

Omarchy Power Guide

by Richard J. Young, Ph.D.

The shortcuts, terminal, programming and AI workflows that matter most, and why each one earns a spot in muscle memory. Taken from the live keybindings on this machine (`omarchy menu keybindings --print`).

The one idea behind Omarchy: your hands stay on the keyboard. `SUPER` (the Windows/⌘ key) is the "OS key": `SUPER` + a key does something to the desktop, and adding `SHIFT`, `CTRL` or `ALT` gives a variation of it. You won't find a start menu, a taskbar to click, or overlapping windows to drag around. Learn about 15 bindings and everything else follows the same pattern.

◆ The grammar: how modifiers change a key

SUPER
Act on the desktop: go, open, focus.
`SUPER+3` → go to WS 3

+ SHIFT
Move the window, or open the "bigger" version.
`SUPER+SHIFT+3` → take window to WS 3

+ ALT
Quiet or fine-grained: no jump, small steps.
`SUPER+SHIFT+ALT+3` → send it silently

+ CTRL
System panels and toggles, or bigger steps.
`SUPER+CTRL+W` → Wi-Fi

🗦 Your keyboard as SUPER sees it

Esc
System

1
WS 1

2
WS 2

3
WS 3

4
WS 4

5
WS 5

6
WS 6

7
WS 7

8
WS 8

9
WS 9

0
WS 10

-
Resize

=
Resize

⌘
Opacity

Tab
Next WS

Q

W
Close

E

R

T
Float

Y

U

I

O
Pop out

P
Pseudo

[

]

\\

Caps

A

S
Scratch

D

F
Full

G
Group

H
Dictate

J
Split

K
Keys!

L
Layout

;

'

↵
Terminal

Shift

Z

X
Cut

C
Copy

V
Paste

B

N

M

,
Notif.

.
Scale

/

Shift

Ctrl

SUPER
the OS key

Alt

Space
Launcher — open anything

←
Focus

↓
Focus

↑
Focus

→
Focus

Window control

Workspaces

Apps & text

Menus & system

Everything shown is **SUPER + key**

★ If you learn only five

SUPER + SPACE
Launch anything. Faster than any icon.

SUPER + K
Searchable list of every shortcut. Your safety net.

SUPER + RETURN
Terminal. The heart of the system.

SUPER + W
Close the window. No hunting for the x.

SUPER + 1...9
Jump to a workspace. One task per screen.

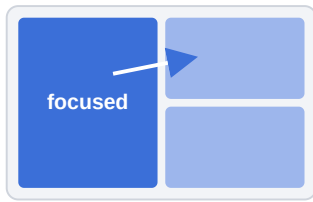
1 Menus & discovery

<code>SUPER + SPACE</code>	Launcher / Omarchy menu	Type a few letters to open any app, so you never need to know where anything is installed.
<code>SUPER + K</code>	Keybindings	A searchable cheat sheet of every binding. Use it whenever you forget one, until you no longer need it.
<code>SUPER + ESCAPE</code>	System menu	Lock, suspend, restart and shut down from one place, all from the keyboard.
<code>SUPER ALT + SPACE</code>	Apps menu	Browse the installed apps when you don't remember an app's name.
<code>SUPER SHIFT CTRL + SPACE</code>	Theme menu	Switches the theme across the terminal, bar, editor and wallpaper in one go. Themes are Omarchy's signature feature.
<code>SUPER CTRL + SPACE</code>	Background switcher	Changes the wallpaper and leaves the theme as it is.
<code>SUPER SHIFT + SPACE</code>	Toggle top bar	Hides the bar so you get every pixel for focused work or a presentation.

2 Apps you'll open all day

<code>SUPER + RETURN</code>	Terminal	Omarchy is built around the terminal. Opening one takes a single keystroke, so you'll do it often.
<code>SUPER ALT + RETURN</code>	Tmux	Sessions that keep running if you close the window. Good for long jobs and SSH.
<code>SUPER SHIFT + RETURN</code> / <code>+ B</code>	Browser	Add <code>ALT</code> for a private window.
<code>SUPER SHIFT + F</code>	File manager	Add <code>ALT</code> to open it in the terminal's current folder, so you don't have to navigate there again.
<code>SUPER SHIFT + N</code>	Editor	Opens your default code editor (Neovim out of the box).
<code>SUPER SHIFT + O</code>	Obsidian	Notes are one keystroke away, so ideas get written down before you lose them.
<code>SUPER SHIFT + SLASH</code>	Passwords	Opens the password manager without leaving what you're doing.
<code>SUPER SHIFT CTRL + A</code>	Agent	Launches your AI coding agent. <code>SUPER SHIFT + A</code> opens ChatGPT.
<code>SUPER SHIFT + D</code>	Docker (lazydocker)	A TUI for containers and logs, instead of memorising <code>docker ps</code> flags.

