

Exiting	
:q	Close file
:qa	Close all files
:w	Save
:wq / :x	Save and close file
ZZ	Save and Quit
:q! / ZQ	Quit without checking changes

Editing	
a	Append
A	Append from end of line
i	Insert
o	Next line
O	Previous line
s	Delete char and insert
S	Delete line and insert
C	Delete until end of line and insert
r	Replace one character
R	Enter Replace mode
u	Undo changes
<C-R>	Redo changes

Character	
fc	Go forward to character c
Fc	Go backward to character c

Line	
0 (zero)	Start of line
^	Start of line (after whites pace)
\$	End of line

Operators Usage	
<i>Operators let you operate in a range of text (defined by motion). These are performed in normal mode.</i>	
d	w
Operator	Motion

Words	
b / w	
ge / e	Previous/next end of word

Text Objects Usage	
Text objects let you operate (with an operator) in or around text blocks (objects)	
v	i p
Operator	[i]inside or [a]around Text object

Text Objects	
p	
w	Word
s	Sentence
[({	A [, ,) or { } block
' ' "	A quoted string
b	A block [(
B	A block in [(
t	A XML tag block

Windows	
z{height}-	Resize pane to {height} lines tall
<Cr>	

Navigation	
%	Nearerst/matching { [()] }
[([[<	Previous (or { or <
])	Next
[m	Previous method start
[M	Previous method end

Folds	
zo / zO	Open
zc / zC	Close
za / zA	Toggle
zv	Open folds for thi line
zM	Close all
zR	Open all
zm	Fold more (foldlevel +=1)
zr	Fold less (foldlevel -=1)

Folds (cont)	
zx	Update folds
Uppercase ones are recursive (eg, zO is open recursively).	

Tags	
:tag	Jump to first definition of Cla
Classname	ssname
<C->	Jump to definition
g]	See all definitions
<C-T>	Go back to last tag
<C-O> /	Back/forward
<C-l>	
:tselect	Find definitions of Classnam
Classname	e
:tjump	Find definitions of Classnam
Classname	e (auto-select 1st)

Exiting Insert Mode	
Esc /	Exit insert mode
<C-]	
<C-C>	Exit insert mode, and abort current command

Clipboard	
x	Delete character
dd	Delete line (Cut)
yy	Yank line (Copy)
p	Paste
P	Paste before
"*p / "+p	Paste from system clipboard
"*y / "+y	Paste to system clipboard

Document	
gg	First line
G	Last line
:{number}	Go to line {number}
{number}G	Go to line {number}
{number}j	Go down {number} lines
{number}k	Go up {number} lines



Window	
zz	Center this line
zt	Top this line
zb	Bottom this line
H	Move to top of screen
M	Move to middle of screen
L	Move to bottom of screen

Operators List	
d	Delete
y	Yank
c	Change (delete then insert)
>	Indent right
<	Indent left
=	Autoindent
g~	Swap case
gU	Uppercase
gu	Lowercase
!	Filter through external program

See `:help operator`

Text Objects Examples	
vip	Select paragraph
vipipip	Select more
yip	Yank inner paragraph
yap	Yank paragraph (including newline)
dip	Delete inner paragraph
cip	Change inner paragraph

Marks	
``	Last position of cursor in insert mode
.'	Last change in current buffer
''	Last exited current buffer
`0	In last file edited
"	Back to line in current buffer where jumped from

Marks (cont)	
``	Back to position in current buffer where jumped from
['	To beginning of previously changed or yanked text
']	To end of previously changed or yanked text
`<	To beginning of last visual selection
`>	To end of last visual selection
ma	Mark this cursor position as a
`a	Jump to the cursor position a
'a	Jump to the beginning of the line with position a
d'a	Delete from current line to line of mark a
c'a	Change text from current line to line of a
y'a	Yank text from current position to position of a
:marks	List all current marks
:delm a	Delete mark a
a	
:delm a-d	Delete marks a, b, c, d
:delm abc	Delete marks a, b, c

Case	
~	Toggle case (Case => cASE)
gU	Uppercase
gu	Lowercase
gUU	Uppercase current line (also gUgU)
guu	Lowercase current line (also gugu)

Do these in visual or normal mode.

Visual Mode	
v	Enter visual mode
V	Enter visual line mode
<C-V>	Enter visual block mode
In visual mode	
d / x	Delete selection
s	Replace selection
y	Yank selection (Copy)

See Operators for other things you can do.

Find & Replace	
:%s/foo/o/bar/g	Replace <i>foo</i> with <i>bar</i> in whole document
:%s/foo/o/bar/gc	Replace <i>foo</i> with <i>bar</i> in whole document interactively

Search	
n	Next matching search pattern
N	Previous match
*	Next whole word under cursor
#	

Operators Examples	
Combine operators with motions to use them.	
dd	(repeat the letter) Delete current line
dw	Delete to next word
db	Delete to beginning of word
2dd	Delete 2 lines
diP	Delete a text object (inside paragraph)
(in visual mode) d	Delete selection

See `:help motion.txt`



Diff

gvimdiff file1 file2 [file3]

Diff

gvimdiff file1 file2 [file3] See differences between files, in HMI

Tab Pages

:tabedit [file] Edit file in a new tab

:tabfind [file] Open file if exists in new tab

:tabclose Close current tab

:tabs List all tabs

:tabfirst Go to first tab

:tablast Go to last tab

:tabn Go to next tab

:tabp Go to previous tab

Miscellaneous

. Repeat last command

]p Paste under the current indentation level

:set ff=unix Convert Windows endings to Unix line endings

Command Line

<C-R><- C-W> Insert current word into the command line

<C-R>" Paste from " register

<C-X><- C-F> Auto-completion of path in insert mode

Text Alignment

:center [width]

:right [width]

:left

See :help formatting

Counters

<C- A> Increase number

<C- X> Decrement number

Calculator

<C-R>=- Shows the result of the division: ' 64 '

Do this in insert mode.

Exiting With an Error

:cq

:cquit

Works like :qa, but throws an error. Great for aborting Git commands.

Spell checking

:set spell Turn on US English spell checking
spelllang=en_us

]s Move to next misspelled word after the cursor

[s Move to previous misspelled word before the cursor

z= Suggest spellings for the word under/after the cursor

zg Add word to spell list

zw Mark word as bad/misspelling

zu / C-X (Insert Mode) Suggest words for bad word under cursor from spellfile

See :help spell

