The AI Bootcamp

27

What AI Can Do Without Asking You

In this chapter you'll learn

The 4 tools it already has, an honest read on how well they work, and one rule for what to permit

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 A LinkedIn carousel here, a YouTube tutorial there, a $2,000 cohort course that goes out of date before the refund window closes.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 can name the tools your assistant already has and which of them act outside its own sandbox, say honestly which jobs are worth handing over and which are not, and apply one rule to decide what it may do without asking you first.

Chapter 26 gave it reach. What follows is not a technical question, it is a question about consequences, and nobody hands you a rule for it. So most people answer it by feel: permit everything and hope, or permit nothing and quietly stop using the thing. Both are expensive, and the second one is more common among people who have read one bad news story.

CONTENTS

27.1 A tool is what turns advice into action

27.2 The demonstration is real, and your Tuesday is different

27.3 A wrong answer you can catch, a wrong action has already happened

27.4 The page it reads can give it instructions too

TOPICS REQUIRED

Chapter 26. You need your assistant from Chapter 2 and at least one connection you have already made to somewhere your work lives. If you turned nothing on in 26, read this chapter first and then go back, because this is the chapter that tells you what you were agreeing to.

AFTER THIS CHAPTER

Chapter 28 assumes you have decided what you are willing to let run without you, because the first thing you build there runs without you.

27.1 A tool is what turns advice into action

Ask it a question and you get words back. Give it a tool and something happens: a search runs, a program executes, a file is written, a button gets clicked in a browser that is signed in as you.

