VIM-PLUG-IN

bash-support.vim

VERSION 4.2.1

HOT KEYS

Key mappings for Vim and gVim.
Plug-in: http://vim.sourceforge.net
Fritz Mehner (mehner.fritz@fh-swf.de)
(i) insert mode, (n) normal mode, (v) visual mode

		Bash
\bps	**p**arameter **s**ubstitution (list)	(n, i)
\bsp	**s**pecial **p**arameters (list)	(n, i)
\ben	**en**vironment (list)	(n, i)
\bbu	**bu**iltins (list)	(n, i)
\bse	**se**t options (list)	(n, i)
\bso	**s**h**o**pts (list)	(n, i)

		Comments
[n]\cl	end-of-line comment	(n, i, v)
[n]\cj	adjust end-of-line comments	(n, i, v)
\cs	set end-of-line comment col.	(n)
[n]\cc	code → comment	(n, i, v)
[n]\cu	uncomment code	(n, i, v)
\cfr	frame comment	(n, i)
\cfu	function description	(n, i)
\ch	file header	(n, i)
\cd	date	(n, i)
\ct	date & time	(n, i)
\css	script sections	(n, i)
\ckc	keyword comments	(n, i)
\cma	plug-in macros	(n, i)
\ce	echo "*actual line*"	(n, i)
\cr	remove echo from actual line	(n, i)

		Statements
\sc	case in ... esac	(n, i)
\sei	elif then	(n, i)
\sf	for in do done	(n, i, v)
\sfo	for ((...)) do done	(n, i, v)
\si	if then fi	(n, i, v)
\sie	if then else fi	(n, i, v)
\ss	select in do done	(n, i, v)
\su	until do done	(n, i, v)
\sw	while do done	(n, i, v)
\sfu	function	(n, i, v)
\se	echo -e ""	(n, i, v)
\sp	printf "%s"	(n, i, v)
\sae	array element ${.[.]}	(n, i, v)
\saa	arr. elem.s (all) ${.[@]}	(n, i, v)
\sas	arr. elem.s (1 word) ${.[*]}	(n, i, v)
\ssa	subarray ${.[@]::}	(n, i, v)
\san	no. of arr. elem.s ${.[@]}	(n, i, v)
\sai	list of indices ${.[*]}	(n, i, v)

		Tests
\ta	arithmetic tests	(n, i)
\tfp	file permissions	(n, i)
\tft	file types	(n, i)
\tfc	file characteristics	(n, i)
\ts	string comparisons	(n, i)
\toe	option is enabled	(n, i)
\tvs	variables has been set	(n, i)
\tfd	file descr. refers to a terminal	(n, i)
\tm	string matches regexp	(n, i)

		IO-Redirection
\ior	IO-redirections (list)	(n, i)
\ioh	here-document	(n, i)

		Pattern Matching
pzo	zero or one, ?(|)	(n, i)
pzm	zero or more, *(|)	(n, i)
pom	one or more, +(|)	(n, i)
peo	exactly one, @(|)	(n, i)
pae	anything except, !(|)	(n, i)
ppc	POSIX classes	(n, i)
pbr	${BASH_REMATCH[0...3]}	(n, i)

		Snippets
\nr	read code snippet	(n, i)
\nv	view code snippet	(n, i)
\nw	write code snippet	(n, i, v)
\ne	edit code snippet	(n, i)
\ntl	edit templates	(n, i)
\ntr	reread templates	(n, i)
\nts	choose style	(n, i)

		Run
\rr	update file, run script	(n, i, v[1])
\ra	set script cmd. line arguments	(n, i)
\rba	set Bash cmd. line arguments	(n, i)
\rc	update file, check syntax	(n, i)
\rco	syntax check options	(n, i)
\rd	start debugger[1]	(n, i)
\re	make script executable/not exec.[1]	(n, i)
\rh	hardcopy buffer	(n, i, v)
\rs	plug-in settings	(n, i)
\rx	set xterm size[1,2]	(n, i)
\ro	change output destination	(n, i)

		Help
\hb	display the Bash manual	(n,i)
\hh	help (Bash builtins)	(n,i)
\hm	show manual (cmd. line utilities)	(n,i)
\hp	help (plug-in)	(n,i)

[1] Linux/U**x only
[2] GUI only