

Sidebar

Programming · VS Code Search and Explorer Cheat Sheet

The search.* and explorer.* settings: what search looks at, how results appear, and how the file tree behaves. Exact ID, what it does, and its default. Sheet 4 in the VS Code series.

SEARCH SHORTCUTS

- Ctrl+Shift+F** Find in Files: search the whole workspace
- Ctrl+Shift+H** Replace in Files
- Alt+C / Alt+W / Alt+R** Toggle match case / whole word / regular expression in the search box
- Ctrl+P** Quick Open: jump to a file by (part of) its name
- Ctrl+Shift+O / Ctrl+T** Go to a symbol in this file / in the workspace
- Ctrl+Shift+E** Focus the Explorer

WHAT SEARCH LOOKS AT

- search.exclude** Globbs skipped by search; default skips `**/node_modules`, `**/bower_components` and `**/*.code-search`
- search.useIgnoreFiles** `true` (default): respect `.gitignore` and `.ignore` files
- search.useParentIgnoreFiles** Also read ignore files in parent folders (default `false`)
- search.useGlobalIgnoreFiles** Also read your global ignore file (default `false`)
- search.followSymlinks** Follow symbolic links while searching (default `true`)
- search.smartCase** `true`; all-lowercase search ignores case, any capital makes it exact (default `false`)
- search.maxResults** Upper limit on results shown (default 20000)

SEARCH RESULTS DISPLAY

- search.sortOrder** `default`, `fileNames`, `type`, `modified`, `countDescending`, `countAscending`
- search.defaultViewMode** `list` (default) or `tree`
- search.collapseResults** `alwaysExpand` (default), `auto`, `alwaysCollapse`
- search.showLineNumbers** Show line numbers beside results (default `false`)
- search.useReplacePreview** Open the preview when you replace a match (default `true`)
- search.actionsPosition** Where the result action buttons sit: `right` (default) or `auto`
- search.decorations.colors / .badges** Colour and badge decorations in results (default `true` for both)

SEARCH TRIGGERS & QUICK OPEN

- search.searchOnType** `true` (default): results update as you type
- search.searchOnTypeDebouncePeriod** Wait in ms before searching while you type (default 300)
- search.seedWithNearestWord** Pre-fill the box with the word at the cursor (default `false`)
- search.seedOnFocus** Pre-fill with the editor selection when the view is focused (default `false`)
- search.quickOpen.includeSymbols** Quick Open also lists workspace symbols (default `false`)
- search.quickOpen.includeHistory** Quick Open lists recently opened files (default `true`)
- search.quickAccess.preserveInput** Remember the last Quick Open text (default `false`)
- search.searchEditor.doubleClickBehaviour** `goToLocation` (default), `selectWord`, `openLocationToSide`

EXPLORER DISPLAY

`explorer.compactFolders` Show single-child folder chains as one line, `a/b/c` (default `true`)

`explorer.sortOrder` `default`, `mixed`, `filesFirst`, `type`, `modified`, `foldersNestsFiles`

`explorer.sortOrderLexicographicOptions` Upper/lower case ordering: `default`, `upper`, `lower`, `unicode`

`explorer.openEditors.visible` How many entries the Open Editors pane shows (default 9)

`explorer.expandSingleFolderWorkspaces` Auto-expand a workspace made of one folder (default `true`)

`explorer.decorations.colors` / `.badges` Git/problem colours and badges in the tree (default `true` for both)

`explorer.copyRelativePathSeparator` Separator when copying a relative path: `/`, `\` or `auto` (default)

EXPLORER BEHAVIOUR

`explorer.autoReveal` Select the active file in the tree: `true` (default), `false`, `focusNoScroll`

`explorer.autoRevealExclude` Files not revealed automatically; default `**/node_modules` and `**/bower_components`

`explorer.confirmDelete` Ask before deleting (default `true`)

`explorer.confirmDragAndDrop` Ask before moving by drag and drop (default `true`)

`explorer.enableDragAndDrop` Allow drag and drop in the tree (default `true`)

`explorer.enableUndo` Ctrl+Z undoes Explorer file actions (default `true`)

`explorer.incrementalNaming` Name for a pasted duplicate: `simple` (default), `smart`, `disabled`

FILE NESTING & HIDING

`explorer.fileNesting.enabled` Group related files under a parent, e.g. `app.js` under `app.ts` (default `false`)

`explorer.fileNesting.expand` Show nested groups expanded (default `true`)

`explorer.fileNesting.patterns` Rules for nesting, e.g. `{ "*.ts": "${capture}.js" }`

`explorer.excludeGitIgnore` Hide anything your `.gitignore` lists (default `false`)

`files.exclude` Globbs hidden from the Explorer entirely; see sheet 2 (Files and Auto Save)

A REASONABLE STARTER SET (MY SUGGESTIONS, NOT DEFAULTS)

`"search.smartCase": true` Case-insensitive until you type a capital

`"search.seedWithNearestWord": true` Ctrl+Shift+F starts with the word under the cursor

`"explorer.confirmDelete": false` Fewer prompts, since the Recycle Bin and Ctrl+Z still protect you

`"explorer.compactFolders": false` Every folder level visible, if the chains confuse you

`"explorer.fileNesting.enabled": true` Tidier tree in web and TypeScript projects

`"search.showLineNumbers": true` Easier to tell matches apart

SEARCH CAN'T FIND A FILE OR TEXT YOU KNOW IS THERE

Three things hide files from search: `search.exclude`, the `files.exclude` setting, and your `.gitignore` (because `search.useIgnoreFiles` is on). In the search panel, open the details with the ... button (Ctrl+Shift+J) and look at the *files to exclude* box and the *Use Exclude Settings and Ignore Files* toggle. Turning that toggle off searches everything, including folders such as `node_modules`.