

The AI Bootcamp

11

Stop repeating yourself

In this chapter you'll learn

The 4-part standing instruction, the project that holds one job,
and the review that keeps both honest

The AI Bootcamp

THE PROBLEM

Most AI education is a pile of disconnected tips. A LinkedIn carousel here, a YouTube tutorial there, a \$2,000 cohort course that goes out of date before the refund window closes. None of it is built to be read in order, so nothing compounds. You finish twelve videos and can recite twelve facts, not perform one new skill.

THE FIX

The AI Bootcamp is built the other way. 90 skills, packed 2 to 3 per chapter, 36 chapters, 3 tracks. Every chapter states what it needs from you and what it hands to the chapter after it, so skill 47 only works because skill 12 already landed. Read it in order and by the end you are not collecting facts about AI, you are running work through it.

LETTER FROM THE EDITOR

We built this because we kept meeting smart people who felt behind on AI for no reason other than nobody had organized the material for them.

This book does not assume you are technical, and it does not water anything down either. Every chapter earns its place: a real skill, a real exercise, a one-page companion you can flip back to in month three when you have forgotten the details but still need the answer.

If a chapter loses you, or a chapter changes how you work, I want to hear about it either way.

Kris

Editor, AI Central

Write to me at editor@thecentral.ai

THE 3 TRACKS

AI 101

Get running

Pick a tool, set it up, write a clear instruction, use it safely at work.

AI 202

Prompt properly

The framework behind every good instruction, and catching AI when it is wrong.

AI 303

Build without watching it

Connect AI to your tools, and automations that run while you are not looking.

IN THIS LESSON YOU'LL LEARN

By the end of this chapter you will have written a standing instruction that every new chat starts with, set up one project holding the files and rules for work you do repeatedly, and you will know how to read and correct what your assistant has decided to remember about you on its own.

Chapters 4, 7 and 10 all left you paying the same tax. Every new chat opens by re-explaining who you are, what you do, who you write for, and how you like the words. Paying that 20 times a week is the quiet reason people stop using the tool: not because it failed them, but because the run-up got tiring.

CONTENTS	11.1	Two things persist, and both fail
	11.2	Four things worth telling it once, forever
	11.3	Some context belongs to one job
	11.4	What persists can be wrong

TOPICS REQUIRED

Chapter 10. You need your account from Chapter 2, the habit of handing over your own material from Chapter 7, and the description of your voice you worked out in Chapter 10. That description is the raw material for most of what you are about to save.

AFTER THIS CHAPTER

Chapter 12 assumes your standing instruction is saved and at least one project exists. It builds the weekly habit on that setup, not on a blank account.

11.1 Two things persist, and both fail

One is a note you write and the assistant reads before every reply. The other is a note the assistant writes about you, on its own, from things you said in passing. Telling them apart is the whole chapter.

Instructions you set

A short standing brief you type into settings once. Every new chat starts having read it.

TRAITS

- You wrote the words, so you can quote them back
- Identical in every chat until you edit it
- Sits in one place you can open in 10 seconds

PROS

- + Predictable: same input, same behaviour
- + A bad line is easy to find and delete

CONS

- Only a few hundred words fit
- Applies everywhere, including where it does not belong

Memory it writes

The assistant's own running notes about you, drawn from what you happened to mention across chats.

TRAITS

- It chose what was worth keeping, and the wording
- Updates itself as you keep talking to it
- On by default in most setups, so it is already running

PROS

- + Costs you nothing: it accumulates while you work
- + Catches context you would never think to write down

CONS

- It can record something wrong and repeat it for months
- You have to go and look to find out what it holds

Fig 11.1 The two mechanisms behind every "it already knows that" moment. The one on the left is text you can point at. The one on the right is text you never wrote, which is why it needs checking.

11.1

You write one. A standing instruction is your words, applied to every chat until you change them.

It writes the other. Memory is the assistant's own note-taking from things you mentioned in passing.

Assume it is running. On all four assistants, memory is on unless you or your IT administrator switched it off.

Here is the friction this chapter removes. You open a new chat and type some version of the same 4 sentences you typed yesterday: this is my role, this is who the output is for, this is roughly how long it should be, please stop opening with a compliment. Then you get to the actual question.

