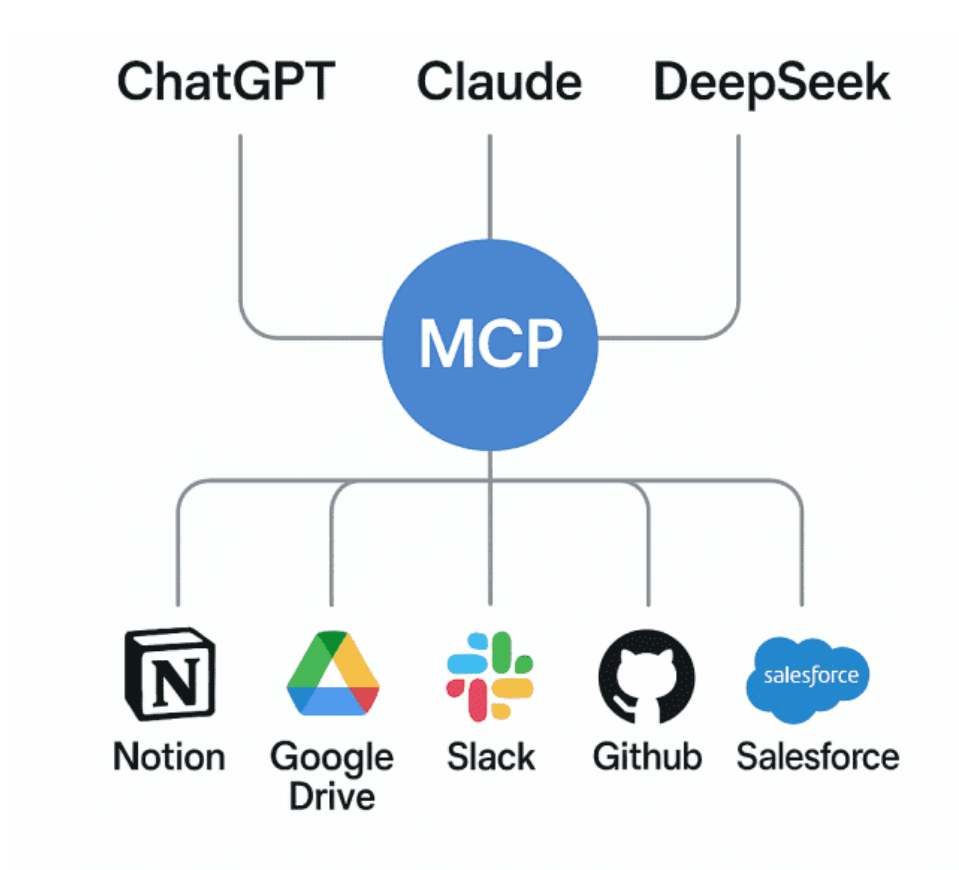


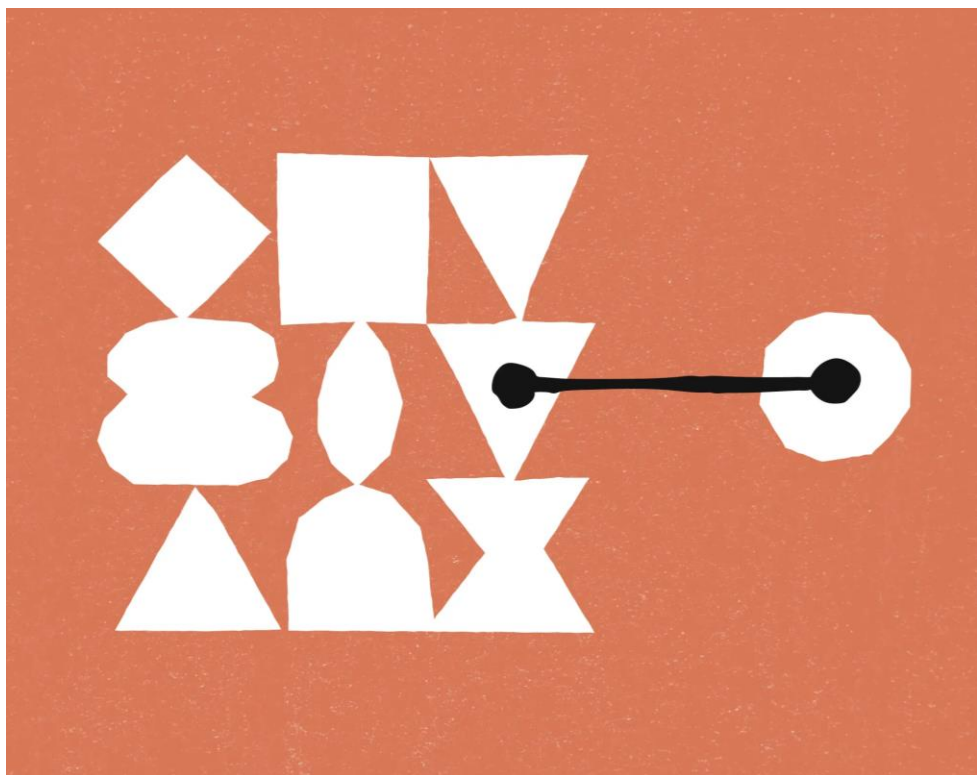


Model Context Protocol



MCP를 선택해야 하나요?

