AITOOLSPECS.COM

RESEARCH PUBLICATION

NO. 03 — PROMPT DIALECTS

"Think step by step" is the most repeated AI advice on the internet. For a growing class of tools, it is now wrong.

Almost everything you have absorbed about writing a good prompt, you learned on one tool. We read the official documentation from the makers of more than 60 of them to find out where those habits quietly stop working — and, in six documented cases, start working against you.

Arnab Bhattacharyya
Founder, AIToolSpecs.com

August 2026
60+ TOOLS · VENDORS' OWN DOCUMENTATION

THE FINDING

There is no such thing as a good prompt. There is only a good prompt for a specific tool.

Prompting advice spread the way folklore spreads. Someone found a phrase that worked, posted it, and it was repeated until it hardened into a rule. That worked well enough when there was essentially one tool that mattered and everyone was learning it at the same time.

There are now dozens of tools that matter, built on genuinely different machinery, and the folklore did not split when the tools did. So people carry habits learned on ChatGPT into Claude, habits learned on Midjourney into a video generator, and assume a prompt that reads well is a prompt that works.

60+

tools whose own official prompting documentation we read directly

6

documented contradictions — one vendor's best practice is another's named mistake

2

areas where unrelated, competing companies independently reached the same answer

> If you have ever added "think through this step by step" to a prompt out of general good practice: on one entire category of current models, the companies that built them now tell you not to.

Both halves of the pattern are useful, in different ways. The contradictions tell you where a habit you trust is about to cost you quality without ever announcing itself. The agreements tell you which habits are genuinely safe to carry everywhere.

HOW TO CITE THIS RESEARCH

Bhattacharyya, A. (2026). *Horses for Courses: What AI Companies' Own Instructions Reveal About Why the Same Prompt Does Not Work Everywhere.* AIToolSpecs.com Research Publication No. 03. aitoolspecs.com/publications/prompt-dialects

01 The most repeated prompting tip on the internet has expired for some tools

"Let's think step by step" is prompting's greatest hit. It has been copied into more templates, courses and posts than any other phrase in the field, and for standard chat models it genuinely earns its reputation — those models do not reason before answering unless you ask them to, so asking is what makes it happen at all.

Then a new category arrived. OpenAI's reasoning models — the o-series and similar "thinking" models — run a hidden reasoning process before they produce a word of output. OpenAI's own guidance for them says plainly: do not tell them to think step by step or explain their reasoning. It is unnecessary, and it can make results worse.

REASONING MODELS · DON'T ASK

```
"Work out which of these three vendors
offers the lowest total cost over two
years, and show the figures you used."
```

STANDARD CHAT MODELS · STILL ASK

```
"Work out which of these three vendors
offers the lowest total cost over two
years. Think through it step by step, and
show the figures you used."
```

The reason is mechanical rather than stylistic. Asking a model to narrate a process it has already completed internally is asking for the work to be done twice — once invisibly and once for show. The second pass adds nothing and can measurably degrade the answer.

WHAT THIS MEANS FOR YOU

If "think step by step" is muscle memory — something you add to every prompt because it is good practice — that habit now silently works against you the moment you are talking to a reasoning-labelled model. And tools increasingly do not make it obvious which kind you have been routed to.

02 Two rivals, working separately, arrived at the same warning: stop shouting

This is the finding we would flag first if you only read one section, because it is the only place in the entire review where two competing companies independently documented the same failure.

Google's current Gemini documentation states it plainly: avoid unnecessary or overly persuasive language. Anthropic's documentation for Claude Opus 4.5 and 4.6 names the failure mode — these models are *more* responsive to forceful phrasing than earlier versions, and prompts like *"CRITICAL: You MUST use this tool when..."* now cause the model to overtrigger.

THE SAME INSTRUCTION, WRITTEN TWO WAYS

OLD HABIT, STILL WIDELY TAUGHT

```
"CRITICAL: You MUST always double-check your
citations before responding. This is
EXTREMELY important."
```

CURRENT GUIDANCE · OPUS 4.5 / 4.6

```
"Double-check citations before responding."
```

Illustrative example written by AIToolSpecs.com to demonstrate the documented principle — not a reproduction of any vendor's own example prompt.

> Two different companies, two different products, one underlying warning: shouting at a current frontier model does not make it more obedient. It makes it less predictable.

Anthropic ties the overtriggering finding specifically to Opus 4.5 and 4.6 — not to every Claude model ever shipped. On this subject, *which exact version* can matter as much as which company.