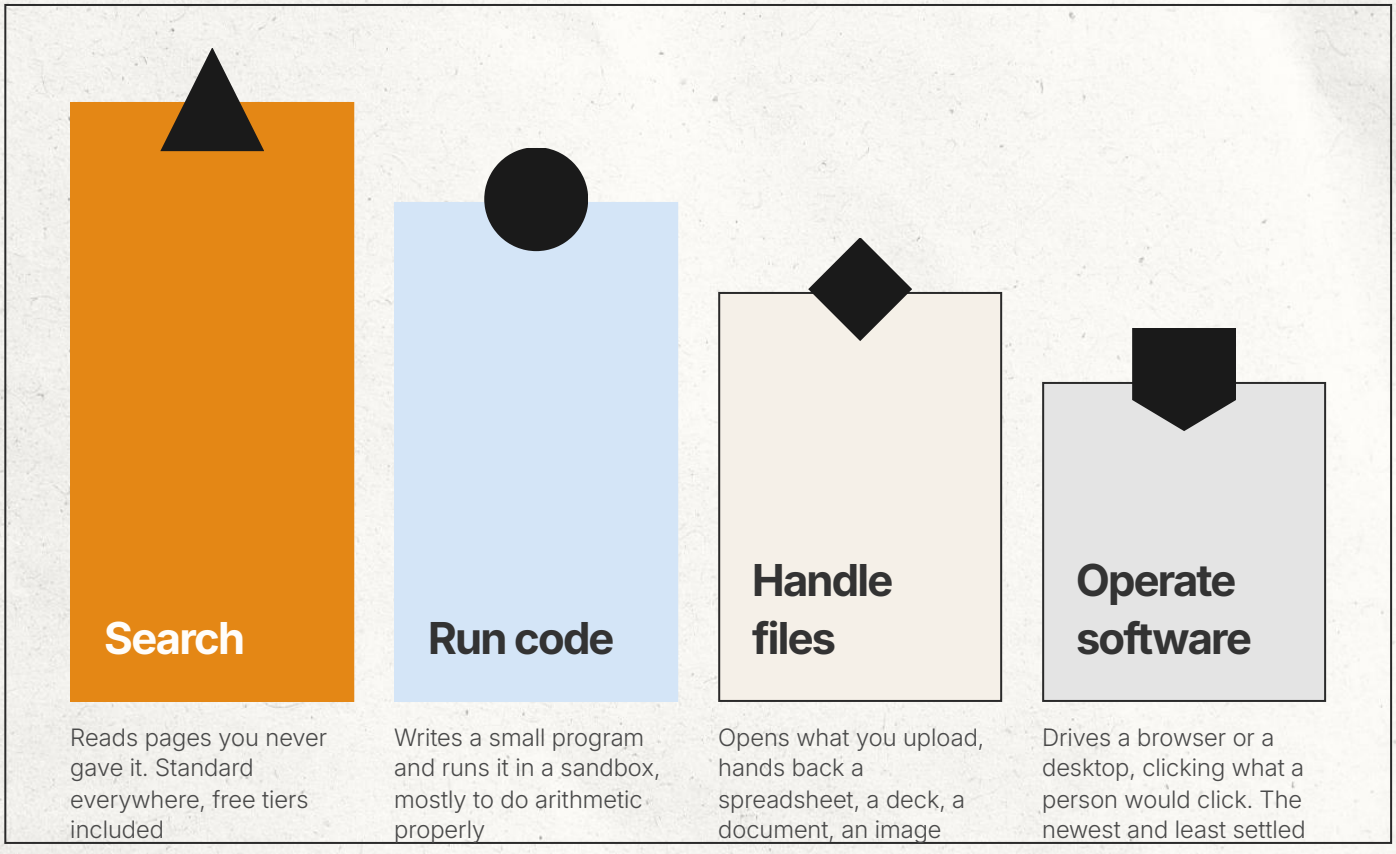


Fig 27.1 The 4 things a mainstream assistant can already do without you. Ordered by how long each has been dependable, oldest first, which is not the same as most useful.

27.1

ChatGPT. Agent mode, on the paid plans, drives a browser on a machine of OpenAI's. The separate Atlas browser was folded into the main desktop app and shut down in August 2026.

Claude. A Chrome extension that clicks inside your own browser, on the paid plans since December 2025, plus a separate desktop tool that works on folders you choose.

Gemini. Spark, its personal agent, on the top Ultra plan in selected countries. Connections to Gmail, Calendar and Drive are off until you turn each one on.

Copilot. Hardest to pin down: Microsoft spent August 2026 merging its consumer and work Copilots into one app, so what you see depends on the month.

Chapter 26 was about reach: which of your systems it can see. This chapter is about the moment it stops describing what it would do and does it. That moment is smaller than you expect. There is no separate agent product to buy and usually no new instruction to learn. You ask for something in the same plain sentence you have been using since Chapter 5, and a line of grey status text appears saying it is searching, or running an analysis, or opening a page.

Underneath, a tool is a boringly simple thing: a capability the model is allowed to call, plus a description of when calling it makes sense. The model decides to reach for it, the software runs it, and the result comes back into the conversation as more text for it to read. That last part matters more than it sounds, and section 27.4 is about why.

Three of the 4 tools in the figure stay inside the vendor's own box. A web search reads public pages. Code runs in a sandbox that is thrown away afterwards, which is how a model that cannot really do arithmetic gets your numbers right: it writes a few lines of Python and runs them rather than guessing. File handling reads what you upload and writes something back for you to download. Image generation belongs in this group too, and it is the least consequential tool on the list, because a picture you do not like costs you nothing but the wait.

The 4th is different in kind. When it operates a browser or a desktop, it is not in a sandbox, it is in your session, signed in as you, with your permissions. It cannot do anything you could not do. That sounds reassuring until you turn it around: it can do anything you can do, and nothing in the software asks whether the person at the keyboard meant it. Every safeguard that exists is the vendor putting a question in front of you, or you having arranged, in advance, for the action to be undoable.

That is the whole shape of this chapter. The first 3 tools are conveniences with a low ceiling on how badly they can go. The 4th is a genuine transfer of authority, and it deserves a decision rather than a shrug.

Before you read on, open the settings page where your assistant lists its tools and connections and read it top to bottom. Most people have never once looked, and almost everybody finds something switched on that they did not switch on.

27.2 The demonstration is real, and your Tuesday is different

Watching an agent book a restaurant in a launch video is impressive. Watching one do your actual errand is a slower and stranger experience, and it is worth knowing the size of that gap before you plan around it.

Worth handing over

Long, dull, repetitive, and cheap to run a second time.

TRAITS

· Pulling the same 5 facts off 30 different pages
· Filling one form the same way, many times over
· Reading a queue and sorting it into piles

PROS

+ You are not needed while it works
+ A bad run costs you a re-run, nothing else

CONS

− Slower per item than you doing one yourself
− You still have to read the result

Worth keeping

Short, judged, one-shot, and expensive to get wrong.

TRAITS

· Anything with a person on the other end of it
· Anything priced, signed, scheduled or sent
· Sites that actively fight automation

PROS

+ You are already fast at these
+ Your judgement is the part nobody can hand over

CONS

− These are exactly what the demos show
− Tempting to automate first, for that reason

Fig 27.2 The dividing line is not how hard the job is. It is what one failed attempt costs you.