MCP(Model Context Protocol)는 Anthropic Claude에서 제공하는 오픈소스 프로토콜로, AI 모델과 개발 환경 간의 통합된 컨텍스트 상호작용을 확립하는 것을 목표로 합니다.



■ 활용 사례

코드 관리 및 개발

MCP 프로토콜을 통해 GitHub에 직접 연결, AI가 코드 작성, 저장소 생성, 코드 푸시, 이슈 생성, 브랜치 생성, PR 생성 등의 작업을 자동화

데이터 관리 및 상호작용

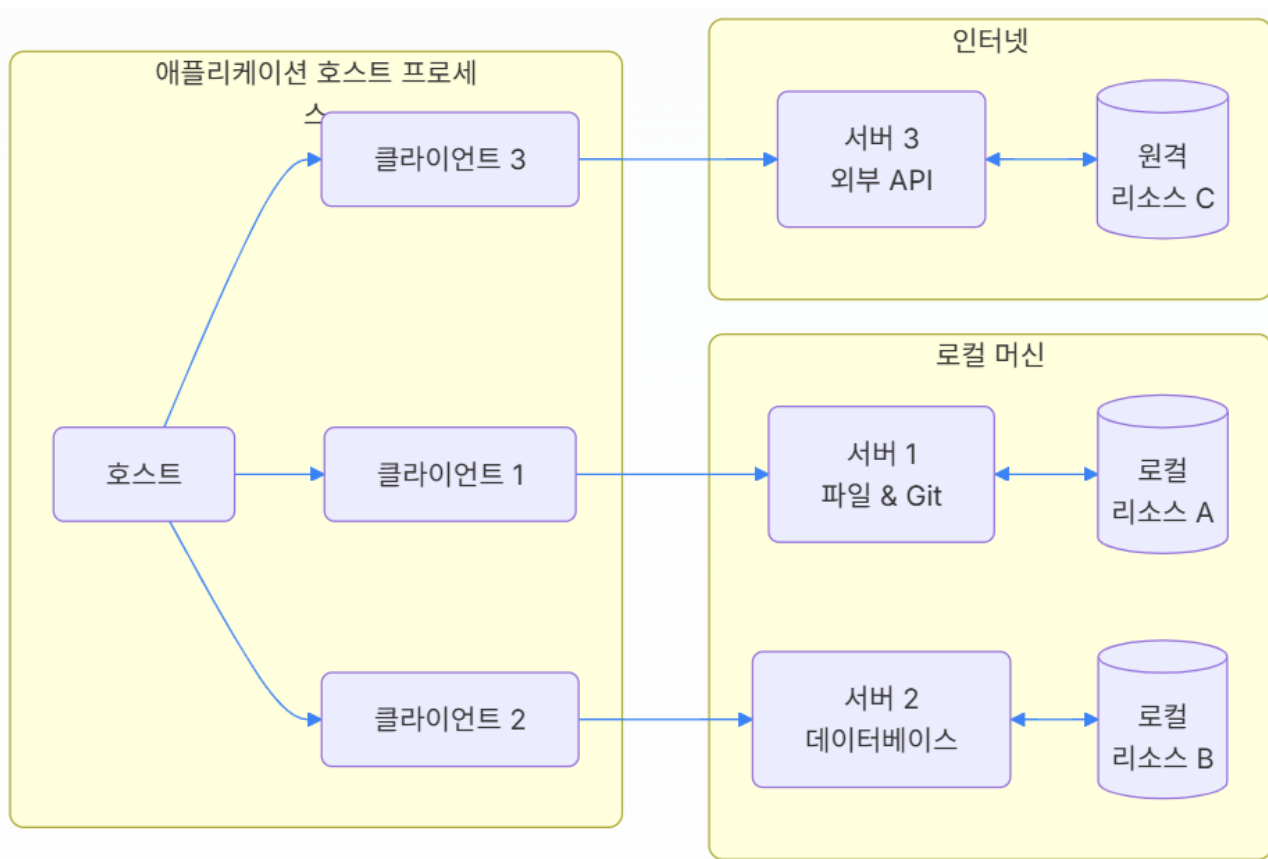
로컬 리소스 관리 : 컴퓨터의 파일, DB 등 로컬 리소스 관리를 지원

원격 리소스 상호작용 : Google Drive, Slack과 같은 원격 리소스를 MCP 프로토콜을 통해 직접 제어하고 접근

지능형 어시스턴트 애플리케이션 구축

MCP 프로토콜을 사용하여 다양한 내부 데이터 리소스(데이터베이스, 파일 서버 등)와 외부 비즈니스 도구(프로젝트 관리 도구 등)를 연결하여 더 포괄적이고 효율적인 서비스를 제공

