(n, i)
<b>\cme</b>	method description	(n, i)
<b>\chpl</b>	file header (.pl)	(n)
<b>\chpm</b>	file header (.pm)	(n)
<b>\cht</b>	file header (.t)	(n)
<b>\chpo</b>	file header (.pod)	(n)
<b>\cd</b>	date	(n, i)
<b>\ct</b>	date & time	(n, i)
<b>\ck</b>	keyword comments	(n, i)
<b>\cma</b>	plug-in macros	(n, i)

<i>Statements</i>		
<b>\sd</b>	do { } while	(n, v, i)
<b>\sf</b>	for { }	(n, v, i)
<b>\sfe</b>	foreach { }	(n, v, i)
<b>\si</b>	if { }	(n, v, i)
<b>\sie</b>	if { } else { }	(n, v, i)
<b>\se</b>	else { }	(n, v, i)
<b>\sei</b>	elsif { }	(n, v, i)
<b>\su</b>	unless { }	(n, v, i)
<b>\sue</b>	unless { } else { }	(n, v, i)
<b>\st</b>	until { }	(n, v, i)
<b>\sw</b>	while { }	(n, v, i)
<b>\sg</b>	given { }	(n, v, i)
<b>\swh</b>	when { }	(n, v, i)
<i>Idioms</i>		
<b>\id</b>	my \$;	(n, i)
<b>\ida</b>	my \$ = ;	(n, i)
<b>\idd</b>	my ( \$, \$ );	(n, i)
<b>\ia</b>	my @;	(n, i)
<b>\iaa</b>	my @ = (, ,);	(n, i)
<b>\ih</b>	my %;	(n, i)
<b>\iha</b>	my % = (=>, =>, );	(n, i)
<b>\ir</b>	my \$rgx_ = q//;	(n, i)
<b>\im</b>	\$ =~ m//xm	(n, i)
<b>\is</b>	\$ =~ s//xm	(n, i)
<b>\it</b>	\$ =~ tr//xm	(n, i)
<b>\isu</b>	subroutine	(n, v, i)
<b>\ip</b>	print "... \n";	(n, i)
<b>\ii</b>	open input file	(n, v, i)
<b>\io</b>	open output file	(n, v, i)
<b>\ipi</b>	open pipe	(n, v, i)
<i>Special Variables</i>		
<b>\vb</b>	basics	(n, i)
<b>\ve</b>	errors	(n, i)
<b>\vf</b>	files	(n, i)
<b>\vid</b>	IDs	(n, i)
<b>\vio</b>	IO	(n, i)
<b>\vr</b>	regexp	(n, i)
<b>\vps</b>	POSIX signals	(n, i)
<b>\vue</b>	use English	(n, i)

<i>Regular Expressions</i>		
<b>\xpc</b>	POSIX classes	(n, i)
<b>\xup</b>	Unicode properties	(n, i)
<b>\xex</b>	extended Regex	(n, i)
<b>\xms</b>	metasymbols	(n, i)
<b>[n]\xr</b>	pick up Regex	(n, v)
<b>[n]\xs</b>	pick up string	(n, v)
<b>\xf</b>	pick up flag(s)	(n, v)
<b>\xm</b>	match	(n)
<b>\xmm</b>	match multiple (Regex/target)	(n)
<b>\xe</b>	explain Regex	(n, v)
<i>POD</i>		
<b>\ppc</b>	pod, cut	(n, i)
<b>\pfc</b>	for, cut	(n, i)
<b>\pbh</b>	begin html, end	(n, i)
<b>\pbm</b>	begin man, end	(n, i)
<b>\pbt</b>	begin text, end	(n, i)
<b>\ph1</b>	head1	(n, i)
<b>\ph2</b>	head2	(n, i)
<b>\ph3</b>	head3	(n, i)
<b>\pob</b>	over, back	(n, i)
<b>\pi</b>	item	(n, i)
<b>\pod</b>	run podchecker	(n, i)
<b>\podh</b>	convert POD data to .html file	(n, i)
<b>\podm</b>	Convert POD data to *roff input	(n, i)
<b>\podt</b>	Convert POD data to ASCII text	(n, i)
<b>\pm</b>	markup sequences	(n, i)
<i>Snippet</i>		
<b>\nr</b>	read code snippet	(n, i)
<b>\nv</b>	view code snippet	(n, i)
<b>\nw</b>	write code snippet	(n, v, i)
<b>\ne</b>	edit code snippet	(n, i)
<b>\ntl</b>	edit local templates	(n, i)
<b>\ntc</b>	edit custom templates	(n, i)
<b>\ntp</b>	edit personal templates	(n, i)
<b>\ntr</b>	reread the templates	(n, i)
<b>\ntw</b>	template setup wizard	(n, i)
<b>\nts</b>	choose template style	(n, i)
<b>\njt</b>	insert jump tag	(n, i)
<b>\nxs</b>	regex snippet template	(n, i)

