

MCP를 이용해 QGIS를 조종하는 방법

시연 동영상: <https://www.youtube.com/embed/dEWOilwgsdo?si=kmJWXJRV6HNDubJt>

구조



{style="border: 0px"}

MCP를 이용해 QGIS를 조종하기 위한 2가지 요소

- LLM에서 MCP 사용하기
- QGIS의 plugin

LLM에서 MCP 사용하기

LLM(Large Language Model)은 chatGPT와 같이 인간의 언어를 이용해 명령을 내리고 답을 들을 수 있는 AI 모델을 말한다. 그런데 우리가 chatGPT에 어떤 변형을 줄 수는 없기 때문에 내 컴퓨터에 작은 LLM을 다운로드 받아 MCP와 연결해 사용해야 한다.

작은 LLM

chatGPT는 무지막지하게 큰 LLM이라 개인이 다운로드 받아서 사용할 수도 없고, chatGPT를 서비스하고 있는 openAI사가 허락하지도 않는다. 그런데 아주 작은, 경량의 LLM을 구글, openAI, meta 등등이 오픈소스로, 무료로 올려 놓은 것들이 있어 그걸 다운로드 받아 사용할 수 있다.

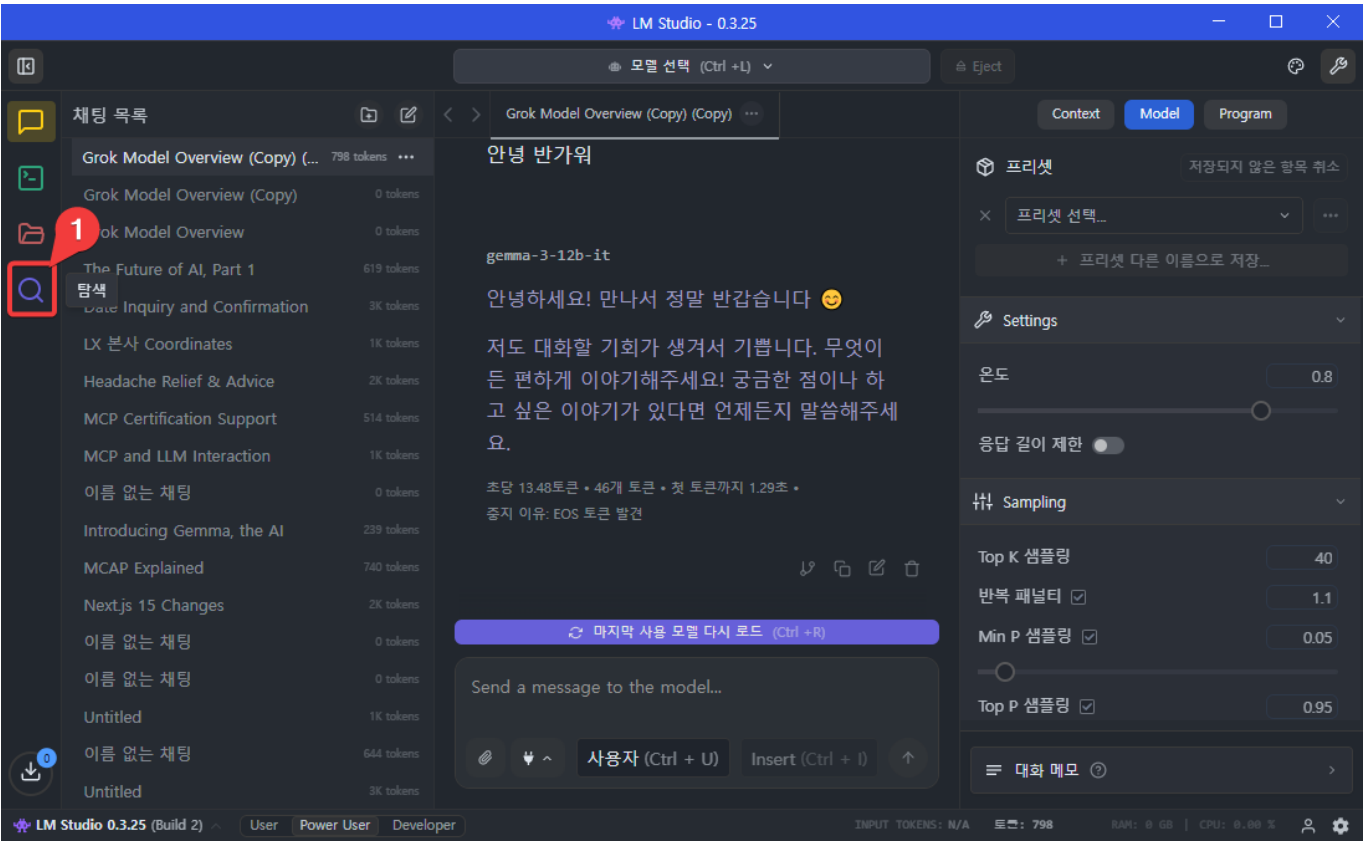
* 작은 LLM을 SLM(Small Language Model)이라 부르기도 한다.

