

CURRENT STABLE 1.130 | VERIFIED 4 AUGUST 2026

Visual Studio Code Visual Cheat Sheet

Large recognition symbols • exact functions • Mac-first shortcuts with Windows equivalents



PRODUCT CHECK VS Code is the lightweight, cross-platform editor. It is not the full Visual Studio Community IDE. Compilers, runtimes, SDKs and many language/debug/test features are separate installations or extensions.

What it is and what it does

Question	Precise answer
What is it?	A free code editor for macOS, Windows and Linux, with built-in editing, terminal, Git UI and Node.js/JavaScript/TypeScript debugging; extensions add many languages and tools.
What is a folder/workspace?	Usually your project root. A workspace can also contain multiple root folders and stores project-specific configuration in .vscode.
Does it compile code?	Not by itself for every language. Install the required compiler/runtime/SDK, then the matching extension or task configuration.
Which version is this?	VS Code Stable 1.130, released 22 July 2026. Insiders changes faster and may look different.

Five rules that prevent mistakes

1. **Open the correct folder:** The Explorer root controls search, Git, tasks, settings and terminal working directory.
2. **Trust carefully:** Keep unfamiliar projects in Restricted Mode until you review them.
3. **Install deliberately:** Check publisher and extension ID; extensions run with VS Code's permissions.
4. **Read the active context:** Shortcuts can be overridden by keyboard layout, extensions, keybindings and when-clauses.
5. **Review before running:** Inspect AI changes, terminal commands, diffs and task definitions before execution or commit.

SYMBOL NOTE Symbols are simplified high-contrast recognition aids, not pixel-perfect Microsoft artwork. The printed control name and command/menu path are the final authority.

02 Interface map

The six main regions stay conceptually stable even when moved, hidden or resized.

Region	Location and job
Activity Bar	Usually far left. Switches major views such as Explorer, Search, Source Control, Run and Debug, Extensions and Testing.
Primary Side Bar	Beside the Activity Bar. Shows the selected view's tree, search results or controls.
Editor	Central area for files, diffs, settings, previews and terminals opened as tabs.
Secondary Side Bar	Usually right; Chat is shown there by default in current VS Code.
Panel	Usually bottom; contains Terminal, Problems, Output and Debug Console.
Status Bar	Bottom strip. Shows branch, errors/warnings, language, encoding, line/column, remote state and context-specific actions.

Symbol	Name	May look like	What it does	Use it for
LAYOUT RECOGNITION				
	Primary Side Bar	vertical view area	Holds the active Activity Bar view.	Browse files, search results or Git controls.
	Panel	bottom drawer	Holds terminal and diagnostic views.	Run commands and read logs.
	Status Bar	thin bottom strip	Reports context and provides quick actions.	Check branch, language, warnings and remote state.
	Editor	central tabs	Edits files and displays rich editors.	Write code, inspect diffs and settings.
	Split Editor	two panes	Creates another editor group.	View files side by side.

FAST LAYOUT RESET Command Palette → View: Reset View Locations. For one hidden region use View → Appearance and choose the named item.

03 Activity Bar symbols

These are the main navigation buttons. Extensions may add more icons.

Symbol	Name	May look like	What it does	Use it for
BUILT-IN AND COMMON VIEWS				
	Explorer	two files	Browse the open folder/workspace and open editors.	Create, rename, move and open files.
	Search	magnifying glass	Searches text across the workspace.	Find or replace across files.
	Source Control	branch graph	Shows repositories and pending Git changes.	Stage, commit, branch and sync.
	Run and Debug	play with bug	Configures and starts debugging.	Run, pause, step and inspect state.
	Extensions	four blocks	Finds and manages extensions.	Add languages, themes and tools.
	Testing	laboratory flask	Shows tests contributed by test extensions.	Discover, run, debug and inspect coverage.
	Chat / Agents	speech bubble	Opens AI chat/agent surfaces when available.	Ask, plan or delegate code changes.
	Accounts	person	Manages sign-ins and Settings Sync.	Switch accounts or sync settings.
	Manage	gear	Opens settings, themes and configuration commands.	Change preferences and keyboard shortcuts.

04 Files, folders and workspaces

VS Code becomes project-aware when a folder or workspace is open.

