The AI Bootcamp

25

What an AI Agent Actually Does

In this chapter you'll learn

What an agent actually is, why 95% a step is not 95% overall, and where the line sits today

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 A LinkedIn carousel here, a YouTube tutorial there, a $2,000 cohort course that goes out of date before the refund window closes.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 explain what an agent is without using the word autonomous, work out from a job's step count how often it will finish clean, and say which parts of your own recurring work are safe to run unwatched and which are not.

This is where the assistant starts doing things you will not watch. That is a change in kind, not a change in degree, and it is the most heavily sold idea in the field, which means you have almost certainly been pitched a version of it that is not true. You get the definition, the arithmetic and the measured evidence here, before Chapter 26 connects anything to your real accounts.

CONTENTS

25.1 The change is not that it got smarter, it is that something happens

25.2 Reliability multiplies, it does not average

25.3 What it can actually finish, measured rather than pitched

25.4 Reversible work can run, irreversible work still needs you

TOPICS REQUIRED

The whole of AI 202, and Chapter 24 in particular: your own prompt playbook, plus the failure taxonomy from Chapter 23. One more thing to have with you, and this chapter is much thinner without it: a job you do every week, the same way every time, that you could describe to someone else as a list of steps.

AFTER THIS CHAPTER

Chapter 26 assumes you have decided this is worth doing and know what you are taking on. It starts connecting the assistant to the drive, the mail and the notes where your work already sits.

25.1 The change is not that it got smarter, it is that something happens

For 24 chapters everything it produced was text on a screen, and nothing was real until you decided it was. An agent closes that gap and takes the step itself.

An assistant that replies

You ask, it answers, and nothing happens in the world until you decide it should.

TRAITS

· One turn: your question in, text out

· The output is a draft, every time

· A mistake costs you the time it took to read it

PROS

+ You see everything before any of it counts

+ A wrong answer stays on the screen where it can do nothing

CONS

– Every step of the actual work is still yours

– It does nothing at all while you are asleep

A system that acts

You give it a goal, and it picks its own steps, uses tools, checks what came back, and keeps going until it decides it is done.

TRAITS

· Many turns, most of them taken without you watching

· The output is a thing that has already happened

· A mistake costs whatever the action cost

PROS

+ The work continues while your attention is elsewhere

+ Handles jobs too long and too dull to sit through

CONS

– Small error rates compound across the steps, see 25.2

– You are accountable for actions you never read

Fig 25.1 The difference is one row deep: what arrives at the end. Everything harder about the right-hand column follows from the output being an event rather than a draft.

25.1

Goal, not question. You stop specifying the answer you want and start specifying the finish line. It picks the route.

Workflow or agent. Anthropic's split: a workflow follows a path fixed in advance, an agent chooses its own. Most things sold as agents are workflows, and that is often the better buy.

The word is marketing. Gartner counted roughly 130 vendors genuinely doing this among the thousands claiming it. Ask what a tool can touch, not what it is called.

Here is what actually changes, and it is worth being exact about it, because nothing else in this book has been sold to you as hard as this has. For 24 chapters, everything the assistant produced was text on a screen. You read it, you judged it, and only then did anything real happen. From here, it does the thing.

The plain definition: an agent is a system you give a goal to rather than a question, which then chooses its own steps to reach that goal, uses tools to take them, looks at what came back, and decides what to do next, until it judges itself finished. Anthropic draws the useful line here. A workflow, in their words, runs through predefined code paths that somebody wrote in advance. An agent is a system where the model "dynamically directs its own processes and tool usage". One follows a route you fixed. The other picks the route.

That distinction is not academic, because it decides how the thing fails. A workflow can only fail in ways you can list beforehand. An agent can fail in ways nobody listed, including by doing something perfectly sensible that you would never have chosen. Most products currently sold to you as agents are workflows, and that is frequently the better purchase: cheaper, faster, and far easier to work out what went wrong.

You should assume the label tells you nothing. When Gartner went looking, it counted somewhere around 130 vendors genuinely offering what it would call agentic, out of thousands making the claim. The industry name for the rest is agent washing. So ignore the word on the box and ask two questions instead: what is this allowed to touch, and what happens if it is wrong. Those two answers are the entire product.

And one thing shifts the moment it acts, which the next three sections keep returning to. Up to now a bad answer was a bad answer, sitting on your screen, costing you a minute. Now a bad answer is an email that went out, a record that changed, a file that was overwritten. The output stopped being a proposal and became an event with your name on it.