LM Studio

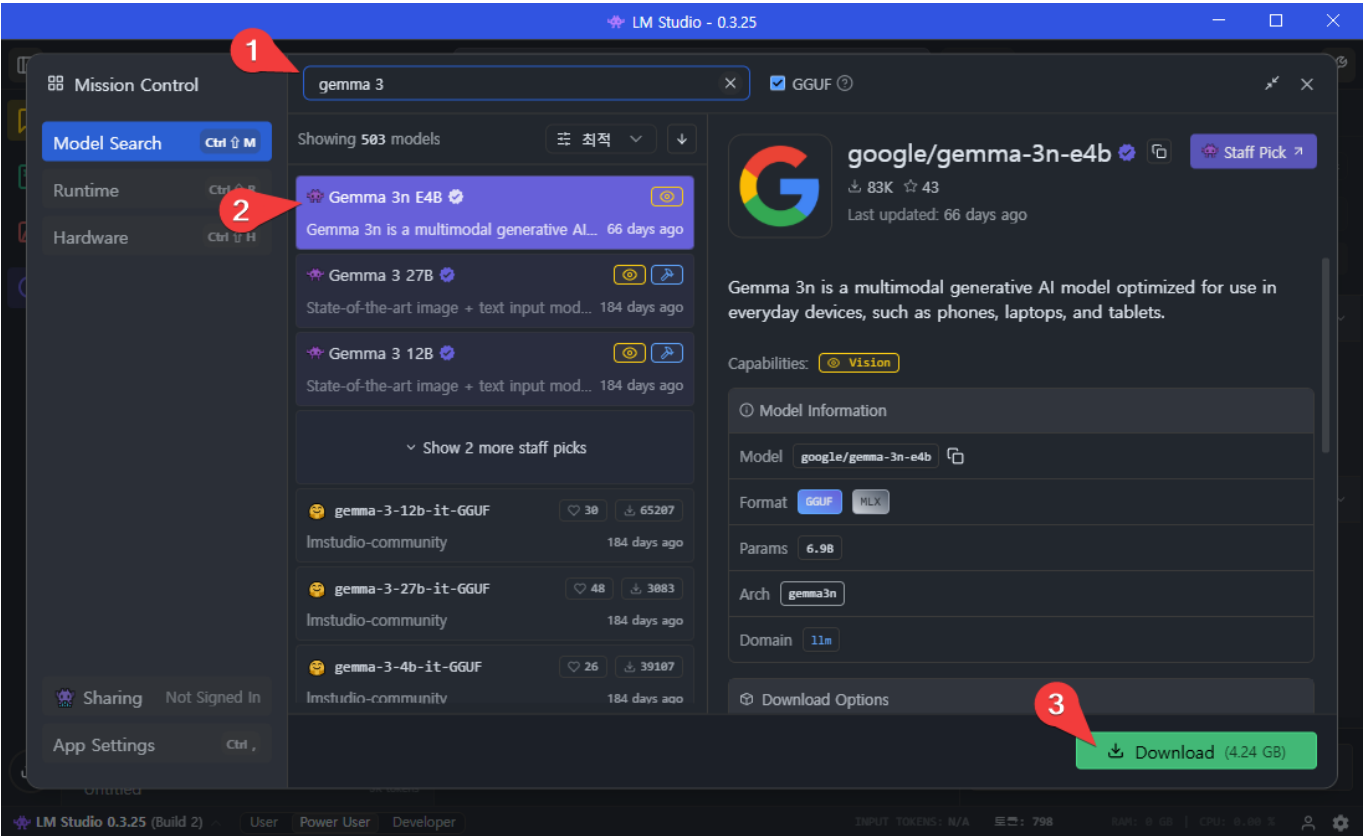
위에서 말한 작은 LLM을 다운로드 받아 chatGPT처럼 쓰기 위해 사용하는 프로그램이 **LM Studio**다.

사용법을 일일이 설명하기엔 너무 많아 큰 흐름만 설명하겠다.

