

ChatGPT Slash Commands in 2026 and How to Create Your Own Shortcuts

TechRounder PDF Edition

Live article: <https://www.techrounder.com/ai/chatgpt-slash-commands-shortcuts/>

By Vipin PG | Published September 2, 2026 | Updated September 2, 2026 | Format: Deep Dive | 10 min read

In brief

ChatGPT does support shortcuts in parts of its interface, but commands such as `/audit:`, `/seoexpert:` or `/developer:` do not automatically activate hidden ChatGPT modes.

ChatGPT does support shortcuts in parts of its interface, but commands such as `'/audit:'`, `'/seoexpert:'` or `'/developer:'` do not automatically activate hidden ChatGPT modes.

You can still use labels like these in a prompt. ChatGPT understands the word after the slash and can use it as an instruction. The result comes from the meaning of the prompt, not from a secret command attached to that label.

There is also a more reliable way to use this idea. You can create your own small set of prompt shortcuts, define what each one should do, and reuse them through Custom Instructions or Project Instructions.

The distinction matters because real ChatGPT shortcuts and user-created prompt shortcuts behave differently.

Does ChatGPT have slash commands?

ChatGPT has interface shortcuts and tool selectors on supported platforms.

For example, OpenAI documents that users with access to ChatGPT Search can type `/` and select Web Search from the available menu. ChatGPT may also decide to search automatically when a request requires current information.

Study Mode provides another example. On the web, OpenAI currently directs users to type `@study` or find Study through the tools menu. On iOS, `/study` may also appear when slash suggestions are available. OpenAI notes that available shortcuts can differ between apps and versions.

This means there is no fixed universal list of slash commands that every ChatGPT user can expect to have.

Some shortcuts belong to the ChatGPT interface. Others are simply text instructions written by the user.

What happens when you type something like `'/audit:'`?

Consider this prompt:

`'/audit: Check this product page for problems.'`

ChatGPT can understand it because "audit" describes the task.

You could write the same request without the slash:

'Audit this product page for problems.'

Both prompts provide roughly the same instruction.

A more detailed version is much more useful:

'/audit: Review this product page for unclear product information, trust problems, purchase friction and missing information. Rank the problems by likely customer impact and explain how each one can be fixed.'

The extra detail gives ChatGPT information about:

- what to examine
- which criteria matter
- how to prioritise findings
- what the response should contain

OpenAI's own prompting guidance recommends clear and specific instructions with enough context about the desired result, format, length and style.

The word '/audit:' can save typing once you have established what it means. It does not replace the rest of the prompt.

A practical formula for personal ChatGPT shortcuts

A useful shortcut can follow this structure:

'shortcut + scope + criteria + required result'

For example:

'/compare: Compare these two Wi-Fi routers for a three-bedroom home with a 1 Gbps connection. Focus on Wi-Fi coverage, wired ports, firmware support and long-term reliability. Finish with one recommendation and explain where the other router would be the better choice.'

The shortcut says what kind of work you want.

The remaining instructions tell ChatGPT what good work looks like.

This is especially useful for tasks you perform repeatedly.

Useful ChatGPT prompt shortcuts you can create yourself

The shortcuts below are examples of personal prompt aliases. They are not official ChatGPT commands.

'/verify:'

Use this when accuracy matters.

Example:

'/verify: Check every factual claim in this draft about Android 17. Verify release dates, supported devices and feature availability using primary sources where possible. Separate confirmed, outdated and unsupported claims.'

This gives ChatGPT a defined verification job instead of simply asking whether an article "looks correct."

‘/compare:‘

Useful when several options need to be judged against the same requirements.

Example:

‘/compare: Compare these three cloud storage plans for a small photography business. Prioritise storage cost, file recovery, sharing controls and desktop sync. Ignore bundled productivity apps unless they affect the recommendation.’

‘/diagnose:‘

Use it when the cause of a problem is unknown.

Example:

‘/diagnose: My website traffic dropped after June 20. Examine the supplied Search Console data and site changes. Separate technical, indexing, ranking and content-related possibilities. Rank them by the evidence available.’

‘/extract:‘

Useful for turning messy information into structured data.

Example:

‘/extract: Read these customer support messages and group recurring complaints. For each group, give the problem, common symptoms and the number of supplied messages that mention it.’

No invented counts should be added. The number must come from the material supplied.

‘/simplify:‘

This can control the level of an explanation.

Example:

‘/simplify: Explain DNS over HTTPS to someone who understands basic Wi-Fi and routers but has never studied networking. Use one realistic home-network example and avoid unnecessary protocol terminology.’

‘/redteam:‘

Use this when you want weaknesses examined rather than a supportive response.

