

COURSE

Copilot, end to end

An eight-module course from your first prompt to scheduled work and an agent you can share. Read a little, try it in your approved environment, then check the result.

8 modules · about 4 hours 20 minutes

An organisation-approved Microsoft 365 Copilot account, current apps and permission to use the practice material. Features depend on your licence, tenant settings and rollout. Use safe copies; ask your administrator if a capability is unavailable.

This reading edition follows the reviewed website content. Examples and exercises remain material for you to adapt to your approved environment.

Contents

01 Getting started

02 01 · What you have been given

03 02 · How it works, and how it fails

04 03 · Prompting that holds up

05 04 · Copilot in the apps you use

06 05 · Working with your own material

07 06 · Researcher and Analyst

08 07 · Prompts that run without you

09 08 · Designing an agent

10 After the course

Getting started

Copilot, end to end

A self-paced course that starts at your first prompt and finishes with scheduled work and an agent you built yourself.

Read a bit, then go and do it

Nothing here needs a trainer in the room. Keep Copilot open in a second window and run each example as you reach it. The exercises are the course; the slides are the setup.

Pace

Eight modules. About four hours twenty minutes if you do every exercise. Stopping between modules costs you nothing.

You Will Need

A licensed Copilot seat, a signed-in Microsoft 365 account, and a real piece of your own work to practise on.

Scope

The exercises focus on Microsoft 365 Copilot. Confirm entitlement and policy before scheduling, sharing agents or enabling additional services. Opening a particular builder alone does not determine billing.

Checkpoints

Decide on an answer before opening the reveal below each question.

Eight modules, first prompt to first agent

Module	Topic	Time
01	What you have been given	20 min
02	How it works, and how it fails	25 min
03	Prompting that holds up	40 min
04	Copilot in the apps you use	45 min
05	Working with your own material	35 min
06	Researcher and Analyst	20 min
07	Prompts that run without you	35 min
08	Designing an agent	40 min

Four things to check first

None of this needs a support ticket. If any check fails, that is worth sorting before module 1 rather than halfway through module 4.

Licence

Sign in to Microsoft 365 Copilot with your work account. Ask your administrator to confirm the assigned licence and enabled features; the visible interface alone is not proof of entitlement.

Applications

Office applications on current builds, signed in with the work account rather than a personal one.

A Real Task

Pick something you actually have to do this week. Practising on invented work teaches you very little.

A Place To Keep Prompts

One note, one file, anywhere you will find it again. You will be saving prompts from module 3 onwards.

01 • What you have been given

The licence turned on more than a chat box. Worth knowing what, before you decide it is not for you.

By the end of this module

Four outcomes. Each is something you can do, not something you can recite.

1. Name the places Copilot appears and say what each one is actually for.
2. Explain the difference between the free chat and the licensed seat you now hold.
3. Choose the right surface for a task instead of using whichever one you saw first.
4. Say plainly what your seat covers and where the paid extras begin.

Six places it now lives

Same assistant, different amounts of context. The surface you choose changes what it can see.

1. The Copilot app

Widest view of your work

2. Inside Word, Excel and PowerPoint

Acts on the open file

3. Outlook and Teams

Mail, meetings, threads

4. Pages and Notebooks

Work you keep and reuse

5. Researcher and Analyst

Slow, thorough, cited

6. Agents you build

A fixed job, done the same way

The day the licence switched on

You Now Have

Grounding in your own mail, files, meetings and chats, inside your tenant.

Copilot embedded in the Office applications rather than only in a browser tab.

Prompts that run on a schedule without you being there.

The ability to build and share an agent that other people can use.

You Already Had

A general chat assistant with web access and no sight of your work.

Perfectly good drafting help, provided you pasted everything in by hand.

No memory of your organisation, your projects or last Tuesday's meeting.

No way to hand a colleague the setup that made your answers good.