MCP 아키텍처



- 각 호스트가 여러 클라이언트 인스턴스를 실행할 수 있는 클라이언트-호스트-서버 아키텍처
- JSON-RPC 를 기반으로 구축된 MCP는 클라이언트와 서버 간의 컨텍스트 교환 및 샘플링 조정에 중점을 둔 상태 유지 세션 프로토콜 제공
- **호스트** : 연결을 시작하는 LLM 애플리케이션 (Claude Desktop, Cursor IDE 등)
- **클라이언트** : 호스트 애플리케이션 내에서 서버와 1:1 연결을 유지
- **서버** : 클라이언트에게 컨텍스트, 도구, 프롬프트를 제공

MCP 메시지 유형과 전송 메커니즘

요청 메시지

```
{
  "jsonrpc": "2.0",
  "id": "string | number",
  "method": "string",
  "param?": {
    "key": "value"
  }
}
```

응답 메시지

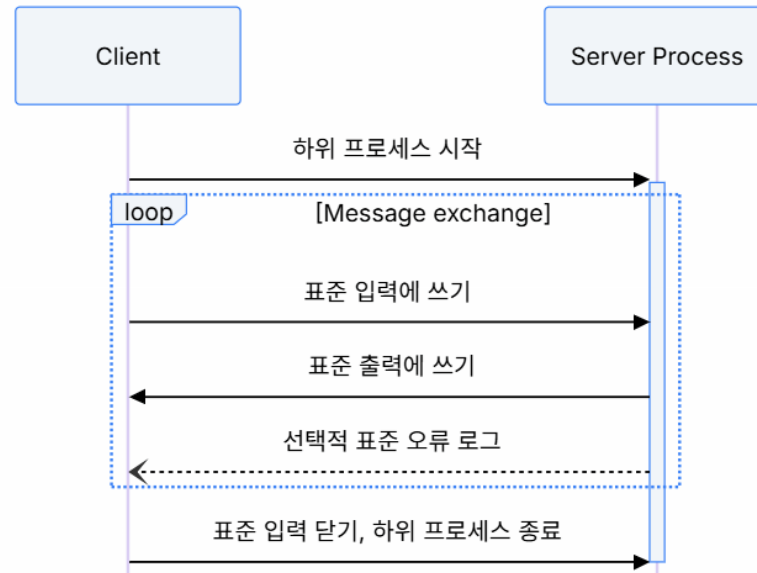
```
{
  "jsonrpc": "2.0",
  "id": "string | number",
  "result?": {
    "[key: string]": "unknown"
  },
  "error?": {
    "code": "number",
    "message": "string",
    "data?": "unknown"
  }
}
```

알림 메시지

```
{
  "jsonrpc": "2.0",
  "method": "string",
  "params?": {
    "[key: string]": "unknown"
  }
}
```

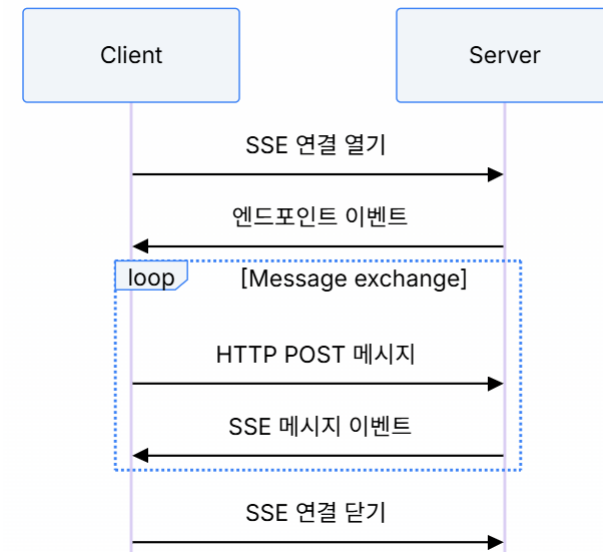
Stdio 전송 메커니즘

클라이언트는 stdio를 가능한 한 지원해야 함



SSE(Server-Sent Events) 전송 메커니즘

서버가 실시간 업데이트를 클라이언트에 전송할 수 있는 HTTP 서버 푸시 기술



서버는 2 개의 엔드포인트를 제공해야 합니다.

