

# SindByte MCP Server

## Complete User Manual

Version 2.3 - March 2026

221 Verified Tools | Trading-AI Calendar | Agent Probe Guide

### Complete Documentation:

- \* 221 Verified Tools
- \* Installation Guides
- \* Trading-AI Calendar
- \* Safety Guard
- \* Agent Probe Guide
- \* HTTP Diagnostics
- \* Interface Screenshot

(C) 2026 Theo Gottwald

Herrenstr.11, 76706 Dettenheim

<http://smart-ai-robot.com>

# 1. Quick Start

---

Welcome to SindByte MCP Server - your AI-powered automation hub with 221 verified tools across 16 categories.

## System Requirements

- Windows 10/11 (desktop environment)

- Ne

## 2. Installation Guide

---

### 2.1 LM Studio (GUI)

- 1. Copy folder to PC

- 2.

### 2.2 OpenAI Codex (CLI)

```
# Start server
SindByte 01.exe

# Add to Codex
codex mcp add SindByte --url http://127.0.0.1:5555/mcp

# Verify
codex mcp list
```

### 2.3 VS Code / Cline

```
{
  "mcpServers": {
    "Sindbyte-MCP-Server": {
      "url": "http://127.0.0.1:5555/mcp",
      "type": "streamableHttp",
      "disabled": false,
      "autoApprove": ["list_tools"],
      "timeout": 300
    }
  }
}
```

### 3. Tool Categories Overview

---

SindByte MCP Server provides 221 verified tools organized into 16 categories:

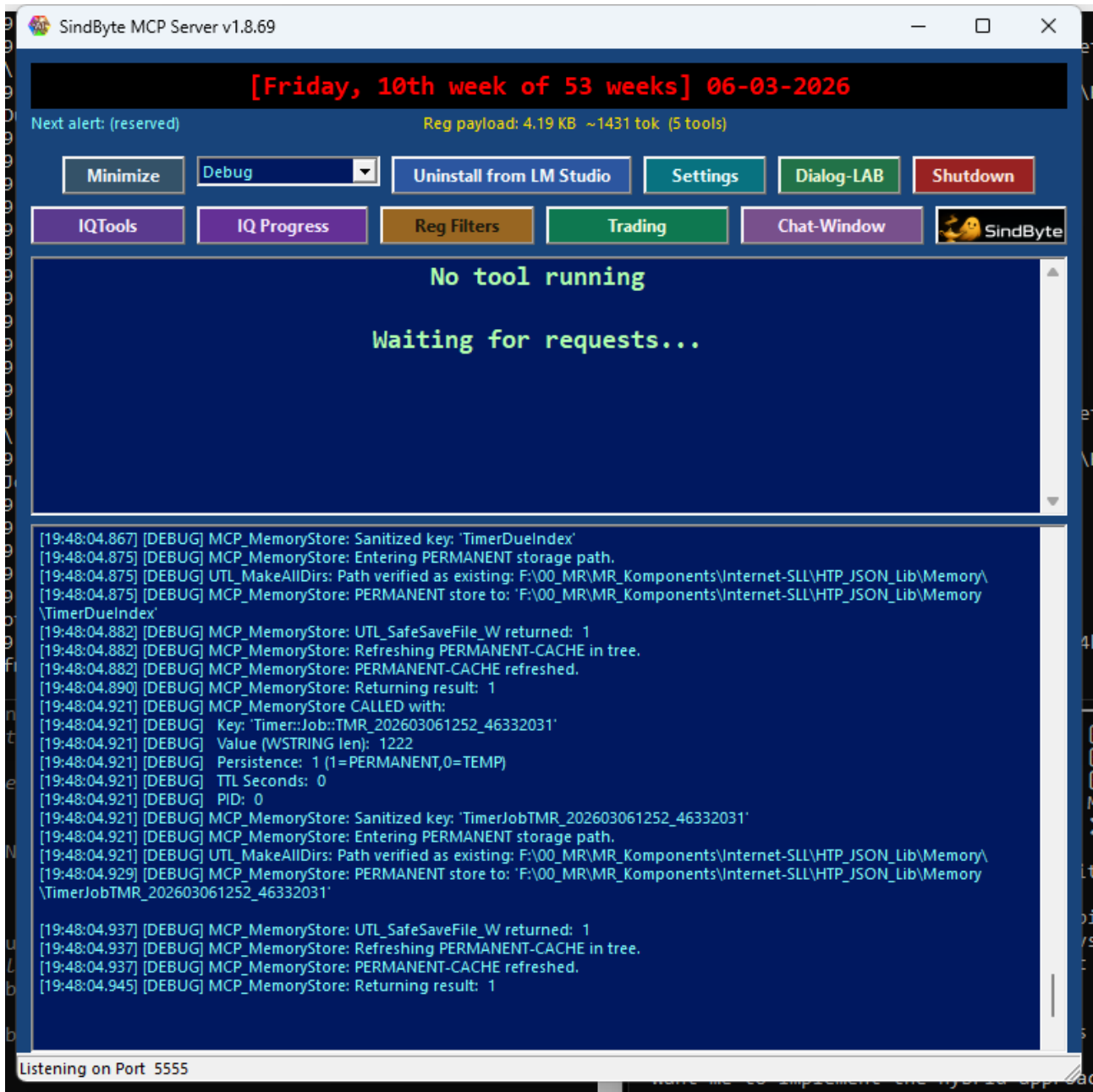
- ClipboardTools: 4 tools

- CU

## 4. SindByte Interface

The main SindByte window provides access to all features:

Figure: SindByte MCP Server Main Window v1.8.69



Key elements: Top buttons (Minimize, Debug, Uninstall, Settings, Dialog-LAB, Shutdown), Feature buttons (IQTools, IQ Progress, Reg Filters, Trading, Chat-Window), Status area, Debug log, Port 5555 indicator.