Two separate features exist to end that, and almost everyone confuses them. The first is a standing instruction: a short brief you type into settings once, which the assistant reads before every new chat. You wrote it, you can see it, and it does not change until you change it. All four assistants have this, on every plan including the free ones. ChatGPT calls it custom instructions, Claude calls it Instructions for Claude, Gemini calls it Instructions for Gemini, Copilot calls it custom instructions.

The second is memory. The assistant reads what you say across your chats, decides for itself which facts are worth keeping, and writes them down in its own words. You did not choose the wording and you were not asked. ChatGPT, Claude, Gemini and Copilot all do this now, and Claude opened it to free accounts in March 2026, so the free tier is not an exemption. One caveat worth knowing if you use Google: on Gemini both features need a personal Google account and do not work when you are signed in with a work, school or supervised one.

The difference matters at the moment things go wrong, which is the only moment worth designing for. A bad standing instruction is a line of text in a settings box: you open it, you read it, you delete the line. A bad memory is a sentence you never wrote, sitting in a screen you have probably never opened, quietly shaping answers you assumed were neutral.

So treat them as two different tools with two different jobs. Standing instructions are for the things that are permanently true and that you want stated exactly your way. Memory is for the accumulated texture you would never bother typing out, and it needs reviewing the way any unattended note-taker needs reviewing.

Before you write anything, go and look at what yours already holds. Open settings, find personalization, and read the memory it has built. Most people are surprised at least once.

11.2 Four things worth telling it once, forever

Most people open the custom instructions box, type "be concise and professional", and get nothing back for it. Concise and professional mean different things to a lawyer and a designer. Here is what actually changes the output.