LM Studio를 다운로드받아 설치한 후,

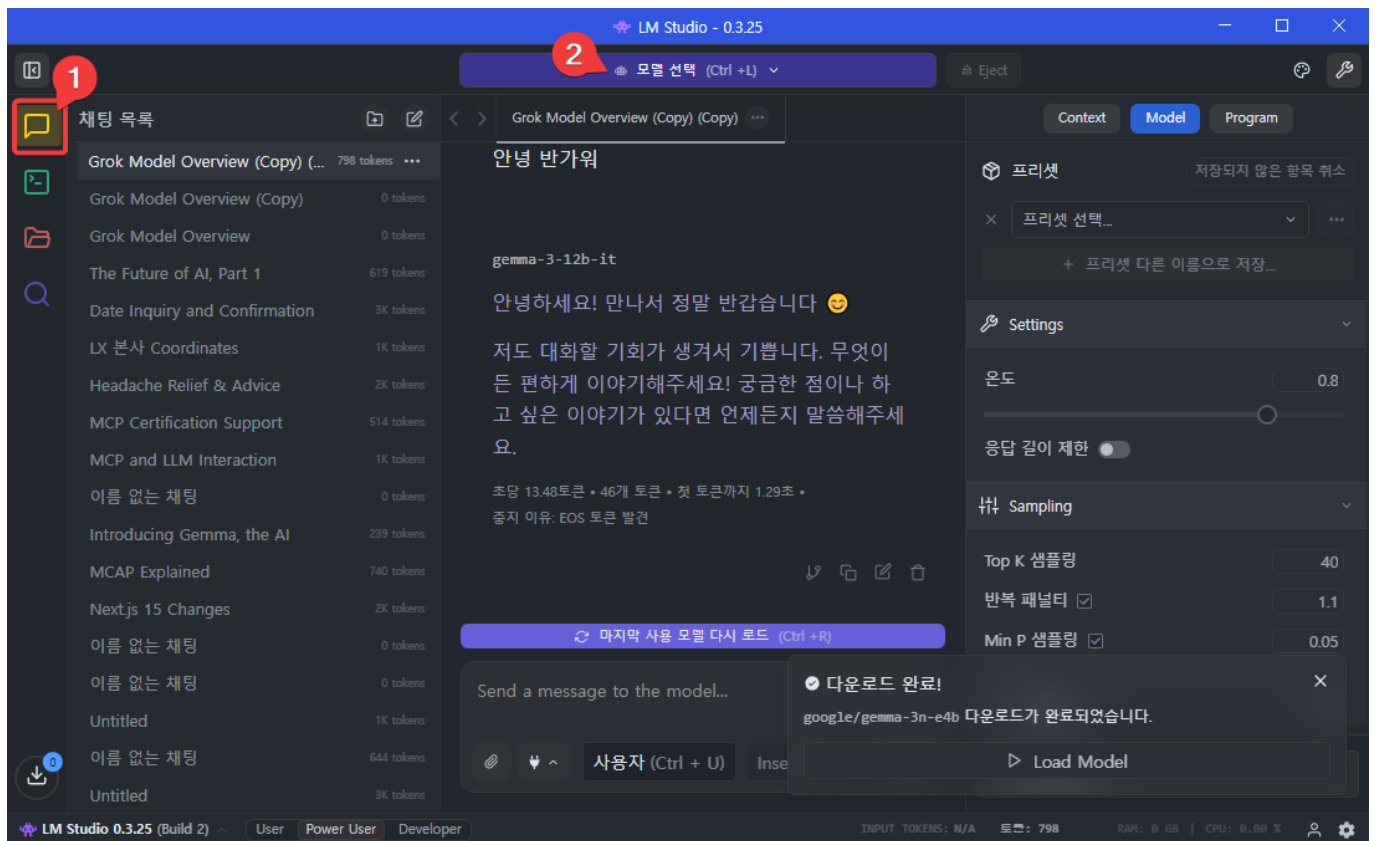


1. 탐색 버튼을 누르고,

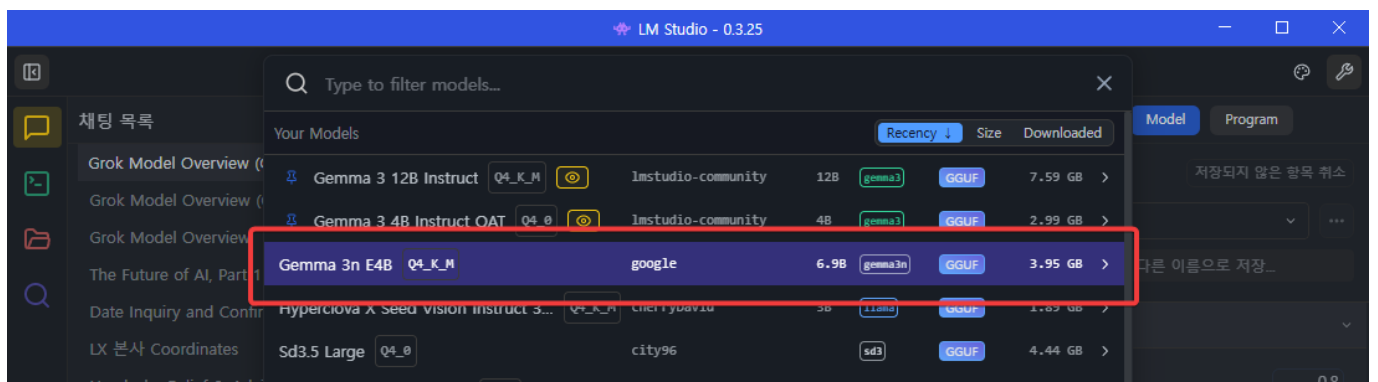


2. "gemma 3"로 검색,
3. "Gemma 3n E4B" 선택
4. [Download] 버튼 클릭

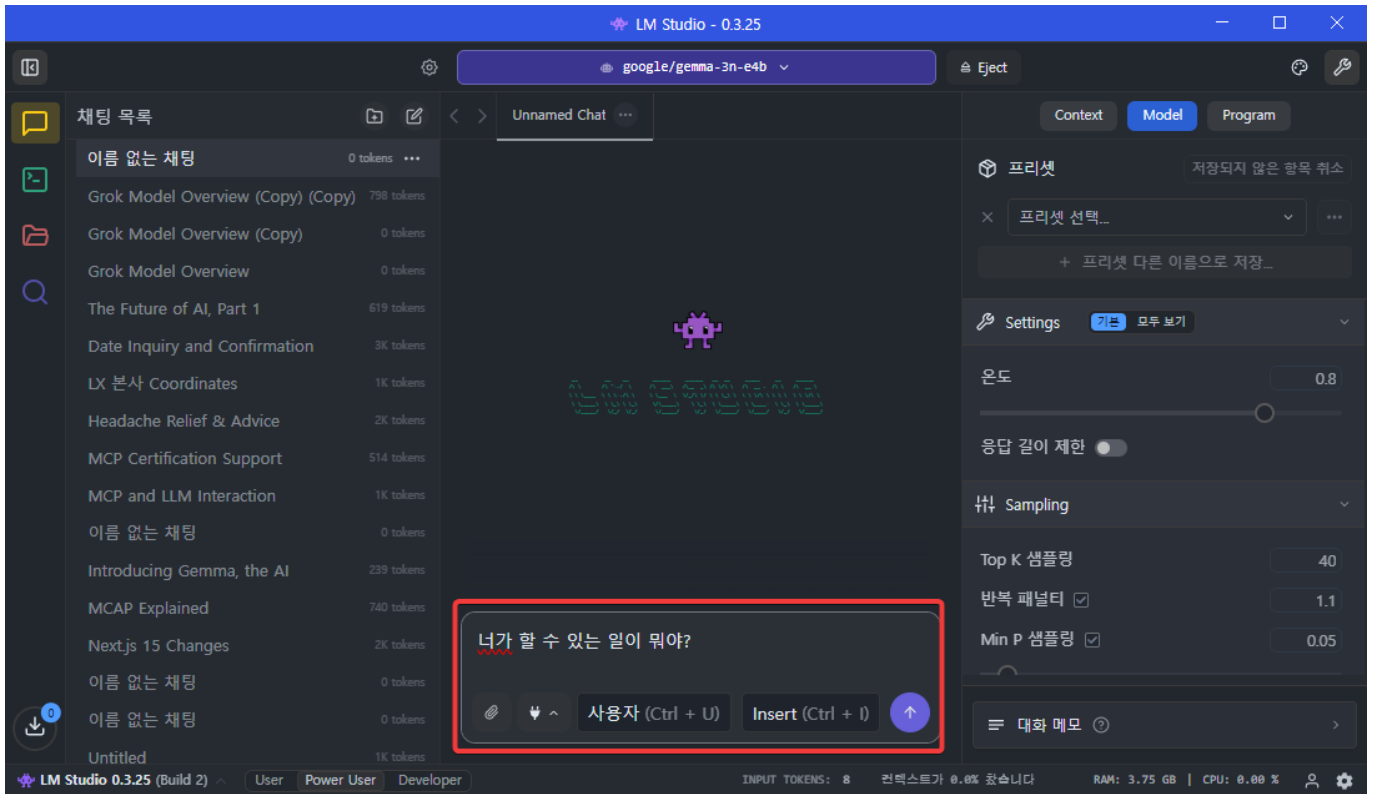
다운로드가 완료되었다는 메시지가 나오면 메인 창을 남기고 모두 닫은 다음,



아래와 같이 "Gemma 3n E4B"를 선택해 Gemma 3n E4B 모델을 읽어들인다.



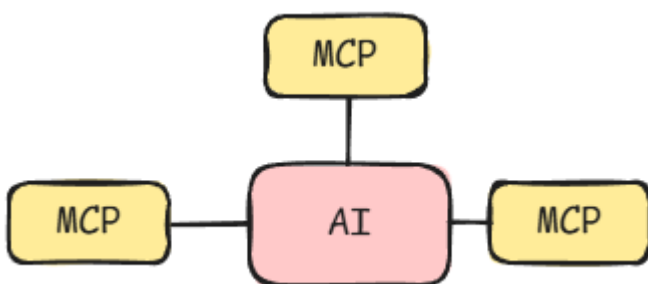
이제 LLM은 준비되었다.



표시한 부분에 chatGPT와 대화하는 것처럼 대화를 입력하고 대화할 수 있다.

MCP

MCP(Model Context Protocol)는 AI 모델을 다른 프로그램 혹은 데이터와 연결해주는 기술이다. AI의 플러그인처럼 동작하며, LLM에 붙여 놓으면 인간의 언어로 다른 프로그램을 조종하거나 데이터에서 정보를 가져올 수 있다.