Pick the surface, then write the prompt

WHEN YOU WANT TO	GO TO	BECAUSE
Pull threads together across mail and meetings	Copilot app	It reaches widest across your work content
Change the document open in front of you	In-app Copilot	It edits in place rather than describing an edit
Catch up on a meeting or a long thread	Teams or Outlook	The transcript and the thread are already there
Produce something cited and structured	Researcher	It works slowly and shows its sources
Repeat the same job every week	A schedule or an agent	Modules 7 and 8 build both

If the answer needs something the surface cannot see, no amount of rewording will fix it. Change surface, or paste the material in.

Checkpoint

You ask the Copilot app about a spreadsheet on your desktop. It says it cannot find it. Why?

- A. The file is too large for it to process.
- B. It is not stored anywhere the assistant can reach.
- C. Spreadsheets need Analyst, not Copilot.
- D. Your licence does not cover Excel files.

Answer and explanation

B. Grounding reaches your work content in the tenant, not your local disk. Save it to OneDrive or SharePoint, or attach it to the prompt.

02 · How it works, and how it fails

Enough mechanics to make good decisions. Not a word more than that.

By the end of this module

The point is calibration: knowing when to relax and when to check.

1. Describe in one sentence what happens between your question and its answer.
2. Say what it can see, what it cannot, and why that is a permissions question.
3. Recognise the four ways it goes wrong before they reach anyone else.
4. Set your own checking effort against the cost of being wrong.

The loop, four steps

Every useful exchange has this shape, whichever surface you started from.

Step 01

You state the job

Task, audience, format. Written the way you would brief a capable new starter.

Step 02

It gathers context

Your prompt, plus whatever it can retrieve that you already have access to.

Step 03

It drafts a likely answer

Fluent by construction. Equally confident whether or not it is correct.

Step 04

You check and sign

Your name goes on the output. This step is the one that is not optional.

What it can see

Grounding is not a search of everything your organisation holds. It runs as you, with your permissions, and it never widens them. If you could not open a document this morning, neither can Copilot on your behalf.

In Reach

Your mail, calendar, Teams messages, and files in OneDrive or SharePoint that you already have rights to open.

In Reach If You Point At It

A specific file, meeting or thread you reference directly. Naming the source is the fastest fix for a vague answer.

Out Of Reach

Local disk, personal accounts, systems with no connector, and anything your permissions do not already cover.

Worth Remembering

Broad permissions become visible fast. A folder shared with everyone was always readable; now it is also findable.

Four ways it goes wrong

What You Will See

A confident answer built on a detail that does not exist anywhere.

A summary that misses the decision, because the decision was never written down.

Numbers that look plausible and do not reconcile with the source.

Guidance that was true a year ago and has quietly been superseded.

What To Do About It

Check any specific claim you would not have known yourself.

Say what you expected to find. Absence is often the real finding.

Open the source and total it yourself before it leaves your hands.

Ask where it got that, then read the thing it points at.

4

small jobs a week is a realistic first target

Teams that stick with it do not hand over their job. They pick a handful of repeated, low-risk chores and give those away, then add more once the checking habit is automatic.

Meeting notes. First drafts of routine replies. Reformatting a list somebody sent badly. Summarising a thread before you join the call.

Module 07 Turns Yours Into Schedules

Checkpoint

Your meeting summary is missing the decision everyone remembers making. Why?

- A. The model needs retraining on your organisation.
- B. The decision was spoken but never captured in anything it can read.
- C. Summaries are always shorter than their source.
- D. It can read calendars but not transcripts.

Answer and explanation

B. It summarises what exists. If the decision only lives in people's heads, no prompt will retrieve it, and the gap is the useful finding.

03 · Prompting that holds up

Not tricks. A writing skill, and a short one.

By the end of this module

You will leave with a prompt you have written and saved, not a technique you have admired.

1. Rewrite a vague request into one with role, context, task, format and an escape hatch.
2. Supply real material rather than describing it and hoping.
3. Improve an answer by continuing the conversation instead of starting again.
4. Keep the prompts that worked so you stop reinventing them.