WHAT THIS MEANS FOR YOU

Treat all-caps, "you MUST", "this is CRITICAL" phrasing as a last resort rather than a default — the one habit two separate, competing companies have independently flagged as backfiring on their current models.

03 Where you put the question changes the answer — in opposite directions

Feed Claude a long document — Anthropic's documentation specifically flags roughly 20,000 words of input and up — and their guidance is to put the document **first** and your actual question at the very end, after it. Anthropic states this ordering can improve answer quality by up to 30% on complex, multi-document tasks.

OpenAI's current guidance for ChatGPT and the API points the other way by default: instructions first, supporting material after, separated with clear delimiters.

SAME CONTRACT, SAME QUESTION, OPPOSITE ORDERING

CLAUDE · LONG DOCUMENT FIRST

```
[ full document ]
↓
"Based on the contract above, what
termination clauses require 60 days'
notice?"
```

CHATGPT · INSTRUCTION FIRST

```
"Review the contract below and identify
which termination clauses require 60 days'
notice."
↓
[ full document ]
```

Illustrative example written by AIToolSpecs.com. Anthropic's up-to-30% figure is their own published claim for complex, multi-document tasks.

Neither is a stylistic preference dressed up as a rule. Anthropic ties its ordering to how Claude's attention behaves across very long inputs specifically — a stated fact about that model's mechanics. Each ordering is the correct one for its own tool, and copying one into the other quietly gives up quality on exactly the tasks where you need it most.

WHAT THIS MEANS FOR YOU

If you move a long-document workflow from one assistant to another, move the question with it. Copying the prompt across unchanged quietly costs you the ordering each tool was tuned for.

04 Two image tools that read prompts in almost opposite languages

Midjourney's convention is close to the opposite of a normal sentence: a short, plain description of the subject, followed by a string of terse, code-like parameter flags. Grammar and politeness are simply not part of how it reads input. OpenAI's image generation through ChatGPT runs the other way entirely: full, well-ordered conversational prose, then iterative follow-up messages refining the same image in place.

MIDJOURNEY

```
elderly ceramicist, sun-etched hands shaping clay, warm studio light --ar 3:2 --stylize 250
```

CHATGPT / DALL·E

```
"A photo of an elderly ceramicist with weathered, sun-etched hands, caught mid-motion shaping wet clay on a wheel, lit by warm afternoon light through a studio window."
```

Neither dialect is more sophisticated than the other. They are built for genuinely different generation approaches — and a prompt written correctly for one reads as either garbled or oddly vague to the other.

05 "Leave this out" is a button in one tool and a sentence in another

Kling AI's video generator gives you a dedicated negative prompt field — a separate box specifically for exclusions, applying to both the video and its audio track. Runway and Google's Veo have no equivalent field; there, exclusions go directly into your main descriptive prompt as ordinary instructions.

KLING · TWO INPUTS

MAIN

"A woman walks through a quiet forest at dawn, soft light filtering through the trees."

NEGATIVE

"blurry, distorted hands, camera shake, background music"

RUNWAY OR VEO · ONE INPUT

MAIN

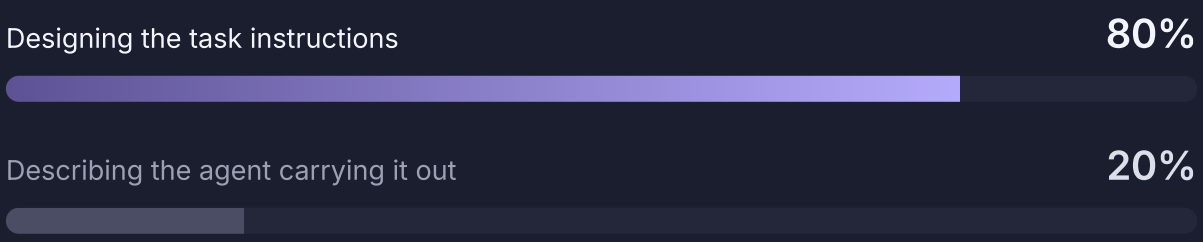
"A woman walks through a quiet forest at dawn, soft light filtering through the trees. Keep the camera steady, no motion blur, no distortion on her hands, and no added music."

The mismatch is not merely inefficient: in Kling's case, product guidance notes that piling too many exclusions into the negative field starts fighting the model's own creative controls rather than cleanly removing anything.

