GO BEYOND SOLUTIONS
TRANSCEND GROWTH BOUNDARIES

PRACTICAL AI / FIELD GUIDE

Give AI a Safe Workspace on Your Computer

6 practical ways to put AI to work on your computer - plus one master setup prompt that walks you through the technical steps.

AI becomes far more useful when it can safely work with your files, folders, and apps. This guide shows what becomes possible, how to reduce risk, and how to set it up step by step.

AI can answer. AI can also act.

Most people first use AI as a conversation: ask a question, get an answer. That is only the beginning.

WITHOUT ACCESS

“How should I organize these photos?”

AI can explain a process, suggest folders, and give you instructions.

WITH THE RIGHT ACCESS

“Help me organize these photos.”

AI can inspect the actual files, find patterns or duplicates, propose a plan, and help carry it out.

WITHOUT ACCESS

“Why is my computer slow?”

AI can list likely causes and tell you what to check.

WITH THE RIGHT ACCESS

“Check what is making it slow.”

AI may be able to inspect the machine, gather evidence, and work through the problem with you.

> **“Giving AI hands” means giving it a controlled path to the place where the work actually lives.**

That path might be a file, a connected app, a browser, a desktop tool, a coding environment, or something else. You do not need to know which one before you start.

Don’t know where to start? Just ask AI to help you.

You do not need to know the right feature, tool, connector, or technical term. Start with what you want.

Try this:
“I’m new to this. Can you help me discover the friction in my day-to-day life that AI could help me reduce? Ask me a few questions about the things I have to manage my life daily across my digital life, apps, computer, etc? Then show me useful ways you could actually remove friction - not just give me information.”

Or start with a real annoyance:

- “My photos are a mess. How could you help me organize them?”
- “My inbox is overwhelming. What could you do if I connected my email?”
- “I keep doing this same task every week. Could you help do some of it?”
- “My computer is slow. What could you actually check if you had the right access?”

> **One of the best ways to learn AI is to ask AI how to help you use AI.**

The follow-up that matters

“What would you need access to in order to actually help me do this - and can you walk me through setting that up?”

Start here: Five steps are enough to get moving.

You do not need to understand AI technology. Use this sequence with the AI you already have.

1. **Describe what you want.** A problem, desired outcome, or recurring annoyance is enough.
2. **Ask how AI can help you.** Make it distinguish current-chat capability from what may be possible with another tool or integration to enable the help.
3. **If AI says it needs something, ask it to help you resolve it for AI.** Ask for specific, detailed steps to enable supported connectors, apps, browser tools, desktop access, coding agents, or MCP when relevant.
4. **Let the AI guide the setup.** Tell it you are a beginner and ask it to use current official instructions, one step at a time.
5. **Do one real task.** Start small, check the result, then decide whether you want to give it more responsibility.

Example use case

"I have about a terabyte of music on my Windows PC. The filenames, artist names, album info, genres, artwork, and metadata are inconsistent. There are probably some duplicates and other cleanup issues I am not even aware of. How could you connect to my PC and inspect my music library for issues? I am new to AI and will need you to walk me through each step in detail assuming I know nothing about music file management and using AI to do things for me. "

If this is all you need, you can stop here.

“I can’t do that” is sometimes a setup problem, not the final answer.

AI may answer based only on what this chat can access right now. That does not always mean the task is impossible.

Ask why.

“Why can’t you do that from this chat? What capability, connection, permission, or tool is missing? Is there a supported way to add it?”

Then ask how to make it possible.

“If there is a way to give you the access you need, explain the simplest current option in plain English and walk me through it step by step using official instructions.”

