

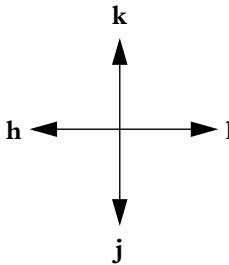


# I

## Quick Reference

The following pages contain the maximum amount of useful information in the minimum space, and thus provide a quick reference. Not all commands are covered, but we have tried to include every command you will encounter in day-to-day editing.

## Basic Commands

<p><b>Movement</b></p> 	<p style="text-align: center;"><b>Inserting Text</b></p> <p>[count]itext&lt;Esc&gt; Insert text (before cursor)                  [count]atext&lt;Esc&gt; Insert text (after cursor)                  o Open line below cursor.                  O Open line above cursor.                  [count]cw Change word                  [count]cc Change entire line                  [count]c{motion} Change from cursor to {motion}.</p>
<p style="text-align: center;"><b>Undo/Redo</b></p> <p>u Undo the last change                  CTRL-R Redo the last change that was undone                  U Undo all changes on the line.</p>	<p style="text-align: center;"><b>Deleting Text</b></p> <p>[count]dd Delete lines.                  [count]dw Delete words.                  [count]x Delete characters                  [count]d{motion} Delete from cursor to {motion}.</p>
<p style="text-align: center;"><b>Getting Out</b></p> <p>ZZ Write file and exit                  :q! Discard changes and exit</p>	<p style="text-align: center;"><b>Help</b></p> <p>:help {topic} Display help on the given topic.                  CTRL-] Follow help “hyper-link.”                  CTRL-T Return to location before last CTRL-].                  ZZ Exit help</p>
<p style="text-align: center;"><b>Searching</b></p> <p>[count]f{char} Forward search for character on a line                  [count]F{char} Backward search for character on a line                  /{string} Search forward for string.                  ?{string} Search backward for string                  [count]n Repeat last search                  [count]N Repeat last search in opposite direction                  / Repeat last search forward                  ? Repeat last search backward</p>	<p style="text-align: center;"><b>Additional Movement</b></p> <p>m{letter} Place mark                  ‘{letter} Go to mark                  G Last line                  gg First line                  [count]G Go to [count] line                  CTRL-U Up 1/2 screen                  CTRL-D Down 1/2 screen                  z&lt;Enter&gt; Make this line the top one on the screen                  zz Center window around the current line.</p>

## Additional Editing Commands

Editing	
.	Repeat last change
[count][“register] <b>d</b> {motion}	Delete from cursor to {motion}
[count][“register] <b>dd</b>	Delete lines
[count][“register] <b>c</b> {motion}	Change from cursor to {motion}
[count][“register] <b>cc</b>	Change lines.
[count][“register] <b>y</b> {motion}	Yank from cursor to {motion}
[count][“register] <b>yy</b>	Yank lines
[count][“register] <b>p</b>	Put after cursor
[count][“register] <b>P</b>	Put before cursor
<b>xp</b>	Twiddle characters
[count] <b>r</b> {char}	Replace one character
[count] <b>R</b> {string}<ESC>	Replace characters until stopped
[count] <b>J</b>	Join lines
Windows	Keyboard Macros
<b>:split</b> {file}	<b>q</b> {register} Record commands into a {register}.
<b>:q</b>	<b>q</b> Stop recording
<b>CTRL-Wj</b>	[count] <b>@</b> {register}
<b>CTRL-Wk</b>	Execute the keyboard macro in
<b>CTRL-Wo</b>	{register}
	window the only one.

## Text Formatting Commands

[count] <b>gq</b> {motion}	Format selected text
[count] <b>gqq</b>	Format lines
[count] <b>gq</b> }	Format paragraphs
<b>:set textwidth</b> ={width}	Set the width of the text line (turn auto wrapping on)
<b>:[range]center</b>	Center the text
<b>:[range]left</b>	Left-align the text
<b>:[range]right</b>	Right-align the text
<b>:set formatoptions</b> ={characters}	Set options which control formatting. Options include:
t	Automatically wrap text
2	When formatting text, use the second line of the paragraph to determine what indent to use.
q	Allow formatting of comments with <b>gq</b> command.