Before you read on, write down one job you would hand over if you trusted it completely. Keep it where you can see it. The rest of this chapter is about whether you should.

25.2 Reliability multiplies, it does not average

This is the single most useful idea in the chapter, and it is arithmetic rather than opinion. A system that is 95% reliable at each step is not a 95% reliable system.

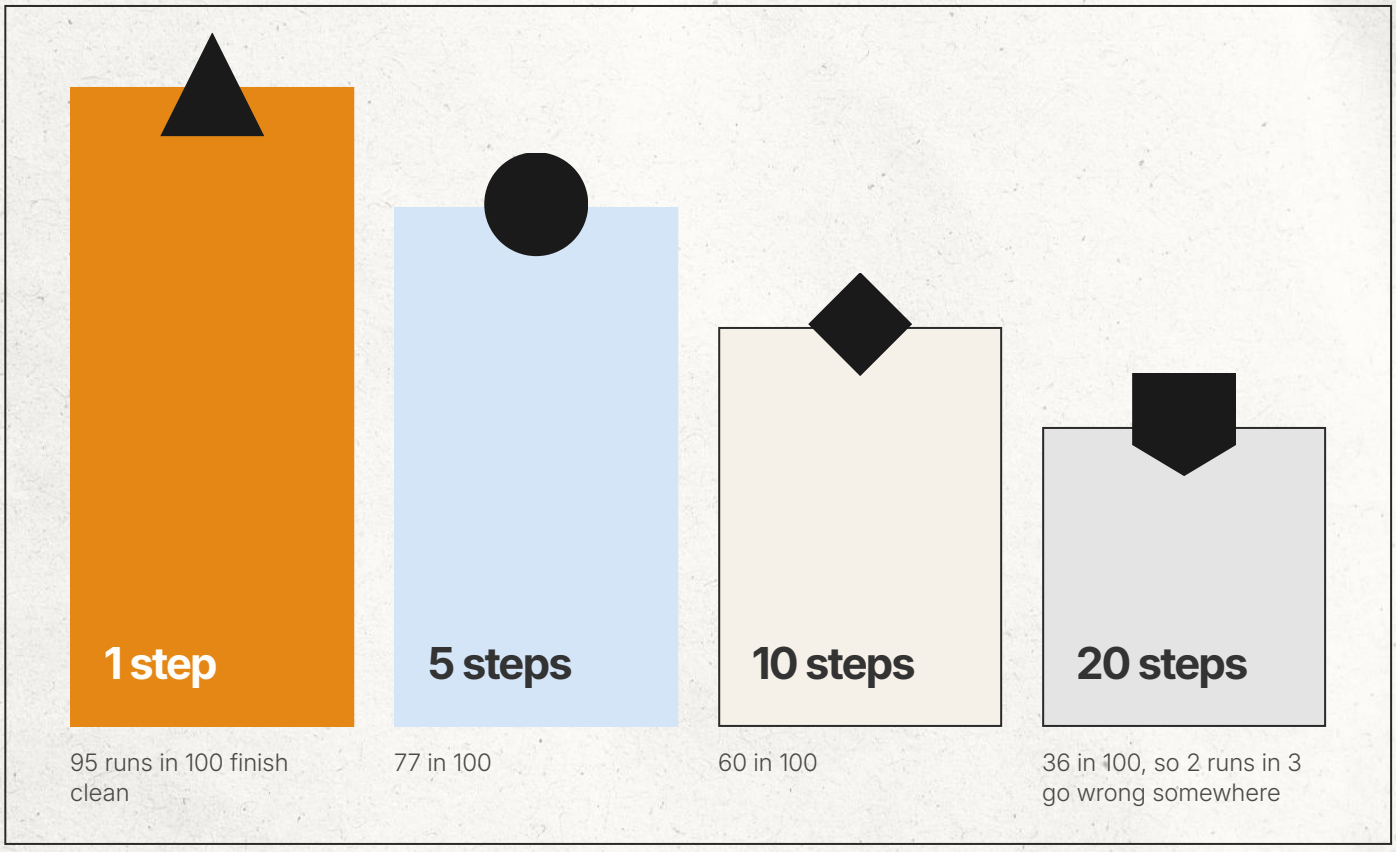


Fig 25.2 One step at 95%, then the same step chained. Bar heights track the numbers, except the last, which is drawn taller than its true share so the label fits. Nothing here is a claim about any product: it is multiplication.

