Artificial Intelligence in Pla...

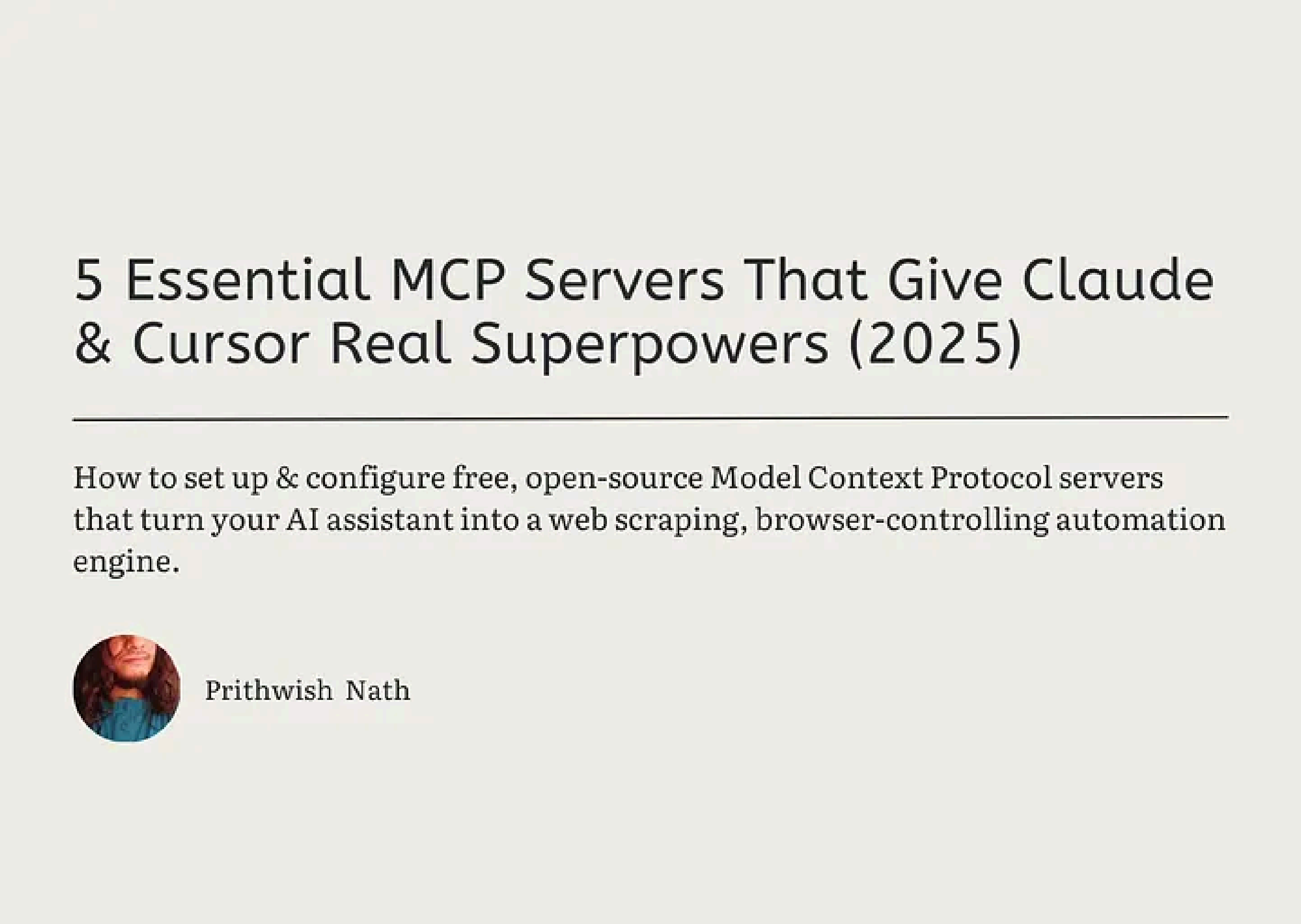
5 Essential MCP Servers That Give Claude & Cursor Real Superpowers (2025)

Transform Claude & Cursor into web scraping, browser-controlling automation engines. 5 essential Model Context Protocol servers with complete setup guides.

Prithwish Nath Follow

15 min read · Sep 27, 2025

371 4



5 Essential MCP Servers That Give Claude & Cursor Real Superpowers (2025)

How to set up & configure free, open-source Model Context Protocol servers that turn your AI assistant into a web scraping, browser-controlling automation engine.

Prithwish Nath

TL;DR: Most AI tools are limited to generating code and answering questions. MCP servers break those walls down. Need to scrape competitor pricing? Want your AI to debug by actually seeing what’s happening in the browser? Process hundreds of PDFs into markdown? There are free and open-source MCP servers for them all.

