

VIM-PLUG-IN
vim-support.vim
 VERSION 2.4
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,
 (T) tab-completion

		<i>Help</i>
<code>\he</code>	English dictionary	(n,i)
<code>\hk</code> <code><S-F1></code>	help (Vim functions)	(n,i)
<code>\hp</code>	help (vim-support)	(n,i)
		<i>Comments</i>
<code>[n]\cl</code>	end-of-line comment	(n, v, i)
<code>[n]\cj</code>	adjust end-of-line comments	(n, v, i)
<code>\cs</code>	set end-of-line comment col.	(n)
<code>[n]\cc</code>	comment code	(n, i, v)
<code>[n]\co</code>	uncomment code	(n, i, v)
<code>\ca</code>	function description (auto)	(n, i, v)
<code>\cfr</code>	frame comment	(n, i)
<code>\cfu</code>	function description	(n, i)
<code>\ch</code>	file description	(n, i)
<code>\cd</code>	date	(n, i)
<code>\ct</code>	date & time	(n, i)
<code>\ck</code>	keyword comments	(n, i, T)
<code>\cma</code>	plugin macros	(n, i, T)

		<i>Statements</i>
<code>\sv</code>	let variable	(n, i)
<code>\sl</code>	let list	(n, i)
<code>\sd</code>	let dictionary	(n, i)
<code>\sf</code>	for	(n, v, i)
<code>\sif</code>	if...endif	(n, v, i)
<code>\sie</code>	if...else...endif	(n, v, i)
<code>\sei</code>	elseif	(n, i)
<code>\sel</code>	else	(n, i)
<code>\sw</code>	while	(n, v, i)
<code>\st</code>	try...catch	(n, v, i)
		<i>Idioms</i>
<code>\ib</code>	builtin functions	(n, i, T)
<code>\ii</code>	iterators	(n, v, i, T)
<code>\if</code>	function	(n, v, i)
		<i>Regular Expressions</i>
<code>\xc</code>	capture	(n, i, v)
<code>\xbc</code>	branch	(n, i, v)
<code>\xbn</code>	branch, no capture	(n, i, v)
<code>\xw</code>	word	(n, i)
<code>\xcc</code>	character classes	(n, i, T)
<code>\xs</code>	switches	(n, i, T)
		<i>Perl</i>
<code>\ps</code>	Perl snippet	(n, i)
<code>\pd</code>	<code>Vim::DoCommand()</code>	(n, i)
<code>\pe</code>	<code>Vim::Eval()</code>	(n, i)
<code>\pm</code>	<code>Vim::Msg("")</code>	(n, i)
<code>\pmc</code>	<code>Vim::Msg("", "Comment")</code>	(n, i)
<code>\pme</code>	<code>Vim::Msg("", "Warning")</code>	(n, i)
<code>\pmw</code>	<code>Vim::Msg("", "ErrorMsg")</code>	(n, i)

		<i>Documentation</i>
<code>\dcc</code>	table-of-contents, chapter	(n, i)
<code>\dcs</code>	table-of-contents, section	(n, i)
<code>\dcu</code>	table-of-contents, subsection	(n, i)
<code>\dtc</code>	text, chapter	(n, i)
<code>\dts</code>	text, section	(n, i)
<code>\dtu</code>	text, subsection	(n, i)
<code>\df</code>	function description	(n, i)
<code>\de</code>	example	(n, v, i)
		<i>Snippet</i>
<code>\nr</code>	read code snippet	(n, i)
<code>\nw</code>	write code snippet	(n, v, i)
<code>\ne</code>	edit code snippet	(n, i)
<code>\ntl</code>	edit local templates	(n, i)
<code>\ntc</code>	edit custom templates	(n, i)
<code>\ntp</code>	edit personal templates	(n, i)
<code>\ntr</code>	reread the templates	(n, i)
<code>\ntw</code>	template setup wizard	(n, i)
<code>\nts</code>	choose template style	(n, i, T)
		<i>Run</i>
<code>\rh</code>	hardcopy buffer to postscript	(n, v, i)
<code>\rs</code>	plugin settings	(n,i)