Symbol	Name	May look like	What it does	Use it for
PROJECT STRUCTURE				
	Folder	folder	A directory opened as the project root.	Use for one project/root.
	Workspace	folder marked W	One or more root folders plus workspace configuration.	Use multi-root projects or shared workspace settings.
	File	page	An individual editable file.	Open it in an editor tab.
	Tabs	tab strip	Shows open editors; a preview tab may be replaced by the next preview.	Double-click to keep an editor open.
	Breadcrumbs	path trail	Shows symbols and folders above the editor.	Navigate the file structure quickly.
	Minimap	small code map	Compressed overview at the editor edge.	Jump through a long file.
	Save	SAVE	Writes current editor changes to disk.	Use before external tools if Auto Save is off.

Exact folder/workspace facts

Item	Meaning
Open Folder...	Makes that folder the workspace root for Explorer, Search, Git discovery and default terminal location.
Multi-root workspace	Adds multiple root folders. Save as a .code-workspace file when you need to reopen/share that arrangement.
.vscode/settings.json	Project-specific settings; override user settings for that workspace.
.vscode/launch.json	Debug configurations for complex or repeated launches.
.vscode/tasks.json	Workspace tasks that invoke external tools; task support requires a workspace folder.
Untitled / dirty dot	Content has not been saved. Closing may prompt; source control sees only saved disk content.

05 Find, navigate and edit

Use Quick Open for files and Command Palette for actions.

Symbol	Name	May look like	What it does	Use it for
NAVIGATION AND LANGUAGE FEATURES				
	Quick Open	⌘P / Ctrl+P	Searches filenames in the workspace.	Open a file without browsing folders.
	Command Palette	CMD	Searches available commands by name.	Run any command and discover its shortcut.
	Find	FIND	Finds text in the current editor.	Navigate matches in one file.
	Replace	REPL	Replaces current-file or workspace matches.	Preview broad replacements first.
	Symbol	SYM	Lists symbols supplied by the language service.	Jump to a method, class or heading.
	Go to Definition	DEF	Navigates to a resolved definition.	Inspect where a symbol is declared.
	References	REF	Lists uses found by the language service.	Assess impact before changing code.
	IntelliSense	ABC	Completion, parameter info and diagnostics from built-in/extension language services.	Write with language-aware assistance.
	Quick Fix	FIX	Shows available fixes/refactorings for the cursor location.	Resolve diagnostics or transform code.
	Rename	F2	Performs semantic rename when the language service supports it.	Rename a symbol across understood references.
	Format	FMT	Runs the configured formatter.	Normalize document/selection layout.
	Multiple cursors	CURS	Places more than one insertion point.	Repeat the same edit in several places.

06 Terminal, Problems, Output and Debug Console

These panel tabs are not interchangeable.

Symbol	Name	May look like	What it does	Use it for
PANEL SURFACES				
	Terminal	prompt >_	Runs a shell installed on your computer, starting at the workspace root by default.	Run compilers, package managers, scripts and Git.
	Problems	warning mark	Aggregates errors, warnings and information from language services, builds and tasks.	Jump to reported locations.
	Output	OUT	Shows logs from the selected output channel.	Read extension, task, Git and language-server detail.
	Debug Console	DBG	Shows debugger output and evaluates expressions while a session is active.	Inspect values; it is not the normal shell.
	Panel	bottom drawer	Container for these views.	Toggle it without closing editors.

TERMINAL SAFETY A terminal command can change or delete files with your account's permissions. Read multiline pastes and AI-generated commands before pressing Return.

Action	Mac	Windows
Toggle terminal	Control+`	Ctrl+`
Create new terminal	Control+Shift+`	Ctrl+Shift+`
External terminal	Shift+Command+C	Ctrl+Shift+C
Split terminal	Command+\	Ctrl+Shift+5
Find in terminal	Command+F	Ctrl+F

07 Start and control debugging

Built-in Node.js/JavaScript/TypeScript debugging is included; other languages usually need an extension.

Symbol	Name	May look like	What it does	Use it for
DEBUG TOOLBAR				
	Start / Continue	green play / F5	Starts the selected configuration or continues from a pause.	Begin/resume debugging.
	Pause	two bars	Requests a pause in running code.	Inspect current state.
	Stop	red square	Ends the debug session.	Terminate debugging.
	Restart	circular arrow	Stops and starts the current configuration again.	Retest from the beginning.
	Step Over	curved arrow / F10	Executes the current statement without entering called code.	Stay in the current function.
	Step Into	down arrow / F11	Enters called code when source/debug info allows.	Follow the call.
	Step Out	up arrow / Shift+F11	Runs until the current function returns.	Return to the caller.