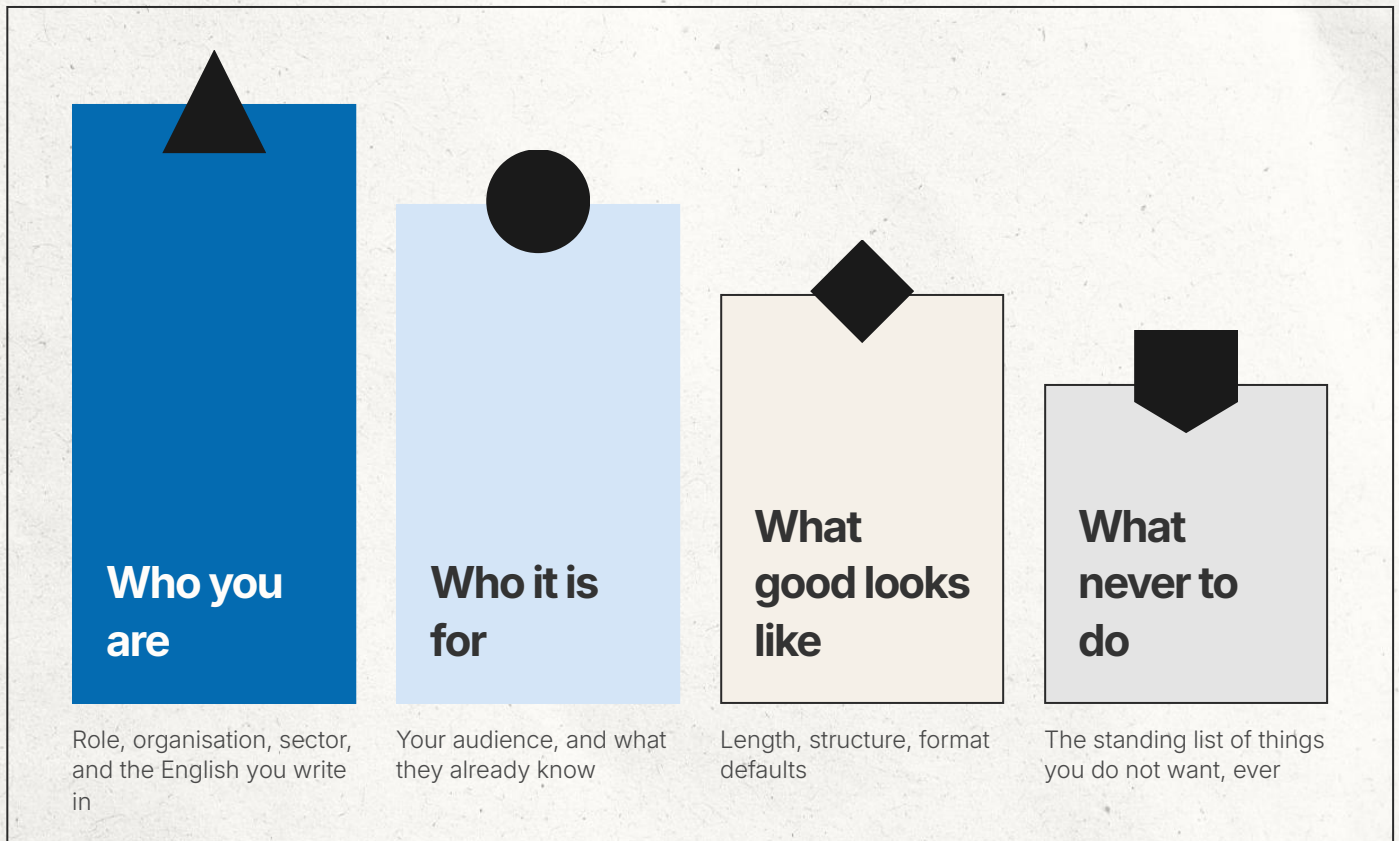


Fig 11.2 The 4 parts of a standing instruction. The bar heights are visual rhythm, not a ranking, though the last one is the part most people skip and the part that changes the most.

11.2

Specific beats polite. "Under 300 words, answer first" changes the output. "Be professional" does not.

Write the nevers. The list of things you do not want is the highest-yield part and the one most people leave out.

Keep it short. A few hundred words is the room you get. This is a brief, not a biography.

The box is small and the instinct is to fill it with adjectives. Resist that. Adjectives are the part an assistant is worst at interpreting, because "professional" is not an observable property of a sentence. Length is. Structure is. A banned opening line is.

Four parts, in this order. Who you are: your role, your organisation or sector, and which English you write in. Who the output is for: the audience, their seniority, and what they already know so it stops explaining your own industry back to you. What good looks like: a default length, whether you want the answer before the reasoning, and whether you want prose or bullets. And what never to do: the standing list.

A worked version, for someone senior enough that most defaults are wrong for them: "I am a commercial director at a mid-sized software company in Europe. Most of what I ask for is read by my executive team, occasionally by our board, sometimes by customers. Assume I know my sector: do not explain the basics back to me. Default to British English. Lead with the answer, then the reasoning. Keep drafts under 300 words unless I ask for longer. Write in plain sentences, and only use bullets when I ask for a list. If a claim rests on a number or a date, say where it came from, and say plainly when you do not know. Never open with a compliment about my question, and never end by offering three more things you could do."

That last sentence is doing more work than the rest combined. The default habits of these tools, the flattering opener, the padded close, the bulleted list of everything, are habits you can simply switch off, permanently, in one line. Nobody does it, because it does not occur to them that the defaults are negotiable.

Where to type it: ChatGPT under Settings then Personalization, Claude under Settings then the Instructions for Claude field on your profile, Gemini under Settings and help then Personal Intelligence then Instructions for Gemini, Copilot under Settings then Personalization. In all four the exact wording of the menu has moved at least once in the past year, so look for the word personalization and you will find it.

Write your version now, in the exercises, and paste it in before you read the next section. You will notice the difference in the very next chat.

11.3 Some context belongs to one job

Your standing instruction applies to everything you ask. That is exactly wrong for the details that only matter on one recurring piece of work, and stuffing them in there ruins the instruction for everything else.