Anatomy of a prompt that works

You are helping me reply to a supplier who has missed a delivery date twice.
Here is the contract clause and the last three emails in the thread. [paste]
Draft a reply: firm, no legal threat, ask for a committed date.
Under 150 words, plain English, no bullet points.
If anything is missing, ask me before you draft.

Role

Who it is helping, and with what

Context

The raw material, pasted in full

Task And Tone

One job, with the edges named

Format

Length and shape of the output

Escape Hatch

Permission to ask instead of guess

The same request, twice

Attempt One

Write an email chasing an invoice.

Returns a template full of square brackets, a subject line nobody would send, and a paragraph about valuing the partnership.

Attempt Two

Invoice 4471, £2,300, due 14 August. Two polite chasers already sent, both ignored. Long-standing client and we want the work to continue. Draft the third email: under 120 words, firm, no legal language, ask for a payment date.

Returns something you edit for thirty seconds and send.

Two habits, side by side

Do This

Paste the real material. A thin prompt earns a generic answer.

Ask for one thing, then refine it in the same conversation.

Say what good looks like: length, tone, who reads it.

Read every line before it leaves your outbox.

Not This

One-line requests, then blaming the tool for guessing.

Five questions in one message, so the answer covers none of them.

Accepting the first draft because it reads confidently.

Shipping facts, figures or law without opening the source.

Steer it, do not restart it

A first answer is a draft to argue with. Most people delete the conversation and retype the whole thing, which throws away all the context they just supplied.

Narrow

"Too long. Half the length, and drop the opening paragraph entirely."

Redirect

"That is aimed at the finance team. Rewrite it for the site engineers."

Challenge

"Which part of that came from the attached report and which did not?"

Lock It In

"Keep that structure. Now do the same for the other two sites."

Rewrite your worst request

1. Find a one-line prompt you typed recently that gave you something useless.
2. Rewrite it with all five parts. Paste the real context this time.
3. Run both versions side by side and keep the better output open.
4. Save the winning prompt into your prompt file. Module 7 will use it.

Time

15 min

Working

On your own. If a colleague is doing the course too, compare afterwards.

Deliverable

One prompt, saved somewhere you will find it again.

Checkpoint

Which single addition usually improves a weak prompt the most?

- A. A more polite tone.
- B. The actual source material, pasted in.
- C. Asking it to think step by step.
- D. Repeating the request twice.

Answer and explanation

B. Context beats phrasing almost every time. Politeness costs nothing and changes nothing; pasted material changes the answer.

04 · Copilot in the apps you use

In the apps you already use

The embedded version is narrower than the app, and much faster for the job in front of you.

By the end of this module

One useful move in each application, which is plenty to start with.

1. Clear an inbox backlog without reading every message twice.
2. Catch up on a meeting you missed and leave with actions, not a transcript.
3. Draft and restructure a document from material you already hold.
4. Get an answer out of a spreadsheet and check the working.

Mail, in a fraction of the time

The gain here is triage rather than writing. Summarising a long thread before you reply is worth more than having a draft written for you.

Summarise

Open a long thread and ask for the decisions and open questions only. Skip the pleasantries.

Draft In Context

Reply with a rough instruction rather than a finished sentence: "decline, offer next Thursday, stay warm".

Find

"What did we agree with the fire alarm contractor about access?" beats scrolling for ten minutes.

Watch For

It drafts in your voice badly at first. Correct it once in the thread and the next one lands closer.

Meetings you did not attend

For a meeting you missed, use an available transcript and check the resulting decisions and actions against it.

Catch Up

Ask what was decided, what is unresolved, and what anyone committed to. Three separate questions beat one.

Your Bit Only

"What was said about the north sites, and by whom?" Filtering by topic is faster than reading the lot.

During The Call

Ask what you missed while you were talking, without stopping the meeting to ask a person.