- SSE 엔드포인트 - 연결을 설정하고 서버에서 메시지를 수신하는 데 사용
- HTTP POST 엔드포인트 - 클라이언트에서 서버로 메시지를 보내는 데 사용

MCP 장단점

■ 장점

- 데이터 연결 단순화 : AI 애플리케이션과 데이터 소스 간의 연결을 단순화
- 향상된 데이터 상호작용 보안 : 표준화된 프로토콜을 통해 데이터 흐름을 더 잘 제어하고 데이터 유출을 방지
- 향상된 AI 애플리케이션 기능 : AI 애플리케이션의 기능 범위를 확장하고 더 많은 비즈니스 시나리오에서 역할을 수행할 수 있게 함
- 우수한 확장성 : MCP 프로토콜은 Prompts, Tools, Sampling과 같은 기능을 제공하여 우수한 확장성을 갖추고 있음
- 다양한 데이터 형식 지원 : 파일 내용, 데이터베이스 레코드, API 응답, 실시간 시스템 데이터, 스크린샷 및 이미지, 로그 파일 등 다양한 데이터 형식을 지원

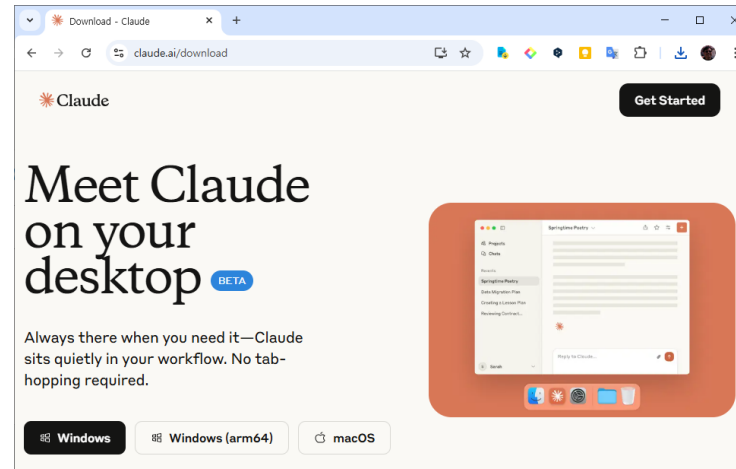
■ 단점

- 높은 산업 표준 경쟁 압력
- 잠재적 호환성 문제
- 프로토콜 본질에 대한 의문
- 현재 제한된 응용 범위 : 여러 기기나 분산 환경에서 데이터 상호작용이 필요한 일부 응용 시나리오의 경우, 현재 MCP 프로토콜이 요구 사항을 충족시키지 못할 수 있음

MCP 실습

<https://www.claudemcp.com/ko/docs/quickstart>

- Claude 데스크톱 앱 설치
<https://claude.ai/download>



- Claude 데스크톱의 설정 파일 열기 : 설정 → 개발자 → 설정 편집
- `claude_desktop_config.json` 에 서버 정보 추가

```
{
  "mcpServers": {
    "sqlite": {
      "command": "uvx",
      "args": ["mcp-server-sqlite", "--db-path", "/Users/Administrator/test.db"]
    }
  }
}
```

MCP 실습

<https://www.claudemcp.com/ko/docs/quickstart>



- DB Browser for SQLite 설치
<https://sqlitebrowser.org/dl/>
- test DB 생성
- products 테이블 생성

```
DB Browser for SQLite - C:\Program Files\DB Browser for SQLite\test.db
파일(F) 편집(E) 보기(V) 도구(T) 도움말(H)
새 데이터베이스(N) 데이터베이스 열기(O) 변경 사항 저장하기
데이터베이스 구조 데이터 보기 Pragma 수정 SQL 실행
SQL 1
1 CREATE TABLE products (
2   id INTEGER PRIMARY KEY,
3   name TEXT,
4   price REAL
5 );
6
```

- products 테이블에 데이터 추가

```
INSERT INTO products (name, price) VALUES
('Widget', 19.99),
('Gadget', 29.99),
('Gizmo', 39.99),
('Smart Watch', 199.99),
('Wireless Earbuds', 89.99),
('Portable Charger', 24.99),
('Bluetooth Speaker', 79.99),
('Phone Stand', 15.99),
('Laptop Sleeve', 34.99),
('Mini Drone', 299.99),
('LED Desk Lamp', 45.99),
('Keyboard', 129.99),
('Mouse Pad', 12.99),
('USB Hub', 49.99),
('Webcam', 69.99),
('Screen Protector', 9.99),
('Travel Adapter', 27.99),
('Gaming Headset', 159.99),
('Fitness Tracker', 119.99),
('Portable SSD', 179.99);
```