Command	Mac default	Windows default	What it does
Start/continue	F5	F5	Starts or resumes debugging.
Run without debugging	Control+F5	Ctrl+F5	Runs without debugger attachment.
Restart	Shift+Command+F5	Ctrl+Shift+F5	Restarts current debug target.
Stop	Shift+F5	Shift+F5	Stops the session.
Step Over / Into / Out	F10 / F11 / Shift+F11	F10 / F11 / Shift+F11	Controls statement execution.
Toggle breakpoint	F9	F9	Adds/removes a line breakpoint.
Debug Console	Shift+Command+Y	Ctrl+Shift+Y	Opens the Debug Console.

08 Breakpoints and debug state

Breakpoints are instructions to the debugger, not changes to source code.

Symbol	Name	May look like	What it does	Use it for
PAUSE CONDITIONS AND STATE				
	Breakpoint	filled red dot	Pauses when execution reaches the bound location.	Inspect a repeatable point.
	Conditional breakpoint	dot with question	Pauses only when its expression/condition is met.	Reduce irrelevant stops.
	Logpoint	diamond	Logs a message without pausing execution.	Trace values with less interruption.
	Variables	VAR	Shows variables in the current scope/frame.	Inspect local state.
	Watch	WATCH	Evaluates saved expressions in the selected frame.	Track important values.
	Call Stack	STACK	Lists active function calls and frames.	Understand how execution arrived here.
	launch.json / tasks.json	JSON	JSON files in .vscode for debug and task configuration.	Make repeatable project commands.

Indicator	Meaning
Filled red breakpoint	Enabled and normally bound.
Gray breakpoint	Disabled.
Gray hollow breakpoint	Unverified/unbound; debugger has not resolved it to executable code.
Yellow/current-line marker	Statement where execution is paused; moving it changes execution position and can skip/repeat side effects.
Inline breakpoint	Shift+F9 creates one at an inline location where supported.

09 Git source control

VS Code uses the Git installation on your machine; the UI and command line operate on the same repository.

Symbol	Name	May look like	What it does	Use it for
CORE GIT ACTIONS				
	Repository	REPO	A folder whose history is stored by Git.	Track versions and collaborate.
	Changes	M	Lists modified, added, deleted and untracked files.	Review work before staging.
	Diff	DIFF	Compares versions side by side/inline.	Inspect exact line changes.
	Stage	plus	Adds selected changes to the next commit snapshot.	Choose what belongs in the commit.
	Commit	check mark	Records staged changes locally with a message.	Create a reviewable history point.
	Branch	branch graph	Moves work to a named line of history.	Isolate a feature or fix.
	Fetch	FETCH	Downloads remote references without integrating them.	Check remote updates safely.
	Pull	down arrow	Fetches then integrates remote changes using configured Git behavior.	Update the current branch.
	Push	up arrow	Uploads local commits to the remote.	Share committed work.
	Sync	up/down arrows	Pulls and pushes according to VS Code/Git configuration.	Synchronize both directions.
	Merge	MERGE	Combines another branch's history into the current branch.	Integrate completed work.
	Conflict	not equal	Marks competing edits Git could not combine automatically.	Resolve each file, then stage and commit.

SAFE GIT LOOP Branch → edit → save → inspect Diff → stage deliberately → commit → fetch/pull → resolve and test → push. Commit is local; Push publishes.

10 Extensions, settings and trust

Extensions can run code with the same permissions as VS Code.

Symbol	Name	May look like	What it does	Use it for
CONFIGURATION AND SAFETY				
	Extensions	four blocks	Marketplace and installed-extension manager.	Install, disable, update or uninstall features.
	Settings	SET	Graphical editor for user and workspace settings.	Change preferences; open JSON for exact keys.
	Workspace Trust	shield with check	Controls whether a folder may run higher-risk features.	Trust only reviewed sources.
	Restricted Mode	shield with warning	Limits agents, terminal, tasks, debugging, workspace settings and extensions.	Inspect unfamiliar projects safely.
	Accounts	USER	Sign-in and Settings Sync entry.	Sync settings, shortcuts and extensions where enabled.

