

quickref.txt Per Vim version 7.3. Ultima modifica: 2013 Jun 29

VIM REFERENCE MANUAL di Bram Moolenaar
Traduzione di questo testo: Antonio Colombo

Breve guida di riferimento

quickref *Contents*

tag	argomento	tag	argomento
Q_ct	Lista file di help	Q_re	Ripetizione comandi
Q_lr	Movimenti: Sinistra/destra	Q_km	Mappatura tasti
Q_ud	Movimenti: Su/giù	Q_ab	Abbreviazioni
Q_tm	Movimenti: Oggetti nel testo	Q_op	Opzioni
Q_pa	Movimenti: Ricerca espressioni	Q_ur	Comandi Undo/Redo
Q_ma	Movimenti: Marcatori	Q_et	Comandi esterni
Q_vm	Movimenti: Vari	Q_qf	Comandi quickfix
Q_ta	Movimenti: Usare tag	Q_vc	Comandi vari
Q_sc	Scorrimento	Q_ce	Ex: Edita su riga-comando
Q_in	Inserimenti: Inserimento testo	Q_ra	Ex: Intervalli
Q_ai	Inserimenti: Tasti	Q_ex	Ex: Caratteri speciali
Q_ss	Inserimenti: Tasti speciali	Q_st	Invocare Vim
Q_di	Inserimenti: Digrafi	Q_ed	Modificare un file
Q_si	Inserimenti: Inserim. speciali	Q_fl	Uso della lista argomenti
Q_de	Modifiche: Cancellazione testo	Q_wq	Salvare e uscire
Q_cm	Modifiche: Copiare e spostare	Q_ac	Comandi automatici
Q_ch	Modifiche: Modificare testo	Q_wi	Comandi multi-finestra
Q_co	Modifiche: Complesse	Q_bu	Comandi lista buffer
Q_vi	Modo Visual	Q_sy	Evidenziazione sintattica
Q_to	Oggetti di testo	Q_gu	Comandi GUI
		Q_fo	Piegature

N indica un contatore opzionale che si può inserire prima del comando.

[Q_lr](#) Movimenti: Sinistra/destra

h	N	h	sinistra (anche: CTRL-H , <BS> , o tasto <Left>)
l	N	l	destra (anche: <Space> o tasto <Right>)
0		0	al primo carattere della riga (anche: tasto <Home>)
^		^	al primo carattere non-bianco della riga
\$	N	\$	all'ultimo carattere della riga (N-1 righe oltre) (anche: tasto <End>)
 g0 		g0	al primo carattere della riga visibile sullo schermo (diverso da "0" se la riga di testo non sta interamente su una riga dello schermo)
 g^ 		g^	al primo carattere non-bianco della riga visibile sullo schermo (diverso da "^" se la riga di testo non sta interamente su una riga dello schermo)
 g\$ 	N	g\$	all'ultimo carattere della riga visibile sullo schermo (diverso da "\$" se la riga di testo non sta interamente su una riga dello schermo)
 gm 		gm	alla metà della riga sullo schermo
 bar 	N		alla colonna N (default: 1)
f	N	f{carat.}	alla N ma ricorrenza di {carat.} alla destra
F	N	F{carat.}	alla N ma ricorrenza di {carat.} alla sinistra
t	N	t{carat.}	prima della N ma ricorrenza del {carat.}, a destra
T	N	T{carat.}	prima della N ma ricorrenza del {carat.}, a sinistra
;	N	;	ripetere ultimo "f", "F", "t", o "T" N volte
,	N	,	ripetere ultimo "f", "F", "t", o "T" N volte nella direzione opposta

[Q_ud](#) Movimenti: Su/giù

 k 	N	k	su N righe (anche: CTRL-P e <Up>)
 j 	N	j	giù N righe (anche: CTRL-J , CTRL-N , <NL> , e <Down>)
 - 	N	-	su N righe, sul primo carattere non-bianco
 + 	N	+	giù N righe, sul primo carattere non-bianco (anche: CTRL-M e <CR>)
 _ 	N	_	giù N-1 righe, sul primo carattere non-bianco
 G 	N	G	vai alla riga N (default: ultima riga), sul primo carattere non-bianco
 gg 	N	gg	vai alla riga N (default: prima riga), sul primo

carattere non-bianco			
N%	N	%	vai alla riga che si trova all' N % del file; N deve essere indicato, altrimenti questo diviene il comando %
gk	N	gk	su N righe di schermo (diverso da "k" per righe del testo più larghe dello schermo)
gj	N	gj	giù N righe di schermo (diverso da "j" per righe del testo più larghe dello schermo)
<hr/>			
Q_tm		Movimenti: Oggetti nel testo	
w	N	w	N parole avanti
W	N	W	N parole separate da spazio avanti WORD
e	N	e	avanti alla fine della N ma parola
E	N	E	avanti alla fine della N ma parola separata da spazio WORD
b	N	b	N parole indietro
B	N	B	N parole separate da spazio indietro WORD
ge	N	ge	indietro alla fine della N ma parola
gE	N	gE	indietro alla fine della N ma parola separata da spazio WORD
<hr/>			
)	N)	N frasi avanti
(N	(N frasi indietro
}	N	}	N paragrafi avanti
{	N	{	N paragrafi indietro
]	N]]	N sezioni avanti, a inizio sezione
[[N	[[N sezioni indietro, a inizio sezione
][N][N sezioni avanti, a fine sezione
[]	N	[]	N sezioni indietro, a fine sezione
[(N	[(N volte indietro a '(' non chiusa
[{	N	[{	N volte indietro a '{' non chiusa
[m	N	[m	N volte indietro a inizio metodo (per Java)
[M	N	[M	N volte indietro a fine metodo (per Java)
])	N])	N volte avanti a ')' non aperta
}]	N	}]	N volte avanti a '}' non aperta
]m	N]m	N volte avanti a inizio metodo (per Java)
]M	N]M	N volte avanti a fine metodo (per Java)
[#	N	[#	N volte indietro a "#if" o "#else" non chiuso
##	N]#	N volte avanti a "#else" o "#endif" non chiuso
[star	N	[*	N volte indietro a inizio commento "/*"
]star	N]*	N volte avanti a fine commento "*/"
<hr/>			
Q_pa		Movimenti: Ricerca espressioni [pattern]	
/	N	/ {espress.} [/ [offset]] <CR>	ricerca in avanti la N ma ricorrenza di {espress.}
?	N	? {espress.} [? [offset]] <CR>	ricerca indietro la N ma ricorrenza di {espress.}
/<CR>	N	/ <CR>	ripeti ultima ricerca, in avanti
?<CR>	N	? <CR>	ripeti ultima ricerca, all'indietro
n	N	n	ripeti ultima ricerca
N	N	N	ripeti ultima ricerca, nella direzione opposta
star	N	*	ricerca in avanti l'identificativo sotto il cursore
#	N	#	ricerca indietro l'identificativo sotto il cursore
gstar	N	g*	come "*", ma trova anche corrispondenze parziali
g#	N	g#	come "#", ma trova anche corrispondenze parziali
gd		gd	vai alla dichiarazione locale dell'identificativo sotto il cursore
gD		gD	vai alla dichiarazione globale dell'identificativo sotto il cursore
<hr/>			
espressione		Caratteri speciali in espressioni di ricerca [pattern]	
significato		magic	nomagic ~
qualsiasi carattere		.	\.
inizio riga		^	^
fine riga <EOL>		\$	\$
inizio di parola		\<	\<
fine di parola		\>	\>
ogni carattere nell'intervallo		[a-z]	\[a-z]
ogni carattere non nell'intervallo		[^a-z]	\[^a-z]
ogni carattere di un identificativo		\i	\i

come sopra ma non le cifre	\I	\I
ogni carattere di una parola chiave	\k	\k
come sopra ma non le cifre	\K	\K
ogni carattere di un nome di file	\f	\f
come sopra ma non le cifre	\F	\F
ogni carattere stampabile	\p	\p
come sopra ma non le cifre	\P	\P
uno spazio bianco	\s	\s
ogni carattere diverso da uno spazio bianco	\S	\S
Escape <Esc>	\e	\e
Tab <Tab>	\t	\t
A capo <CR>	\r	\r
BackSpace <BS>	\b	\b
0 o più occorrenze precedenti	*	*
1 o più occorrenze precedenti	\+	\+
0 o 1 occorrenze precedenti	\=	\=
fra 2 e 5 occorrenze precedenti	\{2,5}	\{2,5}
separa due alternative	\	\
raggruppa espressioni	\(\)	\(\)

| **search-offset** | Spostamenti permessi dopo comandi di ricerca

[num]	[num]	righe in giù, a colonna 1
+[num]	[num]	righe in giù, a colonna 1
-[num]	[num]	righe in su, a colonna 1
e[+num]	[num]	caratteri a destra della fine corrispondenza
e[-num]	[num]	caratteri a sinistra della fine corrispondenza
s[+num]	[num]	caratteri a destra dell'inizio corrispondenza
s[-num]	[num]	caratteri a sinistra dell'inizio corrispondenza
b[+num]	[num]	equivale a s[+num] qui sopra (mnemonico: begin)
b[-num]	[num]	equivale a s[-num] qui sopra (mnemonico: begin)
;{search-command} dopo, esegui {search-command} (comando di ricerca)		

Q_ma Movimenti: Marcature

m	m{a-zA-Z}	marca la posizione corrente con marcatore {a-zA-Z}
`a	`{a-z}	vai a mark {a-z} all'interno del file corrente
`A	`{A-Z}	vai a mark {A-Z} in qualsiasi file
`0	`{0-9}	vai alla precedente posizione di uscita da Vim
`"	`"	vai alla posizione prima dell'ultimo salto
`quote	`"	vai alla posizione dov'eri all'ultima modifica al file
`[`[vai all'inizio del testo appena immesso o modificato
`]	`]	vai alla fine del testo appena immesso o modificato
`<	`<	vai all'inizio della (precedente) area Visual
`>	`>	vai alla fine della (precedente) area Visual
`.	`.	vai alla posizione dell'ultima modifica fatta al file
`'	`{a-zA-Z0-9[]' "<>}.}	come `.` ma al primo carattere non-bianco della riga
:marks	:marks	visualizza i marcatori esistenti
CTRL-O	N CTRL-O	vai alla Nma posizione più vecchia nella lista salti
CTRL-I	N CTRL-I	vai alla Nma posizione più nuova nella lista salti
:ju	:ju[mps]	visualizza la lista salti

