What is MCP?

Model Context Protocol. The USB for AI. The reason Claude (and every other major AI) can use real tools.

USB	100s	Open	2024
FOR AI AGENTS	SERVERS AVAILABLE	STANDARD	ANTHROPIC RELEASED

The USB analogy

Before USB: every device had its own connector. Different cables for printers, mice, cameras. Pain.

USB standardized the plug. Any compliant device works with any compliant computer.

Before MCP: every AI integration was custom. Claude-specific Slack code. ChatGPT-specific GitHub code.

MCP standardizes the connector. Write one Slack integration. Every MCP-compliant AI can use it.

How it works (5 steps)

1
You run a server
An MCP server exposes tools — `read_file`, `send_message`, `query_db`.

2 · AI connects
Claude Code, Cursor, others connect to the MCP server.

3 · AI sees tools
Gets list of available tools + descriptions.

4 · AI asks server
When AI needs a tool, asks server to run it.

5 · Server returns result, AI continues reasoning
The result becomes part of the conversation context. AI uses it to take the next step.

WHY IT MATTERS RIGHT NOW

By itself, an AI is just text. With MCP, it can read your files, query your database, send a Slack message, deploy to Railway. The model + MCP = the agent.

DON'T OVER-EQUIP

Each MCP server costs context-window tokens before any conversation. Keep 5-8 daily-use; install niche ones per-project. 50 servers = bloated context.

DAILY-USE MCPS (IOS BUILDER)

1. **GitHub**
   Issues, PRs, repos from chat
2. **Filesystem** (built-in)
   Read/write project files
3. **App Store Connect**
   Build status, metadata
4. **Railway**
   Deploy + tail logs
5. **Plausible / PostHog**
   "How did traffic do this week?"

INSTALL IN CLAUDE CODE

1. **/mcp install <name>**
   Inside Claude Code
2. **/mcp list**
   See what's loaded
3. **/mcp remove <name>**
   Trim when over-equipped
4. **Browse modelcontextprotocol/servers**
   Reference catalog

Beginner guide: **djenterprises.ai/blog/what-is-mcp** → modelcontextprotocol.io