25.2

It multiplies. 95% a step, across 20 steps, is 36% end to end. Not 95%, and not 75% either.

Errors feed forward. A wrong figure fetched at step 2 is treated as fact by every step after it. Nothing goes back to check.

Why the demo worked. A demo is 3 steps long. Your real job is 15. The demo was honest and still told you almost nothing.

Here is the thing the demos cannot show you, and it is not a criticism of anybody: it is multiplication. Suppose each individual step in an automated job goes right 95% of the time. That sounds like a system you would happily sign off on. Now chain 20 of those steps together. The chance that all 20 go right is 0.95 multiplied by itself 20 times, which is 36%. Roughly 2 runs in 3 fail somewhere.

Put it against a real job. Say your Monday report takes 8 steps: pull this week's numbers, pull last week's file, match the line items, find the 3 biggest movers, look up why the largest one moved, write the commentary, format it, send it to 6 people. Eight steps, each right 95% of the time, gets you a clean report on 66 Mondays out of 100. A third of your Mondays are wrong somewhere, and you did not choose which third.

The reason this surprises people is that they expect errors to average out. They do not. They feed forward. A number fetched wrongly at step 2 is not questioned at step 5, it is written up in confident prose at step 6 and emailed at step 8. The system never says the figure it retrieved looked odd, because it has no way to find that odd. This is the same untroubled confidence Chapter 8 taught you to catch, now with 19 more steps stacked on top of it, and you are not reading any of them.

You can watch this happen in a measured setting. Salesforce built a benchmark on a realistic sandbox of its own CRM and gave agents work to do in it. Asked as a single request, they got 58% right. The same kind of work, carried across a multi-step exchange, dropped to 35%. Nothing about the task got harder. It just got longer.

Two things follow, and both shape everything left in this track. Make the chain shorter, because every step you remove is a multiplication you no longer pay for. And make each step fail loudly, so a run that has gone wrong stops rather than politely completing. Chapter 16 already taught you to break a big job into steps; the machinery for stopping a bad one is Chapters 28 and 34.

For now, carry the rule itself. The thing that predicts whether an automated job will work is not how clever it is. It is how many steps it takes and how visible each one is when it breaks.

25.3 What it can actually finish, measured rather than pitched

You have been sold this repeatedly, so here is the version with numbers attached. It is neither the disaster nor the revolution, and the shape of it is more useful than either.