Q_vm Movimenti: Vari

%	%	trova prossima graffa, quadra, commento, o "#if"/ "#else"/"#endif" su questa riga e vai alla corrispondente apertura/chiusura
H	N H	vai alla Nma riga nella finestra, sul primo non-bianco
M	M	vai alla riga di mezzo della finestra, sul primo non-bianco
L	N L	vai alla Nma riga dal fondo, sul primo non-bianco
go	N go	vai a Nmo byte nel buffer
:go	: [interv.]go[to] [off]	vai a [off] byte nel buffer

Q_ta Movimenti: Usare tag

:ta	:ta[g][!] {tag}	salta a tag {tag}
:ta	: [contat.]ta[g][!]	salta a [contat.]'mo tag più nuova nella

		lista tag
CTRL-	CTRL-	salta al tag sotto il cursore, a meno che siano state fatte modifiche al file
:ts	:ts[elect][!] [tag]	lista tag corrispondenti e scegli a quale saltare
:tjump	:tj[ump][!] [tag]	salta a tag [tag] o scegli dalla lista se ce n'è più d'uno
:ltag	:lt[ag][!] [tag]	salta a tag [tag] e aggiungi il tag alla lista locazioni
:tags	:tags	visualizza lista tag
CTRL-T	N CTRL-T	torna indietro da Nmo tag più vecchio nella lista tag
:po	: [contat.]po[p][!]	torna indietro dal [contat.]'mo tag più vecchio della lista tag
:tnext	: [contat.]tn[ext][!]	salta alla [contat.]'mo prossimo tag corrispondente
:tp	: [contat.]tp[revious][!]	salta alla [contat.]'mo precedente tag corrispondente
:tr	: [contat.]tr[ewind][!]	salta alla [contat.]'mo tag corrispondente
:tl	:tl[ast][!]	salta a ultimo tag corrispondente
:ptag	:pt[ag] {tag}	apri una finestra di pre-visualizzazione per vedere {tag}
CTRL-W_	CTRL-W }	come CTRL- ma visualizza tag in una finestra di pre-visualizzazione
:pts	:pts[elect]	come ":tselect" ma visualizza tag in una finestra di pre-visualizzazione
:ptjump	:ptj[ump]	come ":tjump" ma visualizza tag in una finestra di pre-visualizzazione
:pclose	:pc[lose]	chiudere finestra pre-visualizzazione tag
CTRL-W_z	CTRL-W z	chiudere finestra pre-visualizzazione tag

Q_sc Scorrimento

CTRL-E	N CTRL-E	finestra N righe in giù (default: 1)
CTRL-D	N CTRL-D	finestra N righe in giù (default: 1/2 finestra)
CTRL-F	N CTRL-F	finestra N pagine avanti (in giù)
CTRL-Y	N CTRL-Y	finestra N righe in su (default: 1)
CTRL-U	N CTRL-U	finestra N righe in su (default: 1/2 finestra)
CTRL-B	N CTRL-B	finestra N pagine indietro (in su)
z<CR>	z<CR> o zt	ridisegna, riga corrente in cima a finestra
z.	z. o zz	ridisegna, riga corrente al centro di finestra
z-	z- o zb	ridisegna, riga corrente in fondo a finestra

I comandi seguenti sono attivi solo quando 'wrap' è a off:

zh	N zh	scorre schermo N caratteri a destra
zl	N zl	scorre schermo N caratteri a sinistra
zH	N zH	scorre schermo mezzo schermo a destra
zL	N zL	scorre schermo mezzo schermo a sinistra

Q_in Inserimenti: Inserimento testo

a	N a	aggiungi testo dopo il cursore (N volte)
A	N A	aggiungi testo a fine riga (N volte)
i	N i	inserisci testo prima del cursore (N volte) (anche: <Insert>)
I	N I	inserisci testo prima del primo carattere non-bianco della riga (N volte)
gI	N gI	inserisci testo a colonna 1 (N volte)
o	N o	apri una nuova riga sotto la riga corrente, aggiungi testo (N volte)
O	N O	apri una nuova riga sopra la riga corrente, aggiungi testo (N volte)
:startinsert	:star[tinsert][!]	entra in Modo Insert, aggiungi se si usa [!]
:startreplace	:startr[eplace][!]	entra in Modo Replace, a fine riga <EOL> se si usa [!]

In modo Visual Block:

v_b_I	I	inserisci lo stesso testo davanti a tutte le righe selezionate
v_b_A	A	aggiungi lo stesso testo dopo tutte le righe selezionate

Q_ai Inserimenti: Tasti

| **insert-index** | indice alfabetico dei comandi in Modo Insert

Uscire da Modo Insert:

i_<Esc>	<Esc>	uscita da Modo Insert, ritorno a Modo Normale
i_CTRL-C	CTRL-C	come <Esc> , senza controllare eventuali abbreviazioni
i_CTRL-O	CTRL-O {comando}	esegui {comando} e torna a Modo Insert

Movimenti:

i_<Up>	chiavi cursore	muovi cursore a sinistra/destra/su/giù
i_<S-Left>	shift-sin./dest.	una parola a sinistra/destra
i_<S-Up>	shift-su/giù	una schermata indietro/avanti
i_<End>	<End>	cursore dopo ultimo carattere della riga
i_<Home>	<Home>	cursore al primo carattere della riga

Q_ss Inserimenti: Inserimenti speciali

i_CTRL-V	CTRL-V {carat.}	.. inserisci un carattere scrivendolo letteralmente, o inserisci il valore del byte
i_<NL>	<NL> o <CR> o CTRL-M o CTRL-J	vai a capo (nuova riga)
i_CTRL-E	CTRL-E	inserisci il carattere che si trova nella riga sotto il cursore
i_CTRL-Y	CTRL-Y	inserisci il carattere che si trova nella riga sopra il cursore
i_CTRL-A	CTRL-A	inserisci il testo precedentemente inserito
i_CTRL-@	CTRL-@	inserisci il testo precedentemente inserito ed esci dal Modo Insert
i_CTRL-R	CTRL-R {0-9a-z%#:.-=}	inserisci il contenuto di un registro
i_CTRL-N	CTRL-N	inserisci la prossima corrispondenza di un identificativo prima del cursore
i_CTRL-P	CTRL-P	inserisci la precedente corrispondenza di un identificativo prima del cursore
i_CTRL-X	CTRL-X ...	completa la parola prima del cursore in vari modi
i_<BS>	<BS> o CTRL-H	cancella il carattere prima del cursore
i_		cancella il carattere sotto il cursore
i_CTRL-W	CTRL-W	cancella la parola prima del cursore
i_CTRL-U	CTRL-U	cancella tutti i caratteri inseriti nella riga corrente
i_CTRL-T	CTRL-T	inserisci UNA indentatura davanti alla riga corrente
i_CTRL-D	CTRL-D	togli UNA indentatura davanti alla riga corrente
i_0_CTRL-D	0 CTRL-D	cancella tutta l'indentatura della riga corrente
i_^_CTRL-D	^ CTRL-D	cancella tutta l'indentatura della riga e mettila nella riga seguente

Q_di Inserimenti: Digrafici

:dig	:dig[raphs]	mostra la lista di digrafici a disposisizione
:dig	:dig[raphs] {char1}{char2}{numero} ...	aggiungi digrafico/i alla lista

In modo Insert o modo Linea-Comando:

i_CTRL-K	CTRL-K {char1}{char2}	inserisci digrafico
i_digraph	{char1} <BS> {char2}	inserisci digrafico se l'opzione 'digraph' è impostata

Q_si Inserimenti: Inserimenti Speciali

:r	:r [file]	inserisci il contenuto del [file] il cui nome sta sotto il cursore
:r!	:r! {comando}	inserisci lo "standard output" del {comando} che sta sotto il cursore

Q_de Modifiche: Cancellazione testo

x	N	x	cancella N caratteri sotto e dopo il cursore
	N		cancella N caratteri sotto e dopo il cursore
X	N	X	cancella N caratteri prima del cursore
d	N	d{movimento}	cancella il testo individuato con {movimento}
v_d		{visual}d	cancella il testo evidenziato
dd	N	dd	cancella N righe
D	N	D	cancella fino alla fine della riga (e N -1 righe oltre)
J	N	J	unisci N -1 righe (cancella carattere fine riga <EOL>)
v_J		{visual}J	unisci le righe evidenziate
gJ	N	gJ	come "J", senza inserire spazi unendo le righe
v_gJ		{visual}gJ	come "{visual}J", senza inserire spazi unendo le righe
:d	:	[interv.]d [x]	cancella [interv.] righe [e le mette nel registro x]

Q_cm Modifiche: Copiare e spostare

quote		"{carat.}	usa il registro {carat.} per il prossimo cancella, yank (copia), o put (incolla)
:reg	:	reg	mostra il contenuto di tutti i registri
:reg	:	reg {arg}	mostra il contenuti dei registri elencati con {arg}
y	N	y{movim.}	copia (yank) il testo raggiunto con {movim.} in un registro
v_y		{visual}y	copia il testo evidenziato in un registro
yy	N	yy	copia N righe in un registro
Y	N	Y	copia N righe in un registro
p	N	p	incolla (put) il contenuto di un register dopo la posizione del cursore (N volte)
P	N	P	incolla il contenuto di un registro dopo la posizione del cursore (N volte)
p	N	p	come p, ma adatta indentazione alla riga corrente
[p	N	[p	come P, ma adatta indentazione alla riga corrente
gp	N	gp	come p, ma posiziona cursore dopo il nuovo testo
gP	N	gP	come P, ma posiziona cursore dopo il nuovo testo

Q_ch Modifiche: Modificare testo

r	N	r{carat.}	sostituire N caratteri con {carat.}
gr	N	gr{carat.}	sostituire N caratteri come appaiono sullo schermo [ad es. un singolo carattere <Tab> può essere sostituito da più di un carattere -NdT]
R	N	R	entra in Modo Replace (ripeti N volte il carattere immesso)
gR	N	gR	entra in Modo Virtual Replace: come Modo Replace ma senza modificare l'apparenza dello schermo [vedere NdT qui sopra]
v_b_r		{visual}r{carat.}	in Modo Visual Block: sostituisci tutti i caratteri del testo selezionato con {carat.}

(change = cancella testo ed entra in Modo Insert)

c	N	c{movim.}	modifica il testo coperto con {movim.}
v_c		{visual}c	modifica il testo evidenziato
cc	N	cc	modifica N righe
S	N	S	modifica N righe
C	N	C	modifica fino alla fine della riga (e N -1 righe oltre)
s	N	s	modifica N caratteri
v_b_c		{visual}c	in Modo Visual Block: modifica ognuna delle righe selezionate con il testo immesso
v_b_C		{visual}C	in Modo Visual Block: modifica ognuna delle righe selezionate fino a fine-riga <EOL> con il testo immesso