06 One tool tells you, with a number, that you are spending your time wrong

This one is not about wording at all. It comes from CrewAI, a tool for building multi-step workflows out of several cooperating AI "agents", and it is a claim about where your effort should go.

CREWAI'S DOCUMENTED EFFORT SPLIT



Their reasoning is direct: a well-designed task can carry a mediocre agent description to a good result, while a beautifully written agent persona will still fail against a vague task.

The common instinct with agent tools is to spend the evening crafting an elaborate backstory — *"You are a world-class research analyst with 20 years of experience..."* — and then write the actual instructions in a hurry.

WHAT THIS MEANS FOR YOU

The company that built the tool says that has the ratio backwards. If a multi-step workflow keeps producing mediocre output, the fix is almost never a richer backstory — it is a more specific task, with the expected output described plainly.

PART TWO · WHERE EVERYONE QUIETLY AGREES

The habits that survived contact with every tool we checked

Contradictions make the better story, but they are the less useful half. When competing companies independently land on the same answer without copying, that convergence signals something structural rather than stylistic.

AGREEMENT 01 · THE STANDING INSTRUCTION FILE

Three separate coding-agent tools — Anthropic's Claude Code, Cursor and Windsurf — each independently built the same thing: a standing file holding your project's conventions, loaded automatically every session, kept separate from the instructions for the task in front of you.

Claude Code	Cursor	Windsurf
`CLAUDE.md`	`.cursorrules` · `.cursor/rules/`	`.windsurfrules` · `.windsurf/rules/`

The tell: practitioners report the contents are often 90% identical once tool-specific formatting is stripped out — not a house convention anyone chose, but what the problem's correct answer looks like.

AGREEMENT 02 · GIVE IT A REAL CHECK, AND KNOW WHEN TO STOP CORRECTING

Anthropic's Claude Code guidance states a concrete rule: after two failed correction attempts on the same problem, stop correcting and start over with a clearer instruction that folds in what you learned. By the second failure the context is polluted with failed approaches, and a third correction rarely recovers cleanly.

Paired with it is a principle that shows up independently across essentially every major coding-agent tool's documentation: give the AI something that can actually pass or fail — a test, a build step, a real error message — rather than a description of what good would look like.

— Two corrections is the signal to stop, not to try harder.

THE PATTERN BEHIND THE PATTERN

Machinery diverges. Workflow converges.

Put both lists side by side and the shape of the split is not random at all. It is almost perfectly clean — each set traces back to a different kind of question.

WHAT DIVERGES · MACHINERY

- Whether the model reasons invisibly before answering
- How it distributes attention across a very long input
- Whether a specific version was tuned to respond more strongly to emphasis
- Whether exclusions are a field or a sentence

These are technical facts about how a particular tool was built — and different tools are genuinely built differently.

WHAT CONVERGES · WORKFLOW

- How a person separates standing conventions from one-off requests
- When to stop manually correcting something that is stuck and start over
- How to give an assistant a real pass/fail check

These are solved problems about how humans work with any capable assistant — so separate teams solving them keep arriving in the same place.

> Advice about how the model works is probably tool-specific.
> Advice about how you work is probably portable.

That gives you a rule of thumb you can apply to a tool this report never examined. When you read a prompting tip somewhere, ask which of the two it is. That single question sorts most of the folklore.

WHAT TO ACTUALLY DO

Two habits that cover most of this without memorising anyone's documentation

01

Spend five minutes on the vendor's own prompting page

Before you build a habit around a new tool, read its official tips page if it has one. Most major AI companies publish one and it is usually short. It is the fastest way to find out whether something that serves you well elsewhere is about to quietly work against you here.

02

Drop the forceful language by default

All-caps and "you MUST" is the one habit two separate competing companies have independently documented as backfiring on their current frontier models. Plain and direct is now the safer default everywhere — not politeness, just accuracy.

> There is no such thing as a good prompt. There is only a good prompt for a specific tool — and the companies that build these tools say so themselves.

HOW THIS SHAPES OUR OWN WORK

The research behind our Prompt Generation engine

Everything above is the foundation for the part of AIToolSpecs.com's paid reports that writes the starting prompt for each tool in your recommended stack. It is built on the principle this entire report documents: the right prompt depends on how that specific tool actually works, not on a general-purpose template with the tool's name swapped in. A prompt written for Claude and a prompt written for Kling should not look like the same document with a find-and-replace run over it. In our reports, they do not.