Scope	What it means
User settings	Apply broadly to your VS Code profile.
Workspace settings	Apply only to the current project and override user settings; stored under .vscode for a single-folder workspace.
Disable extension	Temporary; globally or for the current workspace.
Uninstall extension	Removes it and may prompt to restart the extension host.
Publisher trust	First install from a third-party publisher prompts for trust; verify publisher and exact extension ID.

11 Tasks, testing and remote development

These surfaces depend on the folder, installed tools and extensions.

Symbol	Name	May look like	What it does	Use it for
AUTOMATION AND ENVIRONMENTS				
	Task	TASK	Runs a configured external command through VS Code.	Build, lint, package or automate repeatable commands.
	Testing	flask	Central Testing view populated by testing extensions.	Discover, run, debug and inspect tests.
	Remote	REMOTE	Connects the UI to SSH, containers, WSL or tunnels through Remote extensions.	Edit and run where the toolchain lives.
	launch.json / tasks.json	JSON	Workspace-shared configuration files.	Version repeatable debug/task definitions carefully.
	Problems	warning mark	Receives diagnostics from tasks and test/tool integrations.	Jump from failures to source.

Feature	Exact boundary
Tasks	Available when working on a workspace folder, not just an isolated file. Default build task: Shift+Command+B / Ctrl+Shift+B.
Testing	VS Code supplies the Testing UI; test discovery/execution comes from an installed extension for the language/framework.
Remote - SSH	Opens folders on a remote machine/VM over SSH.
Dev Containers	Runs the workspace/toolchain in or mounted into a container.
WSL	Windows-only route into a Linux environment under WSL.
Remote - Tunnels	Connects through a secure tunnel without configuring SSH.

12 AI, Chat and agents

Current VS Code provides agent surfaces, but availability depends on sign-in, plan, policy, extensions and configuration.

Symbol	Name	May look like	What it does	Use it for
AI SURFACES AND SAFEGUARDS				
	Chat / Agents	speech bubble	Opens chat in the workbench; current layouts may place it in the Secondary Side Bar.	Ask about code and start supported sessions.
	AI agent	AI bubble	Can plan, edit files, run commands or use tools subject to permissions.	Delegate a scoped development task.
	Restricted Mode	shield warning	Disables agents in an untrusted workspace.	Prevent project content from driving actions before review.
	Diff	DIFF	Shows proposed or completed file changes.	Review every AI edit before keeping/committing.
	Terminal	prompt	May display commands proposed or run by an agent.	Check targets, flags and side effects before approval.

PRODUCT BOUNDARY GitHub Copilot/VS Code agents are not automatically this ChatGPT conversation. Names, models, plans, permissions and available tools can differ. This ChatGPT can work on files you explicitly provide or connect; it cannot see your local VS Code window by default.

REVIEW GATE Before accepting an AI change: inspect the diff, run relevant tests, read terminal output, check secrets/config files, and commit only the intended files.

13 Mac-first shortcuts: files and layout

Defaults can be overridden; chords are two keypresses in sequence.

Command	Mac default	Windows default	What it does
Command Palette	Shift+Command+P	Ctrl+Shift+P	Searches and runs commands.
Quick Open	Command+P	Ctrl+P	Opens a file by name.
New file	Command+N	Ctrl+N	Creates an untitled editor.
Open file	Command+O	Ctrl+O	Opens a file.
Save / Save As	Command+S / Shift+Command+S	Ctrl+S / Ctrl+Shift+S	Writes current editor / chooses destination.
Close editor	Command+W	Ctrl+W	Closes active editor.
Settings	Command+,	Ctrl+,	Opens Settings editor.
Keyboard Shortcuts	Command+K, Command+S	Ctrl+K, Ctrl+S	Opens keybinding editor; a chord.
Toggle Primary Side Bar	Command+B	Ctrl+B	Shows/hides primary sidebar.
Toggle Panel	Command+J	Ctrl+J	Shows/hides Panel.
Split Editor	Command+\	Ctrl+\	Creates another editor group.
Focus group 1/2/3	Command+1/2/3	Ctrl+1/2/3	Focuses numbered editor group.
Zen Mode	Command+K, Z	Ctrl+K, Z	Hides distractions; a chord.
Open Recent	Control+R	Ctrl+R	Opens recent folders/workspaces.

