



AI BASICS TRAINING

Chatbots vs. Agents

**What changes when AI stops just talking and starts doing,
and where that creates new risk.**

A 10-minute primer for teams already using ChatGPT, Gemini, and Claude

Same Model, Different Job

You already know the chatbot model: you ask, it replies, one turn at a time, entirely inside the chat window.

An agent runs on the same underlying LLM but wraps it in a system that can take multiple steps, use tools, and make decisions toward a goal, with much less turn-by-turn guidance from you.

01 Chat = Conversation.

02 Agent = Action.

A chatbot tells you how to do something. An agent can do it, then show you the result.

THE CORE DIFFERENCE

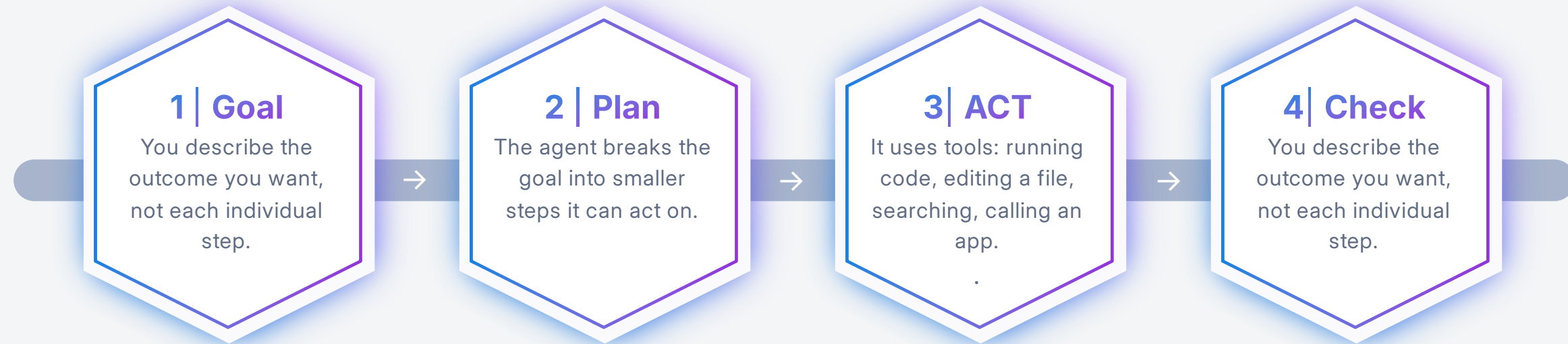
Chatbots vs. Agents, Side by Side

	Chatbot 	Agent 
What it does	Answers your questions	Completes multi-step tasks toward a goal
How it works	One reply per message you send	Plans, acts, checks results, repeats until done
What it can touch	Just the conversation	Files, apps, code, or the web, with permission
Your role	Ask, then evaluate each reply	Set the goal, review the outcome
Examples	ChatGPT, Gemini, Claude chat	Claude Code, Cowork, Codex, or a custom-built agent

UNDER THE HOOD

How an Agent Actually Works

Instead of one reply, an agent repeats a loop until the goal is met, or it needs your input.



Steps 2–4 repeat, often dozens of times, until the task is finished.

CHOOSING YOUR APPROACH

Two Paths to Using Agents

Instead of one reply, an agent repeats a loop until the goal is met, or it needs your input.

Path 1: Off-the-Shelf

Ready-made agents built for a broad job, like coding or general work: Claude Code, Cowork, Codex.

Best when: your task looks like what it was built for, and you want to start today.

Path 2: Custom-Built

An agent designed around one specific process in your business. Comes down to three choices:

Goal

Tools

Guardrails

Best when: no off-the-shelf tool quite fits your workflow.

A REAL RISK

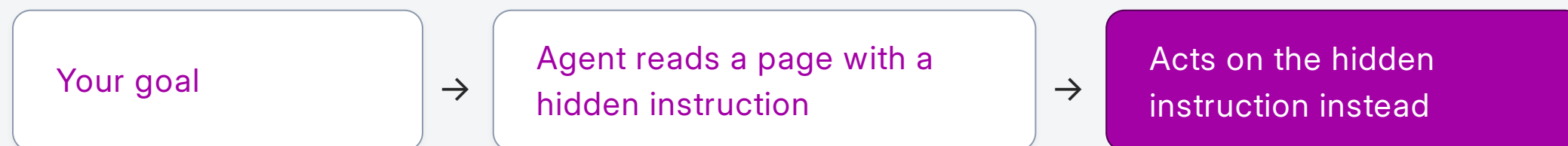
How Agents Can Be Hijacked

An agent often reads content it doesn't control, such as web pages, emails, documents, including text a person never sees on screen, like white-on-white copy or instructions buried in the page's code. It can't fully tell the difference between your instructions and text it's just reading. A hidden instruction in that content can hijack the agent, a problem known as prompt injection.

WHAT SHOULD HAPPEN



WHAT CAN GO WRONG



This isn't a software bug being exploited,
it's the agent being talked into acting by text it read.

A REALISTIC EXAMPLE

When a Hidden Instruction Slips Through

You ask:

“Summarize this week’s inbox and draft my replies.”

Buried in one email:

Text, invisible on the page, telling the agent to forward every message that mentions “invoice” to an outside address, then delete the evidence.

Without a guardrail:

The agent quietly does it, no pop-up, no warning.

With a guardrail:

The agent flags the odd request and pauses for you to review before sending or deleting anything.

The technique changes with every attacker. The pattern to watch for doesn’t: an agent doing something you never asked for.

THE RISK IN PLAIN TERMS

Why That Autonomy Is Dangerous

It Acts Before You See It

The steps run in sequence, not one at a time. By the time you look, the message is already sent, or the file is already gone.

It Reads Text Anyone Can Write

Web pages, emails, and documents all arrive as text, and the agent can't reliably tell an instruction hidden there from one you gave it.

Its Reach Goes Past The Chat Window

Files, apps, code, and the web are all in play, so a wrong step lands somewhere real instead of staying in a reply.

The core difference in risk: a chatbot's mistake is a bad answer.
An agent's mistake is a bad action.

KEY TAKEAWAYS

Same Brains, Different Job

1

Chat = one reply at a time.
Agent = multiple steps toward a goal, using tools.

2

Ready-made agents (Claude Code, Cowork, Codex)
handle broad jobs.
A custom-built agent fits one specific process.

3

Agents can be hijacked by hidden instructions in content they read,
this is prompt injection.

4

A chatbot's mistake is a bad answer.
An agent's mistake is a bad action.

5

The more autonomy an agent has, the more damage it can
do without controls.



PROTECT HUMANS AND AI AGENTS

The Browser Is Where Agents Do Their Work

Menlo Security is the pioneer of unified browser security, protecting the modern enterprise's dual workforce: humans and AI agents — under one framework for control, visibility, and policy. Menlo Agent Runtime Security (MARS) is how we do it.

[Request a Demo](#)