This guide covers 5 essential MCP servers with step-by-step setup instructions for each.

What You’ll Learn

- How to give Claude/Cursor web scraping capabilities
- Setting up browser automation for your AI assistant

- Converting any document to markdown automatically
- Getting up-to-date library documentation in real-time
- Adding structured reasoning to complex problems

Prerequisites

- Claude Desktop, Cursor, or another MCP-compatible client
- Node.js v18+ for most installations, Docker for others
- 10 minutes per server setup

> What Are MCP Servers?
>
> MCP (Model Context Protocol) is a framework that allows MCP-compatible clients like Claude, Cursor, and many others to connect to external data sources and tools through standardized server interfaces that expose those specific tools and resources to Large Language Models (LLMs).
>
> Once connected, AI can use these tools naturally in conversation. Ask it to "scrape that product page and convert the specs to markdown," and it just works.

> To find out more about how MCP works, check out the MCP documentation.

How MCP Servers Actually Work

Think of MCP servers as **specialized microservices for your AI**. Each one:

- **Exposes tools your AI can call** — Every MCP server publishes a set of functions ("tools") the model can invoke. These tools can be as simple as *get a file's contents,* or as powerful as *run a SQL query*. To the AI, they all look like structured, callable APIs.
- **Maintains conversation context** across multiple tool uses.
- **Handles errors gracefully** with meaningful feedback to the AI.
- **Configures via JSON** for easy customization. Adding or removing a server is as simple as editing a JSON config. Each entry describes the server, how to connect to it (command, URL, etc.), and what permissions it has. This makes it dead-easy to customize your AI's toolbox.

The beauty is in the simplicity — add a server to your config, restart Claude/Cursor/Whatever, and suddenly your AI knows your database inside out, can scrape websites, or control browsers.

> 💡 *I'm focusing entirely on the **free and open-source** part here, which means that some paid Model Context Protocol servers which are genuinely useful — like the Figma MCP which turns designs into code, but is only available on Figma's Professional, Organization, or Enterprise plans — didn't make it on my list.*

Ready to give your AI real superpowers? Let's dive in.

1. Bright Data MCP Server — The Ultimate AI Web Access Tool

Repository: https://github.com/brightdata/brightdata-mcp
Documentation: https://docs.brightdata.com/mcp-server/overview
License: MIT
Free Tier: 5,000 requests/month

What it does:

This is *the* web access layer for LLMs and AI agents. It's a MCP wrapper around all Bright Data's offerings — exposing search, crawling, site navigation, and even browser automation through one server. It handles the complexities that break most scraping attempts: bot detection, CAPTCHAs, geo-restrictions, and JavaScript-heavy sites.

GitHub — brightdata/brightdata-mcp: A powerful Model Context Protocol (MCP) server that provides an…

A powerful Model Context Protocol (MCP) server that provides an all-in-one solution for public web access. …

github.com

Why it's essential:

Agents need live data, but if you've ever wired one up to the open web, you know that modern sites use sophisticated anti-bot measures that defeat simple scrapers. **You can't build bespoke solutions at scale.**

Bright Data's MCP server provides residential proxies, automatic CAPTCHA solving, and browser automation — letting you focus on data extraction instead of fighting infrastructure. This MCP abstracts away the pain points so you can focus on the actual agent logic instead of fighting with infrastructure.

The Web MCP by Bright Data - Start with a Free Plan

Connect LLMs and AI agents to real-time web data with Bright Data MCP Server. Search, crawl, and automate web tasks at...

brightdata.com

Reliable, programmatic data extraction, saving me on context size by being both a web search MCP *and* a browser automation MCP in one, with a whopping 5k free tool calls per month? I'll take it.

Key Features:

- **Real-time Web Access:** Access up-to-date information directly from the web, scale as much as needed.
- **60+ specialized tools** including social media, e-commerce, and search engines
- **Geo-restriction Bypass:** Lets you target specific geographies for localized data discovery
- **JavaScript Rendering:** Renders JavaScript remotely to retrieve dynamic content
- **Browser Control:** Remote browser automation capabilities for complex interactions

Quick Start:

Before anything, sign up for an account below — then **grab your API token.**

Bright Data - All in One Platform for Proxies and Web Scraping

Award winning proxy networks, powerful web scrapers, and ready-to-use datasets for download. Welcome to the world's #1...

brightdata.com

New users get a test API key via welcome email, but you can always generate one from **Dashboard -> Settings -> User management.**