Requirement

Post-meeting Copilot needs a retained transcript for the spoken discussion. During-meeting use can work without recording or a saved transcript, depending on meeting settings and organisational policy.

Drafting and restructuring

Copilot in Word is at its best when you give it material rather than a topic. Pointing it at three files you already wrote produces something worth editing; asking for a document about a subject produces filler.

Draft From Sources

Reference the files you want it to use, then say what the document is for and who reads it.

Restructure

"Turn this into a one-page brief for a non-technical reader, keep the figures".

Tighten

Select a paragraph and ask for half the length. Repeat until it stops improving.

Do Not

Ask it to write policy or contract wording from nothing. It will produce something confident and unusable.

Questions, and how it answered them

Ask in plain English against a table with real headers. The important habit here is checking the working rather than the answer, because a wrong formula applied consistently looks entirely convincing.

Ask

"Which sites went over budget in Q2, and by how much?" against a properly headed table.

Build

Describe the formula you want in words and let it write the syntax, then test it on a row you know.

Agent Mode

Give it a multi-step job across the sheet. Read what it changed before you save over anything.

Check

Total one column by hand. If that reconciles, trust the rest a little further.

Slides from something you already wrote

Generating a deck from a prompt gives you filler. Generating one from a document you have already written and edited gives you a starting point, which is a different thing entirely.

From A File

Point it at the brief or report and ask for a deck with a stated audience and slide count.

Per Slide

Ask for speaker notes on a slide you wrote yourself. Often better than the slide generation.

Rework

"This slide has four ideas on it. Split it into two and cut the third point."

Expect

Layout will need your hands. Treat the output as content placed roughly, not as a finished deck.

One move in each application

1. Summarise your longest live email thread and note what it missed.
2. Open a recorded meeting from the past fortnight and ask what was decided.
3. Take a document you wrote and ask for a one-page version for a different reader.
4. Ask one real question of a spreadsheet, then check the answer by hand.

Time

25 min

Working

Use live work, not samples. The point is to see where it helps you specifically.

Deliverable

A note of which of the four you would keep doing.

05 · Working with your own material

Files, pages and notebooks. Where answers stop being disposable.

By the end of this module

The shift here is from asking questions to building something you return to.

1. Point a prompt at specific files instead of hoping it finds them.
2. Turn a good answer into a page you and other people can edit.
3. Group the material for a piece of work so you stop re-attaching it.
4. Find a document by describing it rather than remembering where it went.

Point at the source

Referencing a file directly is the single most reliable way to improve an answer. It removes the retrieval guesswork and leaves the assistant doing the part it is good at.

Name It

Reference the specific document, meeting or thread in the prompt rather than describing it loosely.

Several At Once

Two or three related files works well. A dozen produces an answer that averages everything and says nothing.

Ask What It Used

"Which of those did that figure come from?" If it cannot say, treat the figure as unverified.

Permissions Still Apply

Referencing a file you cannot open does not open it. You get nothing, not an error worth arguing with.

Answers that survive the conversation

A chat answer disappears into history. A page is a working document that keeps the answer, takes your edits, and can be shared with people who were not in your conversation.

Promote

Move a response that is worth keeping onto a page instead of copying it into an email.

Keep Working

Add to it with further prompts, edit it by hand, and both changes stay put.

Share

Colleagues can edit it directly, which is the difference between a page and a screenshot.

Good For

A brief taking shape, a set of notes being assembled, anything with more than one contributor.

One place per piece of work

A notebook holds the files, links and notes for a single piece of work, so every prompt inside it is already grounded in the right material. It removes the re-attaching that quietly eats the time you saved.

Assemble Once

Add the documents, the notes and the links that define the project, then stop thinking about them.

Ask Inside

Prompts run against that collection, so answers stay on topic without you restating scope.

Best Fit

Anything that runs for weeks: a tender, an audit, a migration, a recurring board paper.