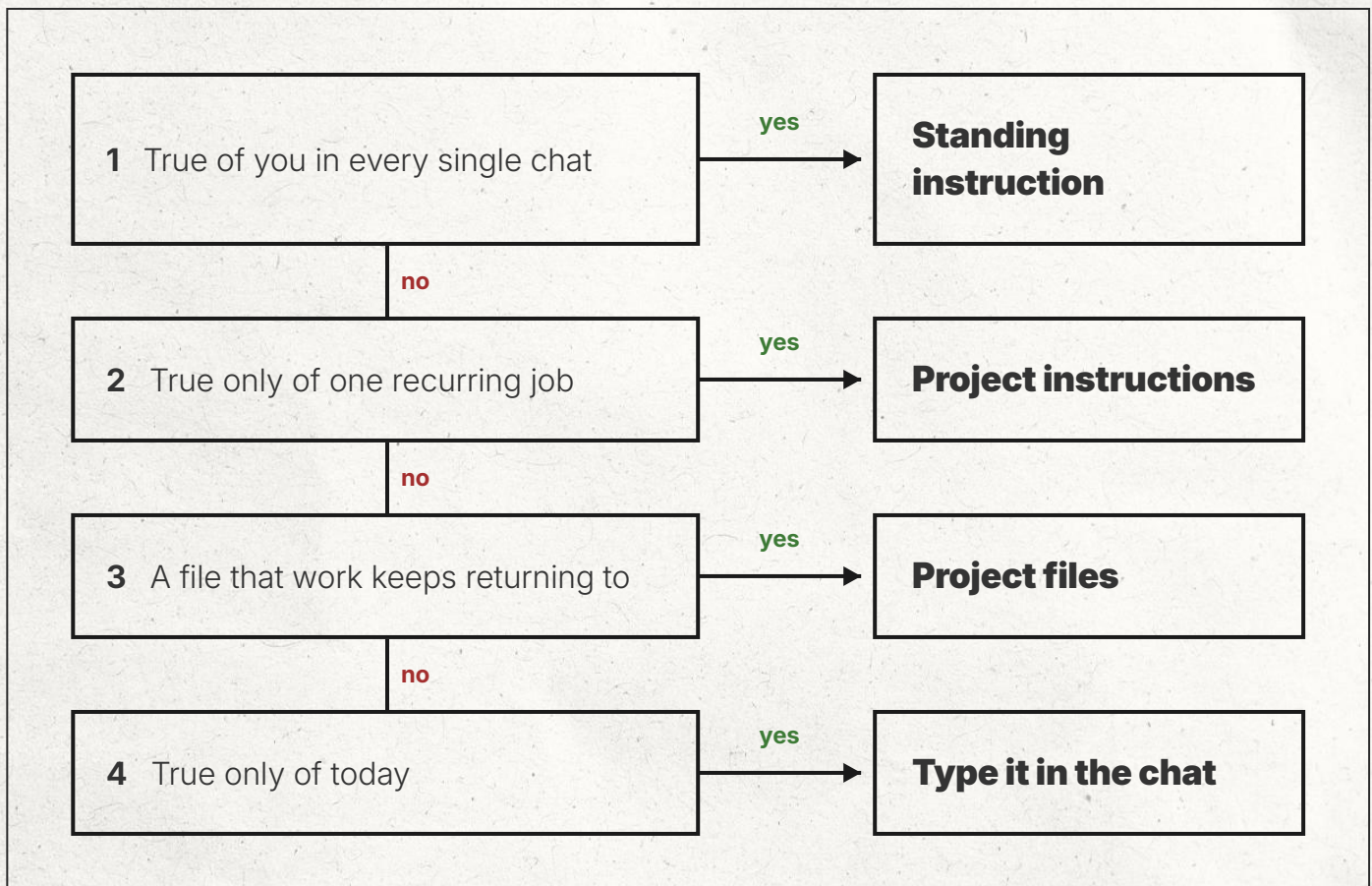


Fig 11.3 Where a piece of context belongs. Read down until a line is true. The test is not how important the context is, it is how often it applies.

11.3

ChatGPT Projects. Files, project-only instructions, and an option to keep that project's memory separate from your account. On free too, with a smaller file cap.

Claude Projects. Its own instructions plus an uploaded knowledge base. Free accounts get up to 5.

Gemini Gems. Fixed instructions plus reference files, free included. Note that your Instructions for Gemini do not apply inside a Gem.

Copilot Notebooks. Gathers the files, pages and chats behind one topic. Availability widened through 2026, so check what your licence includes.

A standing instruction is a blunt tool by design: it fires on every chat you ever open. So the moment you put something in it that is true of only one client, one product or one quarter, you have made every unrelated conversation slightly worse.

The fix is a container. All four assistants now have one: a named workspace that holds a set of chats, a set of files, and its own instructions that apply inside it and nowhere else. ChatGPT and Claude both call it a project. Gemini calls its version a Gem. Copilot calls its version a notebook. The mechanics differ in the details and match in the thing that matters: context scoped to a job rather than to you.

Pick one piece of recurring work to start with, and be strict about "recurring". The board pack you assemble every month. The weekly commercial update. The proposal format you have now written 14 times. Then put 3 things in it: the files that work always refers back to, which is the Chapter 7 habit made permanent instead of re-uploaded every time, the instructions specific to that job, and the chats about it.