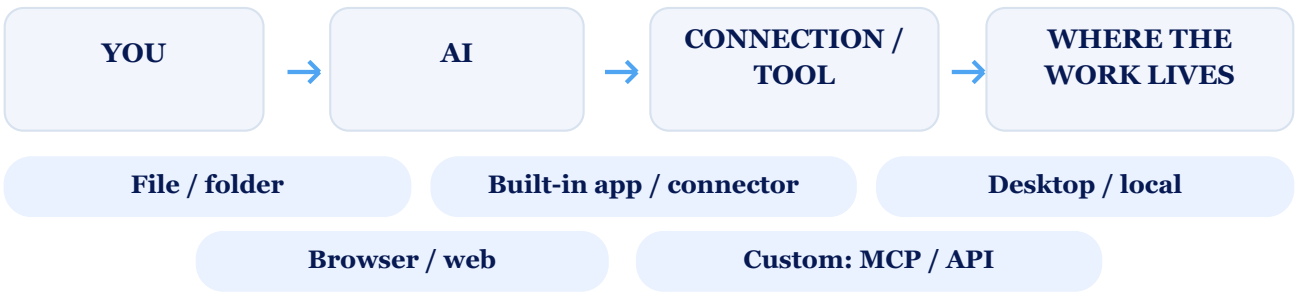
Example: organizing photos on your computer

> **You:** “Can you organize the photos on my computer?”
>
> **AI:** “I can’t access files on your computer from this chat.”
>
> **You:** “Okay. What would let you access them? Do you support a desktop tool, connector, local workspace, or another method? Walk me through the best current option.”

Sometimes the answer really is no. But sometimes the missing step is simply that the right tool exists and is not connected or active in the current conversation.

AI needs a path to where the work lives.

AI does not magically gain access because you asked. Something has to connect the AI to the file, computer, website, app, or service. Start with the simplest connection that already exists.



Three questions explain almost everything.

01

What can it see?

A file? One folder? Your email? A website? Local files or system information?

02

What can it do?

Read only? Draft? Edit? Send? Run a command? Change a setting?

03

Where is it running?

In a browser, desktop app, cloud workspace, connected service, or on your computer through a local tool?

You do not have to memorize product names. Understand what it can see, what it can do, and what connects it to the work.

Start with the easiest doorway.

01
File or folder access

Best for one file or folder: upload it, open it, or use a workspace the AI already reaches.

02
Plugins, apps, and connectors

Usually the easiest path into a service: choose the built-in integration, sign in, and approve access.

03
Browser / computer-use tools

Use when the work lives on the web: research, forms, websites, and web apps.

04
Desktop / remote-computer tools

For PC work, try the desktop AI app or local feature first. Add a local or remote bridge only if needed.

05
Coding / agent environments

For code, repositories, terminals, tests, and software-development workflows.

06
MCP and tool protocols

For custom tools when no simpler built-in connection fits.

> **Prefer the simplest built-in path that works. Add custom setup only when the easy doorway cannot do the job.**

Built-in apps and connectors are prebuilt doorways.

These are ready-made integrations. Usually you choose one, sign in, approve access, and the AI can use the capabilities that connection exposes.

Email

A built-in email connection may let AI search messages, summarize threads, draft replies, label items, or take other supported actions.

Calendar

A calendar connection may read events, identify conflicts, and create or update events when supported and authorized.

Cloud files

Drive, OneDrive, Dropbox, and similar integrations can expose files without you building a custom connection.

Business tools

Slack, GitHub, CRMs, databases, project tools, and many other systems may already have a ready-made integration.

Ask if it already exists.

Before custom setup, ask whether your AI already has a built-in app, plugin, connector, or supported integration for the service you want to use.

Ask the AI

“Is there already a built-in app, plugin, or connector for this? If so, what would it let you do, and can you walk me through connecting it?”

If the work is on your PC, start local.

For work on your PC, try the desktop AI app first. It may already offer local file, screen, or app access. What it can reach varies by product.

WEB / CLOUD SESSION

Runs somewhere else

Useful for websites and cloud apps. It does not automatically inherit the local files, installed software, or desktop state on your PC.