Poor Fit

A one-off question. The setup costs more than the answer is worth.

Describing beats remembering

Ordinary search needs the words that are in the file. Copilot Search works from a description of what the thing was, which is usually all anyone can actually recall.

Describe It

"The capacity paper with the traffic-light table that went to the board in spring".

Search By Person

"What has Priya shared with me about the access control work?"

Then Act On It

Once it surfaces, keep going in the same conversation instead of opening it and starting over.

Limit

It still only searches what you can already open. It is a better index, not a wider one.

Set up one notebook

1. Choose a piece of work that will still be live in a month.
2. Add the three or four documents that genuinely define it. Resist adding everything.
3. Ask two real questions inside it and note whether the answers improved.
4. Promote the better answer onto a page and share it with one colleague.

Time

20 min

Working

Use a real project. A test notebook teaches you nothing about whether this fits your work.

Deliverable

One notebook and one shared page.

06 • Researcher and Analyst

Slower, deeper, cited. Use them when the answer has to survive scrutiny.

By the end of this module

Mostly this is a judgement skill: knowing when the slow tool is the right one.

1. Tell the difference between a chat question and a research question.
2. Brief Researcher properly, which takes longer than briefing chat.
3. Use Analyst on real data and check the code it ran.
4. Spend a limited monthly allowance on the questions that deserve it.

A report, not a reply

Researcher works across the web and your own content in multiple steps, then returns a structured report with its sources named. It takes minutes rather than seconds, and the brief you give it matters more than usual.

Use It For

A market or supplier comparison, a policy landscape, anything you would otherwise spend a morning on.

Brief It Fully

State the decision the report has to support, the scope, and what a useful answer would contain.

Read The Sources

It names them precisely so you can check. Naming a source is not the same as the source being good.

Not For

Quick factual questions. Chat answers those in a second and costs you nothing from the allowance.

Data, with the working shown

Analyst takes files, works through them iteratively and writes Python to do the arithmetic. You can watch the code it runs, which makes it the rare case where checking is genuinely straightforward.

Use It For

Several messy files that need consolidating before any question can sensibly be asked.

Ask Plainly

"Which sites are trending worse this year, and is the change outside normal variation?"

Read The Code

It shows what it ran. Skim it for the assumption you would have questioned in a colleague.

Still Check

Confirm the conclusions against the original data. Correct arithmetic on a misread column is still wrong.

Which one, and what it costs you

THE QUESTION	WHERE IT GOES	WHY
What did we agree last Tuesday?	Chat	Seconds, and no allowance spent
Summarise this thread for my manager	Chat	One source, one pass, nothing to investigate
How are comparable operators handling this?	Researcher	Multi-step, needs sources you can defend
Consolidate these six exports and find the trend	Analyst	Real computation, and it shows the code
Both agents, combined per month	25 queries	Shared allowance included in the licence

Twenty-five a month is generous for real research and trivial to waste on questions chat would have answered instantly.

07 • Prompts that run without you

The first genuinely different thing the licence buys. Also the easiest to get right.

By the end of this module

You will have at least one schedule running before you reach module 8.

1. Schedule a prompt you have already tested, from the surfaces that support it.
2. Write for unattended running, where nobody is there to answer a question.
3. Work inside the real limits on count and on what a schedule can reach.
4. Retire a schedule you stop reading, rather than letting it pile up.

How scheduling works

Run a prompt once in Copilot chat, hover over it in the history and choose to schedule it. Set the day, the time and how you want telling. The result lands in your Copilot history, and optionally as a Teams or email notification.

Where

The Copilot app, Copilot in Teams, or Outlook on the web and desktop. Not from inside Word or Excel.

Requirement

Scheduled prompts require an eligible Microsoft 365 Copilot licence and administrator configuration.

How Many

Ten scheduled prompts per person. That is a healthy ceiling rather than a frustrating one.