Option 1: Remote Server (Easiest)

```
{
  "mcpServers": {
    "Bright Data": {
    "command": "npx",
    "args": [
      "mcp-remote",
      "https://mcp.brightdata.com/mcp?token=YOUR_API_TOKEN_HERE"
      ]
    }
  }
}
```

Option 2: Running Locally + Optional Advanced Config

```
{
  "mcpServers": {
    "Bright Data": {
      "command": "npx",
      "args": ["@brightdata/mcp"],
      "env": {
        "API_TOKEN": "YOUR_API_TOKEN_HERE",
        "PRO_MODE": "true",              // Enable all 60+ tools
        "RATE_LIMIT": "100/1h",          // Custom rate limiting
        "WEB_UNLOCKER_ZONE": "custom",   // Custom unlocker zone
        "BROWSER_ZONE": "custom_browser" // Custom browser zone
        }
     }
    }
  }
```

Pro vs Basic Mode:

By default, you get basic tools (search and scraping). Enable Pro mode to access nearly 60 tools including browser automation and web data extraction — note that this incurs additional PAYG charges.

Tools Provided (~60 total):

- **General Web Scraping:** 3 tools for search engine results and generic webpage scraping with bot detection bypass
- **Browser Automation:** 9 tools for interactive browsing, clicking, typing, navigation, and screenshots
- **E-commerce Platforms:** 10 tools for structured data from Amazon, Walmart, eBay, Home Depot, Zara, Etsy, BestBuy, Booking.com, and Zillow
- **Social Media Platforms:** 16 tools covering LinkedIn, Instagram, Facebook, X/Twitter, TikTok, YouTube, and Reddit with profiles, posts, comments, and marketplace data
- **Professional/Business Data:** 4 tools for LinkedIn job listings, ZoomInfo, Crunchbase company data, and Yahoo Finance
- And more across Maps, Reviews, App Stores.

How To Use Bright Data Web MCP:

- **Market Research:** “*What are the top 10 fintech startups that raised Series A in 2025?*”
- **Real-time Data:** “*What’s Tesla’s current market cap?*”

- **Content Aggregation:** *"What's the Wikipedia article of the day?"*
- **Competitive Analysis:** *"Scrape the pricing pages of the top 5 project management tools"*

Best for: developers building data collection pipelines, AI agents that need web access, market research automation, competitive intelligence tools, or any application requiring reliable web data extraction at enterprise scale.

2. Chrome DevTools MCP Server — Give Your AI Agent Real Browser Vision

Repository: https://github.com/ChromeDevTools/chrome-devtools-mcp
Documentation: https://github.com/ChromeDevTools/chrome-devtools-mcp/?tab=readme-ov-file#chrome-devtools-mcp
License: Apache 2.0
Free Tier: Unlimited

What it does:

This server lets an AI coding agent (Claude, Gemini, Cursor, Copilot, etc.) connect to a running Chrome instance and control it through DevTools. **That means the agent can actually have full context of live, running Chrome tabs** — inspect the DOM, step through console logs, watch network requests, and automate interactions with the browser.

GitHub — ChromeDevTools/chrome-devtools-mcp: Chrome DevTools for coding agents

Chrome DevTools for coding agents. Contribute to ChromeDevTools/chrome-devtools-mcp development by creating…

github.com

Why it's essential:

Normally, coding agents are working blind — they generate code, but can't see what happens when it runs in a real browser. That disconnect is why their "fixes" often miss the mark. By giving them access to Chrome itself, this server removes the blindfold: agents have context of individual tabs and can actually test, debug, and analyze live pages instead of guessing. It shifts them from code-generation-in-the-abstract to proper end-to-end debugging.

Chrome DevTools (MCP) for your AI agent | Blog | Chrome for Developers

Public preview for the new Chrome DevTools MCP server, bringing the power of Chrome DevTools to AI coding assistants.

developer.chrome.com

Key Features:

- **Performance Insights:** Record performance traces and extract actionable insights using Chrome DevTools
- **Advanced Browser Debugging:** Analyze network requests, take screenshots, check console logs
- **Reliable Automation:** Uses Puppeteer to automate actions in Chrome with automatic wait handling
- **Real-time Code Verification:** Generate fixes and automatically verify they work in the browser
- **Live DOM/CSS Inspection:** Debug styling and layout issues with concrete suggestions based on live browser data

Tools Provided (26 total):

- **Input Automation:** 7 tools for form filling, clicking, typing

- **Navigation Automation:** 7 tools for page navigation and URL handling
- **Emulation:** 3 tools for device/viewport simulation
- **Performance:** 3 tools for trace recording and analysis
- **Network:** 2 tools for request monitoring and CORS debugging
- **Debugging:** 4 tools for console inspection and error analysis