{style="border: 0px"}

사용자가 LLM을 통해 "QGIS에서 D:\test.shp 파일을 읽어들여"라고 하면, LLM 입장에서는 QGIS에게 저런 작업하도록 시키는 걸 직접 할 수가 없기 때문에 저런 기능을 해 주는 MCP가 있는지 찾아보고 그 MCP를 통해 shp 파일 읽어들이는 작업을 실행할 수 있다.

MCP가 QGIS plugin에게 명령 내리는 방법

그럼 MCP는 어떻게 QGIS에서 shp 파일을 읽어들이도록 할 수 있을까?



클라이언트

서버

{style="border: 0px"}

우리는 QGIS 플러그인을 서버로 만들거다. QGIS 플러그인이지만 서버처럼 동작하도록 만들거라는 뜻이다.

서버는 클라이언트가 오기를 기다리다가 클라이언트가 접속, 뭔가 요청을 하면 바로 처리해 주는 역할을 한다.

QGIS 플러그인은 파이썬으로 만들기 때문에 매우 쉽게 서버처럼 동작하도록 만들 수 있다. 그냥 파이썬으로 서버를 만드는데 겹데기만 QGIS 플러그인 양식으로 만든다고 생각하자.

QGIS 플러그인에게 접속해서 뭔가 요청을 하는 클라이언트가 바로 MCP이다.

전반적으로 다음 순서로 처리된다.

1. 내가 LLM에게 일 시킨다. (QGIS에 "D:\test\sample.shp 파일 읽어들여")
2. LLM이 MCP들 중에 이 일을 처리할 만한 MCP가 있는지 찾아낸다. (사용자의 명령어에서 유추한다)
3. MCP가 QGIS 플러그인(서버라서 클라이언트 접속이 오기를 기다리고 있다)에게 접속한다.
4. 서버와 클라이언트가 접속된 상태이니 이제 정해진 약속대로 명령을 내린다(통신한다)
5. QGIS 플러그인이 shp 파일을 읽어들인다.

MCP와 QGIS 플러그인이 접속, 통신 과정을 거친다 해도 MCP가 QGIS에 아무 일이나 막 시킬 수 있는 건 아니다. QGIS 플러그인이 정해 놓은 기능만(노출했다고 표현한다) MCP가 실행한다. "shp 파일 읽어들이는 기능 실행해"와 같은 식이다.

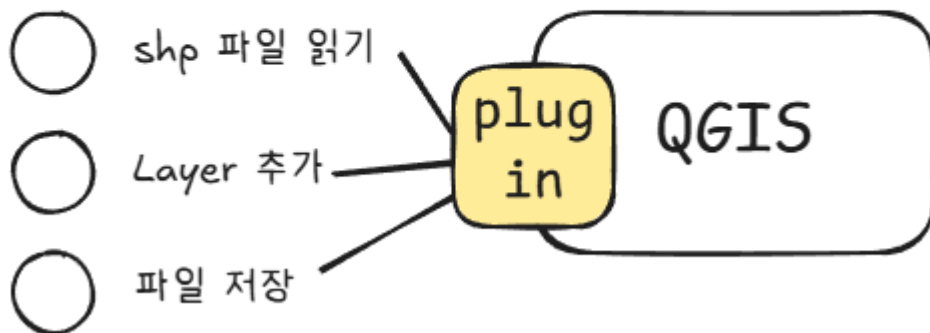
QGIS의 plugin

우리가 만들 QGIS 플러그인은 서버이다. (모든 QGIS 플러그인이 서버인 것은 아니다)

이 말은 MCP 뿐 아니라 그 누구라도 우리가 만들 QGIS 플러그인이 노출한 기능 목록과 사용법을 알고 있다면 이 플러그인에 접속해서 일을 시킬 수 있다는 뜻이다.

다시 말하면, 이 플러그인을 사용할 수 있는 클라이언트는 MCP의 형태로만 만들 수 있는 게 아니라 어떤 형태로도 만들 수 있다는 뜻이다. 파이썬으로 클라이언트 프로그램을 만들 수도 있고, 웹 방식으로 클라이언트 프로그램을 만들 수도 있다.

우리가 만들 플러그인이 노출할 기능들



{style="border: 0px"}

이제 이 서버에 접속한 클라이언트들은 서버가 노출한(공개적으로 제공하는) 기능들을 실행할 수 있다. 물론 서버가 노출할 기능들은 이 외에도 얼마든지 더 만들어 추가할 수 있다.

이 플러그인은 서버를 만드는 기술, 받은 요청을 QGIS에서 실행해 내는 기술 등을 필요로 한다.

StreamLit

이 부분은 삭제된 부분임. 이후 과정은 StreamLit 없이 사용하는 방법으로 소개하겠음. (그래도 LM Studio에 대해 좀 더 알고 싶은 분들을 위해 남겨 놓겠음)

LM Studio는 서버로 동작하는 기능을 제공한다.

