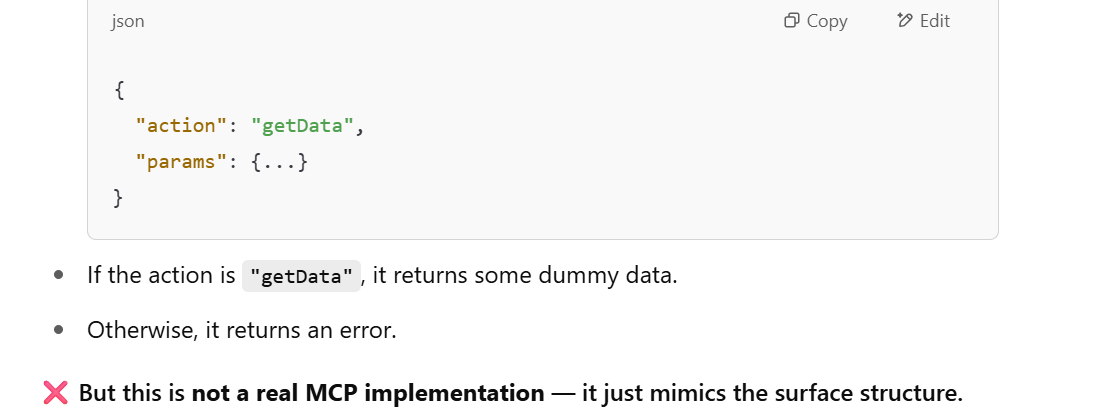
Server **code is:**

* A minimal **JSON API server**
* That accepts POST requests at /mcp/action
* It expects the request body to contain:



**So What *Is* MCP?**

MCP (Model-Context-Protocol), as proposed in emerging GenAI infrastructure, is meant to be a **standardized way to allow LLMs (Large Language Models)** to communicate with tools, APIs, and agents by passing structured **intents and context**.

It involves:

* A **standard message format** (like function-calling or tool-use metadata)
* A **way to define capabilities/tools** the model can invoke
* A **router/server** that can resolve these requests and return structured results

**What a Real MCP Server Would Involve:**

To get closer to **real MCP functionality**, you'd need:

1. **Schemas** to define allowed actions (tools)
2. **Context-handling** (e.g., memory or embeddings)
3. Possibly **OpenAI function calling or LangChain tool calling**
4. An interface for a model (or LangChain Agent) to invoke these endpoints

**If You Want a More Realistic MCP-style Implementation**

I can help you:

* Build an actual tool registry
* Use OpenAI function calling to define the tools
* Route them to this server
* Return structured output to the model

build an actual **end-to-end MCP-like prototype** with OpenAI function calling + Express + LangChain (or similar)