

# Neovim Keymaps Quick Reference

Leader Key: , (comma) | Updated: 2025-11-08 | Config: ~/.config/nvim

---

## Essential Commands

Category	Keymap	Action	Category	Keymap	Action
<b>Editor</b>	,l	Save file	<b>LSP Nav</b>	gd	Go to definition
	,f	Format code		gD	Go to declaration
<b>File Tree</b>	,n	Toggle Neo-tree		gi	Go to implementation
	,F	Reveal file		gr	Find references
<b>Telescope</b>	,e	Focus tree		K	Hover docs
	,t	Find files		<C-k>	Signature help
<b>Git</b>	,g	Live grep	<b>LSP Edit</b>	1gD	Type definition
	,o	Search buffer		,rn	Rename symbol
	;	Buffers		,ac	Code actions
	,j	Find (word)		[d	Previous diagnostic
	,k	Grep (word)		]d	Next diagnostic
	,vg	LazyGit UI		,dd	Diagnostic list
	,vd	Diff view	<b>AI</b>	<C-,>	Claude Code
	,vh	File history		Tab	Copilot (insert)
	,hs	Stage hunk	<b>Comment</b>	,c<space>	Toggle comment
	,hp	Preview hunk			
	]c / [c	Next/Prev hunk			

---

## Neo-tree Commands (when focused)

Navigation	File Ops		Window	Search/Git		
<CR>	Open a file	Add s file	V- / split			Fuzzy find
<Space>	Toggle A node	Add S dir	H- H split			Toggle hid- den
<BS>	Parent d dir	Delete t	New [g tab			Prev git change
.	Set r root	Rename P	Preview ]g			Next git change
R	Refresh y	Copy q	Close </>			Switch source
?	Help p	Paste z/Z	Close/Expand all			

---

## Git Workflow (LazyGit + Gitsigns + Diffview)

Primary (,v)	Hunks (,h)	Navigation		
,vg	LazyGit UI	,hs	Stage hunk	]c
,vs	Git status	,hr	Reset hunk	[c
,vf	LazyGit (file)	,hS	Stage buffer	<b>Inside LazyGit</b>
,vd	Diff view	,hR	Reset buffer	<b>Space</b>
,vc	Close diff	,hu	Undo stage	c
,vh	File history	,hp	Preview hunk	P
,vr	Refresh	,hb	Blame line	p
,vb	Toggle blame	,hd	Diff this	?
				Help / q Quit

**Diff View:** <Tab>/<S-Tab> cycle files · - stage/unstage · gf goto file · [x/]x conflicts · ,co/,ct choose ours/theirs

---

## Tree-sitter Text Objects

**Usage:** <operator><text-object> (e.g., daf = delete entire function, vif = select function body)

Around	Inside	Description	Around	Inside	Description
af	if	Function	ac	ic	Class
al	il	Loop	ai	ii	Conditional
aa	ia	Parameter/Argument			

**Navigation:** ]f/[f (next/prev function start) · ]F/[F (function end) · ]c/[c (class start) · ]C/[C (class end)

**Incremental Selection:** gnn (init) → grn (expand node) → grc (expand scope) → grm (shrink)

---

## Completion (nvim-cmp) - Insert Mode

Key	Action	Key	Action	Key	Action
<C-b>	Scroll docs up	<C-f>	Scroll docs down	<C-o>	Manual trigger
<C-e>	Abort	<CR>	Confirm	<Tab> / <S-Tab>	Next/Prev item

---

## Telescope Picker Controls

Key	Action	Key	Action
<C-n> / <C-p>	History next/prev	<C-d>	Delete buffer

---

## Essential Vim Motions

Movement	Editing		Visual/Search		
h/j/k/l	←↑→	i/a	Insert be- fore/after	v/V	Visual/Line
w/b/e	Word fwd/back/end	I/A	Line start/end insert	<C-v>	Visual block

Movement	Editing		Visual/Search		
0/^/\$	Line start/first/end	o/o	New line below/above	/ / ?	Search fwd/back
gg/G	File start/end	dd/yy/p	Delete/Yank/Paste		Next/Prev
{/}	Para prev/next	u/<C-r>	line Undo/Redo/#		match
%	Match bracket	.	Repeat change	>> / <<	Search word
<C-d>/<C-u>	Scroll half page	<C-o>/<C-i>	Jump back/fwd		fwd/back
					Indent/Unindent

## Window & Tab Management

Windows	Tabs		
<C-w>s	Split horizontal	:tabnew	New tab
<C-w>v	Split vertical	gt/gT	Next/Prev tab
<C-w>c	Close window	<num>gt	Go to tab #
<C-w>h/j/k/l	Navigate windows	:tabclose	Close tab
<C-w>=	Equalize sizes		

## Marks

Key	Action	Key	Action
ma	Set mark 'a'	`a	Jump to mark 'a'
'a	Jump to line of mark 'a'	:marks	List marks

## Quick Command Reference

WORKFLOW	NAVIGATION	CODE EDITING
,l → ,f	Save & format	gd Goto def
,t	Find files	K Docs
,g	Search code	gr References
,n	Toggle tree	[d / ]d Diagnostics
,vg	LazyGit UI	]c / [c Next/Prev hunk
TEXT OBJECTS	SELECTION	GIT HUNKS
vaf	Select function	gnn → grn Smart select
dif	Delete fn body	grm Shrink
vac	Select class	grc Scope

Config Files: lua/config/keymaps.lua · lua/config/lsp/keymaps.lua · lua/plugins/\*.lua