MCP 실습

■ claude_desktop_config.json 에 서버 정보 추가

```
{
  "mcpServers": {
    "filesystem": {
      "command": "npx",
      "args": [
        "-y",
        "@modelcontextprotocol/server-filesystem",
        "/Users/사용자명/Desktop",
        "/Users/사용자명/aiagent"
      ]
    },
    "google-maps": {
      "command": "npx",
      "args": [
        "-y",
        "@modelcontextprotocol/server-google-maps"
      ],
      "env": {
        "GOOGLE_MAPS_API_KEY": "YOUR_GOOGLE_MAPS_API_KEY"
      }
    },
    "tavily-mcp": {
      "command": "npx",
      "args": ["-y", "tavily-mcp@0.1.2"],
      "env": {
        "TAVILY_API_KEY": "YOUR_TAVILY_API_KEY"
      }
    },
    "sequential-thinking": {
      "command": "npx",
      "args": [
        "-y",
        "@modelcontextprotocol/server-sequential-thinking"
      ]
    },
    "mcp-youtube": {
      "command": "npx",
      "args": [
        "@anaisbetts/mcp-youtube"
      ]
    }
  }
}
```

DP 분당 수내중학교에서 분당선 수내역까지 어떻게 가?

분당 수내중학교에서 분당선 수내역까지 가는 방법을 알려드리겠습니다. 정확한 경로를 확인하기 위해 지도 검색을 해보겠습니다.

google-maps의 maps_directions에서 결과 보기(로컬) >

주소를 더 정확하게 검색해 보겠습니다.

google-maps의 maps_geocode에서 결과 보기(로컬) >

google-maps의 maps_geocode에서 결과 보기(로컬) >

google-maps의 maps_directions에서 결과 보기(로컬) >

정확한 위치 정보를 검색해 보겠습니다.

google-maps의 maps_search_places에서 결과 보기(로컬) >

google-maps의 maps_search_places에서 결과 보기(로컬) >

google-maps의 maps_directions에서 결과 보기(로컬) >

정확한 좌표를 사용해 검색해 보겠습니다.

google-maps의 maps_directions에서 결과 보기(로컬) >

다른 방식으로 검색해보겠습니다. 대중교통 옵션으로 시도해보겠습니다.

google-maps의 maps_directions에서 결과 보기(로컬) >

수내중학교에서 분당선 수내역까지 가는 방법을 알려드립니다:

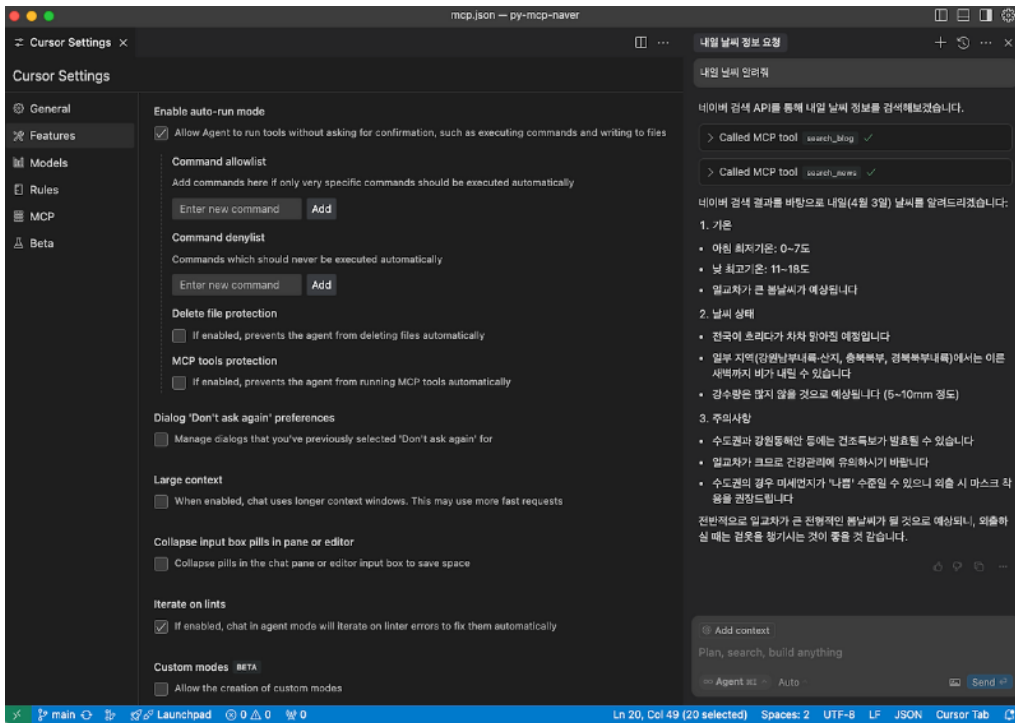
총 거리는 약 1.6km이며, 소요 시간은 약 25분입니다.