27.2

The benchmark. On a 369-task test of real desktop work, people finish about 72%. The best submitted agent runs of 2026 now score in the 80s and above, with a budget of 100 steps per task.

The field. A three-week study published this month logged 1,258 real commands across 12 desktop applications. The best model completed 52.5% of them.

The arithmetic. A step that works 95% of the time works 36% of the time when a job needs 20 of them in a row. Length is the enemy, not difficulty.

Here is the honest picture, and it has moved in the last year, so anything you read in 2025 is now wrong in both directions.

The measured capability is genuinely high. On OSWorld, a standard test of 369 real tasks in real desktop applications, human beings complete about 72%. Through 2025 the best agents managed somewhere between a quarter and a half of that. Through 2026 the leading submitted runs have gone past the human number. If your mental model is still that these things cannot click a button, update it.

Read the small print, though, because it is where your Tuesday lives. Those runs get up to 100 attempts at each task, the tasks are known in advance, nobody is waiting on the result, and the scores are submitted by the people who built the agents. Put the same technology in front of unrehearsed work and the number drops hard: a three-week study published this month logged 1,258 real commands issued to a computer-use agent across 12 desktop applications, and the best model completed just over half.

The second thing the demos hide is the clock. An agent driving a browser is slower than you doing the same thing once, often much slower, because it looks at the screen, decides, clicks, waits for the page, and looks again. Its advantage was never speed per item. It is that you are not in the room. Twenty minutes of it grinding through a list while you do something else is a good trade. Twenty minutes of you watching it grind is not, and watching it is exactly what most people do the first few times.

The failures are rarely dramatic, which is the part worth internalising. It does not crash. It takes a plausible wrong turn at step 4, a cookie banner it reads as a form, a search result that looks like the right company, and then builds the remaining 16 steps carefully on top of that. The arithmetic in the notes is the whole story: reliability that would be excellent in a single answer is mediocre across a long chain, and everything after the wrong turn is confidently wrong.

So pick the work by what a failure costs, not by how clever the task sounds. Choose one job this week that you would be perfectly happy to have done twice.

27.3 A wrong answer you can catch, a wrong action has already happened

This is the change that matters, and it is not really about AI. Everything you learned in Chapter 8 about checking output assumed there was a moment between the answer and the consequence. Acting removes that moment.