3 Windows: tiling without thinking



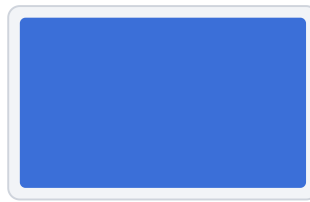
Auto-tiled

SUPER+arrows moves focus



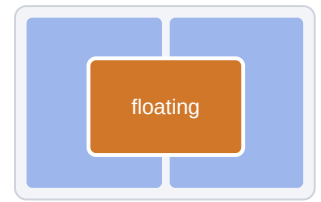
SUPER + J

flip the split direction



SUPER + F

full screen for focus



SUPER + T / O

float it / float & pin on top

You never place windows yourself. The tiler arranges them, and you only change the layout.

SUPER + ← → ↑ ↓

Move focus

Directional focus is faster than Alt-Tab when you can see the window you want.

SUPER SHIFT + arrows

Swap windows

Rearranges the layout while the tiling keeps everything neatly placed.

SUPER + W

Close window

Works the same in every app. **CTRL ALT + DEL** closes all windows for a clean slate.

SUPER + F

Full screen

Focus mode. **SUPER ALT + F** fills the width but keeps the bar. **SUPER CTRL + F** is full screen inside the tile.

SUPER + T

Float / tile

For the odd dialog or calculator that looks wrong when it's tiled.

SUPER + O

Pop out (float & pin)

Keeps a video or reference window on top on every workspace.

SUPER + J

Toggle split

Switches a pair of windows between side by side and stacked.

SUPER + L

Toggle workspace layout

Switches between layout modes (dwindle and scrolling) for the current workspace.

SUPER + MINUS / EQUAL

Resize

Add SHIFT for vertical resizing. ALT resizes a little, CTRL a lot. Or hold SUPER and drag with the right mouse button.

SUPER + G

Group (tabs)

Stacks windows as tabs in one tile. **SUPER ALT + TAB** cycles through them. Useful on small laptop screens.

SUPER + BACKSPACE

Transparency

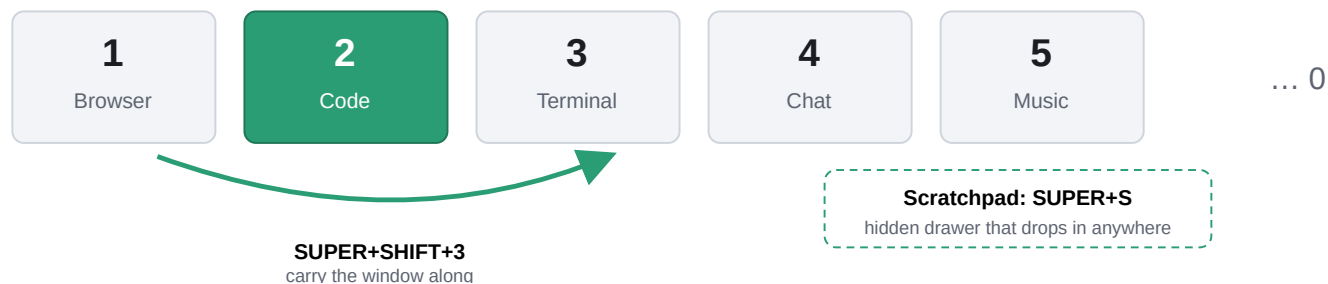
See what's behind a window. **SHIFT** toggles the gaps between windows.

ALT + TAB

Next window

The familiar habit still works. **CTRL ALT + TAB** jumps between monitors.

4 Workspaces: one context per number



Give each workspace a fixed job. Your fingers learn where things are, so you stop searching for windows.