There is a second reason to use one, and it is about separation rather than convenience. ChatGPT lets you set a project to draw only on what is inside it, keeping your account-wide memory out and keeping anything learned inside from leaking back. Claude gives each project its own memory space by default. If you work across clients who should never colour each other, that isolation is the feature, not the file storage.

What you are doing here is personal setup: your assistant, configured for your work. Packaging a repeated job into something a colleague can open and use is a different exercise with different requirements, and it is Chapter 32.

Set up exactly one project today. Not six. The one you would otherwise re-explain on Monday.

11.4 What persists can be wrong

Everything in this chapter cuts both ways. Context that carries between chats without being asked for also carries mistakes between chats without being asked for, and it does it silently.

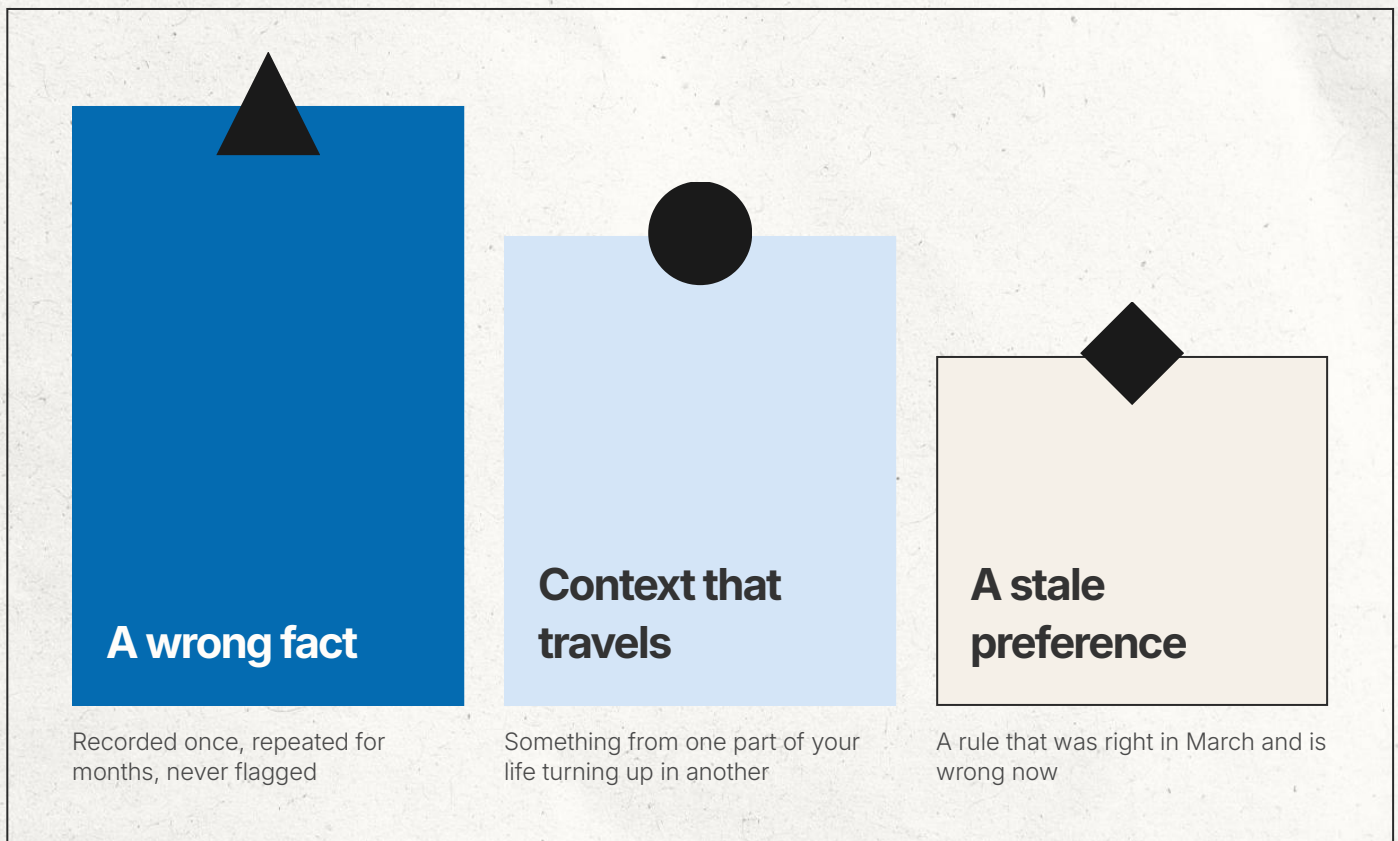


Fig 11.4 The 3 ways persistent context goes bad. Heights are visual rhythm only. None of the three announces itself: the output just gets quietly worse, and you blame the model.