Quick Start:

```
{
  "mcpServers": {
    "chrome-devtools": {
      "command": "npx",
      "args": ["chrome-devtools-mcp@latest"]
    }
  }
}
```

Claude Code CLI:

```
claude mcp add chrome-devtools npx chrome-devtools-mcp@latest
```

VS Code CLI:

```
code --add-mcp '{"name":"chrome-devtools","command":"npx","args":["chrome-devtoo
```

Advanced Configuration:

```
{
  "mcpServers": {
    "chrome-devtools": {
      "command": "npx",
      "args": [
        "chrome-devtools-mcp@latest",
        "--channel=canary",
        "--headless=true",
        "--isolated=true"
      ]
    }
  }
}
```

On first install, try this prompt to verify everything works:

> *“Check the performance of https://developers.chrome.com"*

How To Use Chrome DevTools MCP:

- **Real-time Debugging:** “*A few images on localhost:3000 are not loading. What’s happening?*” — Your AI can inspect network requests, identify CORS issues, and suggest fixes.
- **Performance Optimization:** “*Localhost:8080 is loading slowly. Make it load faster.*” — AI runs performance traces, analyzes LCP metrics, and provides specific optimization recommendations.
- **Visual Bug Fixing:** “*The banner on localhost:3000 keeps overflowing the page. Check what’s happening there.*” — AI inspects live DOM/CSS and suggests layout fixes based on actual browser rendering.

- **Form Flow Testing:** *"Why does submitting the form fail after entering an email address?"* — AI can navigate, fill forms, and debug complex user interactions.
- **Code Verification:** *"Verify in the browser that [change] works as expected."* — AI generates code and immediately tests it in a real browser environment.

Perfect for: Web developers who want AI assistance with debugging, performance optimization teams, and anyone building applications where the AI needs to understand browser behavior.

3. MarkItDown MCP Server — Universal Document-to-Markdown Converter

Repository: https://github.com/microsoft/markitdown/tree/main/packages/markitdown-mcp
Documentation: https://github.com/microsoft/markitdown?tab=readme
License: MIT
Free Tier: Unlimited

What it does:

This server wraps Microsoft's MarkItDown engine and exposes a single function: `convert_to_markdown(uri)` . The URI can point to a file, a URL, or even a data string, and it'll convert the contents into clean Markdown. It supports STDIO, streamable HTTP, and SSE, so it slots neatly into AI agent workflows.

markitdown/packages/markitdown-mcp at main · microsoft/markitdown

Python tool for converting files and office documents to Markdown. — markitdown/packages/markitdown-mcp at main ·...

github.com

Why it's essential:

A lot of valuable data is stuck in formats that LLMs don't handle well — Word docs, PDFs, PowerPoints, etc. MarkItDown flattens those into structured Markdown, which is far easier for agents to process. Instead of trying to patch together ad-hoc parsers, you get one reliable pipeline that turns "legacy document" content into something immediately usable by an AI. **For me, it's less about yet-another-file-converter and more about unlocking documents as a first-class input source for agents.**

Universal URI Support:

- **HTTP/HTTPS URLs:** Direct web content conversion
- **File URIs:** Local document processing
- **Data URIs:** Base64 encoded content processing
- **Any MarkItDown-supported format:** Office docs, PDFs, images, audio, archives, and more

Multiple Server Modes:

- **STDIO:** Default mode for MCP client integration
- **Streamable HTTP:** Network-accessible HTTP server
- **SSE (Server-Sent Events):** Real-time streaming mode
- **Docker:** Containerized deployment with volume mounting

Quick Start:

Python Package:

```
pip install markitdown-mcp
markitdown-mcp # STDIO mode
markitdown-mcp --http --host 127.0.0.1 --port 3001 # HTTP/SSE mode
```

Docker (Recommended for Claude Desktop):

```
docker build -t markitdown-mcp:latest .
docker run -it --rm markitdown-mcp:latest
```

```
{
  "mcpServers": {
    "markitdown": {
      "command": "docker",
      "args": [
        "run",
        "--rm",
        "-i",
        "markitdown-mcp:latest"
      ]
    }
  }
}
```

To Access Local Files:

```
{
  "mcpServers": {
    "markitdown": {
      "command": "docker",
      "args": [
        "run",
        "--rm",
        "-i",
        "-v",
        "/home/user/data:/workdir",
        "markitdown-mcp:latest"
      ]
    }
  }
}
```

Tools Provided (1 total):

- `convert_to_markdown(uri)` — This server exposes a single powerful tool that handles any supported file format.