Example:

‘/redteam: Examine this subscription business idea for weak assumptions. Focus on customer demand, cancellation risk, acquisition cost and dependence on third-party platforms. Identify which assumptions need evidence before money is spent.’

‘/rewrite:‘

Useful when the factual content should remain unchanged while presentation changes.

Example:

/rewrite: Rewrite this product announcement as a neutral technology news article. Preserve confirmed facts and quoted statements. Remove promotional language and do not add claims that are absent from the source.'

'/brief:'

Good for long reports, meetings or research material.

Example:

/brief: Turn this 30-page report into a briefing for a product manager. Keep decisions, risks, deadlines and unresolved questions. Remove background information that does not affect a decision.'

'/decision:'

Useful when the final response needs a recommendation rather than an open-ended comparison.

Example:

/decision: Evaluate these three hosting options for a WordPress news site receiving 500,000 monthly page views. Base the recommendation on the supplied pricing and specifications. Explain the main compromise in the option you select.'

'/plan:'

Turns an objective into an execution sequence.

Example:

/plan: Create a migration plan for moving this WordPress site to a new host while minimising downtime. Include preparation, backup, DNS, testing and rollback steps. Flag anything that depends on information I have not supplied.'

'/challenge:'

Useful for checking your own reasoning.

Example:

/challenge: Review my conclusion about this market opportunity. Identify assumptions I am treating as facts, evidence that could contradict the conclusion and information that would materially change the decision.'

'/sourcecheck:'

This can be useful for publishers and researchers.

Example:

/sourcecheck: Review the sources behind this article. Identify which claims are supported by primary sources, which depend on secondary reporting and which currently have no reliable citation.'

A small set like this is generally easier to maintain than dozens of loosely defined labels.

Why detailed shortcuts work better than role prompts

A common prompt pattern assigns ChatGPT a profession:

'/marketer:'

'/lawyer:'

'/developer:'

'/investor:'

The profession provides some context, but the instruction is broad.

Compare:

'/seoexpert: Check this article.'

with:

'/contentcheck: Review this article for search intent, factual completeness, title accuracy, duplicated sections and information that is likely to become outdated. Do not rewrite it yet. Return the problems in priority order.'

The second prompt defines observable tasks.

The same principle applies to professional roles. Asking ChatGPT to "think like an investor" can influence what it pays attention to, but a prompt that specifies margins, market size, customer acquisition, competition and downside risk gives the model a clearer job.

A role label also does not provide professional credentials or access to information the model does not have.

Use several shortcuts as separate stages for complex work

Large prompts sometimes contain research, analysis, criticism, writing and final decision-making in one request. Splitting those jobs can make the process easier to inspect.

Suppose you are considering a new software product.

Start with:

'/research: Identify the customer problem, current alternatives and evidence of demand using the sources provided.'

Follow with:

'/challenge: Identify unsupported assumptions in the research above and list the evidence still missing.'

Then:

'/decision: Based only on the verified information above, decide whether the idea is ready for a small validation test.'

Finish with:

'/plan: Create the smallest practical validation plan and define what result would justify continuing.'

Each stage receives the output of the previous stage.

OpenAI recommends iterative prompting for tasks that benefit from refinement. Its current ChatGPT prompt guidance advises users to start with a clear request, inspect the response and adjust the prompt with additional context when necessary.

Define your shortcuts in Custom Instructions

A personal shortcut becomes more consistent when ChatGPT has a written definition for it.

Custom Instructions are available across ChatGPT plans on web, desktop, iOS and Android. OpenAI says these instructions are applied to chats so ChatGPT can take the user's preferences into account when responding.

You could define '/verify:' like this:

'When a message begins with /verify:, identify factual claims and check them against reliable sources. Separate confirmed, outdated, contradicted and unsupported information. Prefer primary sources and never fill missing evidence with assumptions.'

You can then write:

'/verify: [paste article]'

The label now has a definition supplied by you.

There is a practical limit to how many detailed aliases you should store this way. OpenAI currently allows Free and Go users to save up to 1,500 characters in Custom Instructions. Plus, Pro, Business, Enterprise and Education users can save up to 5,000 characters.

Keeping only frequently used rules there leaves room for other preferences.

Project Instructions are better for specialised workflows

Projects can keep chats, files and instructions together for a particular subject or type of work. OpenAI says Projects are available across free and paid ChatGPT plans.

A publishing Project, for example, could define:

'/factcheck: Verify names, dates, figures, quotes, product specifications and availability. Use primary sources where possible. Do not rewrite disputed information until its status is clear.'