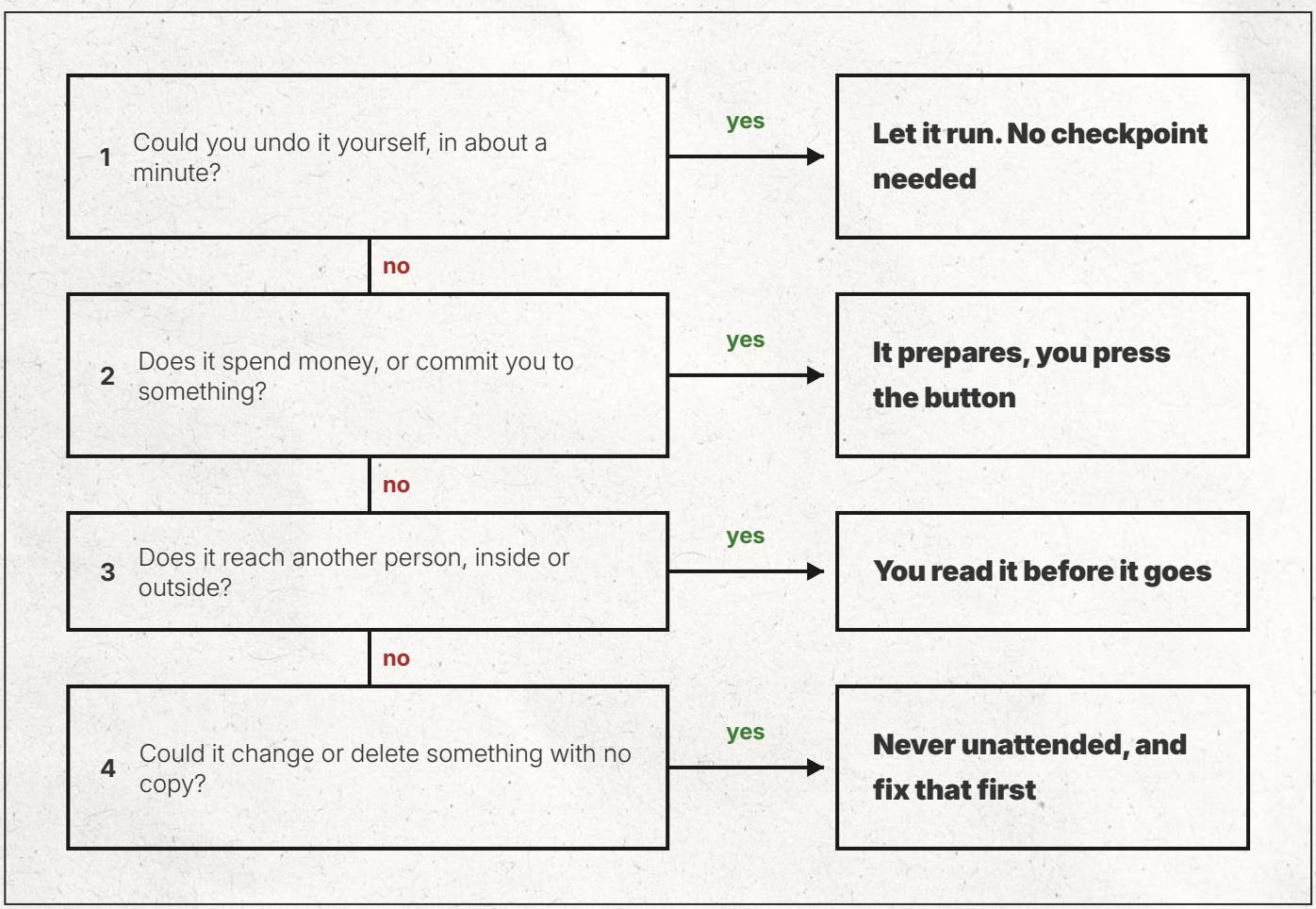


Fig 27.3 Read down until a line is true. Notice that not one of these rows asks how good the AI is.

27.3

Reversible is arranged. Version history, a drafts folder, a bin that keeps things for 30 days, a backup. If none of those exist for this job, the action is permanent however small it looks.

Small is not safe. A 2-line email is tiny and cannot be recalled. A rewrite of a 40-page document is large and sits in version history. Size and reversibility are unrelated.

The vendors agree with you. The confirmation prompts shipped in these products name the same 3 things: publishing, purchasing, and sharing personal data.

Up to now, every mistake this book has warned you about arrived as words on a screen. That is a forgiving failure mode. You read it, you notice the invented statistic, you delete it, and the cost was 30 seconds. An action skips that step. The email is sent, the row is deleted, the order is placed, and your review, however careful, is now an autopsy.

Which means the permission decision is not the question people think it is. They ask "is it good enough to trust yet", which is a question about the technology and has no stable answer. The useful question is "if this particular action is wrong, what does it cost me to put right". That is a question about your own setup, you can answer it in seconds, and the answer will still be right in five years when the models are twice as capable.

So here is the rule, and it fits on a business card. Anything you could undo yourself in about a minute, let it do unsupervised. Anything that spends money, reaches another human being, or otherwise leaves your control, it may prepare and you press the button. Anything that changes or destroys something with no copy anywhere, it does not do at all until you have made a copy.

Notice that reversibility is not a property of the request. It is a property of the environment you put the request in. In February 2026 a man asked Claude's desktop tool to tidy up the files on his wife's computer, and it deleted their family photos. The photos came back, from a cloud backup nobody had made for this occasion. The action was recoverable because somebody, months earlier, had arranged for it to be recoverable. That is the lever you actually control: not how careful the assistant is, but whether the thing it touches has a bin, a backup or a version history behind it.

Do not expect competence to do this job for you. In a test published in August 2026, 7 current models were run against interfaces carrying deliberate traps. Every single one of them clicked Checkout on a $25,000 order without pausing to ask. None of them noticed that a button labelled Optimize was in fact a factory reset. The same models refuse harmless things elsewhere. Their sense of consequence does not match yours and cannot be relied on to, so the checkpoint has to be structural, not vibes.

Write the two lists before you switch anything on. Not a policy document, 6 lines on a page: what it may do while you are not looking, and what waits for your hand. Keeping the second list honest once things are running is Chapter 34's subject. Deciding what goes on it is yours to do now.

27.4 The page it reads can give it instructions too

An assistant cannot reliably tell your instruction apart from an instruction it finds inside the material it is reading. This is not a bug being patched this quarter. It is how the thing works.

