NEOVIM MOVEMENT CHEATSHEET

Be fast as fuck

Basic Motions (The Foundation)

Character & Line Movement

```
h j k l        ← ↓ ↑ →  (left, down, up, right)
w              Next word start
W              Next WORD start (ignores punctuation)
b              Previous word start
B              Previous WORD start
e              Next word end
E              Next WORD end
ge             Previous word end

0              Start of line
^              First non-blank character
$              End of line
g_             Last non-blank character
```

Supercharged Line Movement

```
f{char}        Find next {char} on line (forward)
F{char}        Find previous {char} on line (backward)
t{char}        Till next {char} (cursor before char)
T{char}        Till previous {char}
;              Repeat last f/F/t/T motion
,              Repeat last f/F/t/T motion (opposite direction)

3fa            Find 3rd 'a' on line
df)            Delete forward to )
ct,            Change till comma
```

Vertical Movement (Fast AF)

Screen-Relative

```
H              Top of screen (High)
M              Middle of screen
L              Bottom of screen (Low)
zt             Put current line at top
zz             Center current line
zb             Put current line at bottom
```

Jump Around

```
gg             Top of file
G              Bottom of file
50G            Go to line 50
50gg           Go to line 50
:50            Go to line 50

{              Previous paragraph/block
}              Next paragraph/block
[[             Previous section/function
]]             Next section/function
```

Percentage-Based

```
50%            Jump to 50% through file
```

```
25%             Jump to 25% through file
```

Search Movement (Lightning Fast)

Basic Search

```
/{pattern}      Search forward
?{pattern}      Search backward
n               Next match
N               Previous match
*               Search forward for word under cursor
#               Search backward for word under cursor
g*              Search forward (partial match)
g#              Search backward (partial match)
```

Smart Search Combos

```
*n              Find word, go to next occurrence
cgn             Change next search match
.               Repeat change on next match
```

Code-Specific Movement

Parentheses & Brackets

```
%               Jump to matching bracket/paren
[(              Jump to previous (
])              Jump to next )
[{              Jump to previous {
]}              Jump to next }
```

Functions & Methods (Treesitter)

```
]m              Next method start
[m              Previous method start
]M              Next method end
[M              Previous method end
```

Advanced Motions

Text Objects (combine with d, c, y, v)

```
iw              Inner word
aw              A word (includes space)
iW              Inner WORD
aW              A WORD

is              Inner sentence
as              A sentence
ip              Inner paragraph
ap              A paragraph

i( i) ib        Inner parentheses
a( a) ab        A parentheses (includes parens)
i{ i} iB        Inner braces
a{ a} aB        A braces (includes braces)
i[ i]           Inner brackets
a[ a]           A brackets (includes brackets)
i< i>           Inner angle brackets
a< a>           A angle brackets
i' i" i`        Inner quotes
a' a" a`        A quotes (includes quotes)
```

```
it              Inner tag (HTML/XML)
at              A tag (HTML/XML)
```

Text Object Examples

```
diw             Delete inner word
ci"             Change inside quotes
da(             Delete around parentheses (includes parens)
vi{             Visual select inside braces
yap             Yank a paragraph
```

Marks & Jumps

Quick Marks

```
m{a-z}          Set mark (local to buffer)
m{A-Z}          Set mark (global across files)
'{mark}         Jump to mark line
`{mark}         Jump to exact mark position

''              Jump back to previous position (line)
``              Jump back to previous position (exact)
'.              Jump to last change
`^              Jump to last insert position
```

Jump List

```
Ctrl-o          Jump to older position
Ctrl-i          Jump to newer position
:jumps          Show jump list
```

Change List

```
g;              Go to older change position
g,              Go to newer change position
:changes        Show change list
```

Window & Buffer Movement

Window Navigation

```
Ctrl-w h/j/k/l  Move to left/down/up/right window
Ctrl-w w        Cycle through windows
Ctrl-w p        Previous window
```

Buffer Navigation (From Your Config)

```
<leader><leader> Find existing buffers (Telescope)
```

Scroll Without Moving Cursor

```
Ctrl-e          Scroll down one line
Ctrl-y          Scroll up one line
Ctrl-d          Scroll down half page
Ctrl-u          Scroll up half page
Ctrl-f          Scroll down full page
Ctrl-b          Scroll up full page
```

LSP-Powered Movement (From Your Config)

```
grd             Go to definition
gri             Go to implementation
grt             Go to type definition
```

```
grr             Find references
gO              Document symbols (outline)
gW              Workspace symbols

Ctrl-o          Jump back after goto
```

Telescope Movement (From Your Config)

```
<leader>sf      Search files
<leader>sg      Live grep (search in files)
<leader>sw      Search word under cursor
<leader>sn      Search nvim config files
<leader>/       Fuzzy search in current buffer
```

Speed Combos (Muscle Memory These)

Navigation

```
3w              Move 3 words forward
5j              Move 5 lines down
10k             Move 10 lines up
4f;             Find 4th semicolon on line
```

Edit + Move

```
d3w             Delete 3 words
c$              Change to end of line
y5j             Yank 5 lines down
>ap             Indent a paragraph
=i{             Auto-indent inside braces
```

Find + Change

```
/foo<CR>        Search for 'foo'
cgn             Change the match
.               Repeat on next match
n.              Skip one, change next
```

Word Replace Workflow

```
*               Search for word under cursor
cgn             Change next occurrence
.               Repeat change on next (mash this)
n               Skip one if needed
```

Pro Tips

1. **Count everything**: Most motions accept counts (3w, 5j, 2f;)
2. **Combine operators**: d/c/y + motion = power (d3w, cf), c4j)
3. **Use relative line numbers**: Jump directly with 12j or 8k
4. **Marks for bookmarks**: Use marks for frequent positions
5. **Jump list is your friend**: Ctrl-o to backtrack after jumping around
6. **Learn text objects**: They work everywhere (dab, ci", ya{)
7. **f/t are underrated**: Fastest way to move on a line
8. **Search with /**: Often faster than motion for long distances
9. **Use * for refactoring**: * → cgn → . → . → .
10. **Treesitter motions**:]m and [m for function navigation

Common Patterns

Navigate Function

```
[[              Jump to previous function
```

```
]m                 Jump to next method
%                  Jump between braces
```

Refactor Variable

```
*                  Find all occurrences
cgn                Change next match
.                  Repeat (keep pressing)
```

Navigate Changes

```
g;                 Go to last change
`.                 Jump to last change
Ctrl-o             Jump back
```

Quick Edits

```
ciw                Change inner word
da"                Delete around quotes
yi}                Yank inside braces
```

Practice these daily. Speed comes from muscle memory, not memorization.