SUPER + 1...9, 0	Switch workspace	Give each task a fixed home, for example 1 for the browser, 2 for code, 3 for chat, and switching becomes instant.
SUPER SHIFT + 1...9	Move window there	Sends the window and follows it. Add ALT to send it without following, so you stay where you are.
SUPER + TAB	Next workspace	SHIFT goes to the previous one. CTRL jumps back to the one you just left, which is useful for flipping between two tasks.
SUPER + S	Scratchpad	A hidden workspace you can summon from anywhere. SUPER ALT + S puts a window there. Good for music or a quick terminal.
SUPER SHIFT ALT + arrows	Workspace → monitor	Moves a whole workspace to your external screen at once.

5 Clipboard, capture & text

SUPER + C / V / X	Universal copy/paste/cut	Works the same in terminals and GUI apps, so you don't have to remember CTRL+SHIFT+C for terminals.
SUPER CTRL + V	Clipboard history	Brings back something you copied 10 minutes ago.
PRINT	Screenshot	Select a region, then annotate or copy it. ALT + PRINT records the screen.
SUPER CTRL + PRINT	OCR screenshot	Copies text out of images, videos or error dialogs you can't select.
SUPER + PRINT	Color picker	Grabs the hex value of any pixel on screen.
SUPER CTRL + E	Emoji picker	Search for an emoji by name and paste it.
SUPER + H / F9	Dictation	Voice to text in any app. SUPER + H toggles it (your own binding). Hold F9 for push-to-talk.

6 System & quick toggles

<code>SUPER CTRL + L</code>	Lock	Lock the screen every time you step away.
<code>SUPER CTRL + W / B / A</code>	Wi-Fi / Bluetooth / Audio	Opens each settings panel directly from the keyboard.
<code>SUPER CTRL + D / P</code>	Display / Power	Monitor setup and power profiles (battery saver or performance).
<code>SUPER CTRL + T</code>	Activity	Shows what's using your CPU and RAM right now.
<code>SUPER CTRL + N</code>	Nightlight	Warmer colours in the evening, which is easier on the eyes.
<code>SUPER CTRL + I</code>	Idle-lock toggle	Stops the screen locking during a long build, a video or a presentation.
<code>SUPER + COMMA</code>	Dismiss notification	<code>SHIFT</code> clears all. <code>CTRL</code> turns on do-not-disturb. <code>ALT</code> opens the last notification's action.
<code>SUPER CTRL + Q</code>	Calculator	Quick maths without opening an app.
<code>SUPER CTRL + R</code>	Set reminder	"Check the deploy in 20 min", set with a few keystrokes.
<code>SUPER CTRL + Z</code>	Zoom in	Magnifies the screen for presentations or small text. <code>SUPER CTRL ALT + Z</code> resets it.

7 Terminal commands worth knowing

<code>omarchy update</code>	Update everything	Updates Omarchy and the Arch packages in one step. Run it weekly. It takes a snapshot first, so a bad update can be rolled back.
<code>omarchy snapshot create</code>	Snapshot	Take one before risky experiments. <code>omarchy snapshot restore</code> undoes them.
<code>omarchy pkg install</code>	Find & install	Fuzzy-search the Arch repos. <code>omarchy pkg aur install</code> searches the AUR.
<code>omarchy theme list / set <name></code>	Themes from the CLI	Can be scripted, for example a light theme by day and a dark one at night.
<code>omarchy font list</code>	Fonts	Changes the font everywhere at once.
<code>omarchy restart audio / bluetooth</code>	Fix it fast	Fixes a stuck headset or missing sound without rebooting.
<code>omarchy refresh hyprland</code>	Reset config	Restores the Hyprland config from the defaults if you break it. Your old version is backed up.
<code>omarchy menu keybindings --print</code>	Dump bindings	Pipe it into <code>grep</code> to answer "what does SUPER+X do?"
<code>omarchy commands</code>	Discover everything	Lists every Omarchy command. Add <code>--help</code> to any group, e.g. <code>omarchy toggle --help</code> .
<code>omarchy debug</code>	Diagnostics	Collects the system info to include when you report a bug or ask for help.

8 Ghostty: your terminal



The screenshot shows a Ghostty terminal window with two tabs: 'tab 1 · Alt+1' and 'tab 2 · Alt+2'. The first tab contains a terminal session with the prompt '~ /proj › nvim .' and a blue border. The second tab is split into two panes: the top pane shows '› npm run dev' and the bottom pane shows '› cx (claude)'. A '+ Ctrl+Shift+T' button is visible between the tabs.

