OmniCortex Memory MCP

Quick Start Guide

What is OmniCortex?

OmniCortex is a **universal memory system** for Claude Code that stores what you learn, tracks what you do, and helps you never repeat the same mistakes twice.

> **"Store everything important so you never repeat mistakes"**

Unlike Basic Memory MCPs

- **Activity Logging** - Full audit trail of every tool call
- **Session Continuity** - "Last time you were working on..." context
- **Cross-Project Search** - Find knowledge from any project
- **Auto-Categorization** - Intelligent memory type detection
- **Importance Decay** - Frequently accessed memories surface first

Installation (2 Commands)

Step 1: Install the package

```
pip install omni-cortex
```

Step 2: Run automatic setup

```
omni-cortex-setup
```

For AI-powered semantic search: **pip install omni-cortex[semantic]**

Web Dashboard (Optional)

Launch a visual interface for browsing and managing memories:

```
omni-cortex-dashboard
```

Opens at **http://localhost:8765**

Dual-Layer Storage

Activity Log
- What tools were used
- Success/failure status
- Duration and timing
- Automatic via hooks

Knowledge Store
- Solutions found
- Decisions made
- Warnings learned
- Searchable insights

Smart Features

Session Continuity: When you start a new session, OmniCortex provides context from your previous sessions - what you were working on, key decisions made, and where you left off.

Auto-Categorization: Memories are automatically tagged with types like 'solution', 'warning', 'config', 'troubleshooting' based on their content.

Hybrid Search: Combines keyword matching (FTS5) with optional AI-powered semantic search for finding memories by meaning, not just exact words.

Getting Started

Essential Commands

Command	What It Does
`cortex_remember`	Store important information with auto-categorization
`cortex_recall`	Search your memories by keyword or meaning
`cortex_start_session`	Begin a work session and get previous context
`cortex_global_search`	Search memories across ALL your projects
`cortex_list_memories`	Browse memories with filters and sorting
`cortex_get_timeline`	See chronological history of activities

*See **CommandReference.pdf** for complete tool documentation with all parameters.*

Token/Context Usage

- **Initial Load:** ~2KB for tool schemas (18 tools registered)
- **Per Tool Call:** Minimal overhead, results returned as formatted markdown
- **Memory Storage:** SQLite database, no token cost for stored memories

Storage Locations

Location	Path	Purpose
Per-Project	`.omni-cortex/cortex.db`	Project-specific memories
Global Index	`~/.omni-cortex/global.db`	Cross-project search
Config	`.omni-cortex/config.yaml`	Project settings

Restart Claude Code after installation to activate OmniCortex