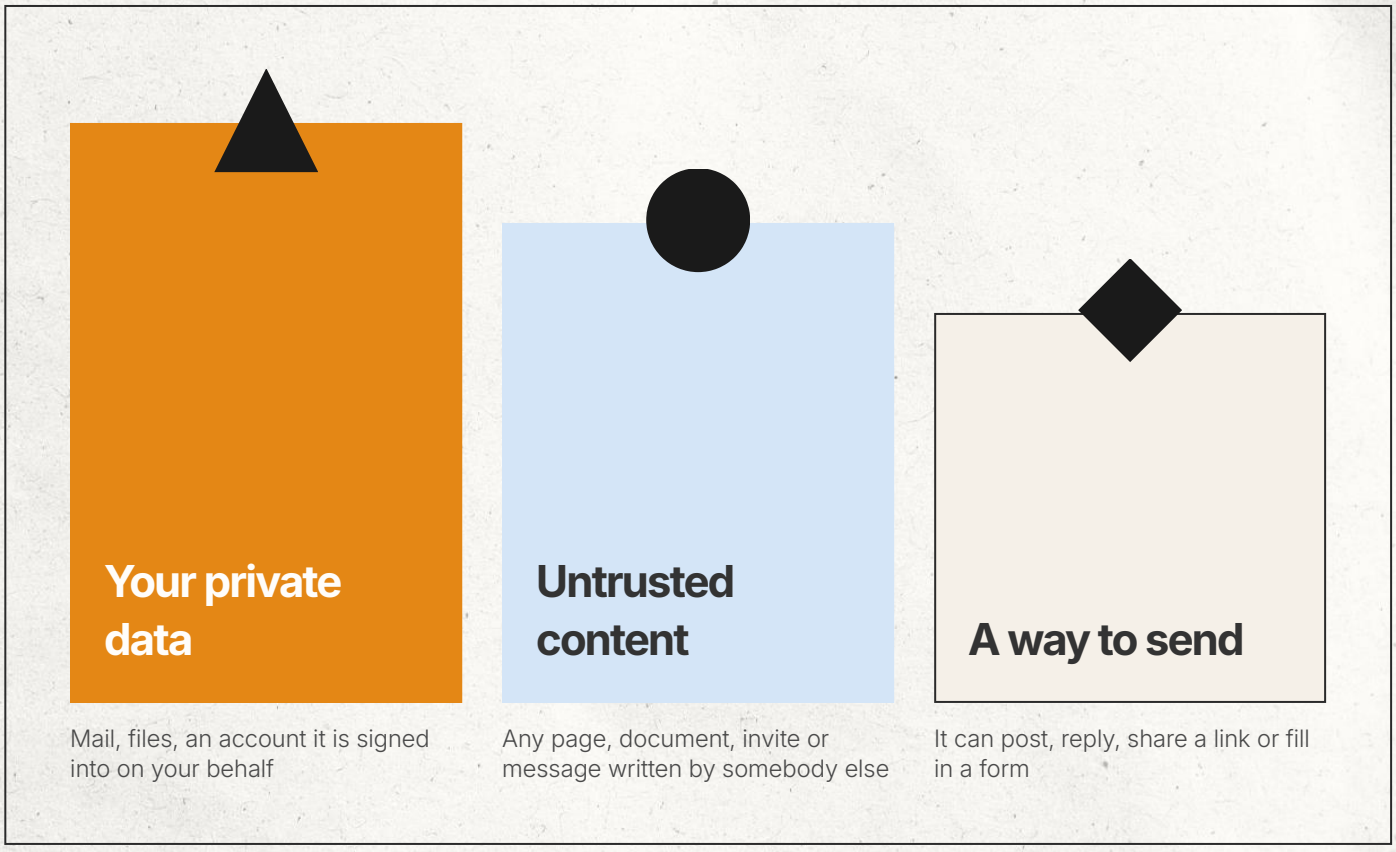


Fig 27.4 Any 2 of these together are ordinary and safe. All 3 in one session is the shape an attacker needs. The bar heights are visual rhythm, not a ranking.

27.4

What it looks like. White text on a white background, a comment in the page source, a line buried in a PDF or a calendar invite. Written for the assistant, invisible to you.

What the fixes buy. One vendor's own red-teaming: 23.6% of targeted attacks worked before mitigations, 11.2% after, and 0% on the browser-specific set. Better, and not solved.

Who says it is unsolved. OWASP ranks this the number 1 risk for these systems and says no fool-proof prevention is known. The UK's national cyber security centre says the same in blunter words.

