AGENTIC SOLUTIONS | FREE RESOURCE

Creating a Company Knowledge Base for Your AI Agents

A practical guide for business owners: what to gather, how to organize it, and how to check that your agents can actually use it.



The software manual is only half the training

We recently spoke with a business owner who had trained an AI agent on her software vendor's manual and training materials. That is a sensible place to start. Vendor materials teach the screens, the menus, and the way the product is meant to work.

They cannot teach the other half: how your business uses that software. The manual explains where the job status field lives. It does not know that your office changes that status only after a certain document is in the file, or who is allowed to approve an exception. Those rules live in your procedures, your templates, and often in the heads of your best people.

A company knowledge base is where that second half gets written down. It is a small, current set of reference material that an employee or an AI agent can look things up in. It is not every file you have ever saved. And having the documents is only the start. An agent also needs real access to them, a way to show where an answer came from, and supervised practice before it touches live work.

The short version

Vendor materials teach the software.

Business materials teach how we use it here.

A folder existing is not the same as an agent being able to read it.

Documents alone do not make an agent competent. Testing and supervised practice do.

Seven steps to a useful knowledge base

1. Pick one workflow or one role

Do not start with the whole company. Choose one process that causes rework when someone new handles it, such as a customer follow up, a job handoff, or a weekly status check. A narrow start gives you something you can actually test.

2. Inventory what already exists

Gather the material that process truly depends on, and only the current versions:

- Official, current vendor manuals and help articles for the software involved
- Training videos you are allowed to use under the vendor's terms
- Your standard operating procedures for this process
- Approved policies and templates (customer wording, estimates, checklists)

- Handoff points: who passes what to whom, and when
- Permissions and exceptions: who can approve what, and what never happens without a person

3. Write down the unwritten process

Most businesses have rules nobody wrote down. Ask the person who does the work best, "What would a new hire get wrong here?" Capture the answers in a short company note, a page or two in plain language. This is often the most valuable document in the whole set.

4. Label everything and retire the old

Give each document an owner, a version, and a next review date. Move replaced or draft versions into a separate archive folder that the agent does not read. If two versions of the same procedure sit side by side, an agent will not know which one you meant, and neither will a new employee.

5. Connect access, read only first

Putting files in a folder does not give an agent access to them. Each agent product supports different connections: some read a shared drive, some take uploaded files, some use a search tool, and some support none of these. Check what your agent actually supports, use storage and permissions you have already approved, and start with read only access.

6. Test retrieval before you trust it

Quiz the agent the way you would quiz a new hire, and check its answers against the documents, not against your memory of them:

- Ask questions that require a source, and confirm the document and section it cites
- Ask something the documents do not cover. A good answer says it does not know.
- Ask about a case where two documents conflict, and see whether it flags the conflict
- Ask about an exception, and see whether it sends you to the right person

7. Run one supervised task, then maintain it

Give the agent one real task with no messages sent and no live changes made until you approve the result. Note every gap, then fix the document or the instructions. After that, keep it current: review dates on the calendar, one named person responsible, and a quick retest whenever a procedure or the software changes. Most agents do not learn from your corrections automatically. If a fix matters, put it in the document.

A worked example

The example below is hypothetical and for illustration only. The rules in it stand in for whatever your own business has decided.

Picture a property restoration business that wants an agent to prepare the handoff from the mitigation crew to the rebuild team. The owner picks that one handoff as the starting workflow.

- **Inventory:** the job management software's current help articles on job status and file notes, the company's handoff SOP, the rebuild scope template, and a list of who can approve schedule changes.
- **Company note:** the office manager explains what the manual does not cover. For example, "We do not mark a job ready for rebuild until the drying records are attached and the project manager has signed off," and "Any change to scope goes to the owner before the customer hears about it."
- **Labels:** each document gets an owner and a review date. An older handoff checklist moves to the archive.
- **Access:** the agent gets read only access to the handoff folder, nothing else, and no customer records.
- **Test:** "Can this job move to rebuild?" The owner checks that the agent cites the handoff SOP and the company note, asks for the missing sign off, and says it does not know when the documents are silent.
- **Supervised task:** the agent drafts a handoff summary for a sample job. A person reviews it. Nothing goes to the rebuild team or the customer until the owner approves.

Limits worth respecting

- **Keep client data out by default.** Customer and employee details belong in the systems already approved for them, with the right permissions.
- **Citations can be wrong.** An agent can point to a real document and still misread it. Check the section it names.
- **Your notes are not the final word.** If an old note disagrees with a regulation, a safety requirement, or a manufacturer's instruction, the official source wins and a person decides.
- **Retrieved text is material, not orders.** Documents can contain outdated or even planted instructions. An agent should treat them as information to check, never as commands that override your rules.
- **Reading is not permission.** What an agent may send, spend, or change is set by your rules and its access, not by what it has read.

A simple, well labeled folder may be all you need. Bigger tools help when the collection grows, but they do not fix stale or conflicting content.

Optional: how the lookup works

You can skip this section. When an agent answers from your documents, it usually does not "remember" them. The system searches your material for the passages that look relevant and places them in front of the model along with your question. That passage set is called context. The technique is often called retrieval augmented generation, or RAG. It reduces guessing, but it does not eliminate mistakes, and it only helps if the right, current material can be found.

This is different from retraining a model, which changes the model itself and is rarely what a small business needs. Some tools go further and keep a searchable memory for agents across many sources. GBrain, an open source project by Garry Tan, is one example. It is a separate retrieval layer that can import markdown notes, including notes kept in Obsidian. You do not need anything like it to begin.

Quick glossary

- **Archive:** everything you have kept, including old and replaced versions. Useful for history, not for answers.
- **Knowledge base:** the current, labeled, owned subset you want people and agents to rely on.
- **Obsidian vault:** a folder of plain text notes on your own computer, used by the Obsidian notes app. It is a way to store and organize notes, not an agent.
- **RAG retrieval:** searching your documents and giving the most relevant passages to the model as context when it answers.
- **Context vs. retraining:** context is what the model is shown for this task. Retraining changes the model itself. Adding documents to a knowledge base is context, not retraining.

Related Agentic Solutions resources

- Hiring Your First AI Agent: how to onboard an agent like a new hire, including reference material and quizzes.
- The Owner's OS: building software around the way your business actually runs.

Appendix: Starter checklist

Work through this for one workflow. Check each box before you move on.

- ☐ I chose one workflow or role to start with.
- ☐ I collected the current vendor manuals and help articles for the software involved.
- ☐ I confirmed any training videos are ones we are allowed to use.
- ☐ I gathered our SOPs, approved policies, and templates for this workflow.
- ☐ I listed the handoffs, permissions, and exceptions.
- ☐ I wrote a short company note covering what is not written down anywhere.
- ☐ Every document has an owner, a version, and a review date.
- ☐ Old and draft versions are moved to an archive the agent does not read.
- ☐ No client or employee data is in the folder unless it is approved storage with the right permissions.
- ☐ I checked which connections my agent actually supports and set read only access.
- ☐ I tested questions with source and section checks, a missing topic, and a conflict.
- ☐ I ran one supervised task with no sends or live changes without my approval.
- ☐ I fixed the gaps in the documents or the instructions.
- ☐ One person owns upkeep, review dates are on the calendar, and I retest after changes.

Sources

- GBrain and its company brain tutorial, Garry Tan, GitHub
- Create a vault, Obsidian Help
- Retrieval augmented generation (RAG), NIST Computer Security Resource Center Glossary
- AI Insights: RAG Systems, UK Government (GOV.UK)