Vim Modal text editor

START HERE

- **i** — Insert before cursor
- **Esc** — Back to Normal mode
- **:w** — Save
- **:q** — Quit
- **d** › **d** — Delete line
- **u** — Undo
- **/** — Search forward
- **.** — Repeat last change

Normal commands `i a o` → **Insert** typing `Esc` → **Normal** commands `v V Ctrl+V` → **Visual** select

Normal mode is the hub

Tab page a layout of windows
Window a view onto a buffer
Buffer a file in memory

CONCEPTS

Vim has modes. Normal mode is home: keys are commands. Press Esc to return to it.

Commands read like a sentence: operator + motion, e.g. d w deletes a word, c i " changes inside quotes.

Prefix a count to repeat: 3 d d deletes three lines, d 2 w two words.

Key sequences are typed one after another, not held together: d w means press d, then w.

MODES & FILES

i Insert before cursor

Esc Back to Normal mode

Ctrl+[or Ctrl+C also leave Insert mode.

v / **V**
Visual: characters / lines

:w Save file

:q Quit window

:wq / **ZZ** Save and quit

ZZ only writes if something changed.

:q! Quit and discard changes

:e › *<file>* Edit another file

:!<cmd>
Run a shell command

MOTIONS

h / **j** / **k** / **l**
Left / down / up / right

w / **b** / **e**
Next word / previous / end

W / **B** / **E**
Same, ignoring punctuation

0 / **^** / **$**
Line start / first char / end

g › **g** / **G** First / last line

{ / **}**
Paragraph back / forward

Ctrl + **D** / **Ctrl** + **U**
Half page down / up

f › *<c>* / **t** › *<c>*
To / before character on line

Repeat with ; and reverse with ,

; / **,** Repeat / reverse f t

% Jump to matching bracket

MOTIONS (cont.)

H / **M** / **L**
Top / middle / bottom of screen

<n> › **G** Go to line n

z › **z** / **z** › **t** / **z** › **b**
Cursor line to middle / top / bottom

m › *<a-z>* Set mark

` › *<a-z>* Jump to mark

Ctrl + **O** / **Ctrl** + **I**
Older / newer jump position

INSERT

a Append after cursor

I / **A** Insert at line start / end

o / **O**
Open line below / above

s / **S** Substitute char / line

Ctrl + **R** › *<reg>*
Insert register contents

Ctrl + **N** / **Ctrl** + **P**
Complete word next / previous

OPERATORS

d › **d** Delete line

d › *<motion>* Delete (cut)

D Delete to end of line

c › *<motion>*
Change: delete and insert

c › **c** / **C**
Change line / to end

c › **w** Change word

cw acts like ce when on a non-blank.

y › *<motion>* Yank (copy)

y › **y** Yank line

x / **X**
Delete char under / before

r › *<c>*
Replace one character

u Undo

Ctrl + **R** Redo

. Repeat last change

Works for changes and puts, not for Ex commands.

p / **P**
Put after / before cursor

> › **>** / **<** › **<**
Indent / outdent line

J Join lines with a space

OPERATORS (cont.)

Ctrl + **A** / **Ctrl** + **X**
Increment / decrement number

Operator + motion

Keys	Effect
`d i w`	delete word
`c i "`	change inside quotes
`y a p`	yank paragraph
`d t ,`	delete up to comma
`> i {`	indent block

TEXT OBJECTS

i › **w** / **a** › **w**
Word / word with space

i › **"** / **a** › **"**
Inside / around double quotes

i' and i` work the same way.

i › **(** / **a** › **(**
Inside / around parentheses

b is a synonym: ib, ab.

i › **{** / **a** › **{**
Inside / around braces

Text objects work from anywhere inside; no need to move to the start first. Use them after an operator or in Visual mode.

VISUAL MODE

v Characterwise Visual

V Linewise Visual

Ctrl + **V** Blockwise Visual

g › **v**
Reselect last Visual area

Ctrl + **V** › **I**
Insert on all block lines

Press Esc to apply to every line.

: Ex command on selection

Vim fills in '<,'> for the range.

Select first, then act: Visual mode takes the same operators (d y c > = and so on).

SEARCH & REPLACE

/ › *<pat>* Search forward

? › *<pat>* Search backward

n / **N** Next / previous match

***** / **#**
Search word under cursor

g and g# skip the whole-word match.*

:noh Clear search highlight

:s/<a>/<b>/
Replace first on line

SEARCH & REPLACE (cont.)

:%s/<a>/<b>/g
Replace in whole file

:%s/<a>/<b>/gc
Replace with confirmation

y yes, n no, a all, q quit, l last.

:g/<pat>/d
Delete matching lines

REGISTERS & MACROS

" › *<a-z>* › **y**
Yank into named register

" › *<a-z>* › **p**
Put from named register

System clipboard

Keys	Effect
`"+y`	yank to clipboard
`"+p`	put from clipboard

" › **0** › **p** Put last yank

Deletes do not overwrite register 0.

q › *<a-z>*
Record macro, q again stops

@ › *<a-z>* Play macro

@ › **@** Replay last macro

WINDOWS & BUFFERS

Ctrl + **W** › **s** / **v**
Split horizontal / vertical

Ctrl + **W** › **h/j/k/l**
Move to window left/down/up/right

:ls List buffers

:b › *<n>* Go to buffer n

Ctrl + **^** Toggle alternate file

FOLDS & MISC

:set › **nu** Show line numbers

:set › **hls** › **is**
Highlight / incrementally search

RECIPES

Open from the shell

```
vim +42 <file>  # at line 42
vim +/<pat> <file>
vim -d a b  # diff
find . -name '*.c' | vim -
```

Rename in file, confirm each

```
:%s/\<old\>/new/gc
```

Edit many lines at once

```
Ctrl+V  # select column
j j j  # extend down
I // Esc  # prefix all
```