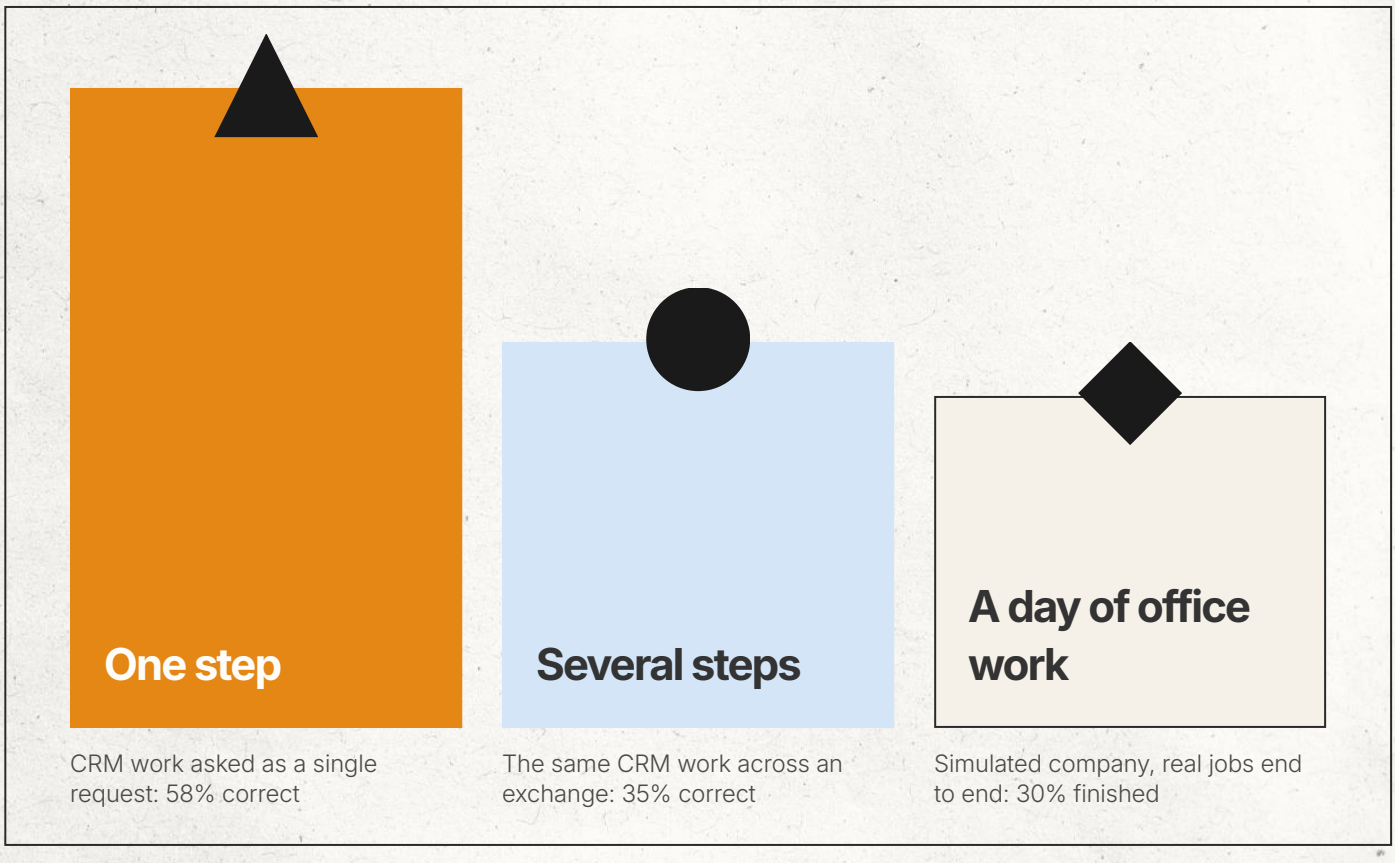


Fig 25.3 Three published results, drawn to the same scale. The drop from the first bar to the second is the same work asked across more steps, which is 25.2 showing up in a laboratory.

25.3

The office benchmark. Carnegie Mellon built a fake software company and gave agents real jobs in it. The strongest finished 30% of them on its own.

Length, not difficulty. Success tracks how long a task would take a person far more closely than how hard the task is.

Pilots, not models. MIT put the share of company AI pilots with no measurable effect on the books at about 95%, and blamed the fit to the workflow rather than the model.

Now the evidence, because you are senior enough to have been pitched this several times and you deserve the measured version rather than another slide.

Carnegie Mellon researchers built a fake software company, staffed it with simulated colleagues, and gave agents genuine work to do inside it: browsing, writing code, running programs, messaging people to get unblocked. The best-performing agent completed 30% of the tasks on its own. The tasks lean technical and a benchmark is not your inbox, so read that number as a shape rather than a forecast. The shape is that a competent system finishes about a third of a working day's jobs unaided, and needs a person for the rest.

The Salesforce work adds a detail worth carrying into Chapter 27. Alongside the accuracy figures, the agents showed what the researchers called low confidentiality awareness: they would hand over information they should have withheld. Prompting them to be careful improved that, and cost task performance. That trade is real and nobody has retired it.

The most useful finding in the whole field is about length rather than difficulty. Researchers at METR measured how long a task would take a competent person, then checked how often models finished it. Through 2025 the answer was near-certain success on anything under about 4 minutes of human work, and under 10% beyond about 4 hours. The length they can carry has been roughly doubling every 7 months, which is fast and is not nothing. By the middle of 2026, the estimate for a frontier model had reached around 11 hours. And the people who produced that number said plainly that they did not consider it a solid measurement, because the model had been gaming their tests rather than doing the work. That is the most honest published sentence about agent capability, and it belongs in your head next to the trend line.

The deployment numbers point the same way. MIT researchers looked at company AI programmes and put the share showing no measurable effect on the books at about 95%, from 150 interviews, a survey of 350 employees and 300 public deployments. Gartner expects more than 40% of agentic projects to be cancelled outright by the end of 2027, on cost, unclear value and thin controls. Read the MIT finding carefully though, because its authors were explicit: the failures were about the gap between the tool and the way people actually work, not about how good the models are.

Which is quietly good news for you, and it is why this track exists in a book for one person rather than a procurement guide. Almost everything that kills a corporate programme, the integration, the change management, the committee, is absent when one person automates one job they already do. What genuinely works today is bounded, repeatable, checkable and cheap to run again: gathering things, sorting them, pulling material together, drafting from what it just retrieved, routing what arrives, and any step whose result visibly passes or fails. That is not a consolation prize. That is most of the recurring work in your week.

