`Vi` Quick Reference

Entering and Exiting vi

vi *file*	Edits *file*.
vi +*n file*	Edits *file*, cursor at line *n*.
vi + *file*	Edits *file*, cursor at end.
vi -r	List system saved files.
vi -r *file*	Edits system saved *file*.
vi *file* ...	Edits first *file*; rest via `:n`.
vi -t *tag*	Edits file through *tag* in tags file.
vi +/*pattern file*	Edits file, cursor at *pattern*.
view *file*	**vi** *file*, no modification.
`ZZ`	Exits from **vi**, saving changes.
`^Z`	Stops **vi** for later resumption.

The Display

Last line	Error messages, echoing input to commands: `/`, `?`, `:`, `!`, feedback about I/O and large changes.
`@` lines	On screen only, not in file.
`~` lines	Lines past end of file.
`^`*x*	Control characters.
TABs	Expands to spaces, cursor on last space.

Counts before vi Commands

Line/column no.	`z, G, |`
Scroll amount	`^D, ^U`
Replicate insert	`a, i, A, I`
Repeat effect	Most rest

Vi States

Command mode   Executes commands; the normal and initial state. Other modes return to command mode. `ESC` cancels a partial command.

Insert mode   Entered by `a, i, A, I, o, O, c, C, s, S, R`, followed by arbitrary text terminated by ESC. An interrupt aborts an insertion.

Last line   Reads user input for commands `/,  ?, :, !`, which terminates in ESC or RETURN. An interrupt aborts.

Basic Commands

`dw`	Deletes a word.
`de`	Deletes a word leaving punctuation.
`dd`	Deletes a line.
`3dd`	Deletes three lines.
`d%`	Deletes to matching parenthesis/bracket
`i`*text*`ESC`	Inserts *text*
`cw`*text*`ESC`	Changes word to *text*
`ea`*s*`ESC`	Pluralizes word.
`xp`	Transposes characters.

Interrupting, Cancelling, Shell Interaction

`ESC`	Ends insert, input, or incomplete command.
`^?`	(DELETE) interrupts **vi**.
`^Z`	Suspends **vi** (*csh* and *ksh*).
`:sh`	Runs default shell, then returns.
`:!`*cmd*	Runs shell-level *cmd*, then returns.

File Manipulation

`:w`	Writes back changes.
`:wq`	Writes and quits from **vi**.
`:q`	Quits from **vi**.
`:q!`	Quits, discards changes.
`:e` *file*	Edits *file*.
`:e!`	Reedits, discards changes.

`:e +` *file*	Edits *file*, starting at the end.
`:e +`*n file*	Edits *file* starting at line *n*.
`:w` *file*	Writes to *file*.
`:w!` *file*	Overwrites *file*.
`:n`	Edits next file in argument list.
`:n` *args*	Specifies new argument list.
`:f`	Displays current file and line.
`^G`	Synonym for `:f`.
`:ta` *tag*	Goes to tags file entry *tag*.
`^]`	Same as `:ta`, but current word is *tag*.

Positioning within File

`^F`	Moves forward a screen.
`^B`	Moves backward a screen.
`^D`	Scrolls down half a screen.
`^U`	Scrolls up half a screen.
k`G`	Goto line *k* or last line (default).
`/`*pattern*`ESC`	Searches forward for *pattern*.
`?`*pattern*`ESC`	Searches backward for *pattern*.
`n`	Repeats last `/` or `?`.
`N`	Repeats last `/` or `?` in other direction.
`/`*pattern*`/+`*n*	Goto *n*th line after *pattern*.
`?`*pattern*`?-`*n*	Goto *n*th line before *pattern*.
`]]`	Next section or function.
`[[`	Previous section or function.
`%`	Finds matching (,), {, or }.

Adjusting the Screen

`^L`	Clears and redisplays.
`^R`	Redisplays, eliminates @ lines.
`z`RETURN	Current line at window top.
`z-`	Current line at window bottom.
`z.`	Current line at window center.

/pattern/z-	Line with *pattern* at bottom.
z*n*	Uses *n* line window.
^E	Scrolls window down one line.
^Y	Scrolls window up one line.

Marking and Returning

' '	Returns to previous context.
' '	Returns to previous context at first nonblank character of the line.
m*x*	Marks position with letter *x*.
'*x*	Returns to marked position *x*.
'*x*	Returns to *x* at first nonblank character in line.

Line Positioning

:.=	Displays current line number
H	Moves to the first window line.
L	Moves to the last window line.
M	Moves to the middle window line.
+	Next line.
-	Previous line.
RETURN	Same as '+'.
↓ or j	Next line, same column.
↑ or k	Previous line, same column.

Character Positioning

^	First nonblank character.
O	(zero) Beginning of line (column one).
$	End of line (last character).
h or ←	Previous character.
l or →	Next character.
^H	Same as *leftarrow*.
SPACE	Same as →.

f*x*	Finds *x* forward in current line.
F*x*	Finds *x* backward in current line.
t*x*	Moves forward upto *x* in current line.
T*x*	Moves backward upto *x* in current line.
;	Repeats last `f F t` or `T`
,	; in the reverse direction.
|	Moves to specified column.
%	Moves to matching (, {,), or }.

Words, Sentences, Paragraphs

w	Next word.
b	Moves backward to the beginning of a word.
e	End of word.
)	Next sentence.
}	Next paragraph.
(	Moves backward to the beginning of a sentence.
{	Moves backward to the beginning of a paragraph.
W	Next blank delimited word.
B	Moves backward to the beginning of a blank delimited word.
E	End of blank delimited word.

Corrections During Insert

^H	Deletes last input character.
^W	Deletes last input word.
erase	Character erase, same as `^H`.
kill	Line kill, deletes entire input line.
\	Escapes character erase and line kill.
ESC	Ends insertion, back to command mode.
interrupt	Aborts insert (usually `^c`).
^D	Backtabs over blanks supplied by autoindent.
^^D	(hat then `^D`) Kills autoindent; autoindent remains effective for next line.
0^D	Kills autoindent; until next indent.
^V	Quotes any special character.

Insert and Replace

`a`	Inserts after cursor.
`i`	Inserts before cursor.
`A`	Inserts at end of current line.
`I`	Inserts before first nonblank character of current line.
`o`	Opens a line for insertion below current line.
`O`	Opens a line for insertion above current line.
`r`x	Replaces current character with x.
`R`	Overwrites characters.
`~`	Changes case of current character.

Operators (Double to affect lines)

`d`	Deletes text object.
`c`	Changes text object.
`<`	Left shifts text object.
`>`	Right shifts text object.
`y`	Yanks text object to buffer.

Miscellaneous Operations

`C`	Changes rest of line.
`D`	Deletes rest of line.
`s`	Substitutes characters.
`S`	Substitutes lines.
`J`	Joins lines.
`x`	Deletes characters.
`X`	Deletes characters before cursor.

Cut and Paste

`p` or `"`x`p`	Pastes deleted or buffer x text after current position/line.
`P` or `"`x	Pastes deleted or buffer x text before current position/line.

`"`x`y`	Copies (Yanks) text object to buffer x.
`"`x`d`	Deletes text object into buffer x.
`yy`	Yanks current line.
n`yy`	Yanks several lines.

Undo, Redo, and Retrieve

`u`	Undoes last change.
`U`	Restores current line.
`.`	Repeats last change.
`"`n`p`	Retrieves nth last delete.