What It Reaches

Test the exact sources in a scheduled run. Do not assume that access in an interactive chat guarantees access during an unattended run; check citations and missing information before relying on it.

Writing for an empty room

Works Unattended

Relative time windows: last 24 hours, since Monday, this week.

Output that stands on its own without a follow-up question.

One clear job, with the format stated so every run looks the same.

A prompt you have already run manually and been happy with.

Fails Quietly

"Today's meeting" or "the email I sent you", which need context you will not be there to give.

Anything that ends by offering to go deeper if you would like.

Four requests bundled together, so each run answers a different one.

Something you wrote optimistically and never tested once.

A Monday morning digest

Review my calendar for the coming week and my email from the last 7 days.

List meetings where I am the organiser and no agenda has been circulated.

List commitments I made in email in the last 7 days with no follow-up since.

Flag anything with a deadline inside the next 10 working days.

Plain list under each heading. No preamble, no closing offer of further help.

Window

Relative, so it works every week

Job One

Meetings needing something from me

Job Two

Promises I have not kept yet

Job Three

What is about to become urgent

Format

Same shape every Monday, no chat

Six worth having

Start with two. Add more once you find you are actually reading them.

1. Monday planning digest

Weekly, before you start

2. Unanswered mail sweep

Daily, late afternoon

3. Commitments I made

Weekly, Friday

4. Meetings with no agenda

Weekly, Thursday

5. Team catch-up before the one to one

Weekly, an hour before

6. Month end approaching

Monthly, five days out

Put one schedule live

1. Take a prompt from your file, or adapt the digest on the previous slide.

2. Run it manually once and fix whatever comes back wrong.

3. Rewrite the time references as relative windows, then schedule it.

4. Diary a reminder for two weeks' time to keep it or delete it.

Time

20 min

Working

Do this now rather than later. An unscheduled good intention is worth nothing.

Deliverable

One live schedule and a note of when you will review it.

Checkpoint

Your scheduled prompt returns nothing about the files it was meant to check. What should you do?

- A. Assume the files contain nothing relevant.
- B. Check supported sources, permissions and citations in the scheduled run.
- C. Make the prompt longer without testing it.
- D. Send the empty result as a verified report.

Answer and explanation

B. An empty result is not evidence that nothing changed. Verify the scheduled capability and its sources, then use a supported approach and test it again.

08 · Designing an agent

A named assistant with a fixed job, built without code and shared with your team.

By the end of this module

You will build one, share it, and know exactly where the free part ends.

1. Say when a job deserves an agent rather than a saved prompt.
2. Write instructions that hold up when somebody else uses them.
3. Attach the right knowledge without over-feeding it.
4. Recognise the point where agent work stops being covered by your seat.

Agent, or just a saved prompt?

An agent is a named Copilot with a fixed role, a set of instructions and optional knowledge sources, available to anyone you share it with. A saved prompt is a piece of text you paste. The difference is who else can use it well.

Saved Prompt

One person, one job, no setup. Perfect until three colleagues start asking you to send it to them.

Agent

The same job done the same way by everyone, with the knowledge attached rather than remembered.

Build It When

The job repeats, the quality depends on context most people will not supply, and more than one person needs it.

Skip It When

You are the only user and the prompt already works. An agent adds maintenance you do not need.

Instructions that hold up

You help our team review supplier responses against our standard requirements.
Always work from the attached requirements document, not from general knowledge.
For each response: state where it meets, partially meets or misses each requirement.
Quote the exact wording you relied on. Never paraphrase a commitment.
If a requirement is not addressed at all, say so plainly rather than inferring intent.

Role

One job, stated in a sentence

Source Of Truth

Where it must ground its answers

Output Shape

Consistent across every user

Evidence Rule

Quote, so claims can be checked

Honesty Rule

Say nothing rather than invent

An agent over a document library

A SharePoint agent answers questions about selected content. Review the library's permissions first; a working agent does not prove the underlying access is appropriate.