## Commands for Programmers

<b>General</b>					
<b>:syntax on</b> % <b>:set autoindent</b> <b>:set cindent</b> <b>CTRL-D</b> [count]>> [count]>{motion} [count]<< [count]<{motion} [count]={motion} <b>CTRL-]</b> [count]CTRL-T [count]CTRL-W CTRL-] <b>K</b>	Turn on syntax highlighting Go to matching brace, comment or <b>#ifdef</b> Indent each new line the same as the previous one Use C style indentation In insert mode, unindent one level of auto indent. Shift lines right Shift right Shift lines left Shift left Indent code from cursor to {motion} Jump to tag (function definition). Go to location before last tag jump. Split the window and jump to tag under the cursor. Run man on the word under the cursor				
<table style="width: 100%; border: none;"> <thead> <tr> <th style="text-align: left; padding: 5px;"><b>Making the program</b></th> <th style="text-align: left; padding: 5px;"><b>Grep</b></th> </tr> </thead> <tbody> <tr> <td style="padding: 5px; vertical-align: top;"> <b>:make</b>     Run <i>make</i> and capture the output  <b>:cc</b>        Jump to current error  <b>:cn</b>        Jump to next error  <b>:cp</b>        Jump to previous error                             </td> <td style="padding: 5px; vertical-align: top;"> <b>:grep</b> '{string}' {files}                      Run <i>grep</i> and capture the output.                      Treat matches like <b>:make</b> errors.                             </td> </tr> </tbody> </table>	<b>Making the program</b>	<b>Grep</b>	<b>:make</b> Run <i>make</i> and capture the output <b>:cc</b> Jump to current error <b>:cn</b> Jump to next error <b>:cp</b> Jump to previous error	<b>:grep</b> '{string}' {files} Run <i>grep</i> and capture the output. Treat matches like <b>:make</b> errors.	
<b>Making the program</b>	<b>Grep</b>				
<b>:make</b> Run <i>make</i> and capture the output <b>:cc</b> Jump to current error <b>:cn</b> Jump to next error <b>:cp</b> Jump to previous error	<b>:grep</b> '{string}' {files} Run <i>grep</i> and capture the output. Treat matches like <b>:make</b> errors.				
<b>Include File Searches</b>					
[count] ]CTRL-D [count] ]CTRL-I [count] ]d [count] ]D	Find the definition of the macro under the cursor Search for the word under the cursor List first macro definition for the macro the cursor is on. List all macro definitions				
<b>Commands for directory searches</b>					
<b>:set path={directory},...</b> <b>:checkpath</b>  <b>:find {file}</b>  <b>gf</b>	Tell <i>Vim</i> where to search for files Make sure that <i>Vim</i> can find all the files that are referenced in <b>#include</b> directives. Edit the {file}. If the file is not in the current directory, search through the <b>'path'</b> to find it. Edit the file who's name is under the cursor. Search through the <b>'path'</b> if the file is not in the current directory.				

## Visual Mode Commands

Basic Commands		Editing Commands	
<b>v</b>	Enter character visual~mode	[ <i>register</i> ] <b>d</b>	Delete the selected text.
<b>V</b>	Enter line visual mode	[ <i>register</i> ] <b>y</b>	Yank the text into register
<b>CTRL-V</b>	Enter block visual mode	<b>&gt;</b>	Shift text right
<b>\$</b>	Move the right side of the block to the end of each line.	<b>&lt;</b>	Shift text left
		<b>=</b>	Run the lines through the indent program
Selection Commands		<b>J</b>	Join the highlighted lines
[ <i>count</i> ] <b>a</b> (	Select () block	<b>gq</b> { <i>motion</i> }	Format text to ' <b>textwidth</b> '
[ <i>count</i> ] <b>a</b> [	Select [] block	<b>U</b>	Convert text to all UPPER CASE
[ <i>count</i> ] <b>a</b> {	Select {} block	<b>u</b>	Convert text to all lower case.
[ <i>count</i> ] <b>aw</b>	Select word and space after	<b>~</b>	Invert the case of the text.
[ <i>count</i> ] <b>iw</b>	Select word only.		
[ <i>count</i> ] <b>as</b>	Select sentence and following space		
[ <i>count</i> ] <b>is</b>	Select sentence only.		

## Insert Mode Commands

Insert Mode Commands	
<b>CTRL-V</b> { <i>char</i> }	Insert character literally .
<b>CTRL-D</b>	Unindent one level
<b>0 CTRL-D</b>	Remove all automatic indentation
<b>CTRL-T</b>	Insert one ' <b>shiftwidth</b> '
<b>CTRL-Y</b>	Copy the character above the cursor
<b>CTRL-E</b>	Copy the character below the cursor
<b>CTRL-N</b>	Find next match for word before the cursor
<b>CTRL-P</b>	Find previous match for word before the cursor
<b>CTRL-R</b> { <i>register</i> }	Insert contents of a register
<b>CTRL-K</b> { <i>char1</i> }{ <i>char2</i> }	Insert digraphs
<b>CTRL-W</b>	Delete word before cursor
Abbreviations	
<b>:abbreviate</b> { <i>abbr</i> } { <i>expansion</i> }	Define abbreviation.