~	N	~	cambia maiuscolo in minuscolo e viceversa per N caratteri e sposta in avanti il cursore
v_~		{visual}~	cambia maiuscolo in minuscolo e viceversa nel testo evidenziato
v_u		{visual}u	cambia il testo evidenziato a minuscolo
v_U		{visual}U	cambia il testo evidenziato a maiuscolo
g~		g~{movim.}	cambia maiuscolo in minuscolo e viceversa nel testo coperto dal movimento {movim.}
gu		gu{movim.}	cambia a minuscolo il testo coperto dal movimento

gU		gU{movim.}	cambia a maiuscolo il testo coperto dal movimento {movim.}
v_g?		{visual}g?	cifra col metodo rot13 il testo evidenziato
g?		g?{movim.}	cifra col metodo rot13 il testo coperto dal movimento {movim.}
CTRL-A	N	CTRL-A	aggiungi N al numero sotto il cursore, o subito dopo
CTRL-X	N	CTRL-X	sottrai N dal numero sotto il cursore, o subito dopo
<	N	<{movim.}	sposta le righe coperte dal movimento {movim.} a sinistra di un livello di indentazione (shiftwidth)
<<	N	<<	sposta N righe a sinistra di un livello di indentazione
>	N	>{movim.}	sposta le righe coperte dal movimento {movim.} a destra di un livello di indentazione
>>	N	>>	sposta N righe a destra di un livello di indentazione
gq	N	gq{movim.}	riaggiusta le righe coperte dal movimento {movim.} alla larghezza specificata in 'textwidth'
:ce	:	[interv.]:ce[nter] [width]	centra le righe in [interv.] tra [width] colonne
:le	:	[interv.]:le[ft] [indent]	allinea a sinistra le righe in [interv.] (con [indent])
:ri	:	[interv.]:ri[ght] [width]	allinea a destra le righe in [interv.] tra [width] colonne

Q_co Modifiche: Complesse

!	N	!{movim.}{comando}<CR>	filtra (sostituisci) le righe coperte dal movimento con quelle che risultano dall'esecuzione di {comando} [che le prende in input]
	N	!!{comando}<CR>	filtra N righe attraverso {comando}
v_!		{visual}!{comando}<CR>	filtra le righe evidenziate attraverso {comando}
:range!	:	[interv.]:!{comando}<CR>	filtra [interv.] righe attraverso {comando}
=	N	={movim.}	filtra le righe coperte dal movimento attraverso 'equalprg'
==	N	==	filtra N righe attraverso 'equalprg'
v_=		{visual}=	filtra le righe evidenziate attraverso 'equalprg'
:s	:	[interv.]:s[ubstitute]/{espress.}/{stringa}/[g][c]	sostituisci {espress.} con {stringa} in [interv.] righe; con [g], sostituisci tutte le ricorrenze di {espress.}; con [c], chiedi conferma prima di ogni sostituzione
:s	:	[interv.]:s[ubstitute] [g][c]	ripeti il precedente ":s" con un nuovo intervallo e nuove opzioni
&		&	ripeti il precedente ":s" sulla riga corrente senza opzioni
:ret	:	[interv.]:ret[ab][!] [tabstop]	imposta 'tabstop' a un nuovo valore e ricalcola gli spazi bianchi in corrispondenza

Q_vi Modo Visual

visual-index		lista comandi del Modo Visual
v	v	inizia evidenziazione caratteri
V	V	inizia evidenziazione per riga
CTRL-V	CTRL-V	inizia evidenziazione a blocco
v_o	o	inverte posizione cursore portandolo a inizio evidenziazione
gv	gv	inizia evidenziazione sulla precedente area visuale

v_v	v	evidenzia caratteri o ferma evidenziazione
v_V	V	evidenzia per riga o ferma evidenziazione
v_CTRL-V	CTRL-V	evidenzia a blocchi o ferma evidenziazione
<hr/>		
Q_to	Oggetti di testo (solo in Modo Visual o dopo un operatore)	
v_aw	N aw	Seleziona "a word" (parola)
v_iw	N iw	Seleziona "inner word" (anche gli spazi contano per "una parola", in questo caso)
v_aW	N aW	Seleziona "a WORD " (PAROLA, comprende anche segni di interpunzione etc.)
v_iW	N iW	Seleziona "inner WORD " (parola e spazio)
v_as	N as	Seleziona "a sentence" (frase, termina con [.!?!])
v_is	N is	Seleziona "inner sentence" (frase e spazi)
v_ap	N ap	Seleziona "a paragraph" (paragrafo)
v_ip	N ip	Seleziona "inner paragraph" (paragrafo e spazi)
v_ab	N ab	Seleziona "a block" (blocco, da "[(" a ")")
v_ib	N ib	Seleziona "inner block" (blocco e spazi, da "[(" a ")")
v_aB	N aB	Seleziona "a Block" (Blocco, da "[{" a "]")
v_iB	N iB	Seleziona "inner Block" (Blocco e spazi, da "[{" a "]")
v_a>	N a>	Seleziona "a <> block" (blocco)
v_i>	N i>	Seleziona "inner <> block" (blocco e spazi)
v_at	N at	Seleziona "a tag block" (blocco di tag, da <aaa> a </aaa>)
v_it	N it	Seleziona "inner tag block" (blocco di tag e spazi, da <aaa> a </aaa>)
v_a'	N a'	Seleziona "una stringa fra apici singoli"
v_i'	N i'	Seleziona "una stringa fra apici singoli e spazi"
v_aquote	N a"	Seleziona "una stringa fra apici doppi"
v_iquote	N i"	Seleziona "una stringa fra apici doppi e spazi"
v_a`	N a`	Seleziona "una stringa fra apici all'indietro"
v_i`	N i`	Seleziona "una stringa fra apici all'indietro e spazi"
<hr/>		
Q_re	Ripetizione comandi	
.	N .	ripeti ultima modifica (con contatore sostituito da N)
q	q{ a-z }	registra caratteri immessi nel registro { a-z }
Q	Q{ A-Z }	registra caratteri immessi, aggiungendoli in fondo al registro { a-z }
q	q	termina la registrazione
@	N @{ a-z }	esegui i contenuti del registro { a-z } (N volte)
@@	N @@	ripeti il precedente comando @{ a-z } (N volte)
:@	:@{ a-z }	esegui i contenuti del registro { a-z } come comando Ex
:@@	:@@	ripeti il precedente comando :@{ a-z }
:g	: [interv.] g[lobal]/{ espress. }/[cmd]	esegui comando Ex [cmd] (default: ":p") nelle righe comprese nell'intervallo [interv.] per le quali è verificata una corrispondenza di { espress. }
:g	: [interv.] g[lobal]!/({ espress. })/[cmd]	esegui comando Ex [cmd] (default: ":p") nelle righe comprese nell'intervallo [interv.] per le quali NON è verificata una corrispondenza di { espress. }
:so	:so[urce] { file }	leggi ed esegui comandi Ex dal file { file }
:so	:so[urce]! { file }	leggi ed esegui comandi Vim dal file { file }
:sl	:sl[ee]p [sec]	non fare nulla per [sec] secondi
gs	N gs	go to sleep (vai a nanna) per N secondi
<hr/>		
Q_km	Mappatura tasti	

Nota: {**lhs**} Left Hand Side - parte sinistra dell'espressione
 {**rhs**} Right Hand Side - parte destra dell'espressione

|**:map**| :ma[p] {**lhs**} {**rhs**} mappa {**lhs**} a {**rhs**} in Modo Normal e Modo Visual

|**:map!**| :ma[p]! {**lhs**} {**rhs**} mappa {**lhs**} a {**rhs**} in Modo Insert e Modo

		Riga-comando
:noremap	:no[remap][!] {lhs} {rhs}	come ":map", ma {rhs} non è soggetto a ulteriori rimappature
:unmap	:unm[ap] {lhs}	annulla mappatura di {lhs} in Modo Normal e Modo Visual
:unmap!	:unm[ap]! {lhs}	annulla mappatura di {lhs} in Modo Insert e Modo Riga-comando
:map_l	:ma[p] [lhs]	lista mappature (che iniziano con [lhs]) per Modo Normal e Modo Visual
:map_l!	:ma[p]! [lhs]	lista mappature (che iniziano con [lhs]) per Modo Insert e Modo Riga-comando
:cmap	:cmap/:cunmap/:cnoremap	come ":map!"/":unmap!"/":noremap!" ma solo per Modo Riga-comando
:imap	:imap/:iunmap/:inoremap	come ":map!"/":unmap!"/":noremap!" ma solo per Modo Insert
:nmap	:nmap/:nunmap/:nnoremap	come ":map"/":unmap"/":noremap" ma solo per Modo Normale
:vmap	:vmap/:vunmap/:vnoremap	come ":map"/":unmap"/":noremap" ma solo per Modo Visual
:omap	:omap/:ounmap/:onoremap	come ":map"/":unmap"/":noremap" ma solo quando non si è ancora specificato su quale testo deve agire un operatore
:mapc	:mapc[lear]	rimuovi mappature per Modo Normal e Visual
:mapc	:mapc[lear]!	rimuovi mappature per Modo Insert e Modo Riga-comando
:imapc	:imapc[lear]	rimuovi mappature per Modo Insert
:vmapc	:vmapc[lear]	rimuovi mappature per Modo Visual
:omapc	:omapc[lear]	rimuovi mappature per MOdo Operator-pending
:nmapc	:nmapc[lear]	rimuovi mappature per Modo Normale
:cmapc	:cmapc[lear]	rimuovi mappature per Modo Riga-comando
:mkexrc	:mk[exrc][!] [file]	scrivi mappature correnti, abbreviazioni e impostazioni a [file] (default: ".exrc"; usare ! per sovrascriverlo)
:mkvimrc	:mkv[imrc][!] [file]	come ":mkexrc", ma con default ".vimrc"
:mksession	:mks[ession][!] [file]	come ":mkvimrc", ma scrivi anche le informazioni relative ai file correntemente in modifica, alle finestre, etc. per essere in grado di continuare questa sessione di modifica più tardi

Q_ab Abbreviazioni

Nota: {lhs} Left Hand Side - parte sinistra dell'espressione
 {rhs} Right Hand Side - parte destra dell'espressione

