

# VS code cheat sheet

Jarmo Hurri

feature	action / command	description
any	<b>ctrl-shift-p</b>	run vs code command
terminal type	<b>ctrl-shift-p</b> terminal profile → <b>git bash</b>	set <b>bash</b> as terminal
focus/visibility		
	<b>ctrl-l</b>	focus editor
	<b>ctrl-tab</b>	cycle editor
	<b>ctrl-w</b>	close (editor window)
	<b>ctrl-b</b>	toggle (left) sidebar visibility
	<b>ctrl-shift-`</b>	terminal
	<b>ctrl-j</b>	toggle (bottom) panel visibility
	<b>ctrl-alt-i</b>	toggle copilot visibility
	<b>ctrl-shift-e</b>	explorer
files		
	<b>ctrl-n</b>	new
	<b>ctrl-s</b>	save
	<b>ctrl-w</b>	close
selection		
	<b>ctrl-a</b>	select all code
	<b>shift-&lt;arrow&gt;</b>	select code
(un)comment	<b>ctrl-/</b>	toggle commenting region
python eval		
	<b>&gt;</b>	run
	<b>shift-enter</b>	eval line or selection in repl
git		
	<b>ctrl-shift-p</b> init	initialize git repository
	<b>ctrl-shift-g</b>	source control
packages		
	<b>ctrl-shift-p</b> create env	create (reactivate) <b>venv</b>
	<b>python -m pip install &lt;pkg&gt;</b>	install <b>&lt;pkg&gt;</b> in project
	<b>python -m pip freeze &gt; requirements.txt</b>	record dependencies in file

ai feature	action	description
terminal	<b>ctrl-i</b>	inline chat within terminal
participants		
	<b>@vscode</b>	vs code
	<b>@github</b>	github
	<b>@workspace</b>	current workspace
	<b>@terminal</b>	integrated terminal
commands		
	<b>/explain</b>	explain concept / code
	<b>/tests</b>	generate tests
	<b>/fix</b>	fix code or resolve errors
	<b>/doc</b>	generate documentation comments
	<b>/new</b>	new program element
	<b>/setupTests</b>	help setting up testing
test error	find failed test → fix test failure	analyses test problem and fix
tools		
	<b>#usages</b>	find usages
	<b>#new</b>	create new project
	<b>#fetch</b>	fetch file (e.g. url)
	<b>#openSimpleBrowser</b>	open web app in built-in browser
variables / context		
	<b>#&lt;file&gt;, #&lt;folder&gt;, #selection, #&lt;function&gt;</b>	specific code
	<b>#&lt;class&gt;, #block, #line, #&lt;symbol&gt;</b>	
	<b>#codebase</b>	entire project or workspace
	<b>#problems</b>	problems panel
	<b>#terminal</b>	integrated terminal
	<b>#output</b>	output panel
	<b>#git</b>	git status / related files
	<b>#changes</b>	source control changes
	<b>#diff</b>	code changes
	<b>#githubRepo</b>	specific github repo
	<b>#searchResults</b>	search view results
	<b>#testFailure</b>	test failure information as context

ai feature	action	description
chat		
	<b>ctrl-i</b> (editor/terminal/repl)	inline chat
	<b>ctrl-alt-i</b>	chat view
	<b>ctrl-alt-shift-ℓ</b>	quick chat
chat view		
	<b>ctrl-.</b>	toggle between chat modes
	<i>Add Context...</i>	attach context to prompt
	drag & drop file	attach file as context
	drag & drop folder	attach files within as context
	drag & drop problem	problem as context
	[history icon]	access history of chat sessions
	[mic icon]	enter chat prompt by using speech
generate		
	[smart-apply icon]	smart-apply generated code
	[insert icon]	insert code at cursor
	[copy icon]	copy code block to clipboard
	[terminal icon]	insert code in terminal
editor		
	code comments	prompt by instructions in comment
	<b>ctrl-.</b>	review / rewrite / extract