Good Candidates

A policy library, a project site, a technical standards area, a set of contracts your team keeps re-reading.

Permissions

It inherits the library's. Nobody sees anything through the agent that they could not open directly.

Keep It Narrow

Point it at the library that answers a question, not at the whole site because it was easier.

Maintain It

The agent is only as current as the library. Stale documents produce confidently stale answers.

Where your licence stops paying

WHAT YOU ARE DOING	IN THE SEAT	NOTES
Running an agent you built in Copilot chat	Yes	Grounded in your content, on your own seat
A SharePoint agent over a library you can open	Yes	Permissions inherited, nothing extra to buy
Researcher, Analyst, scheduled prompts	Yes	Within the licence, allowances aside
Copilot Studio agents	Depends on use	Eligible employee use under a licensed user identity can be included; other scenarios may consume credits.
Autonomous runs and additional services	Check first	Trigger, identity, channel and capabilities affect billing. Obtain administrator approval.

Check the current licensing guide with your administrator before expanding beyond these exercises. The product used to build an agent is not, by itself, the billing boundary.

Build and share one agent

1. Pick a job you do repeatedly where the quality depends on context others lack.
2. Write the instructions, including an evidence rule and an honesty rule.

3. Attach only the knowledge that job needs, then test it with three real questions.
4. Share it with one colleague and watch them use it without coaching them.

Time

25 min

Working

The watching step is the valuable one. What they type first tells you what the instructions missed.

Deliverable

One working agent, and a list of what you would change.

Checkpoint

A colleague without a Copilot licence wants to use your agent. What should you expect?

- A. It works identically for them at no cost.
- B. It is blocked outright until they are licensed.
- C. Their usage may be metered separately, which is an admin decision.
- D. The agent stops working for everyone.

Answer and explanation

C. Your own licensed use sits on your seat. Unlicensed access to agents grounded in tenant data is billed on consumption, and whether that is even enabled is for your administrators to decide.

After the course

The risk was never that it writes badly. It is that it writes confidently.

Every Output Still Needs A Named Human Owner

Your first thirty days

Adoption fails from doing everything at once far more often than from doing too little.

Week 01

Use it daily for four small jobs

Pick the four now. Do not wait for a better idea to arrive, because it will not.

Habit before ambition

Week 02

Keep every prompt that worked

One file. Paste in anything that produced something you actually sent.

Your library starts here

Week 03

Put two schedules live

Test each one manually first, then diary a fortnightly review to prune them.

Work that happens without you

Week 04

Build one agent and share it

Choose the job colleagues keep asking you to do, then watch one of them use it.

Where the team gain starts

What to take back to your desk

1. Context is the work

The quality of an answer is set by what you supply and what it is allowed to reach, not by clever phrasing.

2. Check in proportion

Cheap to be wrong, glance at it. Expensive to be wrong, open the source. The habit matters more than the rule.

3. Repetition is the win

Small jobs done every week beat one impressive demonstration nobody repeats.

4. You sign the output

Schedules and agents change who does the typing. They do not change whose name is on it.

Sources and further reading

Microsoft — Copilot in Teams meetings

<https://support.microsoft.com/en-US/teams/copilot/catch-up-on-meetings-with-microsoft-365-copilot-in-teams>

Microsoft — scheduled prompt requirements

<https://learn.microsoft.com/en-us/microsoft-365/copilot/scheduled-prompts>

Microsoft — scheduling and current limits

<https://support.microsoft.com/en-us/microsoft-365-copilot/schedule-your-most-used-copilot-prompts>

Microsoft — agent billing and included usage

<https://learn.microsoft.com/en-us/microsoft-copilot-studio/requirements-messages-management>

Microsoft — Researcher and Analyst allowance

<https://www.microsoft.com/en-us/microsoft-365/blog/2025/06/02/researcher-and-analyst-are-now-generally-available-in-microsoft-365-copilot/>