11.4

Deleting the chat is not enough. Deleting a conversation does not delete the memory it produced. They are two separate jobs.

One chat, no memory.

Temporary chat in ChatGPT and Gemini, incognito chat in Claude, for anything you do not want carried forward.

Book the review. 10 minutes a quarter, in the diary. Read what it holds, delete what is wrong.

Mention once, in passing, that you are launching in March, and an assistant with memory may write that down as a fact about you. Six months later, long after the date moved, it is still quietly shaping answers on that basis. Nothing warns you. There is no error message for a stale belief, and the output does not look wrong, it just looks slightly off in a way you attribute to the model.

The second failure is context arriving somewhere you did not want it. Something you mentioned about your health, your family or your job hunt surfacing in a draft for work. One client's situation informing a reply about another. This is not a malfunction: it is the feature working exactly as built, in a context you did not have in mind when you fed it. The escape hatch is the single-chat mode every assistant now has, called temporary chat in ChatGPT and Gemini and incognito chat in Claude, which leaves no trace in history and feeds nothing forward.

Chapter 9 gave you a rule about what never goes in the box. Extend it to the settings box: a standing instruction and a memory both persist, and on a company account an administrator can search and export them, so anything too sensitive for a chat is too sensitive for your custom instructions.

One more thing worth knowing without losing sleep over it. Because assistants can read documents and web pages, security researchers have shown it is possible to hide instructions inside a file or a page that get written into an assistant's memory and then act on every later conversation. Specific cases have been demonstrated and patched. It is a reason to read your memory occasionally, not a reason to turn everything off.

The habit that covers all of this costs 10 minutes a quarter. Open the memory screen and read it as if someone else wrote it, because someone else did. Delete anything out of date, correct anything wrong, and check your standing instruction still matches the job you actually have now. Note while you are there that clearing your chat history is a different action: deleting the conversations does not delete what was learned from them.

Put the review in your diary now, dated 3 months out, before you close the book. It will not happen otherwise.

READING ENDS HERE

Everything before was reading. Everything after is doing.

3 exercises, About 15 minutes and one real decision that opens Chapter 12.

Do these before Chapter 12

3 exercises. About 15 minutes

Draft the first 3 parts of your standing instruction. Short phrases, not sentences you are proud of. The 4th part is the next exercise.

WHO YOU ARE

WHO IT IS FOR

WHAT GOOD LOOKS LIKE

Write your never list: the things you never want it to do, however you phrase them. Then type all 4 parts into your assistant's personalization settings, and note here where that setting lives so you can find it again.

Name one piece of work you do repeatedly, and list the 2 or 3 files it always needs. Set up a project, Gem or notebook for it, and put those files in.

Chapter 11 on one page

THE TWO KINDS

Instructions you set (your words, every chat, until you edit them) and memory it writes (its words, chosen by it, from things you mentioned)

THE 4-PART BRIEF

Who you are · who the output is for · what good looks like (length, order, format) · what never to do

WHERE IT LIVES

ChatGPT: Settings, Personalization · Claude: Instructions for Claude · Gemini: Personal Intelligence, Instructions for Gemini · Copilot: Settings, Personalization

THE CONTAINER

Projects (ChatGPT, Claude), Gems (Gemini), Notebooks (Copilot). Recurring job, its files, its own instructions. One job per container

THE QUARTERLY REVIEW

Read the memory, delete what is wrong, recheck the instruction. Deleting a chat does not delete the memory it produced

BRING TO CHAPTER 12

Your saved standing instruction and the one project you set up. Chapter 12 turns the setup into a rhythm you keep.

BOTTOM LINE

You have told your assistant once who you are and what good work looks like, and you know how to check what it decided to remember on its own.