:abbreviate	:ab[breviate] {lhs} {rhs}	aggiungi abbreviazione per {lhs} a {rhs}
:abbreviate	:ab[breviate] {lhs}	lista abbreviazioni che iniziano con {lhs}
:abbreviate	:ab[breviate]	lista tutte le abbreviazioni
:unabbreviate	:una[breviate] {lhs}	rimuovi abbreviazione per {lhs}
:noreabbrev	:norea[bbrev] [lhs] [rhs]	come ":ab", ma non rimappare [rhs]
:iabbrev	:iab/:iunab/:inoreab	come ":ab", ma solo in Modo Insert
:cabbrev	:cab/:cunab/:cnoreab	come ":ab", ma solo in Modo Riga-comando
:abclear	:abc[lear]	rimuove tutte le abbreviazioni
:cabclear	:cabc[lear]	rimuove tutte le abbreviazioni per il Modo Riga-comando
:iabclear	:iabc[lear]	rimuove tutte le abbreviazioni per il Modo Insert

Q_op Options

:set	:se[t]	lista tutte le opzioni modificate [diverse dal valore di default]
------	--------	---

:set	:set all	lista tutte le opzioni, tranne quelle relative alle proprietà del terminale (termcap)
:set	:set termcap	lista tutte le opzioni relative al al terminale (termcap)
:set	:set {opzione}	imposta un'opzione booleana (binaria), (attivandola - on), lista un'opzione di tipo stringa o numerica
:set	:set no{opzione}	annulla un'opzione booleana (binaria) (annullandola - off)
:set	:set inv{opzione}	inverti il valore di un'opzione booleana (binaria)
:set	:set {opzione}={valore}	imposta opzione di tipo stringa o numerica a {valore}
:set	:set {opzione}+={valore}	aggiungi {valore} a un'opzione di tipo stringa, aggiungi {valore} a un'opzione di tipo numerico
:set	:set {opzione}-={valore}	rimuovi {valore} da un'opzione di tipo stringa, sottrai {valore} da un'opzione di tipo numerico
:set	:set {opzione}?	visualizza valore di {opzione}
:set	:set {opzione}&	reimposta {opzione} al suo valore di default
:setlocal	:setl[ocal]	come ":set" ma imposta il valore locale per opzioni che ne abbiano uno
:setglobal	:setg[lobal]	come ":set" ma imposta il valore globale di un'opzione locale
:fix	:fix[del]	imposta valore di 't_kD' a seconda del valore di 't_kb'
:options	:opt[ions]	apri una nuova finestra per visualizzare e impostare opzioni, divise a seconda della loro funzione, con una riga di descrizione e collegamenti alla pagina di aiuto

Breve spiegazione di ciascuna opzione:

option-list

'aleph'	'al'	codice ASCII della lettera Aleph (Ebraico)
'allowrevins'	'ari'	consenti CTRL-__ nei Modi Insert e Riga-comando
'altkeymap'	'akm'	per secondo linguaggio di default (Farsi/Ebraico)
'ambiwidth'	'ambw'	gestione caratteri Unicode di larghezza ambigua
'antialias'	'anti'	Mac OS X: usa font nitidi, non confondibili
'autochdir'	'acd'	cambia a directory del file in finestra corrente
'arabic'	'arab'	per Arabo come secondo linguaggio di default
'arabicshape'	'arshape'	applica conformazione per caratteri Arabic
'autoindent'	'ai'	indenta nuova riga a seconda della precedente
'autoread'	'ar'	rileggi file in automatico, se cambiato fuori Vim
'autowrite'	'aw'	scrivi file in automatico, dopo alcune modifiche
'autowriteall'	'awa'	come 'autowrite', ma scatta più spesso
'background'	'bg'	"dark" o "light", usato per evidenziazione
'backspace'	'bs'	cosa fa tasto <backspace> a inizio riga
'backup'	'bk'	tieni file di backup dopo aver riscritto file
'backupcopy'	'bkc'	fai backup copiando, invece che rinominando file
'backupdir'	'bdir'	lista directory usabili per il file di backup
'backupext'	'bex'	estensione usata per il file di backup
'backupskip'	'bsk'	nessun backup per file con questi nomi
'balloondelay'	'bdlay'	ritardo in mS prima di visualizzare menù dinamico
'ballooneval'	'beval'	attiva valorizzazione riquadro (balloon)
'balloonexpr'	'bexpr'	espressione da visualizzare nel riquadro
'binary'	'bin'	leggi/scrivi/modifica file in modo binario
'bioskey'	'biosk'	MS-DOS: usa chiamate a BIOS per input caratteri
'bomb'		metti un Byte Order Mark a inizio file
'breakat'	'brk'	caratteri che possono far andare a capo
'browseidir'	'bsdir'	directory da cui cominciare esplorazione
'bufhidden'	'bh'	che fare se un buffer non è più visibile
'buflisted'	'bl'	se il buffer è incluso nella lista buffer
'buftype'	'bt'	tipo speciale di buffer

'casemap'	'cmp'	specifica cambio minuscolo/minuscolo
'cdpath'	'cd'	lista directory cercate con comando ":cd"
'cedit'		tasto usato per aprire finestra riga-comando
'charconvert'	'ccv'	espressione per conversione codifica caratteri
'cindent'	'cin'	indentazione per programmi C
'cinkeys'	'cink'	tasti di indentazione se 'cindent' impostato
'cinoptions'	'cino'	come indentare se 'cindent' impostato
'cinwords'	'cinw'	parole dove 'si' e 'cin' aggiungono indentazione
'clipboard'	'cb'	usa la clipboard come registro senza nome
'cmdheight'	'ch'	numero di righe da usare per la riga-comandi
'cmdwinheight'	'cwh'	altezza della finestra riga-comandi
'colorcolumn'	'cc'	colonne da evidenziare
'columns'	'co'	numero di colonne dello schermo
'comments'	'com'	stringhe che indicano inizio righe commento
'commentstring'	'cms'	modello per commenti; per marcare piegature
'compatible'	'cp'	comportamento il più possibile compatibile-Vi
'complete'	'cpt'	specifica uso completamento in Modo Insert
'completefunc'	'cfu'	funzione da usare per completamento in Modo Insert
'completeopt'	'cot'	opzioni per completamento in Modo Insert
'concealcursor'	'cocu'	se testo nascosto visibile sulla linea del cursore
'conceallevel'	'cole'	se il testo nascondibile è visibile o nascosto
'confirm'	'cf'	che fare con file non salvati/in sola lettura
'conscopy'	'consk'	ricevi tasti direttamente da console (solo MS-DOS)
'copyindent'	'ci'	'autoindent' usi struttura indentazione corrente
'cptions'	'cpo'	flag per comportamenti compatibili-Vi
'cryptmethod'	'cm'	tipo di cifratura da usare in scrittura file
'cscopepathcomp'	'cspc'	quanti componenti mostrare del nome-file intero
'cscopeprg'	'csprg'	comando per eseguire cscope
'cscopequickfix'	'csqf'	usa finestra quickfix per risultati cscope
'cscoperelative'	'csre'	usa nome-directory path cscope.out come prefisso
'cscopetag'	'cst'	usa cscope per tag comandi
'cscopetagorder'	'csto'	definisci ordine di ricerca per ":cstag"
'cscopeverbose'	'csverb'	dà messaggi per aggiunta database cscope
'cursorbind'	'crb'	muove cursore in sincronia fra tutte le finestre
'cursorcolumn'	'cuc'	evidenzia la colonna di schermo del cursore
'cursorline'	'cul'	evidenzia la riga di schermo del cursore
'debug'		impostare a "msg" per vedere ogni messaggio errore
'define'	'def'	espress. da usare per trovare definizione macro
'delcombine'	'deco'	cancella uno a uno ogni byte di carat. multibyte
'dictionary'	'dict'	lista di nomi-file da usare per completamenti
'diff'		usare Modo Diff per la finestra corrente
'diffexpr'	'dex'	espressione usata per ottenere un file Modo Diff
'diffopt'	'dip'	opzioni per usare Modo Diff
'digraph'	'dg'	abilita immissione digrafici in Modo Insert
'directory'	'dir'	lista di nomi directory per il file di swap
'display'	'dy'	lista di flag per modalità visualizzazione testo
'eadirection'	'ead'	direzione in cui agisce 'equalalways'
'edcompatible'	'ed'	attiva flag per il comando ":substitute"
'encoding'	'enc'	codifica usata internamente
'endofline'	'eol'	scrivi <EOL> all'ultima riga del file
'equalalways'	'ea'	finestre automaticamente di uguali dimensioni
'equalprg'	'ep'	programma esterno da usare per comando "="
'errorbells'	'eb'	emetti suono assieme a messaggi di errore
'errorfile'	'ef'	nome del file-errori per Modo QuickFix
'errorformat'	'efm'	descrizione righe nel file-errori
'esckey'	'ek'	riconoscimento tasti funzionali in Modo Insert
'eventignore'	'ei'	eventi da ignorare per autocomandi
'expandtab'	'et'	usa spazi quando si inserisce <Tab>
'exrc'	'ex'	leggi .vimrc e .exrc dalla directory corrente
'fileencoding'	'fenc'	codifica file per testo multi-byte
'fileencodings'	'fencs'	determina automaticamente codifiche caratteri
'fileformat'	'ff'	formato file usato per I/O
'fileformats'	'ffs'	determina automaticamente valori di 'fileformat'
'fileignorecase'	'fic'	ignora maiuscolo/minuscol nei nomi di file
'filetype'	'ft'	tipo-file, usato per autocomandi
'fillchars'	'fcs'	caratteri da usare per mostrare caratt. speciali
'fkmap'	'fk'	mappature tastiere Farsi
'foldclose'	'fcl'	chiudere una piegatura se il cursore la lascia
'foldcolumn'	'fdc'	larghezza colonna usata per indicare piegature
'foldenable'	'fen'	se impostato visualizza tutte le piegature
'foldexpr'	'fde'	espressione usata con 'foldmethod' "expr"
'foldignore'	'fdi'	ignora righe con 'foldmethod' "indent"
'foldlevel'	'fdl'	chiudere piegature a livello maggiore di questo

