Does Claude Have Memory? How Context, Projects, and Memory Features Work

Edition of November 13, 2026 — facts verified November 13, 2026. Current edition always at https://tygartmedia.com/does-claude-have-memory/.

Direct Answer (November 13, 2026): Within a single session, current Claude models (Fable 5.2, Opus 5.5, Sonnet 5.5, and Haiku 5.5) hold up to **1,000,000 tokens** of active conversation context. Across sessions, **Projects** (up to 5 on Free; unlimited on Pro, Max, Ultra, Team, and Enterprise per claude.com/pricing) and Claude.ai Memory preserve instructions, files, and user facts.

Last refreshed: November 13, 2026 · Sources: Anthropic models overview & claude.com/pricing.

Claude AI · Fitted Claude

Claude doesn't have persistent memory by default — each conversation starts fresh, with no recollection of previous sessions. But there are several ways to give Claude memory, both through Anthropic's built-in features and through how you structure your interactions. Here's exactly how memory works in Claude and what your options are.

Claude Memory Features by Plan (November 2026)

Feature	Free (up to 5 Projects)	Pro/Max/Ultra	Team	API
Conversation history (within session)	✓	✓	✓	✓ (you manage)
Projects (persistent memory across sessions)	✓ (up to 5 Projects)	✓	✓	Manual via system prompt
Memory (auto-remembered facts)	✗	✓ (beta)	✓ (beta)	✗
Cross-conversation recall	✗	✓ (via Projects/Memory)	✓	You build this
Remembers between devices	✗	✓ (cloud sync)	✓	N/A

The Three Types of Claude Memory

The three types of Claude memory.

Memory Type	What it does	Persists across sessions?
In-conversation context	Everything said in the current chat	No — resets when conversation ends
Projects	Custom instructions + uploaded knowledge	✓ Yes — available in every session
Memory feature	Facts Claude learns about you over time	✓ Yes — grows over time

In-Conversation Context: What Claude Remembers Right Now

Within a single conversation, Claude has full access to everything that's been said — all your messages, all its responses, any files you've uploaded. This is the context window, which for all current 5.x Claude models (Fable 5.2, Opus 5.5, Sonnet 5.5, Haiku 5.5) supports up to **1,000,000 tokens** (200,000 tokens on legacy Haiku 4.5). That covers very long conversations, large documents, and extensive back-and-forth without Claude losing track of earlier details.

When the conversation ends, that context is gone. Start a new conversation and Claude has no knowledge of the previous one.

Projects: Persistent Context Across Sessions

Projects — persistent context across sessions.

Projects are Claude's primary mechanism for persistent memory. A Project is a workspace where you can:

- Set custom instructions that apply to every conversation in that Project
- Upload documents, style guides, or knowledge files that Claude can reference
- Keep your conversation history organized by topic or client

Every conversation you start within a Project has access to those instructions and documents from the beginning — without you having to re-explain your context every time. This is the practical solution for most persistent memory use cases: tell Claude who you are, what you're working on, and what you need once in the Project settings, and it carries forward.

The Memory Feature: Claude Learning About You

The memory feature — Claude learning about you.

Claude.ai has a Memory feature (found in Settings → Memory) where Claude automatically extracts and stores facts about you from your conversations — your job, preferences, ongoing projects, communication style. These memories surface in future conversations to make Claude more personalized without you having to re-introduce yourself.

You can view, edit, and delete individual memories from the settings page. You can also turn the feature off entirely if you'd rather start fresh each time. When Memory is active, Claude may reference things you mentioned in past conversations — "you mentioned you work in restoration…" — which can feel surprisingly persistent for a tool that otherwise has no cross-session recall.

Memory in the API

For developers building on Claude via the API, there's no built-in persistent memory — each API call is stateless by default. Persistent memory for API applications requires building it yourself: storing conversation history in a database and injecting relevant context into each new request. Anthropic's system prompt is the standard mechanism for doing this — load relevant facts or history into the system prompt at the start of each call.

Frequently Asked Questions

Does Claude remember previous conversations?

Not by default. Each new conversation starts fresh. You can enable persistent memory through Projects (custom instructions and uploaded knowledge that apply to every session) or through Claude's Memory feature (which stores facts about you across conversations).

How do I give Claude memory between sessions?

Use Projects: create a Project, add custom instructions describing your context, and upload any relevant documents. Every conversation within that Project will have access to that information from the start — no re-explaining required.

What is Claude's memory feature?

Claude's Memory feature (Settings → Memory) automatically extracts facts about you from conversations and stores them to personalize future interactions. You can view, edit, or delete individual memories, or disable the feature entirely.

Does Claude remember within a conversation?

Yes, fully. Within a single conversation Claude has access to everything said — up to **1,000,000 tokens** of context across current models (Fable 5.2, Opus 5.5, Sonnet 5.5, Haiku 5.5; legacy Haiku 4.5 supports 200,000 tokens).

Does Claude remember things between conversations?

By default, chats start fresh unless you use Projects (up to 5 on Free; unlimited on Pro, Max, Ultra, Team, and Enterprise) or the Memory feature on paid plans.

How does Claude Projects work for persistent memory?

Projects are persistent workspaces in claude.ai (up to 5 on Free; unlimited on Pro, Max, Ultra, Team, and Enterprise). You can upload files, write custom instructions, and share that context across sessions.

Can I give Claude memory via the API?

The Claude API is stateless. To add memory to API-based applications, developers store conversation history in a database and include the relevant history in each API call's system prompt or messages array. Tools like mem0 and LangChain provide memory layers for Claude API applications.

Need this set up for your team?

Talk to Will →

Primary & Reference Links Cited on This Page

- Anthropic models overview — `https://platform.claude.com/docs/en/about-claude/models/overview`
- claude.com/pricing — `https://claude.com/pricing`
- Tygart Media: context window
- Tygart Media: Talk to Will →