<i>Profiling</i>		
<code>\rps</code>	run <b>SmallProf</b>	(n, i)
<code>\rpss</code>	sort <b>SmallProf</b> report	(n, i)
<code>\rpso</code>	open existing <b>SmallProf</b> results	(n, i)
<code>\rpf</code>	run <b>FastProf</b>	(n, i)
<code>\rpfs</code>	sort <b>FastProf</b> report	(n, i)
<code>\rpfo</code>	open existing <b>FastProf</b> results	(n, i)
<code>\rpn</code>	run <b>NYTProf</b>	(n, i)
<code>\rpns</code>	sort <b>NYTProf</b> report	(n, i)
<code>\rpno</code>	open existing <b>NYTProf</b> results	(n, i)
<code>\rpnh</code>	browse HTML files ( <b>NYTProf</b> )	(n, i)
<i>Run</i>		
<code>\rr</code>	update file, run script	(n, i)
<code>\rs</code>	update file, check syntax	(n, i)
<code>\ra</code>	set command line arguments	(n, i)
<code>\rw</code>	set Perl cmd. line switches	(n, i)
<code>\re</code>	make script executable/not exec.	(n, i)
<code>\rd</code>	start debugger	(n, i)
<code>\ri</code>	show installed Perl modules	(n, i)
<code>\rg</code>	generate Perl module list	(n, i)
<code>\ry</code>	run <b>perltidy</b>	(n, v, i)
<code>\rpc</code>	run <b>perlcritic</b>	(n, i)
<code>\rpcs</code>	set <b>perlcritic</b> severity	(n, i)
<code>\rpcv</code>	set <b>perlcritic</b> verbosity	(n, i)
<code>\rpco</code>	set <b>perlcritic</b> options	(n, i)
<code>\rt</code>	save buffer with timestamp	(n, i)
<code>\rh</code>	hardcopy buffer	(n, v, i)
<code>\rk</code>	settings and hotkeys	(n, i)
<code>\rx</code>	set xterm size	(n, i <small>GUI only</small> )
<code>\ro</code>	change output destination	(n, i)
<i>Tool Box</i>		
<code>\rm</code>	run <b>make</b>	(n, i)
<code>\rcm</code>	choose makefile	(n, i)
<code>\rmc</code>	<b>make clean</b>	(n, i)
<code>\rma</code>	command line arguments for <b>make</b>	(n, i)

## Run

Specify command line arguments for the script in the current buffer. Use tab expansion to choose a file or a directory.

`:PerlScriptArguments`

Specify command line switches for the Perl interpreter.

`:PerlSwitches`

## Perlcritic

Ex commands for **perlcritic** (version 1.01+)

Use tab expansion to choose the severity or the verbosity.

`:CriticSeverity` 1 2 3 4 5  
brutal cruel harsh stern gentle

`:CriticVerbosity` 1 ... 11

`:CriticOptions` option(s), see **perlcritic(1)**

## Regular Expression Tester

Ex command for the regular expression tester. Set control character replacements for newline and tabulator used to display the results of a match, e.g.:

`:RegexSubstitutions` '\$~'

## Profiling

The following ex commands can be used to sort a profiler report in the quickfix window.

Use tab expansion to choose the sort criterion or the file name.

For `Devel::SmallProf`

`:SmallProfSort` file-name|line-number|line-count|time|ctime

For `Devel::FastProf`

`:FastProfSort` file-name|line-number|time|line-count

For `Devel::NYTProf`

`:NYTProfCSV` Read a CSV-file.

`:NYTProfHTML` Read the HTML-reports with an external viewer (GUI only).

`:NYTProfSort` file-name|line-number|time|calls|time-call