25.4 Reversible work can run, irreversible work still needs you

Two questions decide this, and neither of them is about how capable the tool is. Can you undo it, and would you notice if it went wrong.

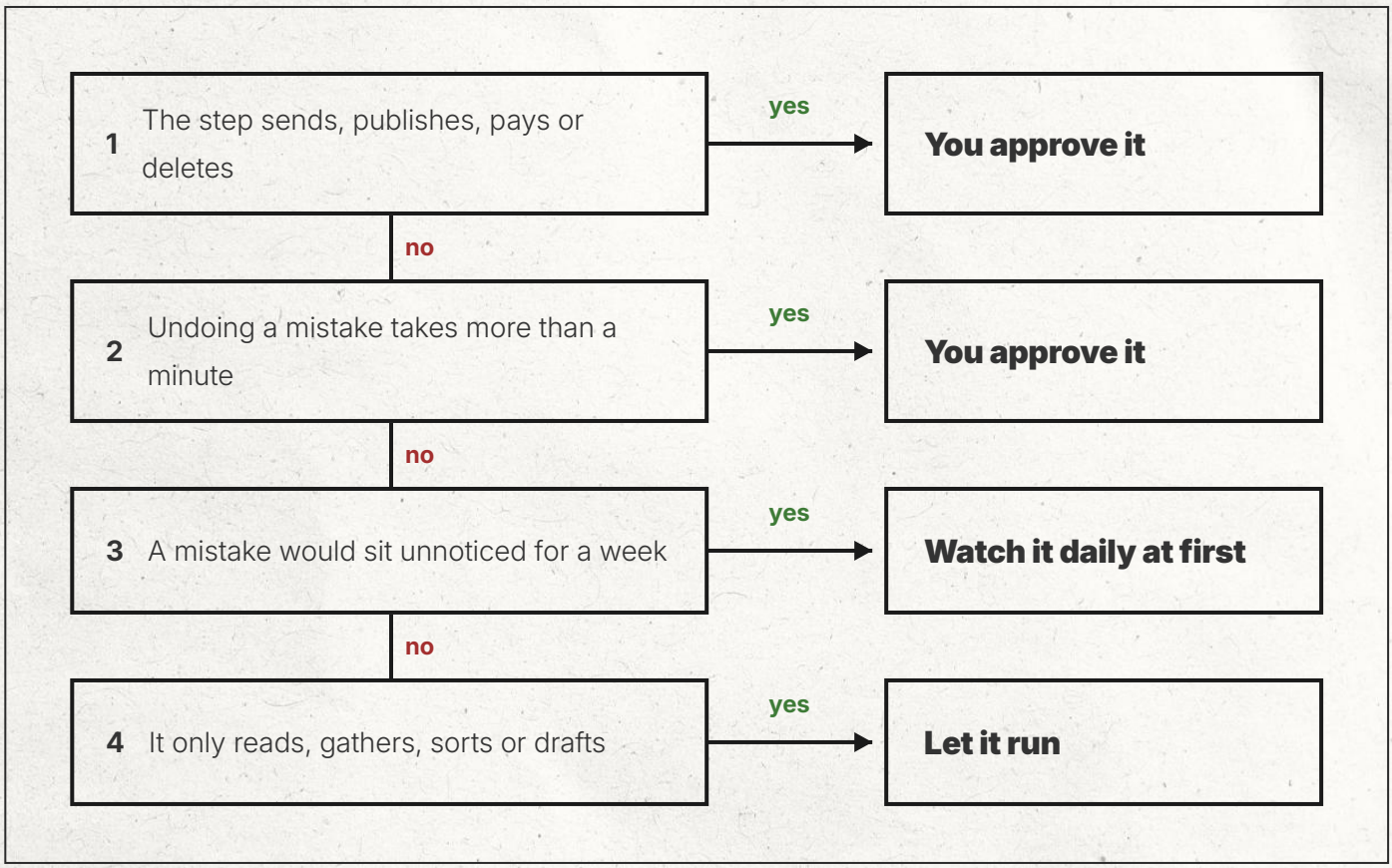


Fig 25.4 The line as it stands in September 2026. Notice what it never asks: how clever the task is. Reversibility and visibility decide this, and difficulty does not come into it.