1. 수내중학교에서 출발하여 푸른마을관리사무소 방향으로 도보로 이동합니다 (약 3분, 0.2km)
2. 롯데백화점 방향 버스를 탑승합니다 (약 7분, 1.4km)
3. 하차 후 수내역까지 도보로 이동합니다 (약 1분, 42m)

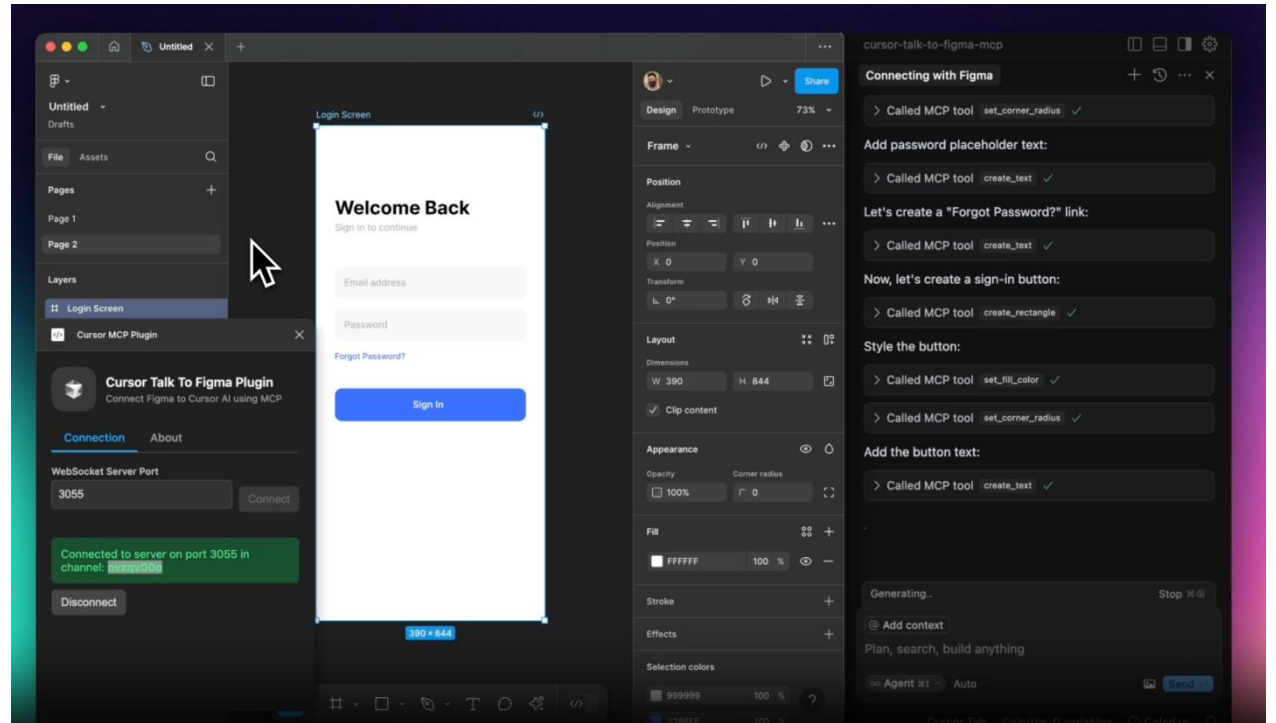
만약 버스를 이용하지 않고 도보로만 이동하고 싶으시다면, 약 1.6km 거리이므로 약 20-25분 정도 소요될 것으로 예상됩니다. 수내중학교에서 분당선 수내역 방향으로 걸어가시면 됩니다.

MCP 실습

- Cursor 에서 MCP 사용하기
<https://dytis.tistory.com/114>



- Figma 에서 MCP 사용하기
<https://maly.so/design.pick/posts/d5rylkv0o1w>
<https://apidog.com/kr/blog/figma-mcp/>



MCP 서버 구축

<https://www.devshorts.in/p/how-to-build-your-own-mcp-server>

1. 필수 조건:

a) mcp 패키지 설치:

서버 구축을 위한 MCP 패키지를 설치하는 방법은 두 가지가 있습니다.

pip 사용:

```
pip install mcp
```

uv를 사용하는 것이 좋습니다. (더 빠른 Rust 기반 Python 패키지 관리자):

```
uv add "mcp[cli]"
```

b) Gmail 및 캘린더 API 액세스를 위해 OAuth 2.0 설정:

- OAuth 2.0 구성 가이드를 따르고 Gmail 및 캘린더의 API 범위에 대한 OAuth 동의를 설정합니다.
- 다운로드 `credentials.json`.
- 처음 실행하면 코드는 `token.pickle` 인증된 세션을 저장하기 위해 생성됩니다.