NEXT CHAPTER

Build Your Weekly AI Routine

Chapter 12 closes AI 101. It reduces the first 11 chapters to the 5 uses you will genuinely repeat, and the weekly rhythm that keeps them from quietly falling out of your week by November.

BRING WITH YOU

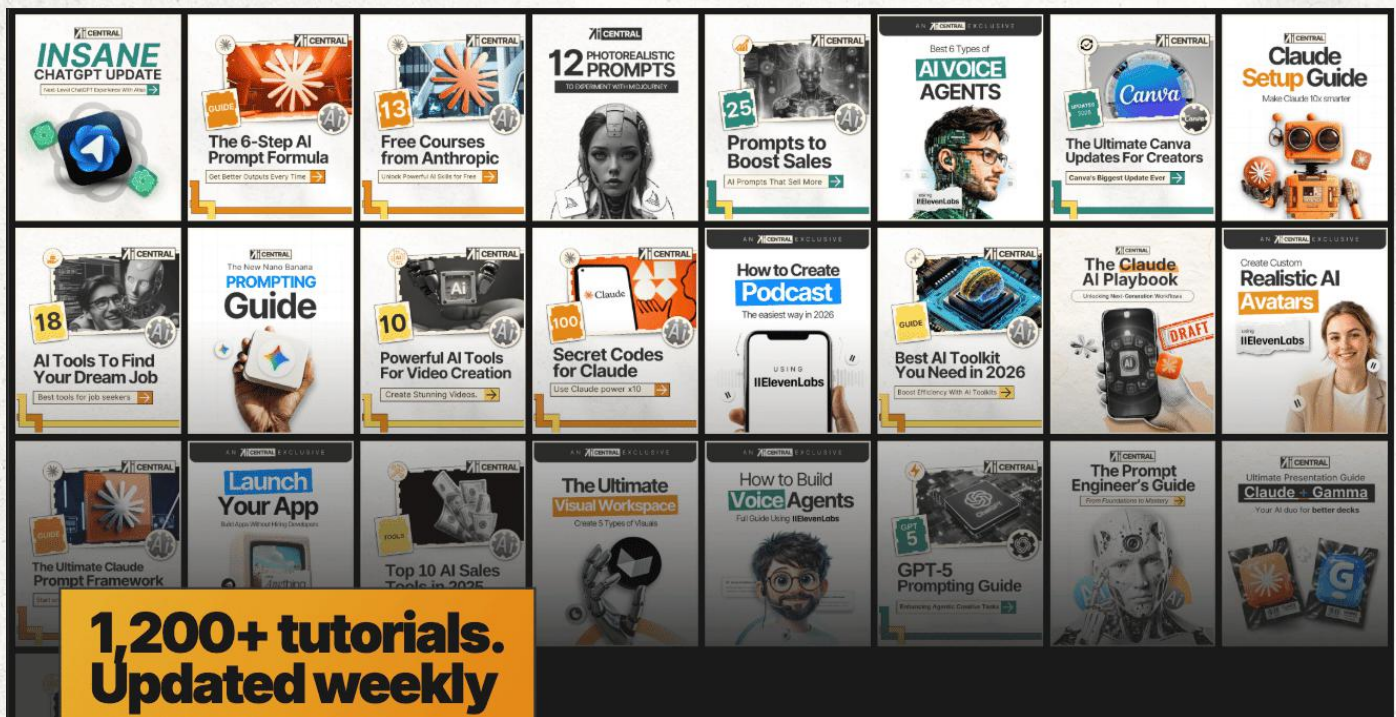
- ☐ Your standing instruction, saved in settings
- ☐ The project you set up, with its files already in it

GO FURTHER

The Ultimate AI Library

Every guide, prompt pack and playbook AI Central has published, in one place. The reference shelf this course keeps pointing you to.

explore the library



WHO MAKES THIS



AI Central Media is a London-based editorial team. Our flagship publication covers practical AI for senior professionals, and 100+ AI companies, SaaS platforms and growth teams have advertised with us since 2023.

100+

AI companies, SaaS platforms, education brands and growth teams have advertised with us since 2023

613K

Accounts reached every month, across the 7 channels we operate

151

Countries our readers are in, plus all 50 US states

PROUD PARTNER OF

GAMMA

IIElevenLabs

guidde.

Outskill

Luma AI

HubSpot

attio

SSynthflow

Notion

Replit

We turn attention into pipeline.

visit thecentral.ai