## Ex Mode Commands

<b>Basic Commands</b>	
<b>Q</b>	Enter Ex mode
<b>:vi</b>	Enter normal mode
<b>:[range]s/{old}/{new}/{flags}</b>	Substitute {new} for {old}. If a flag of “g” is present substitute all occurrences on a line. Otherwise just change the first one.
<b>:[range]write [file]</b>	Write text to a file
<b>:[line]read [file]</b>	Read text from another file into the current file.
<b>File Selection Commands</b>	
<b>:rewind</b>	Edit the first file
<b>:next</b>	Edit next file
<b>:prev</b>	Edit previous file
<b>:last</b>	Edit the last file in the list
<b>:args</b>	List the files in the edit list.
<b>Editing</b>	
<b>:[range]delete</b>	Delete the specified lines
<b>:[range]copy[line]</b>	Copy the specified lines to [line] (default = after current line)
<b>:[range]move[line]</b>	Like:copy but delete the lines as well.
<b>Miscellaneous Commands</b>	
<b>:[range]print</b>	Print the specified lines
<b>:[range]number</b>	Print the specified lines with line numbers
<b>:[range]list</b>	Print the specified lines with ‘list’ option on.
<b>:dlist {name}</b>	List definitions for {name}
<b>:[range]retab[!] {tabstop}</b>	Change the tabbing from the existing setting to the new {tabstop}.
<b>:exit[!]</b>	Close the current file. If it’s the last one, exit the editor
<b>:suspend</b>	Suspend the editor.
<b>:source {file}</b>	Read commands from the specified file.
<b>:redir &gt;{file}</b>	Record the output of the commands in the specified file.

## Options

<b>Setting</b>	
<b>:set</b> {option}= {value}	Set an option
<b>:set</b> {option}?	Display the value of an option
<b>:set</b>	Display the value of options that are set to something other than the default.
<b>:set all</b>	Display the value of all options
<b>:set</b> {option}&	Set an option to the default
<b>:browse set</b>	Set options using a full screen based dialog.
<b>:set</b> {option}+= {value}	Add a value to a list of options (string option). Add a number to a numeric option.
<b>:set</b> {option}-= {value}	Remove a value from a list of options (string option). Subtract a number from a numeric option.
<b>:set</b> {option}	Turn on a boolean option
<b>:set no</b> {option}	Turn off a boolean option
<b>:set inv</b> {option}	Invert a boolean option
<b>Indent / Tabbing</b>	
<b>:set cindent</b>	Turn on C style indentation
<b>:set autoindent</b>	Indent each line the same as the previous one
<b>:set expandtabs</b>	Turn all tabs into space
<b>:set softtabstop</b>	Set the amount of space that the <Tab> key uses. (Note: This is not the same as the number of spaces for a Tab.)
<b>Listing Options</b>	
<b>:set list</b>	Turn on list mode where everything is visible
<b>:set number</b>	Display line numbers
<b>Searching Options</b>	
<b>:set hlsearch</b>	Highlight matching search strings
<b>:set incsearch</b>	Do incremental searches
<b>:set wrapscan</b>	If a search reaches the bottom of the file wrap to the top and keep searching. (Also will wrap from the top to the bottom for reverse searches)
<b>:set ignorecase</b>	Make searches case insensitive
<b>:set autowrite</b>	Automatically write files as needed to preserve data.

## Regular Expressions

<b>Simple Atoms</b>	
<code>x</code>	The literal character “x”
<code>^</code>	Start of line
<code>\$</code>	End of line.
<code>.</code>	A single character .
<code>\&lt;</code>	Start of a word.
<code>\&gt;</code>	End of word.
<b>Range Atoms</b>	
<code>[abc]</code>	Match either “a”, “b”, or “c”.
<code>[^abc]</code>	Match anything except “a”, “b”, or “c”.
<code>[a-z]</code>	Match all characters from “a” through “z”.
<code>[a-zA-Z]</code>	Match all characters from “a” through “z” and “A” through “Z”.
<b>Sub Patterns</b>	
<code>\(pattern\)</code>	Mark the pattern for later use. The first set of <code>\(. \)</code> marks a pattern as <code>\1</code> , the second <code>\2</code> and so on.
<code>\1</code>	Matches the same string that was matched by the first sub-expression in <code>\</code> (and <code>\)</code> . Example: “ <code>\([a-z]\)\. \1</code> ” matches “ata”, “ehe”, “tot”, etc.
<code>\2</code>	Like “ <code>\1</code> ”, but uses second sub-expression,
<code>\9</code>	Like “ <code>\1</code> ”, but uses ninth sub-expression.
<b>Modifiers</b>	
<code>*</code>	Match the previous atom 0 or more times. As much as possible.
<code>\+</code>	Match the previous atom 1 or more times. As much as possible.
<code>\=</code>	Match the previous atom 0 or 1 times.
<code>\{ }</code>	Match the previous atom 0 or more times. (Same as the “ <code>*</code> ” modifier.)
<code>\{n\}</code>	Match the previous atom <i>n</i> times.
<code>\{n,m\}</code>	Match the previous atom <i>n</i> to <i>m</i> times.
<code>\{n, }</code>	Match the previous atom <i>n</i> or more times.
<code>\{,m\}</code>	Match the previous atom from 0 to <i>m</i> times.
<code>\{-n,m\}</code>	Match the previous atom <i>n</i> to <i>m</i> times. Match as little as possible.
<code>\{-n, }</code>	Match the previous atom at least <i>n</i> times. Match as little as possible.
<code>\{-,m\}</code>	Match the previous atom up to <i>m</i> times. Match as little as possible.
<code>\{- }</code>	Match the previous atom 0 or more times. Match as little as possible.
<code>str1   str2</code>	Match <i>str1</i> or <i>str2</i> .