A programming Project might use:

'/debug: Reproduce the reported failure from the supplied code and logs. Identify the likely cause before suggesting changes. Preserve working behaviour outside the affected area.'

Project Instructions apply within that Project. OpenAI states that they override global Custom Instructions there.

This allows the same account to have different working rules for publishing, development, research or other recurring work.

Skills can handle more structured reusable workflows

OpenAI now has a feature called Skills for eligible workspaces.

A Skill can contain instructions, examples and code. ChatGPT can use an installed Skill when it is relevant to a task. OpenAI currently describes Personal Skills as generally available for Business, Enterprise, Healthcare and Edu users.

This is useful when a workflow involves more than a short prompt definition.

A fact-checking Skill, for example, could specify that ChatGPT should:

1. extract factual claims from a draft;

2. identify the strongest available source for each claim;
3. compare dates, figures and quotations;
4. mark conflicting evidence;
5. return a standard verification report.

The workflow can remain consistent without the user typing those instructions every time.

Availability and workspace permissions still matter, so Skills should not be presented as a feature available to every personal ChatGPT account.

Custom GPTs have different availability in 2026

Custom GPTs can also contain persistent instructions, knowledge and selected capabilities. They remain usable by ChatGPT users who have access to them.

The creation rules changed in 2026.

OpenAI currently says personal Free, Go, Plus and Pro accounts cannot create or publish new GPTs. Existing GPTs remain available, while new GPT creation continues in eligible Business, Enterprise and Edu workspaces when permissions allow it.

For a personal ChatGPT user building a new shortcut system today, Custom Instructions and Projects are therefore the more broadly available options.

Personal shortcuts do not create new ChatGPT capabilities

A shortcut can change the instructions given to ChatGPT. It cannot grant a tool or capability that the current chat does not have.

For example:

`‘/searchweb:‘`

does not guarantee access to live web search.

`‘/database:‘`

does not create access to a private database.

`‘/photoshop:‘`

does not install image-editing software.

`‘/lawyer:‘`

does not turn the response into professional legal advice.

Tools depend on the ChatGPT plan, interface, permissions, connected services and product features available in the conversation.

This is one reason genuine ChatGPT interface commands should be kept separate from personal prompt aliases when documenting them.

A shortcut does not need a slash

There is no requirement to use `‘/‘`.

You could define:

'VERIFY:'

#verify'

'[verify]'

or even:

'CHECK FACTS:'

ChatGPT can interpret all of them as text.

The slash is useful because it makes a personal command visually easy to recognise and quick to type. It also resembles familiar command syntax used in software.

Consistency matters more than the symbol.

If you choose '/verify:', use the same label instead of alternating between '/verification:', '/checkfacts:' and '/factchecker:' for the same job.

How to build a shortcut that stays useful

Start with a task you repeat frequently.

Suppose you often compare technology products. Write down the requirements you normally repeat:

- use confirmed specifications;
- separate meaningful differences from marketing claims;
- consider price;
- identify missing information;
- recommend an option only when the evidence supports it.

Turn those requirements into one definition:

'When I use /productcompare:, compare products using verified specifications and current pricing where available. Focus on meaningful differences. Flag missing data instead of guessing. Finish with recommendations for the use cases supported by the evidence.'

Your future prompt becomes shorter:

'/productcompare: Pixel A vs Pixel B for photography and battery life in India.'

You still need to provide context that changes from request to request. The stored instruction handles the repeated part.

Which ChatGPT shortcuts should you create first?

There is little benefit in maintaining a catalogue of shortcuts that you rarely use.

Five well-defined shortcuts may cover most repetitive work.

A writer might use:

'/verify:' '/sourcecheck:' '/rewrite:' '/brief:' '/outline:'

A developer could instead keep:

'/debug:' '/reviewcode:' '/explain:' '/testplan:' '/refactor:'

Someone evaluating businesses might prefer:

`/research:` `/compare:` `/challenge:` `/decision:` `/plan:`

The names matter less than their definitions.

ChatGPT shortcuts work best as reusable instructions

ChatGPT has genuine interface shortcuts, and typing `/` can expose supported tools such as Web Search on eligible interfaces. Other features have their own selectors, including `@study` and, on some iOS versions, `/study`. Those options come from ChatGPT itself and may change as the product changes.

Labels such as `/verify:`, `/compare:` or `/audit:` can still be useful. They are personal prompt conventions whose behaviour comes from the instructions attached to them.

The most reliable setup is to choose a small number of repeated tasks, give each one a clear definition, and store the appropriate definitions in Custom Instructions or Project Instructions. Eligible workspace users can use Skills when the process needs a more structured workflow.

That approach saves typing while keeping the instructions specific enough for ChatGPT to know what result you expect.