How To Use MarkItDown MCP::

- **Documentation Pipeline:** “*Convert all our legacy Word docs in this folder to Markdown for our new developer portal*” — Perfect for migrating documentation systems.

- **Content Analysis:** “*Extract and convert all PDFs from our quarterly reports to analyze trends*” — Enables AI analysis of previously locked content.

- **Meeting Notes Processing:** “*Convert these PowerPoint presentations and audio recordings from our strategy meetings into searchable Markdown*” — Transforms mixed-media content into analyzable text.

- **Knowledge Base Migration:** “*Take these Excel spreadsheets with our product specs and convert them to Markdown tables*” — Modernizes data formats for wiki-style documentation.

- **Research Paper Analysis:** “*Convert these academic PDFs to Markdown so I can analyze research trends across 50 papers*” — Unlocks structured analysis of research documents.

Perfect for: Technical writers, content managers, researchers, and developers building document processing pipelines.

4. Context7 MCP Server — Up-to-Date Documentation For Your Coding Agent

Repository: https://github.com/upstash/context7
Website: https://context7.com
License: MIT
Free Tier: Unlimited for personal use

What it does:

Built by Upstash, Context7 gives AI agents access to always up-to-date, version-specific documentation pulled straight from official sources. It's not just static docs — it also includes working code examples, formatted so LLMs can actually consume and use them.

GitHub — upstash/context7: Context7 MCP Server — Up-to-date code documentation for LLMs and AI…

Context7 MCP Server — Up-to-date code documentation for LLMs and AI code editors — upstash/context7

github.com

Why it's essential:

Most LLMs are stuck with whatever docs were available at training time, which means newer releases (Next.js 15, Tailwind 4, some entirely new library etc.) are either missing or misrepresented (ever tried to tell your LLM to use `Framer Motion` vs. use `Motion` ? It *will* get confused by their renaming.)

That’s why you often get broken snippets or hallucinated APIs. Context7 sidesteps this by letting agents fetch the actual documentation for the exact version you’re using. No hallucinations, no trial-and-error, and working code on the first pass.

Introducing Context7: Up-to-Date Docs for LLMs and AI Code Editors

Articles and tutorials on serverless technologies from Upstash and community

upstash.com

Key Features:

- **Always Up-to-date Documentation:** Version-specific, current information on thousands of libraries, fetched on demand
- **Real Code Examples:** Working snippets straight from official docs
- **Concise Information:** Relevant details with no filler
- **Free Personal Use:** No cost for individual developers
- **Has the most comprehensive installation coverage of any MCP server,** supporting 20+ clients including Cursor, VS Code, Claude Desktop, Windsurf, Cline, JetBrains, Warp, and more.

Quick Start:

Remote (recommended):

```
{
  "mcpServers": {
    "context7": {
      "url": "https://mcp.context7.com/mcp"
    }
  }
}
```

Local:

```
{
  "mcpServers": {
    "context7": {
      "command": "npx",
      "args": ["-y", "@upstash/context7-mcp"]
    }
  }
}
```

Tools Provided (2 Total):

- **`resolve-library-id`**: Converts library names to Context7-compatible IDs
- **`get-library-docs`**: Fetches up-to-date documentation with optional topic filtering

How To Use Context7 MCP:

Simply add `use context7` to any prompt:

```
Create a Next.js middleware that checks for JWT in cookies. use context7
```

```
Configure a Cloudflare Worker to cache JSON responses. use context7
```

Context7 automatically fetches current docs and injects them into your AI's context.

Or, if you want an exact library, use the slash syntax:

```
implement basic authentication with supabase. use library /supabase/supabase
```

You can also set up automatic Context7 usage in Cursor or Windsurf:

```
[[calls]]
match = "when the user requests code examples, setup or configuration steps, or
tool = "context7"
```

Before Context7: Ask Claude about Upstash Redis stream trim → Get hallucinated code that doesn't work → Spend time debugging → Multiple back-and-forth iterations

After Context7: Add `use context7` → Get working code from official docs on first try → Ship faster

Perfect For:

- **New Library Adoption:** Working with post-training-cutoff libraries
- **Version Upgrades:** Migrating to newer framework versions
- **Complex APIs:** Libraries with frequently changing interfaces

- **Niche Packages:** Lesser-known libraries LLMs weren’t trained on
- **Production Code:** When correctness matters more than speed

And anyone tired of debugging AI-generated code that doesn’t work.