LM Studio의 기본 동작은, 우리가 대화창에 입력하면 AI가 그에 대한 답을 바로 보여주는건데, 이 동작을 네트워크 연결을 통해 할 수 있도록 해주는 거다.

PC에서 LM Studio를 실행해 LLM을 사용 가능하게 해 놓고 서버 기능을 켜 놓은 다음, 다른 PC에서 접속해 그 LLM을 사용할 수 있다.

다른 PC 뿐 아니라 같은 PC 내에서도 다른 프로그램들이 접속해 사용할 수 있다.

LM Studio가 서버로 동작하는 기능을 제공하기 때문에 정말 많은 응용 프로그램을 만들어 사용할 수 있다. 예를 들면 엑셀에서도 LM Studio가 제공하는 서버 기능에 접속해 질문을 보내고 LLM이 만들어 준 답을 받아올 수 있다.

StreamLit은 이 서버 기능을 이용해 LLM과 연동하는 웹페이지를 쉽게 만들 수 있는 기술이고, LM Studio와 연결해 MCP 기능을 쉽게 사용하도록 만들 수 있다.

몇 달 전만 해도 LLM에서 MCP를 사용하기 위해 이 StreamLit을 이용했는데, 최근 2025. 6월 말쯤에 LM Studio가 업데이트되면서 자체적으로 MCP를 지원하기 시작해서 StreamLit 없이 LM Studio에서 바로 MCP를 사용할 수 있게 되었다. 그래서 이후 과정은 StreamLit 없이 사용하는 방법으로 안내하겠음.

소스코드 적용

필요한 코드는 다음과 같다.

- MCP를 만들기 위한 코드
- QGIS 플러그인을 만들기 위한 코드

그리고 간단한 몇몇 설정을 해줘야 한다.

- LM Studio에 우리의 MCP를 등록하기 위한 설정
- QGIS에 우리의 플러그인을 등록하기 위한 설정

MCP 소스코드

다운로드:

우리가 만들 MCP는 내부적으로 2개로 나뉘어 있다.

1. MCP 그 자체
2. QGIS와 통신하고 명령을 전달하는 클래스

MCP는 FastMCP 라이브러리를 이용해 만들어졌다. 먼저 FastMCP를 설치해야 한다.

```
pip install FastMCP
```

이 아래에 언급한 파일은 모두 같은 폴더에 저장되어야 한다. 여기서는 d:\test\mcp_qgis 폴더를 사용했다.

아래 두 개의 파일을 mcp_qgis.zip 파일로 압축했다.

파일 다운로드: [mcp_qgis.zip](#)

server.py

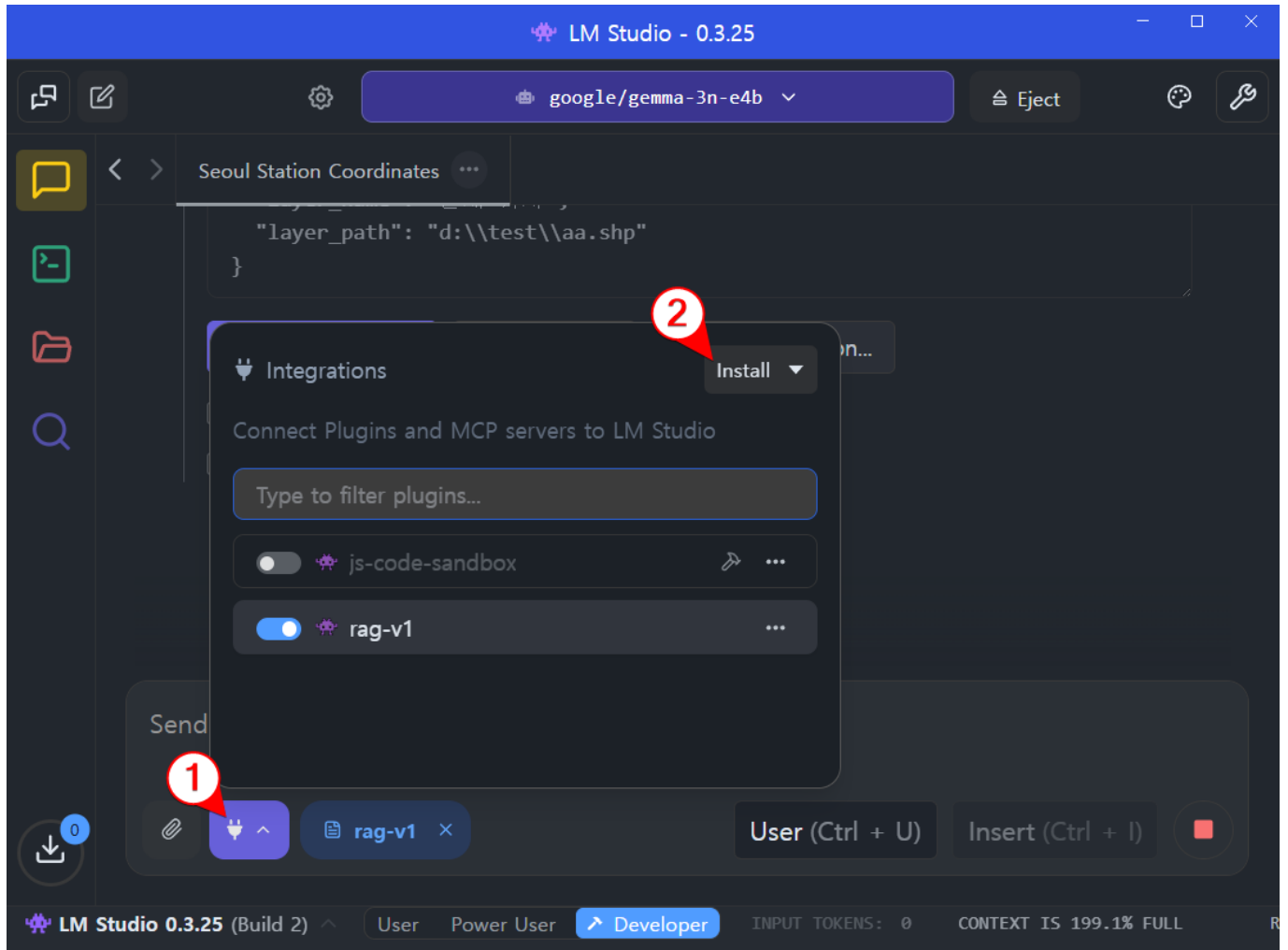
MCP 서버. AI에서 넘어온 정보를 받아서 위에서 활성화한 qgis 플러그인에 명령을 전달하는 역할. qgis_client.py 를 이용한다.

