Windows Terminal

Tabbed Windows terminal

START HERE

Ctrl + Shift + T
New tab

Ctrl + Shift + P
Command palette

Ctrl + Shift + Space
Pick a profile

Ctrl + ,
Settings

Alt + Shift + Plus
Split right

Alt + Shift + Minus
Split down

Alt + Arrows
Move between panes

Ctrl + Shift + W
Close pane

Window named or numbered; wt -w targets it

Tab one entry in the tab row

Pane one profile's shell in a split

CONCEPTS

A profile is one shell plus its look: PowerShell, Command Prompt, each WSL distro. New tabs and panes open a profile.

Keys shown are the shipped defaults. A fresh settings.json also adds Ctrl+C copy, Ctrl+V paste and Alt+Shift+D duplicate pane.

TABS

Ctrl + Shift + T
New tab, default profile

Ctrl + Shift + 1-9
New tab with profile 1-9
Numbers follow the order of the new-tab dropdown.

Ctrl + Shift + Space
Open the new-tab dropdown

Ctrl + Tab / Ctrl + Shift + Tab
Next / previous tab

Ctrl + Alt + 1-8
Jump to tab 1-8

Ctrl + Alt + 9
Jump to last tab

Ctrl + Shift + D
Duplicate tab

TABS (cont.)

Ctrl + Shift + N
New window

Alt + F4 Close window

closeTab has no default key: Ctrl+Shift+W closes the tab once its last pane is gone.

PANES

Alt + Shift + Plus
Split: new pane to the right
Duplicates the focused pane's profile.

Alt + Shift + Minus
Split: new pane below

Alt + Shift + D
Duplicate pane, longest side
Comes from the settings.json written on first run, not the built-in defaults.

Alt + Arrows
Move focus to neighbour pane

Ctrl + Alt + Left
Focus last used pane

Alt + Shift + Arrows
Resize focused pane

Ctrl + Shift + W
Close pane (or tab if only one)
Closing the last tab also closes the window.

Alt + Click
New-tab button: open as pane

PANES (cont.)

Zoom, swap, move to another tab, read-only and broadcast input have no keys; run them from the palette or bind togglePaneZoom, swapPane, movePane.

CLIPBOARD AND SELECTION

Ctrl + Shift + C / Ctrl + Insert
Copy selection

Ctrl + Shift + V / Shift + Insert
Paste

Enter
Copy the active selection

Ctrl + Shift + A
Select the whole buffer

Right-click
Copy selection, else paste

Alt + Drag
Block (rectangle) selection

Shift + Click
Extend selection to here

Shift + Arrows
Grow selection by a character

Esc Clear selection

MARK MODE

Ctrl + Shift + M
Toggle mark mode (keyboard select)
Starts a selection at the cursor; no mouse needed.

Arrows Move the cursor

Ctrl + Left / Ctrl + Right
Move by word

Tab / Shift + Tab
Next / previous hyperlink

SCROLLBACK AND SEARCH

Ctrl + Shift + Up / Ctrl + Shift + Down
Scroll one line

Ctrl + Shift + PageUp / Ctrl + Shift + PageDown
Scroll one page

Ctrl + Shift + Home / Ctrl + Shift + End
Top / bottom of history

Ctrl + Shift + K
Clear screen and scrollback

Ctrl + Shift + F
Search the pane's buffer
Searches only the focused pane; case match and direction are buttons.

WINDOW AND VIEW

Ctrl + Shift + P
Command palette
Delete the leading > to type a wt command line instead.

Ctrl + , Open the Settings UI

Ctrl + Shift + ,
Open settings.json

Ctrl + Alt + ,
Open defaults.json (read only)

Ctrl + Plus / Ctrl + Minus
Bigger / smaller font

Ctrl + 0 Reset font size

Alt + Enter / F11
Toggle fullscreen

Win + `
Quake mode: drop-down window
Summons a window named _quake from the top of the screen; same key hides it.

WT.EXE COMMAND LINE

`wt`
Open a window, default profile

`wt -p "<profile>"`
Open with a named profile

`wt -d <dir>`
Start in a directory
wt -d . opens where you are.

WT.EXE COMMAND LINE (cont.)

`wt -w 0 nt`
New tab in the latest window
-w new (or -1) forces a new window; -w <name> creates or reuses a named one.

`wt nt -p <a> ; sp -H -p <b>`
Tab, then a pane below it
nt = new-tab, sp = split-pane; -V splits side by side.

In PowerShell escape the separator as `; (or put --% after wt). From WSL run cmd.exe /c wt.exe and use \;.

SETTINGS.JSON ACTIONS

actions defines what to run (command + id); keybindings maps keys such as ctrl+alt+plus to an id.

```
{ "keys": "ctrl+shift+z",
"id":
"Terminal.TogglePaneZoom" }
```
Bind a built-in action

```
{ "keys": "ctrl+v", "id":
null }
```
Free a key for the shell or vim
"id": "unbound" works too; the key then reaches the app.