## 11. Agent MCP Server Probe Guide

Purpose: This guide shows how to check the running SindByte MCP Server directly via HTTP without modifying VS Code or Cline configuration. For pure runtime diagnostics, use direct HTTP requests from PowerShell.

### 11.1 Location and Prerequisites

- Project folder: F:\00\_MR\MR\_Komponents\Internet-SLL\HTP\_JSON\_Lib

- Se

### 11.2 SSE Handshake Check

This probe shows whether the server responds to GET /mcp with an open SSE handshake:

```
$req=[System.Net.HttpWebRequest]::Create('http://127.0.0.1:5555/mcp')
$req.Method='GET'
$req.Accept='text/event-stream'
$req.Timeout=5000
$req.ReadWriteTimeout=5000
$resp=$req.GetResponse()
[pscustomobject]@{
    StatusCode = [int]$resp.StatusCode
    ContentType = $resp.ContentType
    Headers = $resp.Headers
}
$resp.Close()
```

Expected: StatusCode = 200, ContentType = text/event-stream; charset=utf-8

### 11.3 Initialize via POST

This request checks if the server correctly answers modern MCP initialization via JSON:

```
$initBody='{ "jsonrpc": "2.0", "id": 0, "method": "initialize", "params": { "protocolVersion": "2025-11-25", "capabilities": {}, "clientInfo": { "name": "CodexProbe", "version": "1.0" } } }'

$init=Invoke-WebRequest `
    -Uri 'http://127.0.0.1:5555/mcp' `
    -Method Post `
    -ContentType 'application/json' `
    -Headers @{
        Accept='application/json, text/event-stream'
        'MCP-PROTOCOL-VERSION'='2025-11-25'
    } `
    -Body $initBody `
    -UseBasicParsing `
    -TimeoutSec 20

$init.StatusCode
$init.Headers['Mcp-Session-Id']
$init.Content
```

Expected: HTTP 200, Header Mcp-Session-Id, JSON-RPC result with protocolVersion, serverInfo, capabilities

## 11.4 Tools List Query

With the session from initialize, tools/list can be checked directly:

```
$sid=$init.Headers['Mcp-Session-Id']
$body='{ "jsonrpc": "2.0", "id": 1, "method": "tools/list" }'

$resp=Invoke-WebRequest `
  -Uri 'http://127.0.0.1:5555/mcp' `
  -Method Post `
  -ContentType 'application/json' `
  -Headers @{
    Accept='application/json, text/event-stream'
    'MCP-PROTOCOL-VERSION'='2025-11-25'
    'Mcp-Session-Id'=$sid
  } `
  -Body $body `
  -UseBasicParsing `
  -TimeoutSec 30

$json=$resp.Content | ConvertFrom-Json

[pscustomobject]@{
  SessionId      = $sid
  StatusCode      = $resp.StatusCode
  ContentLength   = $resp.Headers['Content-Length']
  ToolCount       = @($json.result.tools).Count
  ToolNames       = @($json.result.tools | Select-Object -ExpandProperty name)
}
```

Freshly verified reference: StatusCode = 200, Content-Length = 43782, ToolCount = 221

## 11.5 Debug.txt Verification

After a probe, always cross-check Debug.txt:

```
Get-Content '.\Debug.txt' -Tail 200
```

Important lines for POST probes: POST /mcp HTTP/1.1, K=ACCEPT V=application/json, text/event-stream, K=MCP-PROTOCOL-VERSION V=2025-11-25, INT\_ProcessCommand: Handling as a REQUEST for method: initialize/tools/list, POST done (bytes)

Important lines for SSE: GET /mcp HTTP/1.1, legacy SSE handshake, Sent legacy endpoint event: /mcp?session\_id=...

## 11.6 Dual-Mode Understanding

The server is currently dual-mode: GET /mcp delivers SSE handshake. POST /mcp delivers modern JSON-MCP responses.

This means: The server is not SSE-only. The server is not pure Streamable HTTP-only. Both paths must be considered for Cline/VS Code problems.

## 11.7 Common Misinterpretations

- A successful initialize body alone does not prove complete client integration is healthy

- Du

## 12. Troubleshooting

---

### Common Issues

- LM Studio does not show SndByte: Ensure server running, verify URL `http://127.0.0.1:5555/mcp`, restart LM Studio, check `mcp.json` syntax

- Ins



# 13. Legal Information

---

- Use is at your own responsibility

- No

*Document Version: 2.3 - March 2026*

*Includes content from AGENT\_MCP\_SERVER\_PROBE\_GUIDE.md*