Split right

Ctrl+Shift+O

Split down

Ctrl+Shift+E

Move between splits

Ctrl+Alt+arrows

New tab / go to tab

Ctrl+Shift+T / Alt+1...9

Search scrollbar

Ctrl+Shift+F

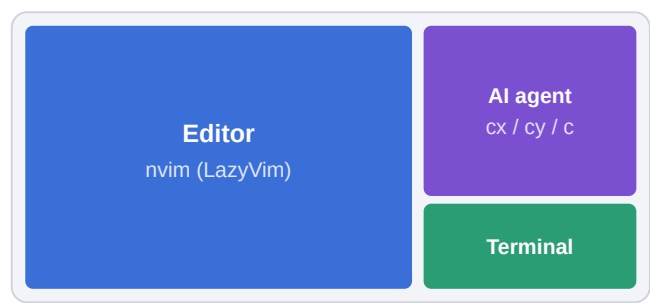
Jump prompt ↑ / ↓

Ctrl+Shift+PgUp / PgDn

Quick splits and tabs inside one window. For sessions that survive closing the window, use tmux (SUPER ALT + RETURN).

<code>SUPER + C / V</code>	Copy / paste	Omarchy's universal copy and paste work in Ghostty as well. <code>CTRL SHIFT + C / V</code> is the terminal's own version.
<code>CTRL + = / - / 0</code>	Font size	Zoom in for demos or pairing. Your font is set to 8pt, so this helps. <code>CTRL + 0</code> resets it.
<code>CTRL SHIFT + J</code>	Screen → file path	Saves everything on screen to a file and pastes its path. Useful for giving a long error log to an AI agent.
<code>SHIFT + HOME / END</code>	Top / bottom of scrollbar	Jumps to the start of a long build log, or back to the bottom.
<code>SHIFT + ENTER</code>	Newline in TUIs	Omarchy sets this up so AI CLIs (Claude, Codex) take multi-line prompts without sending early.
<code>CTRL SHIFT + ,</code>	Reload config	Edit <code>~/.config/ghostty/config</code> and see the change straight away. Colours follow the Omarchy theme automatically.
<code>CTRL SHIFT + I</code>	Inspector	Shows which key codes and escape sequences the terminal receives, for when a binding "doesn't work".

9 Programming: the dev loop



The diagram shows a layout with three panes: a large blue pane on the left labeled 'Editor' with 'nvim (LazyVim)' below it; a purple pane on the top right labeled 'AI agent' with 'cx / cy / c' below it; and a green pane on the bottom right labeled 'Terminal'.

Terminal tricks Omarchy sets up

`n` open nvim in this folder
`cd proj` zoxide jumps to it from anywhere
`ff / eff` fuzzy-find a file / open it
`lt` tree view with git status
`g · gcm "msg"` git · commit
`t` attach to (or create) tmux "Work"
`mise use node@lts` per-project versions

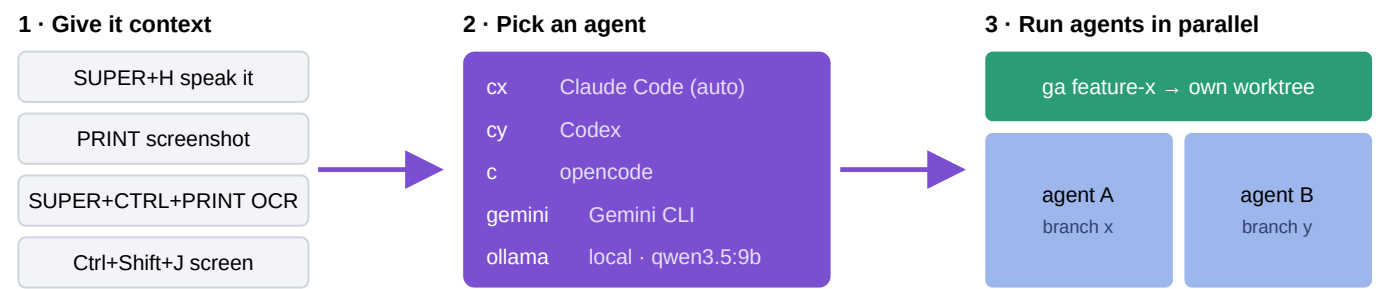
`tdl cx` → a full IDE in one command

One command lays out editor, AI and shell side by side in tmux. `hdl` does the same in Herdr.