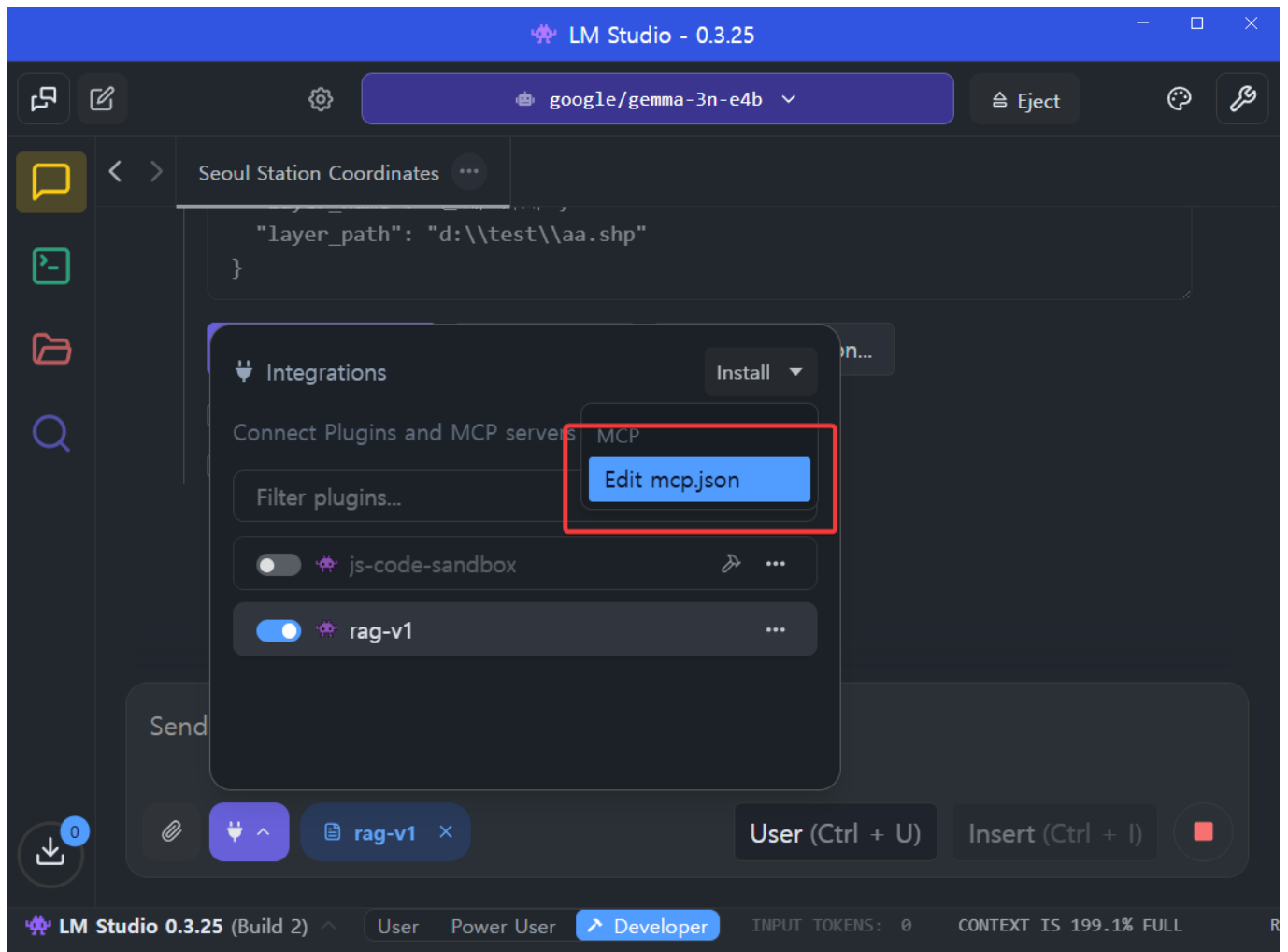
qgis_client.py

qgis 플러그인에 명령을 쉽게 보내도록 준비된 클래스