5. Sequential Thinking MCP Server — Structured Reasoning for Complex Problems

Repository:
https://github.com/modelcontextprotocol/servers/tree/main/src/sequentialthinking
License: MIT
Free Tier: Unlimited

What it does:

This server, built by Anthropic themselves, gives agents a framework for step-by-step reasoning. Instead of jumping straight to an answer, it encourages them to break a problem down, revisit earlier steps as they learn more, and keep track of context across multiple stages. It’s basically “think before you talk” for AI — a scaffold for more deliberate problem solving.

servers/src/sequentialthinking at main · modelcontextprotocol/servers

Model Context Protocol Servers. Contribute to modelcontextprotocol/servers development by creating an accou…

github.com

Why it matters:

Most LLMs rush to the first plausible answer, which is fine for simple queries but falls apart on harder problems. By enforcing a structured process — *break it down, check assumptions, revise if needed* — this MCP server helps agents handle complexity more like a human analyst would. It's not about making them "smarter," but about giving them a reliable way to organize their reasoning and avoid shallow one-shot responses.

Key Features:

- **Step-by-Step Breakdown:** Decompose complex problems into manageable, sequential thoughts
- **Dynamic Revision:** Revise and refine thoughts as understanding deepens
- **Alternative Reasoning Paths:** Branch into different approaches when needed

- **Adaptive Planning:** Adjust the total number of thoughts dynamically based on problem complexity
- **Hypothesis Generation:** Generate and verify solution hypotheses systematically
- **Context Maintenance:** Preserve thinking context across multiple steps

Tools Provided (1 total):

- **`sequential_thinking(params)`** —This tool helps analyze problems through a flexible thinking process that can adapt and evolve. Each thought can build on, question, or revise previous insights as understanding deepens.

Parameters:

- `thought (string)` : The current thinking step
- `nextThoughtNeeded (boolean)` : Whether another thought step is needed
- `thoughtNumber (integer)` : Current thought number
- `totalThoughts (integer)` : Estimated total thoughts needed
- `isRevision (boolean)` : Whether this revises previous thinking
- `revisesThought (integer)` : Which thought is being reconsidered
- `branchFromThought (integer)` : Branching point for alternative reasoning
- `branchId (string)` : Branch identifier for tracking different paths
- `needsMoreThoughts (boolean)` : If additional thoughts are required

Quick Start:

NPX (Recommended):

```
{
  "mcpServers": {
    "sequential-thinking": {
    "command": "npx",
      "args": [
        "-y",
        "@modelcontextprotocol/server-sequential-thinking"
      ]
    }
  }
}
```

Docker:

```
{
  "mcpServers": {
    "sequentialthinking": {
    "command": "docker",
    "args": [
        "run",
        "--rm",
        "-i",
        "mcp/sequentialthinking"
      ]
    }
  }
}
```

Advanced Configuration:

Set `DISABLE_THOUGHT_LOGGING=true` environment variable to disable logging of thought information for privacy.

How To Use Sequential Thinking MCP:

- **System Architecture Design:** "*Design a microservices architecture for an e-commerce platform*" — The AI can break this down into: infrastructure considerations, service boundaries, data flow, security implications, and deployment strategies, revising each as the design evolves.
- **Complex Debugging:** "*Debug this multi-threaded performance issue*" — Systematic analysis through: symptom identification, hypothesis formation, testing strategies, result interpretation, and solution refinement.
- **Strategic Planning:** "*Plan our product launch strategy*" — Step through: market analysis, competitive positioning, timeline development, resource allocation, and risk mitigation, with ability to revise based on new insights.
- **Code Refactoring:** "*Refactor this legacy codebase to modern patterns*" — Methodical approach: code analysis, pattern identification, migration planning, dependency mapping, and execution strategy.
- **Research Analysis:** "*Analyze the implications of this new AI regulation*" — Structured thinking: regulation breakdown, industry impact assessment, compliance requirements, implementation timeline, and strategic recommendations.

Perfect for: Senior developers tackling architectural decisions, technical leads planning major changes, academics planning research, and any complex problem at all where the solution path isn't immediately obvious.

Quick Comparison: Which Server Fits Your Needs?

- **Bright Data MCP** — enables AI agents to access, navigate, and extract from real websites with proxy / CAPTCHA handling; good for production web data.

