

File Commands

<code>:e filename</code>	open a file in vim
<code>:E</code>	open a file explorer window
<code>:wq</code>	save changes in the current file and quit
<code>:w filename</code>	save changes in the current file or in the new file if filename was provided
<code>:saveas filename</code>	save changes in the new file
<code>:q</code>	quit Vim
<code>:q!</code>	quit Vim without saving changes

Navigation

<code>h</code>	move left
<code>j</code>	move down
<code>k</code>	move up
<code>l</code>	move right
<code>w</code>	move to the start of the next word
<code>gg</code>	go to the start of a file
<code>G</code>	go to the end of a file
<code>50G</code>	jump to the line 50
<code>:50</code>	jump to the line 50

Edit mode

<code>i</code>	enter insert mode at cursor
<code>I</code>	enter insert mode at the start of the line
<code>a</code>	enter append mode after cursor
<code>A</code>	enter append mode at the end of the line

Deleting Text

<code>x</code>	delete character
<code>dd</code>	delete line
<code>D</code>	delete to end of line
<code>daw</code>	delete a word (regardless of cursor position within that word)
<code>das</code>	delete entire sentence (ending with a . / ? / !)
<code>dap</code>	delete entire paragraph
<code>dat</code>	delete a tag (while cursor position is on that tag)

Deleting Text (cont)

<code>da}</code> or <code>di}</code>	delete section between curly braces (while cursor position is within them)
--------------------------------------	--

Undo & Redo

<code>u</code>	undo
<code>CTRL+r</code>	redo

Copying and Pasting

<code>yy</code>	yank/copy line
<code>yaw</code>	yank/copy a word
<code>p</code>	paste after cursor
<code>P</code>	paste before cursor

Find & Replace

<code>/</code>	start pattern search
<code>n</code>	repeat search in the same direction
<code>N</code>	repeat search in the opposite direction
<code>f<C HAR ACT E</code>	find character forward
<code>R></code>	
<code>F<C HAR ACT E</code>	find character backward
<code>R></code>	
<code>:s/foo /bar/g</code>	replace all 'foo' with 'bar' in the current line
<code>:%s/fo o/bar/g</code>	replace all 'foo' with 'bar' in the whole document
<code>g</code>	
<code>:retab</code>	replace tab with spaces

Macros

<code>qa</code>	start recording a macro in the a registry
<code>q</code>	stop recording a macro
<code>@a</code>	replay a macro stored in a registry
<code>:reg a</code>	look at the contents of a registry
<code>O</code>	as a first step of the macro to start on the beginning of the line
<code>j</code>	used at the end of macro to move the cursor to the next line, so that it would be easy to repeat a macro
<code>:20,100 normal @a</code>	apply a macro from an a registry for the lines in range 20 - 100

Windows and Files

<code>:split</code> or <code>:sp</code>	split the window horizontally
<code>:vsplit</code> or <code>:vs</code>	split the window vertically
<code>CTRL + w</code>	navigate between split windows (can be followed with a motion key)
<code>:bn</code>	go to the next file
<code>:bp</code>	go to the previous file
<code>:bd</code>	close the current file

Text alignment

<code>:center</code>	center text (can be selected in visual mode) to the width of 80
<code>:left</code>	left align
<code>:left 5</code>	left align text with 5 spaces indentation
<code>:right</code>	right align text to the width 80

Visual mode

<code>v</code>	start visual mode
<code>V</code>	start linewise visual mode
<code>CTRL + v</code>	start visual block mode (column edit)
<code>gv</code>	reselect the same bit of text
<code>V, select, J</code>	start linewise visual mode, select few lines and join them into a single line with <code>J</code>
<code>~</code>	switch case
<code>c</code>	change text
<code>d</code>	delete text
<code>y</code>	yank
<code>r</code>	replace text
<code>x</code>	delete
<code>I</code>	insert
<code>A</code>	append
<code>J</code>	join
<code>U</code>	make uppercase
<code>u</code>	make lowercase
<code>></code>	shift right
<code><</code>	shift left