Everything an assistant reads arrives as text, and it treats text as something to be understood and acted on. It has no reliable way to mark one paragraph as your instruction and the next as somebody else's content. So a web page can carry a line addressed to it rather than to you, and it may simply comply. The trade calls this prompt injection. You can call it the page talking back.

The demonstration that made this concrete came from the Brave browser team. Hidden instructions on an ordinary-looking page told an agentic browser to go and look at the user's own account, read a one-time login code out of their email, and post it publicly. It did. Nothing was hacked in the traditional sense. The agent was signed in as the user, it read some text, and it followed it. Decades of browser security rest on the assumption that a page cannot reach into your other tabs. An agent with your permissions walks straight through that assumption.

The mitigations are real and worth turning on: per-site permissions, whole categories of site blocked outright, and a confirmation before anything is published, purchased or shared. They move the numbers a long way, from roughly a quarter of targeted attacks succeeding to roughly a tenth in one vendor's own testing. In security, a tenth is not a pass mark. Independent work through 2026 has been blunter still. One study this August split a hostile instruction into innocuous-looking pieces across a chain of pages and pushed the average success rate against 6 leading agents from 31% to 37%, with a single model going from 42% to 73%. Another concluded that agentic browsers need rearchitecting before they are ready for the web as it actually exists.

The practical defence for you is not technical and does not require you to understand any of that. Keep the 3 ingredients in the figure apart. An agent reading pages written by strangers should not be doing so in a browser signed into your bank, your mail and your company drive. Use a separate browser profile, or a narrow account, and grant it sites one at a time. The most valuable thing you can do costs nothing: reduce what it is holding while it reads things you do not control.

None of this makes the answer "never let it act". The answer is the one from the last section, which happens to survive contact with this section unchanged: permit the work where being wrong is cheap, put yourself in front of the work where being wrong is not. An injected instruction that makes an agent re-sort a folder is a nuisance. The same instruction reaching an agent that can send mail from your address is a different afternoon.

Go and look at what your assistant is allowed to reach right now, and take one thing off the list. Not because something has gone wrong, but because the list grew by accident and nobody has ever pruned it.

READING ENDS HERE

Everything before was reading. Everything after is doing.

3 exercises, About 15 minutes and one real decision that opens Chapter 28.

Do these before Chapter 28

3 exercises. About 15 minutes

Open the settings where your assistant lists its tools and connections. Read the whole list before you write anything.

SWITCHED ON BY DEFAULT	CONNECTIONS YOU ADDED YOURSELF	ANYTHING HERE YOU CANNOT EXPLAIN

Take one job you would genuinely hand over this month. Answer these about the action, not about the AI.

THE ACTION, IN ONE LINE	HOW YOU WOULD UNDO IT	HOW LONG UNDOING TAKES

Now run the smallest version of that job and stay in the room while it works. Write down the first thing it did that you did not expect, and whether the rule in 27.3 would have caught it.

Chapter 27 on one page

THE PERMISSION RULE	Undo it yourself in a minute, let it run · spends money, reaches a person, or leaves your control, you press the button · no copy exists anywhere, it does not touch it until one does
THE 4 TOOLS	Search the web · run code in a sandbox · read and write files · operate a browser or desktop. The first 3 stay in the vendor's box. The 4th acts as you, with your permissions.
REVERSIBILITY IS ARRANGED	Version history, a drafts folder, a 30-day bin, a backup, a spare account. If none exists for this job, treat the action as permanent no matter how small it is
THE 3 INGREDIENTS	Private data, content written by someone else, and a way to send things out. Any 2 is an ordinary day. All 3 in one session is the risky shape
NEEDED BEFORE CHAPTER 28	One job you have decided may run unattended, and one you have decided may not

BRING TO CHAPTER 28

Your two lists: what may run while you are not looking, and what waits for your hand. Chapter 28 builds the first thing that runs without you, and it should come off the first list.

BOTTOM LINE

You know what your assistant can do on its own, roughly how well it does it, and exactly where you intend to stand in the way.

NEXT CHAPTER

Build Your First AI Automation

Chapter 28 is the build. One trigger, one action, one result, running without you being there, and it comes off the list of permitted work you just wrote rather than off a list of impressive demos.

BRING WITH YOU

- ☐ The list of jobs you decided it may do unattended
- ☐ One recurring task from your week that is small, dull and reversible

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

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 · Synthflow · Notion · Replit

We turn attention into pipeline.

visit thecentral.ai ↗