- **Chrome DevTools MCP** — gives a live coding agent full browser control & inspection via DevTools.
- **MarkItDown MCP** — turns virtually any document (PDF, Word, Excel, audio, images, HTML, YouTube, ZIP, etc.) into clean Markdown.
- **Context7 MCP** — provides version-specific, up-to-date docs and working code examples for any library, for AI agents.
- **Sequential Thinking MCP** — gives agents a structured reasoning / multi-step problem solving framework instead of jumping to answers.

```
| MCP Server                     | What it does
|--------------------------------|------------------------------------------------
| **Bright Data MCP Server**     | Agents can search, browse, navigate, and extrac
| **Chrome DevTools MCP Server** | Expose a live Chrome browser via DevTools so 
| **MarkItDown MCP Server**      | Convert any document (PDF, Word, Excel, Audio, 
| **Context7 MCP Server**        | Provide versioned, up-to-date documentation and
| **Sequential Thinking MCP Server** | Offer a structured reasoning engine so ag
```

Pro Tips

- **The best tip I can give you would be to stack MCP servers.** Use Sequential Thinking to plan out a research path, Bright Data to scrape based on that plan, the LLM to analyze, and finally, Context7 MCP to know how best to store findings with metadata to your ChromaDB instance using its SDK — all in one workflow.
- **Start small:** The MCP ecosystem is young but growing fast. There are thousands of MCP servers out there. Don't get overwhelmed — pick one server that matches your immediate need and start building solutions. You can experiment with combinations later.

- **Watch context limits:** More servers mean more active tools bloating your context. Five servers work well together, but too many can degrade performance, or cause strange/unpredictable LLM behavior.

As the ecosystem grows, the developers who start experimenting now will be the ones building the next generation of AI-powered applications.

Have you tried any of these MCP servers? What has your experience been like? Found others worth recommending? Share in the comments below 👇.

Programming · Artificial Intelligence · Machine Learning · Ai Agent · Web Development

Published in Artificial Intelligence in Plain English

30K followers · Last published 13 hours ago

Follow

New AI, ML and Data Science articles every day. Follow to join our 3.5M+ monthly readers.

Written by Prithwish Nath

842 followers · 9 following

Follow

Fullstack developer, audiophile, strategy game enthusiast.

Responses (4)

Bgerby

What are your thoughts?

Keith

20 hours ago

Terrific help, thanks!

1 Reply

Riccardo Tartaglia he/him

10 hours ago

Can I add artiforge.ai ?

Code with AI in easy mode!

Reply

Donkey

17 hours ago

It was really helpful, I have tested Context7 MCP server and it was amazing, thanks for sharing.

Reply

See all responses

More from Prithwish Nath and Artificial Intelligence in Plain English

I Built a Documentation Agent With N8N + MCP That Turns GitHub READMEs Into Full Stack Best Practices

How to automatically generate a living “Best Practices Bible” from your dependencies using AI + N8N + MCP – it’s like a fully local Dependabot, but for documentation.

Prithwish Nath

AI In Artificial Intelligence in Plain E... by Prithwish N...

How To Build an AI Documentation Agent with N8N + MCP that Turns...

Step-by-step guide to automatically generate a living “Best Practices Bible” from your...

Sep 23 👏 142 💬 2

AI In Artificial Intelligence in Plain ... by Simranjeet S...

RAG is Hard Until I Know these 12 Techniques → RAG Pipeline to 99...

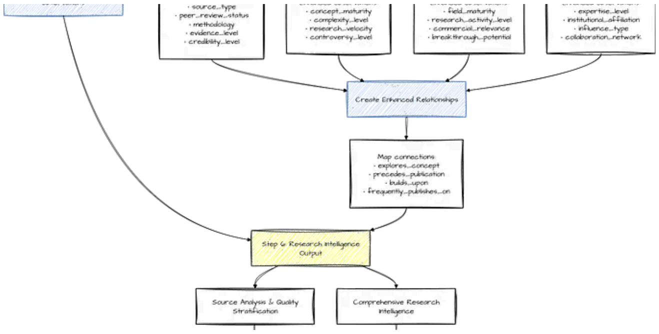
RAG is Hard Until I Know these 12 Techniques → RAG Pipeline to 99% Accuracy. Best Blog...

✦ Sep 27 👏 426 💬 9

AI In Artificial Intelligence in Plain ... by Simranjeet S...

OpenAI ML Engineer Interview Questions 2025

A mock interview with an OpenAI ML engineer covering LLM deployment, low-...



AI In Artificial Intelligence in Plain E... by Prithwish N...

I Connected 3 MCP Servers to Claude & Built a No-Code Researc...

A step-by-step tutorial to building a research assistant that can’t hallucinate, by connectin...