2. Gmail, GCalendar MCP 서버 구축

Python SDK를 사용하여 Gmail + Google 캘린더 MCP 서버를 구축해 보겠습니다.

a) mcp 패키지 가져오기

```
from mcp.server.fastmcp import FastMCP
mcp = FastMCP("gmail_mcp_server")
```

b) server.py에서 Gmail 함수 정의

Gmail의 세 가지 기능을 구현할 예정입니다.

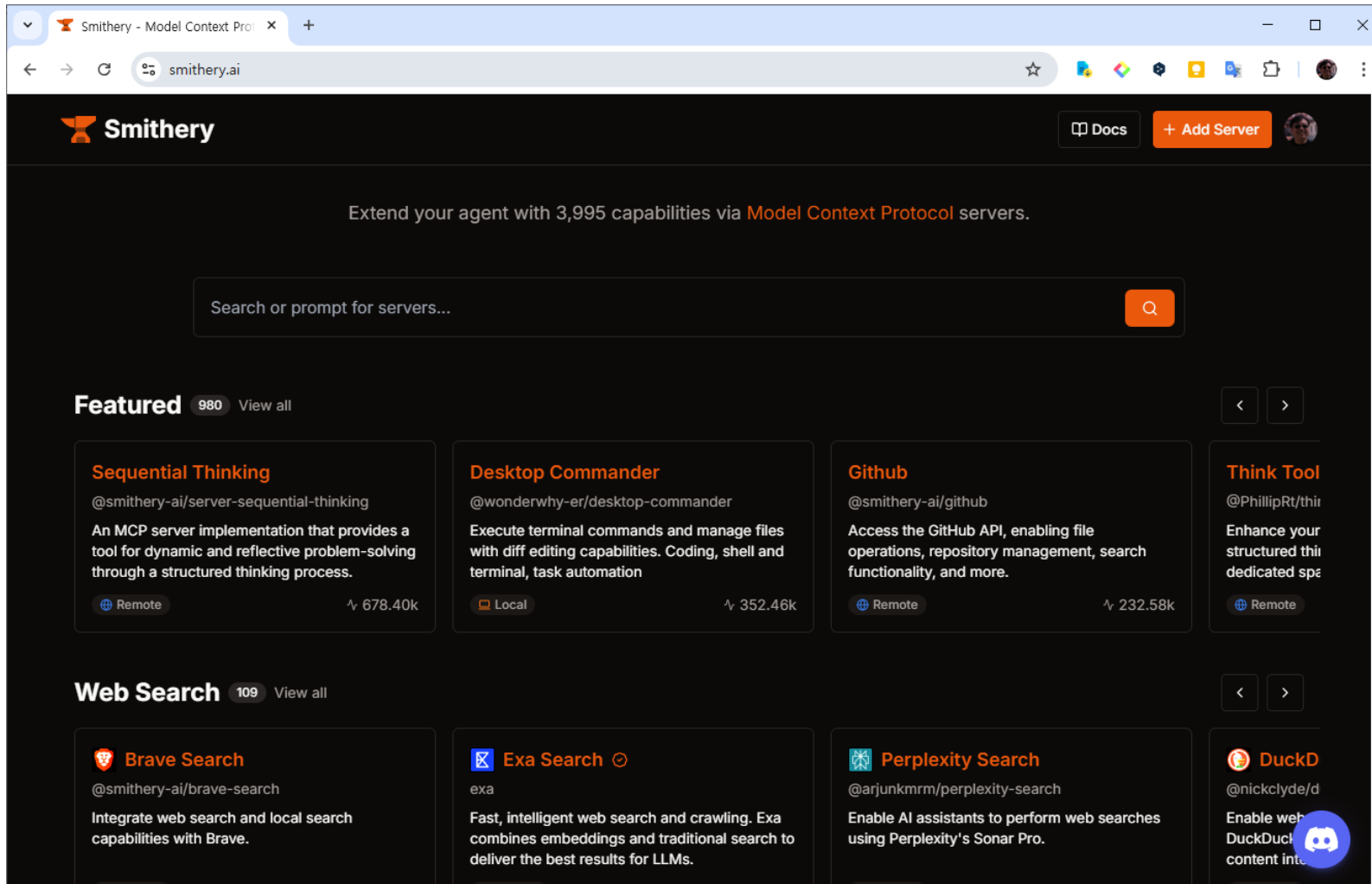
- 최신 이메일 받기
- 특정 이메일 검색
- 이메일 내용 받기

```
@mcp.resource("gmail://inbox")
def get_emails() -> str:
    """Get recent emails from inbox"""
    try:
        service = get_gmail_service()
        results = service.users().messages().list(
```