'foldlevelstart'	'fdls'	'foldlevel' all'inizio della modifica di un file
'foldmarker'	'fmr'	marcatori usati con 'foldmethod' "marker"
'foldmethod'	'fdm'	tipo piegatura
'foldminlines'	'fml'	minimo numero righe per chiudere piegatura
'foldnestmax'	'fdn'	massima profondità piegature
'foldopen'	'fdo'	per quali comandi una piegatura verrà aperta
'foldtext'	'fdt'	espress. visualizzata su una piegatura chiusa
'formatlistpat'	'flp'	espress. usata per riconoscere inizio di una lista
'formatoptions'	'fo'	come fare la formattazione automatica
'formatprg'	'fp'	nome programma esterno da usare col comando "gq"
'formatexpr'	'fex'	espressione usata col comando "gq"
'fsync'	'fs'	se invocare fsync() dopo aver scritto un file
'gdefault'	'gd'	flag 'g' di ":substitute" attiva per default
'grepformat'	'gfm'	formato output di 'grepprg'
'grepprg'	'gp'	programma da usare per ":grep"
'guicursor'	'gcr'	GUI: impostazioni forma e lampeggiamento cursore
'guifont'	'gfn'	GUI: nome(i) di font da usare
'guifontset'	'gfs'	GUI: nome(i) di font multi-byte da usare
'guifontwide'	'gfw'	lista nomi font per caratteri larghi
'guiheadroom'	'ghr'	GUI: numero pixel per decorazione finestra
'guioptions'	'go'	GUI: che componenti e opzioni sono usate
'guity'		GUI: prova a usare pseudo-tty per comandi "!!"
'guitablabel'	'gtl'	GUI: etichetta personalizzata per linguetta
'guitabtooltip'	'gtt'	GUI: suggerimento personalizzato per linguetta
'helpfile'	'hf'	nome completo del file di help principale
'helpheight'	'hh'	altezza minima di una nuova finestra di help
'helplang'	'hlg'	linguaggi preferiti per l'help
'hidden'	'hid'	non scaricare buffer abbandonato abandon
'highlight'	'hl'	imposta evidenziazione per varie situazioni
'hlsearch'	'hls'	evidenzia corrispondenze ultima espress. cercata
'history'	'hi'	numero di righe comando che sono memorizzate
'hkmmap'	'hk'	mappature tastiera Ebraica
'hkmapp'	'hkp'	mappature fonetiche tastiera Ebraica
'icon'		Vim imposta tesa finestra iconizzata
'iconstring'		stringa da usare per testo di icona Vim
'ignorecase'	'ic'	ignora maiuscolo/minuscolo nella ricerca espress.
'imactivatekey'	'imak'	tasto che attiva Metodo di Input (IM) X
'imactivatefunc'	'imaf'	funzione che abilita/disabilita Metodo di Input X
'imcmdline'	'imc'	usare Metodo di Input per editare riga-comando
'imdisable'	'imd'	non usare Metodo di Input in nessun caso
'iminsert'	'imi'	usare :lmap o Metodo di Input in Modo Insert
'imsearch'	'ims'	usare :lmap o Metodo di Input per espress. ricerca
'imstatusfunc'	'imsf'	funzione che ottiene status da Metodo di Input X
'include'	'inc'	espress. da usare per trovare file incluso
'includeexpr'	'inex'	espress. da usare per localizzare file include
'incsearch'	'is'	evidenzia corrispondenze durante ricerca espress.
'indentexpr'	'inde'	espressione usata per far indentare una riga
'indentkeys'	'indk'	tasti che provocano indentazione con 'indentexpr'
'infercase'	'inf'	adatta maiuscolo/minuscolo nei completamenti
'insertmode'	'im'	inizia a modificare un file in Modo Insert
'isfname'	'isf'	caratteri inclusi in nomi-file e percorsi
'isident'	'isi'	caratteri inclusi in identificativi
'iskeyword'	'isk'	caratteri inclusi in parole chiave (keywords)
'isprint'	'isp'	caratteri stampabili
'joinspaces'	'js'	due spazi dopo un punto usando comando join
'key'		chiave di cifratura
'keymap'	'kmp'	nome di una mappatura tastiera
'keymodel'	'km'	abilita inizio/fine scelta con tasti
'keywordprg'	'kp'	programma da usare per il comando "K"
'langmap'	'lmap'	caratt. alfab. usati in modo alternativo lingua
'langmenu'	'lm'	linguaggio da usare nei menù
'laststatus'	'ls'	determina presenza di una riga-status in basso
'lazyredraw'	'lz'	non ridisegnare durante esecuzione macro
'linebreak'	'lbr'	vai a capo per righe lunghe a fine parola
'lines'		numero di righe sullo schermo
'linespace'	'lsp'	numero di righe di pixel di interlinea
'lisp'		indentazione automatica per Lisp
'lispwords'	'lw'	parole che determinano indentazione Lisp
'list'		visualizza <Tab> e <EOL>
'listchars'	'lcs'	caratteri da usare in modo 'list'
'loadplugins'	'lpl'	carica scritti plugin alla partenza
'macatsui'		GUI Mac: usa visualizzazione ATSUI
'magic'		modifica caratteri speciali in espress. ricerca

'makeef'	'mef'	nome del file-errori per ":make"
'makeprg'	'mp'	programma da usare per il comando ":make"
'matchpairs'	'mps'	coppie di caratteri gestiti da comando "%"
'matchtime'	'mat'	decimi di secondo visualizz. parentesi corrisp.
'maxcombine'	'mco'	massimo numero caratteri multi-byte visualizzati
'maxfuncdepth'	'mfd'	massima profondità ricorsiva per funzioni utente
'maxmapdepth'	'mmd'	massima profondità ricorsiva per mappature
'maxmem'	'mm'	massima memoria (in Kbyte) usata per un buffer
'maxmempattern'	'mmp'	massima memoria (in Kbyte) usata x espress. ricer.
'maxmemtot'	'mmt'	massima memoria (in Kbyte) usata x tutti i buffer
'menuitems'	'mis'	tempo massimo per considerare clic come doppio
'mkspellmem'	'msm'	memoria usata prima che :mkspell comprima albero
'modeline'	'ml'	riconosci modeline a inizio o fine file
'modelines'	'mls'	numero di righe controllate cercando modeline
'modifiable'	'ma'	modifiche al testo inibite
'modified'	'mod'	un buffer è stato modificato
'more'		sosta nelle liste una volta riempito lo schermo
'mouse'		abilita uso di clic del mouse
'mousefocus'	'mousef'	spostamento mouse provoca cambio di buffer
'mousehide'	'mh'	nascondi puntatore mouse mentre si inserisce testo
'mousemodel'	'mousem'	cambia funzione ai bottoni del mouse
'mouseshape'	'mouses'	cambia forma mouse per nei differenti Modi di Vim
'mousetime'	'mouset'	tempo massimo per considerare clic come doppio
'mzquantum'	'mzq'	intervallo fra i controlli di sottotask MzScheme
'nrformats'	'nf'	numero formati riconosciuti dal comando CTRL-A
'number'	'nu'	premetti un numero di riga a ogni riga
'numberwidth'	'nuw'	numero di colonne usato per inserire numero righe
'omnifunc'	'ofu'	funzione per completamento relativo a tipo-file
'opendevic'	'odev'	consenti lettura/scrittura diretta in MS-Windows
'operatorfunc'	'opfunc'	funzione da invocare per operatore g@
'osfiletype'	'oft'	non più supportato
'paragraphs'	'para'	macro nroff che separa paragrafi
'paste'		consenti di incollare (PASTE) testo
'pastetoggle'	'pt'	tasto che abilita/disabilita 'paste'
'patchexpr'	'pex'	espressione usata per applicare patch a file
'patchmode'	'pm'	tieni la versione più vecchia di un file
'path'	'pa'	lista directory cercate con "gf" et al.
'preserveindent'	'pi'	preserva struttura indentazione re-indentando
'previewheight'	'pvh'	altezza finestra di anteprima
'previewwindow'	'pvw'	identifica finestra di anteprima
'printdevice'	'pdev'	nome stampante da usare per :hardcopy
'printencoding'	'penc'	codifica da usare per stampare
'printexpr'	'pexpr'	espressione usata in PostScript per :hardcopy
'printfont'	'pfn'	nome del font da usare per :hardcopy
'printhead'	'pheader'	formato intestazione usata per :hardcopy
'printmbcharset'	'pmbcs'	font CJK da usare per :hardcopy
'printmbfont'	'pmbfn'	nomi font da usare per output CJK di :hardcopy
'printoptions'	'popt'	controlla formato di output :hardcopy
'pumheight'	'ph'	massima altezza menù dinamico (popup)
'quoteescape'	'qe'	caratteri protezione usato in una stringa
'readonly'	'ro'	inibisci scrittura buffer
'redrawtime'	'rdt'	ritardo evidenziazione 'hlsearch' e :match
'regexpengine'	're'	codice di default per valutare espress. regolari
'relativenumber'	'rnu'	mostra numero riga relativo prima di ogni riga
'remap'		permette ricorsività nelle mappature
'report'		soglia per dare messaggio numero righe cambiate
'restorescreen'	'rs'	Win32: ripristina schermata quando si esce
'revins'	'ri'	inserimento caratteri funzionerà anche indietro
'rightleft'	'rl'	finestra orientata da destra a sinistra
'rightleftcmd'	'rlc'	comandi con modifiche da destra a sinistra
'ruler'	'ru'	mostra cursore, riga e colonna in riga-status
'rulerformat'	'ruf'	formato personalizzato riga-status
'runtimepath'	'rtp'	lista directory usate per file runtime
'scroll'	'scr'	numero righe da scorrere con CTRL-U e CTRL-D
'scrollbind'	'scb'	scorrimento sincronizzato tra finestre
'scrolljump'	'sj'	numero minimo di righe da scorrere
'scrolloff'	'so'	numero minino di righe sopra e sotto il cursore
'scrollopt'	'sbo'	specifica comportamento 'scrollbind'
'sections'	'sect'	macro nroff che separano sezioni
'secure'		modo sicurezza per lettura .vimrc in direct. corr.
'selection'	'sel'	tipo di selezione da usare
'selectmode'	'slm'	quando usare Modo Select al posto di Modo Visual
'sessionoptions'	'ssop'	opzioni per :mksession