14 Mac-first shortcuts: views and navigation

If one differs, open Keyboard Shortcuts and search the printed command name.

Command	Mac default	Windows default	What it does
Explorer	Shift+Command+E	Ctrl+Shift+E	Focuses Explorer.
Search	Shift+Command+F	Ctrl+Shift+F	Searches across files.
Source Control	Control+Shift+G	Ctrl+Shift+G	Focuses Source Control.
Run and Debug	Shift+Command+D	Ctrl+Shift+D	Focuses Run and Debug.
Extensions	Shift+Command+X	Ctrl+Shift+X	Focuses Extensions.
Problems	Shift+Command+M	Ctrl+Shift+M	Opens Problems.
Output	Shift+Command+U	Ctrl+Shift+U	Opens Output.
Find / Replace	Command+F / Option+Command+F	Ctrl+F / Ctrl+H	Finds/replaces in active editor.
Replace in files	Shift+Command+H	Ctrl+Shift+H	Workspace replacement.
Go to line	Control+G	Ctrl+G	Moves to line number.
Symbol in file	Shift+Command+O	Ctrl+Shift+O	Lists symbols.
Definition / Peek	F12 / Option+F12	F12 / Alt+F12	Opens / previews definition.
References	Shift+F12	Shift+F12	Lists references.
Rename	F2	F2	Semantic rename when supported.

15 Mac-first shortcuts: editing

Keyboard layout and extensions can change bindings; the Keyboard Shortcuts editor is authoritative.

Command	Mac default	Windows default	What it does
Format document	Shift+Option+F	Shift+Alt+F	Runs configured formatter.
Line comment	Command+/	Ctrl+/	Toggles line comment.
Block comment	Shift+Option+A	Shift+Alt+A	Toggles block comment.
Move line	Option+Up/Down	Alt+Up/Down	Moves current line/selection.
Copy line	Shift+Option+Up/Down	Shift+Alt+Up/Down	Duplicates line/selection.
Delete line	Shift+Command+K	Ctrl+Shift+K	Deletes line.
Insert below	Command+Enter	Ctrl+Enter	Opens line below.
Insert above	Shift+Command+Enter	Ctrl+Shift+Enter	Opens line above.
Next occurrence	Command+D	Ctrl+D	Adds next matching occurrence to selection.
All occurrences	Shift+Command+L	Ctrl+Shift+L	Selects all current matches.
Arbitrary cursor	Option+click	Alt+click	Adds/removes a cursor.
Cursor above/below	Option+Command+Up/Down	Ctrl+Alt+Up/Down	Adds cursor vertically.
Quick Fix	Command+.	Ctrl+.	Shows code actions.
Trigger suggestions	Control+Space	Ctrl+Space	Invokes completion.

16 Troubleshooting and official verification

Use the command name—not memory—when an icon or shortcut differs.

Symptom	Check in this order
Command missing	Open a folder if required → install/enable the correct extension → verify language mode in Status Bar → check Workspace Trust → reload window.
Shortcut does nothing	Keyboard Shortcuts → search command name → inspect conflicts and when-clause → test keyboard layout/Fn key.
Run/debug will not start	Read first complete error → confirm runtime/SDK and debugger extension → select configuration → inspect launch.json → check Restricted Mode.
Terminal differs from instructions	Check selected terminal profile/shell and current directory; commands are shell- and OS-specific.
Git view empty	Confirm the folder is a Git repository and Git is installed; open Source Control and inspect Git Output.
Test view empty	Install/configure a supported testing extension, open the workspace root and confirm test discovery settings.
Extension caused trouble	Disable it for the workspace, restart extensions, inspect its Output channel, then update or uninstall.
AI action uncertain	Stop, review diff/command/permissions, revert only intended changes through source control, and run tests before commit.

Official Microsoft sources checked

- [Stable 1.130 release notes](#)
- [User interface](#)
- [Keyboard shortcuts](#)
- [Tips and tricks](#)
- [Debugging](#)
- [Source control](#)
- [Terminal](#)
- [Extensions](#)
- [Settings](#)
- [Workspace Trust](#)
- [Tasks and testing](#)
- [Remote development](#)
- [Agents](#)

FINAL AUTHORITY Command Palette shows available commands; Keyboard Shortcuts shows the installed binding and conflicts; Settings shows current scope. Those live views override a printed guide.