DESKTOP / LOCAL

Best for work on this PC

The desktop app may already support local files, screen context, apps, or other machine capabilities. If it cannot reach enough, add a local bridge only when needed.

> **Check what the desktop app already supports before installing another tool.**

A useful follow-up is:

"Can your desktop app handle this directly? If not, what is the simplest supported local connection that would let you?"

Remote Desktop Commander is one example of an additional local bridge when built-in desktop access is not enough. Other AI products may provide different local-access options.

Use the easiest doorway first.

Do not build a custom connection if a simpler supported one already exists.

A practical order

1. Built-in app or connector.
2. Desktop/local for PC; browser for web.
3. Custom tools only if needed.

Where the work lives tells you what to try:

- File / folder: upload.
- Services: built-in connector.
- PC: desktop app or local tool.
- Web: browser tool.
- Custom: coding agent, MCP, or API.

Use the doorway that matches where the work lives.

> **Ask: “What is the simplest supported way to connect to this? Prefer a built-in option before custom setup.”**

When built-in access is not enough.

CODING / AGENT ENVIRONMENT

Built to work with projects

Useful when the work is code, repositories, terminals, tests, or repeatable development tasks. Some environments run locally; others use managed cloud workspaces.

DESKTOP / REMOTE-COMPUTER AGENT

Built to reach a machine

Useful when AI needs a path to local files, commands, or installed software that the desktop app cannot reach on its own.

When you may need more.

- No built-in app, plugin, or connector exists.
- AI still cannot reach the local files or software you need.
- The service exposes what you need only through API, MCP, or another technical interface.
- You need repeatable access beyond a one-time upload.
- You are connecting several tools into a custom workflow.

Useful follow-up

“Is there a built-in option first? If not, what is the simplest next step - desktop/local agent, coding environment, MCP, API, or something else?”

Where AI runs changes what it can reach.

LOCAL

On or directly connected to your device

A desktop AI app or local agent may work with files, commands, installed software, screen context, or other machine-specific information - depending on product and permissions.

CLOUD

On remote infrastructure

Web chat and cloud workspaces are convenient for general AI work, websites, research, and managed tools, but they do not automatically inherit access to your PC.

CONNECTED SERVICE

Inside another app

Built-in connectors let AI reach email, cloud files, calendars, CRMs, databases, and other services after you authenticate and approve access.

HYBRID

A combination

Many workflows combine them: cloud AI, built-in connectors, desktop/local access, and browser tools only where useful.

> **A useful question for any workflow: “Where is the work, what can you already reach, and what is the simplest connection that closes the gap?”**

The best users keep asking questions.

Once AI can act, keep asking what would make the workflow easier, more capable, or more repeatable.

When it says no

"Why not? Is this a hard limitation, or is there another built-in or supported way to do it?"

When work is manual

"What built-in access, desktop capability, or tool would let you do more of this yourself?"

When you repeat yourself

"Could we turn this into a reusable workflow or automation once we know it works?"

When the tool is confusing

"Explain what this app, connector, agent, or integration is doing in plain English."

When something fails

"What actually failed: the AI, the permission, the connection, the environment, or the tool?"

When you want more capability

"What is the next useful thing you could do if I gave you the right access - and what is the easiest way to provide it?"

And keep one basic boundary.

Know what the AI can see, what it can change, and where it is working. Prefer the easiest supported doorway, verify it, and add complexity only when it earns its keep.

> **You do not need to learn the whole AI ecosystem before you benefit from it. Know what you want. Ask questions. Start with the easiest doorway. Make the AI help you understand the rest.**

GO BEYOND SOLUTIONS

TRANSCEND GROWTH BOUNDARIES

KEEP EXPLORING

The best next question is usually the one you already have.

Tell the AI what you want to accomplish. If it cannot do it yet, ask why, what is missing, and how to make it possible.

Use the web guide for the latest version as AI products, connectors, and tools change.

gobeyondsolutions.ai