'shell'	'sh'	nome shell da usare per eseguire comandi esterni
'shellcmdflag'	'shcf'	flag alla shell per eseguire un comando
'shellpipe'	'sp'	stringa per inserire in file-errori output ":make"
'shellquote'	'shq'	carattere/i fra cui includere comandi per la shell
'shellredir'	'srr'	stringa per inserire output di filtro in file temp
'shellslash'	'ssl'	usare "/" per nomi-file per la shell
'shelltemp'	'stmp'	se usare file temporaneo per comandi della shell
'shelltype'	'st'	Amiga: influenza esecuzione di una shell
'shellxescape'	'sxe'	carattere da proteggere se ' shellxquote ' è (
'shellxquote'	'sxq'	come ' shellquote ', ma comprende ridirezione
'shiftround'	'sr'	arrotonda indentazione a multipli di shiftwidth
'shiftwidth'	'sw'	numero spazi da usare per ogni (auto)indent
'shortmess'	'shm'	lista flag, per ridurre lunghezza messaggi
'shortname'	'sn'	fuori da MS-DOS: nomi-file supposti 8.3 caratteri
'showbreak'	'sbr'	stringa per marcare inizio righe "spezzate"
'showcmd'	'sc'	mostra comando (parziale) nella riga-status
'showfulltag'	'sft'	mostra espressione tag durante completamento
'showmatch'	'sm'	mostra parentesi corrispondente durante inserzione
'showmode'	'smd'	messaggio su riga-status che indica Modo corrente
'showtabline'	'stal'	dice quando visualizzare riga delle linguette
'sidescroll'	'ss'	numero minimo colonne per scorrimento orizzontale
'sidescrolloff'	'siso'	num. min. colonne a sinistra e destra del cursore
'smartcase'	'scs'	passa a ' noic ' se espress. ha maiuscolo/minuscolo
'smartindent'	'si'	autoindentazione intelligente per programmi C
'smarttab'	'sta'	usare ' shiftwidth ' inserendo <Tab>
'softtabstop'	'sts'	numero spazi usati da <Tab> durante modifica
'spell'		abilita controllo ortografico
'spellcapcheck'	'spc'	espressione per individuare fine frase
'spellfile'	'spf'	file ai quali zg e zw aggiungono parole
'spelllang'	'spl'	lingua/e per cui fare controllo ortografico
'spellsuggest'	'sps'	metodi/i usato per suggerire correzioni ortograf.
'splitbelow'	'sb'	nuova finestra da split sotto quella corrente
'splitright'	'spr'	nuova finestra da split a destra della corrente
'startofline'	'sol'	comandi di movimento al primo non-bianco di riga
'statusline'	'stl'	formato personalizzato per riga-status
'suffixes'	'su'	suffissi da ignorare in corrispondenze multiple
'suffixesadd'	'sua'	suffixes aggiunti nella ricerca di un file
'swapfile'	'swf'	se usare come buffer un file di swap
'swapsync'	'sws'	come sincronizzare il file di swap
'switchbuf'	'swb'	imposta comportamento nel passare ad altro buffer
'synmaxcol'	'smc'	colonna massima ricerca colorazione sintattica
'syntax'	'syn'	sintassi da caricare per buffer corrente
'tabstop'	'ts'	numero di spazi usati da <Tab> nel file
'tabline'	'tal'	formato personalizzato linea linguette console
'tabpagemax'	'tpm'	massimo numero di linguette per ~p e "tab all"
'tagbsearch'	'tbs'	usa ricerca binaria in file tag
'taglength'	'tl'	numero caratteri significativi in un tag
'tagrelative'	'tr'	nomi-file nei file tag sono relativi
'tags'	'tag'	lista nomi-file usati dal comando tag
'tagstack'	'tgst'	aggiunge tag sulla pila tag
'term'		nome del terminale in uso
'termbidi'	'tbidi'	il terminale si occupa della bi-direzionalità
'termencoding'	'tenc'	codifica carattere usata dal terminale
'terse'		abbrevia alcuni messaggi
'textauto'	'ta'	obsoleta, usare ' fileformats '
'textmode'	'tx'	obsoleta, usare ' fileformat '
'textwidth'	'tw'	massima larghezza di una riga
'thesaurus'	'tsr'	lista file thesaurus per completamento parole
'tildeop'	'top'	comando tilde "~" si comporta come un operatore
'timeout'	'to'	tempo di time-out su mappature e tasti speciali
'timeoutlen'	'tm'	tempo di time-out in millisecondi
'title'		consenti a Vim di impostare titolo della finestra
'titlelen'		percentuale di ' columns ' usato per finestra titolo
'titleold'		vecchio titolo, ripristinato uscendo da Vim
'titlestring'		stringa da usare per il titolo della finestra Vim
'toolbar'	'tb'	GUI: elementi da mostrare nella toolbar
'toolbariconsze'	'tbis'	dimensione icone in toolbar (solo per GTK 2)
'ttimeout'		tempo di time-out su mappature
'ttimeoutlen'	'ttm'	tempo di time-out su tasti speciali (millisecondi)
'ttybuiltin'	'tbi'	usare termcap interna prima di quella esterna
'ttyfast'	'tf'	indica connessione a un terminale veloce
'ttymouse'	'ttym'	tipo di codici mouse generati
'ttyscroll'	'tsl'	massimo numero di righe per uno scorrimento

'ttytype'	'tty'	alias per 'term'
'undodir'	'udir'	dove posizionare file undo
'undofile'	'udf'	salvare informazioni undo in un file
'undolevels'	'ul'	massimo numero di modifiche annullabili
'undoreload'	'ur'	max num. righe salvate per undo su ricaricamento
'updatecount'	'uc'	dopo 'uc' caratteri, scrivere il file di swap
'updatetime'	'ut'	dopo 'ut' millisecondi, scrivere il file di swap
'verbose'	'vbs'	invia messaggi dettagliati
'verbosefile'	'vfile'	file su cui scrivere i messaggi
'viewdir'	'vdir'	directory dove posizionare file con :mkview
'viewoptions'	'vop'	specifica cosa salvare con :mkview
'viminfo'	'vi'	usare file .file viminfo a inizio e fine edit
'virtualedit'	've'	quando usare Virtual Edit
'visualbell'	'vb'	usare segnale luminoso invece che suono
'warn'		avvisa per comando shell se buffer modificato
'weirdinvert'	'wiv'	per terminali con strani modi di inversione
'whichwrap'	'ww'	consente ad alcuni tasti di uscire dalla riga
'wildchar'	'wc'	carattere per innescare espansione su riga-comando
'wildcharm'	'wcm'	come 'wildchar' ma funziona anche nelle mappature
'wildignore'	'wig'	nomi file di questo tipo non sono espansi
'wildignorecase'	'wic'	ignora maiuscolo/minuscolo espandendo nomi file
'wildmenu'	'wmnu'	usare menù per espansione riga-comando
'wildmode'	'wim'	modalità espansione 'wildchar' su riga-comando
'wildoptions'	'wop'	specifica come fare espansione su riga-comando
'winaltkeys'	'wak'	specifica come gestire tasti ALT per i menù
'window'	'wi'	numero righe da scorrere per CTRL-F e CTRL-B
'winheight'	'wh'	numero minimo di righe per la finestra corrente
'winfixheight'	'wfh'	mantieni altezza finestra aprendone/chiudendone
'winfixwidth'	'wfw'	mantieni larghezza finestra aprendone/chiudendone
'winminheight'	'wmh'	numero minimo di righe per ogni finestra
'winminwidth'	'wmw'	numero minimo di colonne per ogni finestra
'winwidth'	'wiw'	numero minimo di colonne per la finestra corrente
'wrap'		righe lunghe piegate, su più di una riga schermo
'wrapmargin'	'wm'	caratteri a destra dove si va su nuova riga
'wrapscan'	'ws'	ricerca a fine file, ricomincia da inizio file
'write'		scrittura di un file consentita
'writeany'	'wa'	scrivi al file senza conferme "!" per riscrittura
'writebackup'	'wb'	fai un backup prima di ricoprire un file
'writedelay'	'wd'	ritarda 'wd' millisecondi per ogni caratt. (debug)

Q_ur Comandi Undo/Redo

u	N	u	undo (annulla) ultime N modifiche
CTRL-R	N	CTRL-R	redo (riapplica) ultime N modifiche annullate
U		U	ripristina ultima linea modificata

Q_et Comandi esterni

:shell	:sh[ell]	inizia una shell
:!	::! {comando}	esegui {comando} all'interno di una shell
K	K	cerca la parola sotto il cursore usando il programma 'keywordprg' (default: "man")

Q_qf Comandi quickfix

:cc	:cc [nr]	mostra errore [nr] (default ancora l'ultimo)
:cnext	:cn	mostra il prossimo errore
:cprevious	:cp	mostra l'errore precedente
:clist	:cl	lista tutti gli errori
:cfile	:cf	leggi errori dal file 'errorfile'
:cgetbuffer	:cgetb	come :cbuffer ma non saltare a primo errore
:cgetfile	:cg	come :cfile ma non saltare a primo errore
:cgetexpr	:cgete	come :cexpr ma non saltare a primo errore
:caddfile	:caddf	aggiungi errori dal file errori alla lista quickfix corrente
:caddexpr	:cad	aggiungi errori da un'espressione alla lista quickfix corrente
:cbuffer	:cb	leggi errori da testo in un buffer
:cexpr	:cex	leggi errori da un'espressione
:cquit	:cq	esci senza scrivere e restituisci codice di errore (al compilatore)
:make	:make [args]	inizia make, leggi errori, e salta al primo errore

:grep	:gr[ep] [args]	esegui 'grepgrg' per trovare corrispondenze, e salta alla prima di esse
<hr/>		
Q_vc	Comandi vari	
CTRL-L	CTRL-L	cancella e ridisegna lo schermo
CTRL-G	CTRL-G	mostra nome completo file corrente e posizione cursore
ga	ga	mostra valore ascii del carattere sotto il cursore, e quello decimale, esadecimale e ottale
g8	g8	per codifica utf-8: mostra sequenza di byte del carattere sotto il cursore, in esadecimale
g_CTRL-G	g CTRL-G	mostra colonna, riga e posizione del carattere sotto il cursore
CTRL-C	CTRL-C	durante ricerche: interrompe ricerca
dos-CTRL-Break	CTRL-Break	MS-DOS: durante ricerche: interrompe ricerca
		mentre si inserisce un contatore: cancella ultimo carattere
:version	:ve[rsion]	mostra informazioni sulla versione di Vim
:mode	:mode N	MS-DOS: imposta modo schermo a N (numero, C80, C4350, etc.)
:normal	:norm[al][!] {comandi}	esegue comandi in Modo Normale
Q	Q	passa a Modo "Ex"
:redir	:redir >{file}	ridirige messaggi a {file}
:silent	:silent[!] {comando}	esegue {comando} senza dare messaggi
:confirm	:confirm {comando}	nei comandi quit, write, etc., chiede a riguardo di modifiche non salvate o file in sola lettura
:browse	:browse {comando}	apre/legge/scrive un file, usando un dialogo per scegliere il file
<hr/>		
Q_ce	Ex: Edita su riga-comando	
c_<Esc>	<Esc>	lascia riga-comando (se 'wildchar' è <Esc>, va battuto due volte)
c_CTRL-V	CTRL-V {carat.}	inserisci {carat.} letteralmente
c_CTRL-V	CTRL-V {numero}	inserisci valore decimale di un carattere (fino a tre cifre)
c_CTRL-K	CTRL-K {char1} {char2}	inserisci digrafico (Vedere Q_di)
c_CTRL-R	CTRL-R {0-9a-z"%#:=}	inserisci contenuto di un registro
c_<Left>	<Left>/<Right>	cursore a sinistra/destra
c_<S-Left>	<S-Left>/<S-Right>	cursore una parola a sinistra/destra
c_CTRL-B	CTRL-B/CTRL-E	cursore a inizio/fine della riga-comando
c_<BS>	<BS>	cancella il carattere davanti al cursore
c_		cancella il carattere sotto il cursore
c_CTRL-W	CTRL-W	cancella la parola davanti al cursore
c_CTRL-U	CTRL-U	rimuovi tutti i caratteri da inizio riga fino al cursore
c_<Up>	<Up>/<Down>	riprendi precedente/seguito riga-comando che inizia col comando corrente
c_<S-Up>	<S-Up>/<S-Down>	riprendi precedente/seguito riga-comando dalla storia dei comandi
:history	:his[tory]	mostra righe-comando (salvate) più vecchie
Completamento dipendente-dal-contesto sulla riga-comando:		
c_wildchar	'wildchar' (default: <Tab>)	completa espressione davanti al cursore; se ci sono più corrispondenze, emetti suono e mostra la prima; ulteriori 'wildchar' mostreranno le corrispondenze seguenti
c_CTRL-D	CTRL-D	lista tutti i nomi che corrispondono alla espressione davanti al cursore

c_CTRL-A	CTRL-A	inserisci tutti i nomi che corrispondono alla espressione davanti al cursore
c_CTRL-L	CTRL-L	inserisci la parte comune più ampia dei nomi che corrispondono all'espressione
c_CTRL-N	CTRL-N	dopo ' wildchar ' con corrispondenze multiple: vai alla prossima corrispondenza
c_CTRL-P	CTRL-P	dopo ' wildchar ' con corrispondenze multiple: vai alla corrispondenza precedente