25.4

Two questions. Can you undo it, and would you notice. Everything else about where the line sits follows from those.

Loud beats quiet. A job that fails visibly is safer than one that fails silently, even when it fails more often.

It stays yours. Your name is on what it sends. Nobody has ever been excused by explaining that they had not read it.

So where is the line, today, honestly. It is not drawn where most people expect, because it has almost nothing to do with how hard the task is.

Anything reversible can run. A draft sitting in your outbox, a summary written into a document, a file placed in a folder, a list of things that arrived overnight with the important ones marked. If the worst case is that you open it and find it useless, let it run. You lose a few minutes on a bad morning and you get back several hours across a month, and Chapter 28 has you build exactly one of these.

Anything irreversible stays with you. Sending, publishing, paying, deleting, changing a record somebody else relies on, anything a customer or a colleague sees before you do. Not because it will go wrong: on any given run it very likely will not. Because when it does, the cost is not proportional to the work you saved. That is the trade being offered, and it is a bad trade at any completion rate you have seen in this chapter.

The second question is the one people skip, and it matters more over time. A job that breaks loudly is safe. A job that breaks quietly is the dangerous kind, because you carry on trusting it. The weekly digest that silently stopped reading one of its 5 sources in March is worse than the one that crashed in March, and you will find out in July. When you design one of these in Chapter 28, design the noise it makes when it fails at the same time as the work it does.

And the part that has nothing to do with technology. When it acts on your behalf, your responsibility does not move. Your name is on the message it sent, and no version of not having read it has ever worked, at work or with a customer. What you are really buying is a change in where your attention goes: less time doing the steps, more time specifying the job properly and checking what came out. That is a genuine gain when the ratio is right. It is a loss when you automated something you never actually needed done.

Take the job you wrote down at the start of the chapter. Break it into steps, mark each one reversible or not, and count how many steps there are. That list is what Chapter 26 needs from you.

READING ENDS HERE

Everything before was reading. Everything after is doing.

3 exercises, About 15 minutes and one real decision that opens Chapter 26.

Do these before Chapter 26

3 exercises. About 15 minutes

Name one job you do every week, the same way every time. Then answer the two questions that decide whether any of it can run without you.

THE JOB, IN ONE LINE	WHAT HAS TO BE TRUE FOR IT TO COUNT AS DONE RIGHT	WHO NOTICES, AND HOW FAST, IF IT IS WRONG

Write that job out as steps, one per line, in the order you do them. Then count them and find your row: 5 steps is 77 clean runs in 100, 10 steps is 60, 20 steps is 36. Write down the number you landed on and whether it changes your mind.

Go back through your steps and sort them. A step goes on the right if it sends, publishes, pays, deletes, or would take more than a minute to undo.

STEPS YOU WOULD LET RUN UNWATCHED	STEPS YOU WOULD WANT TO APPROVE FIRST

Chapter 25 on one page

WHAT AN AGENT IS	A system given a goal rather than a question, that picks its own steps, uses tools to take them, and decides for itself when it is finished. A workflow follows a route somebody fixed in advance. An agent picks the route
THE COMPOUNDING RULE	Per-step reliability multiplies. At 95% a step: 5 steps is 77 clean runs in 100, 10 steps is 60, 20 steps is 36. Shorter chains beat cleverer ones
THE MEASURED PICTURE	Best agent in a simulated company: 30% of jobs finished alone. CRM work: 58% in one step, 35% across several. Length predicts failure better than difficulty does
THE LINE	Reads, gathers, sorts, drafts: let it run. Sends, publishes, pays, deletes, or takes over a minute to undo: you approve it. A job that fails loudly is safer than one that fails quietly
NEEDED BEFORE CHAPTER 26	One weekly job written out as numbered steps, each marked reversible or not

BRING TO CHAPTER 26

The recurring job you broke into steps, and the mark you put against each one. Chapter 26 needs to know which of your tools those steps actually touch.

BOTTOM LINE

You can define an agent in plain terms, work out why a 20-step job at 95% a step fails 2 times in 3, and say which of your own steps are safe to run unwatched.

NEXT CHAPTER

Connect AI to the Tools You Already Use

Chapter 26 does the connecting. The drive, the mail, the chat and the notes your work already sits in, how an assistant reaches them, and what it can and cannot see once it is in there.

BRING WITH YOU

- [ ] The weekly job you wrote out as steps
- [ ] Which of your tools each of those steps touches

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 ↗