<code>tdl cx / tdl cx cy</code>	Tmux dev layout	Opens editor, AI and terminal panes in one go (or two AIs side by side). Run it in tmux. <code>tds</code> adds a live diff pane.
<code>tdlm cx</code>	Layout per sub-project	One tmux window per subfolder. Useful for monorepos or several repos.

<code>SPACE</code> in nvim	LazyVim leader	Press it and wait. A menu (which-key) shows every command, so you don't have to memorise them.
<code>SPACE SPACE</code> · <code>SPACE /</code>	Find file · grep project	The two moves you'll use most in nvim.
<code>gd</code> · <code>K</code> · <code>SPACE c a</code>	Definition · docs · fix	Go to definition, hover docs and code actions come from the LSP, as in VS Code.
<code>SPACE g g</code> / <code>lazygit</code>	Git UI	Stage lines, commit, rebase and push without typing git flags.
<code>SUPER SHIFT + D</code>	lazydocker	Container logs, restarts and stats in one TUI.
<code>CTRL SPACE</code> (tmux)	Tmux prefix	Common moves need no prefix: <code>ALT + ENTER</code> splits, <code>ALT + 1...9</code> switches windows, <code>CTRL ALT + arrows</code> changes pane. <code>SUPER ALT + K</code> lists them all.
<code>omarchy install</code>	Dev environments	One-shot setup for languages, databases and editors (VS Code is already installed). Tab-complete to see the options.

10 AI: agents on tap



Worktrees give each agent its own copy of the repo, so they never overwrite each other's changes. `gd` cleans up.

Voice, screenshots and terminal output feed the agent. Worktrees let several agents work at once.

<code>SUPER SHIFT CTRL + A</code>	Launch agent	Opens your default coding agent in a new terminal from anywhere. In the shell, <code>a</code> runs it in the current window.
<code>cx</code> · <code>cy</code> · <code>c</code>	Claude · Codex · opencode	Each starts with auto-approval turned on, so the agent works without asking at every step. Use them in repos you trust.
<code>SUPER + H</code> · <code>F9</code>	Dictate prompts	Speaking is about three times faster than typing, and long, detailed prompts get better results.
<code>ga <branch></code> / <code>gd</code>	Worktree per agent	Creates a branch in a sibling folder, trusts its mise config and <code>cd</code> s into it. <code>gd</code> removes it when you're done.
<code>tsl 4 cx</code>	Agent swarm	Four panes running the same agent, for example one per task or one per worktree. (Must be run in tmux. <code>hsl</code> is the Herdr version.)
<code>ollama run qwen3.5:9b</code>	Local model	Private, offline and free. Good for quick questions or sensitive data that shouldn't leave the machine.

<code>SUPER SHIFT + A</code> · <code>+ALT</code>	ChatGPT · Grok web apps	Chat assistants one keystroke away for non-coding questions.
<code>omarchy agent prompt "..."</code>	One-shot ask	Starts the default agent with a prompt already filled in. <code>omarchy default agent claude</code> changes which agent is the default.

11 Make it yours

- **Your own shortcuts** go in `~/.config/hypr/bindings.lua`, e.g. `o.bind("SUPER + SHIFT + R", "SSH", "alacritty -e ssh my-server")`. To change a default binding, run `hl.unbind("...")` on it first.
- **Settings are split by topic** in `~/.config/hypr/`: `monitors.lua`, `input.lua` (keyboard layout, touchpad), `looknfeel.lua` (gaps, borders, animations) and `autostart.lua`.
- **Why edit these files and not the defaults?** Omarchy updates replace the defaults but leave your files alone, so your changes survive every `omarchy update`.
- **Habit tip:** for one week, use `SUPER + SPACE` instead of the mouse to open anything, and press `SUPER + K` whenever you're stuck. After that the keyboard will be your default.

Generated 25 Sep 2026 from the keybindings configured on this Omarchy install. Hardware keys (volume, brightness, media) work as you'd expect. Hold `ALT` for finer steps and `SHIFT` to go straight to the minimum or maximum.