Q_ra Ex: Intervalli [sui quali opera un comando]

:range	,	separa due numeri di linea
:range	;	idem, posiziona il cursore sulla prima riga indicata, prima di interpretare la seconda
:range	{numero}	un numero di linea in formato assoluto
:range	.	la riga corrente
:range	\$	l'ultima riga nel file
:range	%	come 1,\$ (tutte le righe del file)
:range	*	come '<,>' (area scelta in Modo Visual)
:range	't	posizione del marcatore t
:range	/ {espress.} [/]	prossima riga in cui {espress.} corrisponde
:range	? {espress.} [?]	riga precedente in cui {espress.} corrisponde
:range	+{num}	aggiungi [num] al numero di riga precedente (default: 1)
:range	-{num}	sottrai [num] dal numero di riga precedente (default: 1)

Q_ex Ex: Caratteri speciali

:bar		separa due comandi (non per ":global" e ":!")
:quote	"	segnala inizio di un commento
:_%	%	nome del file corrente (solo dove si suppone ci debba essere un nome-file)
:_#	#[num]	nome-file alternativo [num] (solo dove si suppone ci debba essere un nome-file)
Nota: I prossimi sette comandi vanno inseriti come sono scritti, la notazione NON indica tasti speciali!		
:<abuf>	<abuf>	numero buffer, da usare in un autocomando (solo dove si suppone ci debba essere un nome-file)
:<afile>	<afile>	nome-file, da usare in un autocomando (solo dove si suppone ci debba essere un nome-file)
:<amatch>	<amatch>	la corrispondenza all'espressione, da usare in un autocomando (solo dove si suppone ci debba essere un nome-file)
:<cword>	<cword>	parola sotto il cursore (solo dove si suppone ci debba essere un nome-file)
:<cWORD>	<cWORD>	PAROLA sotto il cursore (solo dove si suppone ci debba essere un nome-file) (vedere WORD)
:<cfile>	<cfile>	nome-file sotto il cursore (solo dove si suppone ci debba essere un nome-file)
:<sfile>	<sfile>	nome-file di un file ":source"d (eseguito come una sequenza di comandi Vim), all'interno di quel file (solo dove si suppone ci debba essere un nome-file)

Dopo "%", "#", "<cfile>", "<sfile>" o "<afile>"

:p	:p	nome-file completo
:h	:h	testa (nome-file rimosso)
:t	:t	coda (solo nome-file)
:r	:r	radice (ultima estensione rimossa)
:e	:e	estensione
:s	:s/ {espr} / {rimp} /	sostituisce {espr} con {rimp}

Q_st Invocare Vim

-vim	vim [opzioni]	inizia edit con un buffer vuoto
-file	vim [opzioni] {file} ..	inizia edit con uno o più file
--	vim [opzioni] -	leggi file in edit da "stdin"
-tag	vim [opzioni] -t {tag}	edita file associato con {tag}
-qf	vim [opzioni] -q {fname}	inizia edit in Modo QuickFix, visualizza il primo errore

Argomenti di Vim più utili (per la lista completa vedere `|startup-options|`)

<code> -gui </code>	<code>-g</code>	utilizza GUI (altre opzioni specificabili)
<code> -+ </code>	<code>+ [num]</code>	metti cursore a riga <code>[num]</code> (default: ultima riga)
<code> -+c </code>	<code>+ {comando}</code>	esegui <code>{comando}</code> dopo aver caricato il file
<code> -+ / </code>	<code>+ / {espr.} {file} ..</code>	metti cursore alla prima occorrenza di <code>{espr.}</code>
<code> -v </code>	<code>-v</code>	Modo Vi, inizia ex in Modo Normale
<code> -e </code>	<code>-e</code>	Modo Ex, inizia vim in Modo Ex
<code> -R </code>	<code>-R</code>	Modo Sola-lettura, implica <code>-n</code>
<code> -m </code>	<code>-m</code>	modifiche non permesse (annulla opzione <code>'write'</code>)
<code> -d </code>	<code>-d</code>	Modo Diff <code> diff </code>
<code> -b </code>	<code>-b</code>	Modo Binario
<code> -l </code>	<code>-l</code>	Modo Lisp
<code> -A </code>	<code>-A</code>	Modo Arabo (<code>'arabic'</code> è impostato)
<code> -F </code>	<code>-F</code>	Modo Farsi (<code>'fkmap'</code> e <code>'rightleft'</code> sono impostati)
<code> -H </code>	<code>-H</code>	Modo Ebraico (<code>'hkmap'</code> e <code>'rightleft'</code> sono impost.)
<code> -V </code>	<code>-V</code>	Modo Verbose, invia molti messaggi informativi
<code> -C </code>	<code>-C</code>	Modo Compatible, imposta opzione <code>'compatible'</code>
<code> -N </code>	<code>-N</code>	Modo Non-Compatible, annulla opzione <code>'compatible'</code>
<code> -r </code>	<code>-r</code>	fornisce (solo) una lista di file di swap
<code> -r </code>	<code>-r {file} ..</code>	recupera una sessione di edit andata male
<code> -n </code>	<code>-n</code>	non creare un file di swap (tutto in memoria)
<code> -o </code>	<code>-o [num]</code>	aprire <code>[num]</code> finestre (default: una per ogni file)
<code> -f </code>	<code>-f</code>	GUI: processo in questo terminale, non innescare un processo indipendente (fork)
		Amiga: non far ripartire Vim per aprire una finestra (per esempio, mail)
<code> -s </code>	<code>-s {scriptin}</code>	prima esegui comandi letti dal file <code>{scriptin}</code>
<code> -w </code>	<code>-w {scriptout}</code>	aggiungi caratteri inseriti in fondo al file <code>{scriptout}</code> (append)
<code> -W </code>	<code>-W {scriptout}</code>	sostituisci coi caratteri inseriti il file <code>{scriptout}</code> (overwrite)
<code> -T </code>	<code>-T {terminal}</code>	imposta nome del terminale
<code> -d </code>	<code>-d {device}</code>	Amiga: aprire <code>{device}</code> da usare come console
<code> -u </code>	<code>-u {vimrc}</code>	leggi personalizzazioni da <code>{vimrc}</code> invece che dai file "normali"
<code> -U </code>	<code>-U {gvimrc}</code>	idem, serve se si usa una GUI
<code> -i </code>	<code>-i {viminfo}</code>	leggi informazioni da <code>{viminfo}</code> invece che dai file "normali"
<code> --- </code>	<code>--</code>	fine delle opzioni, gli argomenti ulteriori sono dei nomi-file
<code> --help </code>	<code>--help</code>	mostra lista argomenti possibili ed esce
<code> --version </code>	<code>--version</code>	mostra informazioni sulla versione ed esce
<code> -- </code>	<code>-</code>	legge file da "stdin"

Q_ed Modificare un file

Senza `!:` Non eseguire se il buffer corrente è stato modificato.

Con `!:` Non salvare le modifiche al buffer corrente.

<code> :edit_f </code>	<code>:e[dit][!] {file}</code>	modifica <code>{file}</code>
<code> :edit </code>	<code>:e[dit][!]</code>	ricarica il file corrente
<code> :enew </code>	<code>:ene[w][!]</code>	modifica un nuovo buffer, senza nome
<code> :find </code>	<code>:fin[d][!] {file}</code>	trova <code>{file}</code> in <code>'path'</code> e modificalo
<code> CTRL-^ </code>	<code>N CTRL-^</code>	modifica file alternativo <code>N</code> (equivalente a <code>" :e #N"</code>)
<code> gf </code>	<code>gf o lf</code>	modifica il file il cui nome è sotto il cursore
<code> :pwd </code>	<code>:pwd</code>	lista nome della directory corrente
<code> :cd </code>	<code>:cd [path]</code>	cambia la directory corrente a <code>[path]</code>
<code> :cd- </code>	<code>:cd -</code>	torna alla precedente directory corrente
<code> :file </code>	<code>:f[ile]</code>	lista nome-file corrente e posizione del cursore
<code> :file </code>	<code>:f[ile] {nome}</code>	imposta nome-file corrente a <code>{nome}</code>
<code> :files </code>	<code>:files</code>	lista nomi-file alternativi

Q_fl Uso della lista argomenti `|argument-list|`

<code> :args </code>	<code>:ar[gs]</code>	lista lista argomenti, con il file corrente tra <code>"[]"</code>
------------------------	----------------------	---