Sep 24 139 8

Sep 16 182

See all from Prithwish Nath

See all from Artificial Intelligence in Plain English

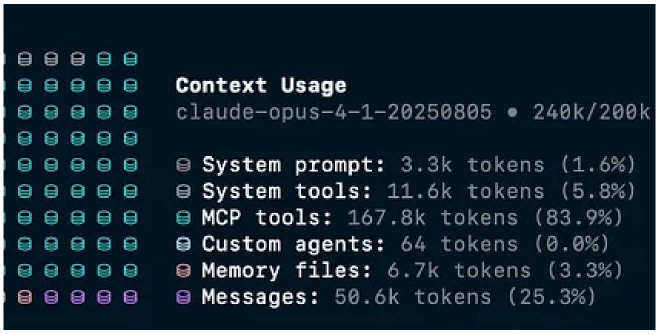
Recommended from Medium

In Data Science Collective by Erdogan T

Build Your Private Language Model: Local and Specialized For...

A complete step-by-step guide from setup to deployment of local language models, makin...

Oct 3 726 4



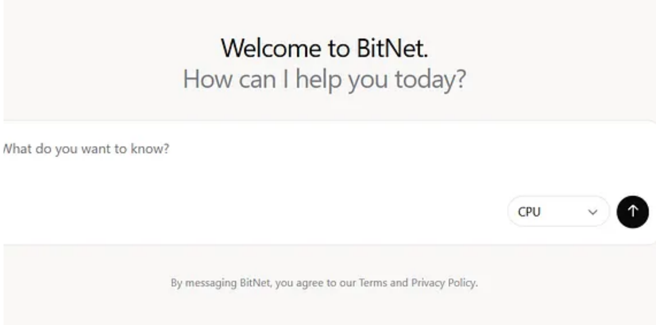
```
Context Usage
claude-opus-4-1-20250805 • 240k/200k

System prompt: 3.3k tokens (1.6%)
System tools: 11.6k tokens (5.8%)
MCP tools: 167.8k tokens (83.9%)
Custom agents: 64 tokens (0.0%)
Memory files: 6.7k tokens (3.3%)
Messages: 50.6k tokens (25.3%)
```

Sevak Avakians

Claude Code Limit Hit on Max Plan?! What to do next:

$200/m plan blocks me for a week!

Oct 3 87 17



Welcome to BitNet.
How can I help you today?

What do you want to know?

By messaging BitNet, you agree to our Terms and Privacy Policy.

In Coding Nexus by Algo Insights

Microsoft Just Declared War on the GPU Mafia: Meet bitnet.cpp

Bitnet.cpp will break the GPU Lock

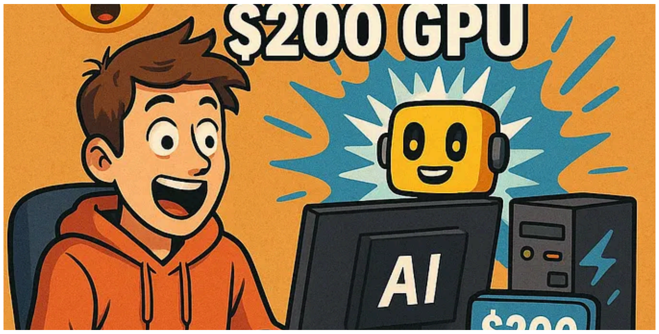
5d ago 334 5

In Level Up Coding by Yasas Sandeepa

Google's New AI – Nano Banana: Top 10 Use Cases That Will Blow...

How Nano Banana is redefining the future of AI Image Gen

Oct 2 208 2

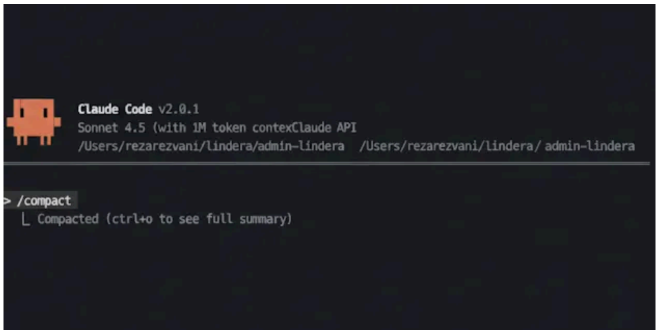


In Towards Deep Learning by Sumit Pandey

Meet oLLM: The Secret Sauce to Run Huge AI on Tiny Hardware

oLLM slashes LLM memory use: Run 100k context GPTs on 8GB GPUs. A lightweight...

Oct 2 162 9



```
Claude Code v2.0.1
Sonnet 4.5 (with 1M token contexClaude API
/Users/rezarezvani/lindera/admin-lindera  /Users/rezarezvani/lindera/ admin-lindera

> /compact
  ⎿ Compacted (ctrl+o to see full summary)
```

Reza Rezvani

The Complete Claude Code 2.0

Claude Code 2.0 isn't just an update—it's a complete reimagining of how AI assists...

Sep 30 295 11

See more recommendations