METHODOLOGY

A note on scope and sources

A curated set, not the whole market

This covers AI tools we researched directly — not every tool available, and not every model version each vendor currently offers.

First-party documentation, read directly

Every specific claim above about a named vendor's guidance was checked against that vendor's own current official documentation at the time of writing (August 2026) — not a summary, a blog post, or a third party's paraphrase.

Where convention stands in for documentation, we say so

Midjourney's parameter-flag style rests on widely repeated practitioner convention rather than formal published documentation, because Midjourney does not document with the same formality as Anthropic, OpenAI or Google. We flag that rather than present it with false certainty.

Every example prompt is ours

All worked "this way, not that way" prompt examples were written by us to illustrate the underlying documented principle. They are original illustrations, not reproductions of any vendor's own examples.

A snapshot, not a permanent rule

Vendor guidance changes as new model versions ship. A quirk tied to one specific version — the Opus 4.5/4.6 finding above is exactly this — may not apply to whatever that vendor releases next.

No financial relationship influenced any of this

Naming a tool's documented quirk, whether it works in a prompt's favour or against it, has no connection to whether we have an affiliate relationship with that tool elsewhere on our site.

REFERENCES

Bibliography

01 **Claude — long-context document placement, up to 30% improvement —** Anthropic, "Long context prompting tips," official prompt engineering documentation.
docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/long-context-tips

02 **OpenAI's instructions-first, delimiter-based default ordering —** Prompt Builder, "ChatGPT & OpenAI Prompt Engineering Guide," citing OpenAI's own published guidance, 2026.
promptbuilder.cc/blog/openai-prompt-engineering-guide-best-practices-2026

03 **OpenAI reasoning models — avoid chain-of-thought prompting, try zero-shot first —** OpenAI, "Reasoning best practices," official API documentation.
developers.openai.com/api/docs/guides/reasoning-best-practices

04 **Gemini — avoid overly persuasive language —** Google, "Prompt design strategies," official Gemini API documentation.
ai.google.dev/gemini-api/docs/prompting-strategies

05 **Claude Opus 4.5 / 4.6 — forceful language causing tool overtriggering —** Anthropic, "Prompting best practices," official Claude Platform documentation.
platform.claude.com/docs/en/build-with-claude/prompt-engineering/claude-prompting-best-practices

06 **Kling's dedicated negative-prompt field vs. Runway/Veo's inline approach —** Artlist, "Negative prompts for Kling, Veo, and Wan," 2026.
artlist.io/blog/negative-prompts-ai-video

07 **Midjourney's parameter-flag dialect vs. DALL·E/ChatGPT's conversational prose —** SurePrompts, "AI Image Prompting: The Complete 2026 Guide."
sureprompts.com/blog/ai-image-prompting-complete-guide-2026

08 **CrewAI's 80/20 task-vs-agent design rule —** CrewAI, "Crafting Effective Agents," official documentation.
docs.crewai.com/en/guides/agents/crafting-effective-agents

09 **Claude Code — the two-failed-corrections rule; verification-first workflow —** Anthropic, "Best practices for Claude Code," official engineering documentation.
anthropic.com/engineering/claude-code-best-practices

10 **Convergent persistent-context file pattern across Claude Code, Cursor and Windsurf —** The Prompt Shelf, "AGENTS.md vs CLAUDE.md vs .cursor/rules: The Complete 2026 Three-Way Comparison," 2026.
thepromptshelf.dev/blog/agents-md-vs-claude-md-vs-cursorrules

ALSO IN THIS SERIES

No. 01 · Tool Churn — In one pass through 225 AI tools, 11 had already changed underneath us: what shut down, what was rebranded, and what was quietly repriced. August 2026

No. 02 · Risk Patterns — We scored 225 AI tools on risk: which categories are minefields, the nine tools we will not recommend, and where the billing surprises come from. August 2026

ABOUT THE AUTHOR

Arnab Bhattacharyya

Founder of AIToolSpecs.com. Twenty-three years in global operations and business transformation, including seventeen years at HSBC, with senior positions such as VP and Senior Director at HSBC and FIS. Alumnus of the Indian Institute of Management, Kolkata, and a Google Cloud Digital Leader.

Connect on LinkedIn

We did the homework, so you don't have to.

Tell us your budget, goal, skill level and hardware, and we will match you to tools that actually fit — with a starting prompt written for how each specific tool really works, not a template with the name swapped in.

Get your own plan at aitoolspecs.com

Free to use · no card required

AITOOLSPECS.COM