:all	:all o :sall	apri una finestra per ogni file nella lista argomenti	
:wn	:wn[ext][!]	scrivi un file e inizia a editare il prossimo file	
:wn	:wn[ext][!] {file}	scrivi un {file} e inizia a editare il prossimo file, a meno che {file} esista già; aggiungendo "!", sovrascrivi il file anche se già esistente	
:wN	:wN[ext][!] [file]	scrivi {file} e inizia a editare il file precedente	
<hr/>			
	finestra corrente	nuova finestra	~
:argument	:argu[ment] N	:sar[gument] N	edita file N
:next	:n[ext]	:sn[ext]	edita prossimo file
:next_f	:n[ext] {arglist}	:sn[ext] {arglist}	imposta nuova lista argomenti ed edita il primo file
:Next	:N[ext]	:sN[ext]	edita file precedente
:first	:fir[st]	:sfir[st]	edita primo file
:last	:la[st]	:sla[st]	edita ultimo file
<hr/>			
Q_wq		Salvare e uscire	
:w	: [interv.]w[rite][!]	scrivi su file corrente	
:w_f	: [interv.]w[rite] {file}	scrivi su {file}, se non esiste già	
:w_f	: [interv.]w[rite]! {file}	scrivi su {file}. Sovrascrivi il file se già esistente	
:w_a	: [interv.]w[rite][!] >>	aggiungi in fondo al file corrente	
:w_a	: [interv.]w[rite][!] >> {file}	aggiungi in fondo a {file}	
:w_c	: [interv.]w[rite]! {cmd}	esegui comando {cmd} con [interv.] righe come "stdin" (standard input)	
:up	: [interv.]up[date][!]	riscrivi il file corrente solo se è stato modificato	
:wall	:wa[ll][!]	scrivi tutti i buffer che sono stati modificati	
:q	:q[uit]	lascia il buffer corrente, se non è stato modificato. Esci da Vim se non ci sono altri buffer diversi da quello di help	
:q	:q[uit]!	lascia il buffer corrente in ogni caso, scartando eventuali modifiche. Esci da Vim se non ci sono altri buffer diversi da quello di help	
:qa	:qa[ll]	esci da Vim, se non ci sono modifiche da salvare	
:qa	:qa[ll]!	esci da Vim in ogni caso, scartando eventuali modifiche	
:cq	:cq	esci senza scrivere eventuali modifiche e ritorna un codice di errore	
:wq	:wq[!]	scrivi il file corrente ed esci	
:wq	:wq[!] {file}	scrivi su {file} ed esci	
:xit	:x[it][!] [file]	come ":wq" ma scrivi solo se sono state fatte modifiche	
ZZ	ZZ	come ":x"	
ZQ	ZQ	come ":q!"	
:xall	:xa[ll][!] o :wqall[!]	scrivi tutti i buffer modificati ed esci	
:stop	:st[op][!]	sospendi Vim o inizia una shell nuova. Se l'opzione 'aw' è impostate e [!] non è specificato, scrivi il buffer	
CTRL-Z	CTRL-Z	come ":stop"	
<hr/>			
Q_ac		Comandi automatici	
viminfo-file	legge i registri, i marcatori, la storia, a inizio sessione Vim, e li salva all'uscita da Vim		
:rviminfo	:rv[iminfo] [file]	leggi info da file viminfo [file]	
:rviminfo	:rv[iminfo]! [file]	idem, sovrascrivi info esistenti	
:wviminfo	:wv[iminfo] [file]	aggiungi info a file viminfo [file]	

:wviminfo	:wv[iminfo]! [file]	scrivi info a file viminfo [file]
modeline		impostazione automatica di opzioni per file in modifica
modeline	vim:{set-arg}: ..	Nelle prime e ultime righe del file (vedere opzione 'ml'), {set-arg} è usato come argomento per ":set"
autocommand		Esecuzione automatica di comandi al verificarsi di alcuni eventi
:autocmd	:au	lista tutti gli autocomandi
:autocmd	:au {evento}	lista tutti gli autocomandi per {evento}
:autocmd	:au {evento} {pat}	lista tutti gli autocomandi per {evento} con {pat}
:autocmd	:au {evento} {pat} {cmd}	inserisci nuovi autocomandi per {evento} con {pat}
:autocmd	:au!	rimuovi tutti gli autocomandi
:autocmd	:au! {evento}	rimuovi tutti gli autocomandi per {evento}
:autocmd	:au! * {pat}	rimuovi tutti gli autocomandi per {pat}
:autocmd	:au! {evento} {pat}	rimuovi tutti gli autocomandi per {evento} con {pat}
:autocmd	:au! {evento} {pat} {cmd}	rimuovi tutti gli autocomandi per {evento} con {pat} e inseriscine di nuovi
<hr/>		
Q_wi	Comandi multi-finestra	
CTRL-W_s	CTRL-W s o :split	dividi finestra in due parti
:split_f	:split {file}	dividi finestra ed edita {file} in una di esse
:vsplit	:vsplit {file}	idem, ma dividi verticalmente
:vertical	:vertical {cmd}	se {cmd} divide la finestra, fallo verticalmente
:sfind	:sf[ind] {file}	dividi finestra, trova {file} in 'path' ed editalo
CTRL-W_]	CTRL-W]	dividi finestra e salta a tag sotto il cursore
CTRL-W_f	CTRL-W f	dividi finestra ed edita nome-file sotto il cursore
CTRL-W_^	CTRL-W ^	dividi finestra ed edita file alternativo
CTRL-W_n	CTRL-W n o :new	crea una nuova finestra vuota
CTRL-W_q	CTRL-W q o :q[uit]	lascia edit e chiudi finestra
CTRL-W_c	CTRL-W c o :cl[ose]	nascondi buffer e chiudi finestra
CTRL-W_o	CTRL-W o o :on[ly]	rendi la finestra corrente l'unica visibile sullo schermo
CTRL-W_j	CTRL-W j	sposta cursore a finestra sotto
CTRL-W_k	CTRL-W k	sposta cursore a finestra sopra
CTRL-W_CTRL-W	CTRL-W CTRL-W	sposta cursore a finestra sotto (gira circolarmente)
CTRL-W_W	CTRL-W W	sposta cursore a finestra sopra (gira circolarmente)
CTRL-W_t	CTRL-W t	sposta cursore a finestra più alta
CTRL-W_b	CTRL-W b	sposta cursore a finestra più bassa
CTRL-W_p	CTRL-W p	sposta cursore alla finestra attiva precedentemente
CTRL-W_r	CTRL-W r	ruota finestre in giù
CTRL-W_R	CTRL-W R	ruota finestre in su
CTRL-W_x	CTRL-W x	cambia finestra corrente con la prossima
CTRL-W_=	CTRL-W =	tutte le finestre di altezza uguale
CTRL-W_-	CTRL-W -	diminuisci altezza finestra corrente
CTRL-W_+	CTRL-W +	aumenta altezza finestra corrente
CTRL-W_	CTRL-W _	imposta altezza finestra corrente (default: molto ampia)

Q_bu

Comandi lista buffer

:buffers|

:ball|

:unhide|

:badd|

:bunload|

:bdelete|

:buffer|

:bnext|

:bNext|

:bprevious|

:bfirst|

:blast|

:bmodified|

:syn-on|

:syn-off|

:syn-keyword|

:syn-match|

:syn-region|

:syn-sync|

:syntax|

:syn-clear|

:highlight|

:highlight|

:filetype|

:filetype|

:buffers o :files

:ball o :sball

:unhide o :sunhide

:badd {fname}

:bunload[!] [N]

:bdelete[!] [N]

:buffer [N]

:bnext [N]

:bNext [N]

:bprevious [N]

:bfirst

:blast

:bmod [N]

:syntax on

:syntax off

:syntax keyword {nome-gruppo} {keyword} ..

:syntax match {nome-gruppo} {espress.} ...

:syntax region {nome-gruppo} {espress.} ...

:syntax sync [ccomment | righe {N} | ...]

:syntax [lista]

:syntax clear

:highlight clear

:highlight {nome-gruppo} {key}={arg} ..

:filetype on

:filetype plugin indent on

lista tutti i buffer e nomi-file noti

edita tutti gli argomenti/buffer

edita tutti i buffer esistenti

aggiungi nome-file {fname} alla lista

scarica buffer [N] dalla memoria

scarica buffer [N] e cancellalo dalla lista buffer

vai ad arg./buf. N

a Nmo prossimo arg./buf.

a Nmo preced. arg./buf.

a Nmo preced. arg./buf.

a primo arg./buf.

a ultimo arg./buf.

a Nmo buf. modificato

inizia uso evidenziazione sintattica

termina uso evidenziazione sintattica

aggiungi un elemento keyword

aggiungi un elemento match

aggiungi un elemento region

specifica come sincronizzare sintassi (quante righe considerare)

lista elementi di sintassi correnti

azzerà ogni informazione di sintassi

azzerà ogni informazione di evidenziazione

imposta evidenziazione per {nome-gruppo}

attiva determinazione tipo-file, senza evidenziazione sintattica

attiva determinazione tipo-file, con indentazione automatica e impostazioni

Q_sy

Evidenziazione sintattica

Q_gu

Comandi GUI

Nota: {lhs}

{rhs}

Left Hand Side - parte sinistra dell'espressione

Right Hand Side - parte destra dell'espressione

:gui|

:gui|

:menu|

:menu|

:menu|

:menu|

:tmenu|

:unmenu|

:gui

:gui {fname} ..

:menu

:menu {mpath}

:menu {mpath} {rhs}

:menu {pri} {mpath} {rhs}

:menu ToolBar.{nome} {rhs}

:tmenu {mpath} {text}

:unmenu {mpath}

UNIX: inizia ad usare GUI

idem, e inoltre edita {fname} ..

lista tutti i menù

lista i menù iniziando da {mpath}

aggiungi menù {mpath}, dando {rhs}

idem, con priorità {pri}

aggiungi elemento toolbar, dando {rhs}

aggiungi suggerimento a menù {mpath}

rimuovi menù {mpath}

Q_fo

Piegature

'foldmethod'|

set foldmethod=manual

piegatura manuale

	set foldmethod=indent	piegatura in base alla rientranza
	set foldmethod=expr	piegatura in base a 'foldexpr'
	set foldmethod=syntax	piegatura in base alla sintassi
	set foldmethod=marker	piegatura in base a 'foldmarker'
zf	zf{movim.}	operatore: Definizione di una piegatura in modo manuale
:fold	:{range}fold	definisce una piegatura per {range} righe
zd	zd	cancella una sola piegatura sotto il cursore
zD	zD	cancella tutte le piegature sotto il cursore
zo	zo	apre una sola piegatura sotto il cursore
zO	zO	apre tutte le piegature sotto il cursore
zc	zc	chiude una sola piegatura sotto il cursore
zC	zC	chiude tutte le piegature sotto il cursore
zm	zm	aumenta piegature: diminuisci 'foldlevel'
zM	zM	chiudi tutte le piegature: imposta 'foldlevel' a zero
zr	zr	riduci piegatura: aumenta 'foldlevel'
zR	zR	aprire tutte le piegature: imposta 'foldlevel' al valore massimo
zn	zn	nessuna piegatura: annulla 'foldenable'
zN	zN	imposta 'foldenable' , le piegature tornano quelle che erano prima
zi	zi	inverti impostazione 'foldenable'

vim:tw=78:ts=8:ft=help:norl:

Per segnalazioni scrivere a vimdoc.it at gmail dot com
oppure ad Antonio Colombo azc100 at gmail dot com