MCP 서버 Registry

<https://smithery.ai/>

Smithery는 MCP 서버의 중앙 HUB 역할을 하며, MCP 서버를 쉽게 검색하고 활용할 수 있도록 지원합니다



1 MCP 서버 검색 및 관리

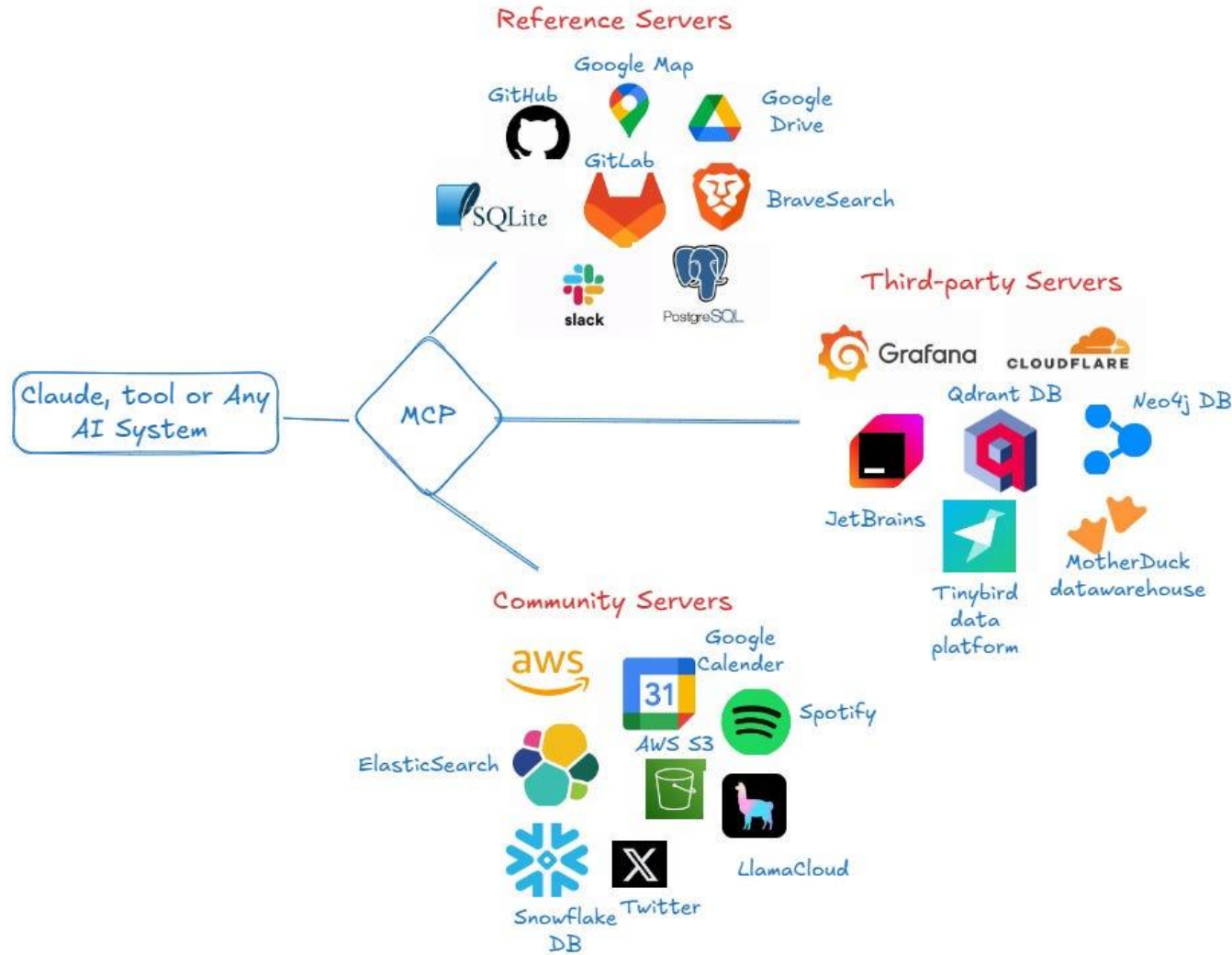
2 MCP 서버 호스팅 및 배포

3 표준화된 인터페이스 제공

MCP 서버 목록

<https://github.com/modelcontextprotocol/servers>

사전 구축된 MCP 서버는 GitHub, Cloudflare, Slack과 같은 주요 플랫폼에서 사용할 수 있습니다.



MCP Python SDK

Python implementation of the Model Context Protocol (MCP)

`pypi` `v1.6.0` `license` `MIT` `python` `3.10 | 3.11 | 3.12 | 3.13` `docs` `modelcontextprotocol.io`
`spec` `spec.modelcontextprotocol.io` `discussions` `0 total`

Table of Contents

- [MCP Python SDK](#)
 - [Overview](#)
 - [Installation](#)
 - [Adding MCP to your python project](#)
 - [Running the standalone MCP development tools](#)
 - [Quickstart](#)
 - [What is MCP?](#)
 - [Core Concepts](#)
 - [Server](#)
 - [Resources](#)
 - [Tools](#)
 - [Prompts](#)

Thank you 😊