Terminally-Incoherent.com Vim Cheat-Sheet

Movements

w	next word
W	next white space delimited word
b	previous word
B	previous space delimited word
e	end of word
E	end of white space delimited word
0 or ^	to the physical beginning of the line
$	to the physical end of the line (ie. the newline char)
g0	to the beginning of the virtual screen line
gm	to the middle of virtual screen line
g$	to the end of the virtual screen line
gk	up screen line
gj	down screen line
fchar	till the next occurrence of character 'char' (inclusive)
Fchar	till the previous occurrence of character 'char' (inclusive)
tchar	till the previous occurrence of character 'char' (exclusive)
Tchar	till the previous occurrence of character 'char' (exclusive)
G	end of file
gg	start of file
gd	go to the definition (first occurrence of the word under cursor)
{}	begging/end of paragraph
()	begging/end of sentence
[{ }]	beginning/end of a code block
[[ ]]	beginning/end of a method
[*]*	beginning/end of a comment block
%	find next open block char ([{ and jump to it's matching close char
'.	go to the line with the latest change
*	search to the next occurrence of the word under cursor
#	search to previous occurrence of the word under cursor
/pattern	search to next occurrence of "pattern"
?pattern	search to previous occurrence of "pattern"
n	repeat last search (*,#,/,?) forward
N	repeat last search (*,#,/,?) backward

In the examples below I indicate a movement specified above as mov

Entering Insertion Mode

i	insert at the cursor
I	insert at the beginning of the line
gI	insert in column 1 of the line
a	append after the cursor
A	append at the end of the line
o	open a blank line below cursor
O	open a blank line above cursor
r	replace character under cursor and exit insert mode
R	insert at the cursor in overstrike mode
cmov	delete (change) movement and enter insert mode
cc	replace entire line
C	delete (change) line and enter insert mode
s	delete character under cursor and enter insert mode
S	delete line under cursor and enter insert mode

Deleting

x	delete char under cursor
X	delete char before cursor
dmov	delete range of movement m
dd	delete current line (text moves up)
D	delete to the end of line
J	join the line below to the current
gJ	append the line below to the end of current line

Block Modifiers

When dealing with blocks of text delimited by parentheses (), brackets [], braces {} or <> you can apply the command to either the contents of the block or the whole thing by following it with:

iblk	inner block (everything inside braces, parenthesis etc..)
ablk	all (whole parenthesized block)

Where 'blk' is),], } or >

Copy/Paste

yy	copy current line
ymov	copy movement m
p	paste after cursor
P	paste before cursor
vipy	visual interior paragraph yank - copy the current paragraph

Visual Mode

v	enter visual mode highlighting 1 char at a time
V	enter visual mode highlighting 1 line at a time
Ctrl+V	enter visual block mode
aw	highlight word
as	highlight sentence
ap	highlight paragraph
ab	highlight a parenthesized block
aB	highlight a {} blocks
ib	highlight contents of a parenthesized block
iB	highlight contents of a {} block

Special

< >	indent left/right
.	repeat last command
u	undo last command
U	undo all changes on the current line
Ctrl+R	redo last undo
Ctrl+O	jump outwards (up-back) to the previous cursor position
Ctrl+I	jump inwards (down-forward) to the previous cursor position

Completion

Ctrl+X	enter completion mode
Ctrl+N	auto complete word in insert and completion mode
Ctrl+P	auto complete word in insert and completion mode
Ctrl+L	auto complete whole line in completion mode
Ctrl+K	dictionary completion in completion mode
Ctrl+F	filename completion in completion mode

Useful Tricks

g~mov	toggle uppercase/lowercase on the range of 'mov'
gumov	lowercase the range of 'mov'
guu	lowercase the current line
gUmov	uppercase the range of 'mov'
gUU	upercase the whole line
g?mov	rot13 on the range of 'mov'

Regexp Replace

:ranges/foo/bar/arg - replace foo with bar in 'range' with

Values of 'range':

%	whole file
number	that particular line
none	apply to current line only

Values of 'arg':

none	apply to first occurrence
g	global (all occurrences)
i	ignore case
I	don't ignore case
c	confirm each substitution
p	print the last line containing substitution
e	ignore errors

Macros

qchar	start recording macro storing it in register 'char'
q	end recording
@char	replay the macro stored in 'char'
:1,10 norm! @char	run the macro stored in 'char' over the 1-10 line range