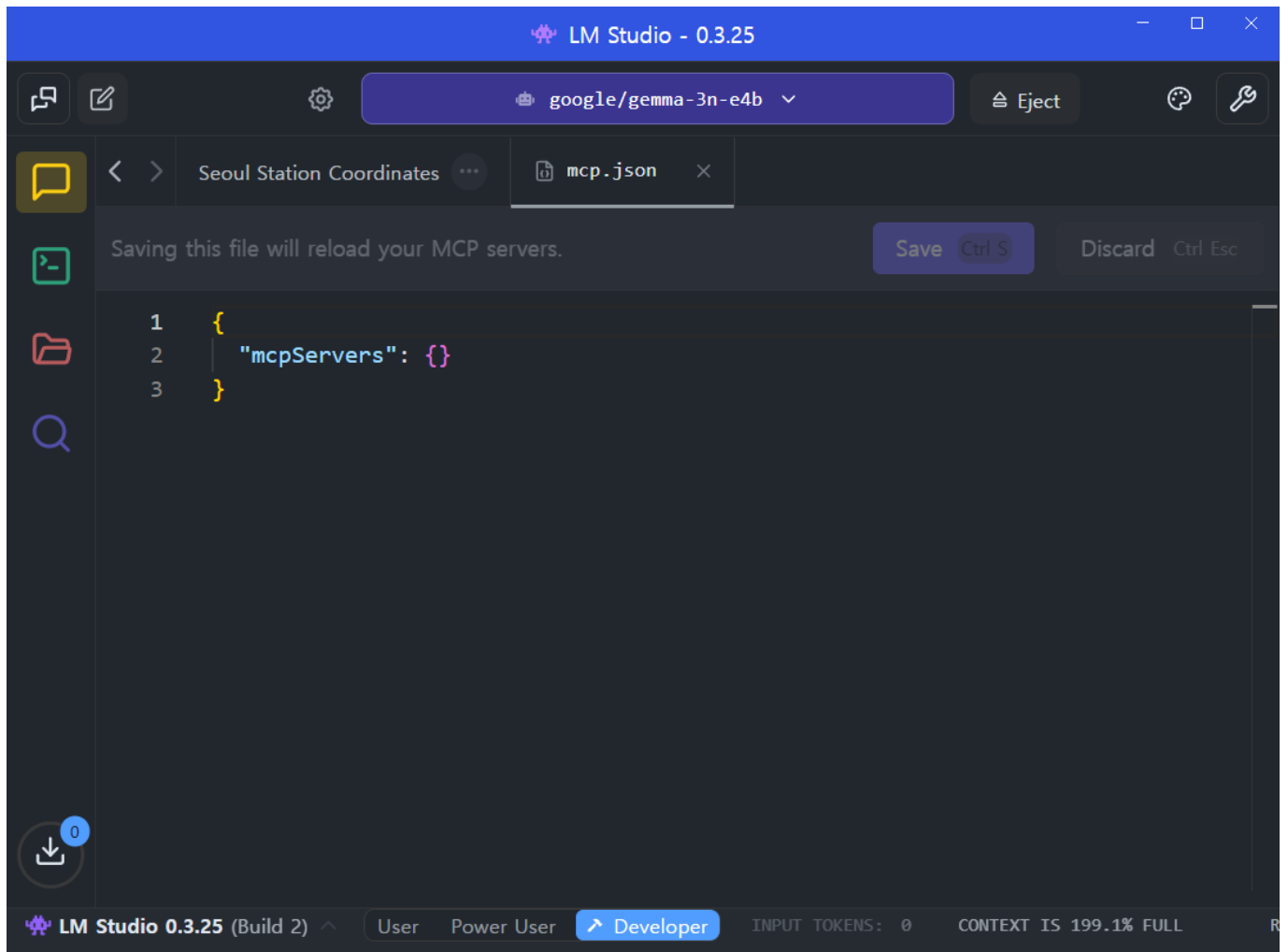
설정

LM Studio에서 이렇게 만든 QGIS용 MCP를 사용하려면 등록하는 설정을 해야 한다.





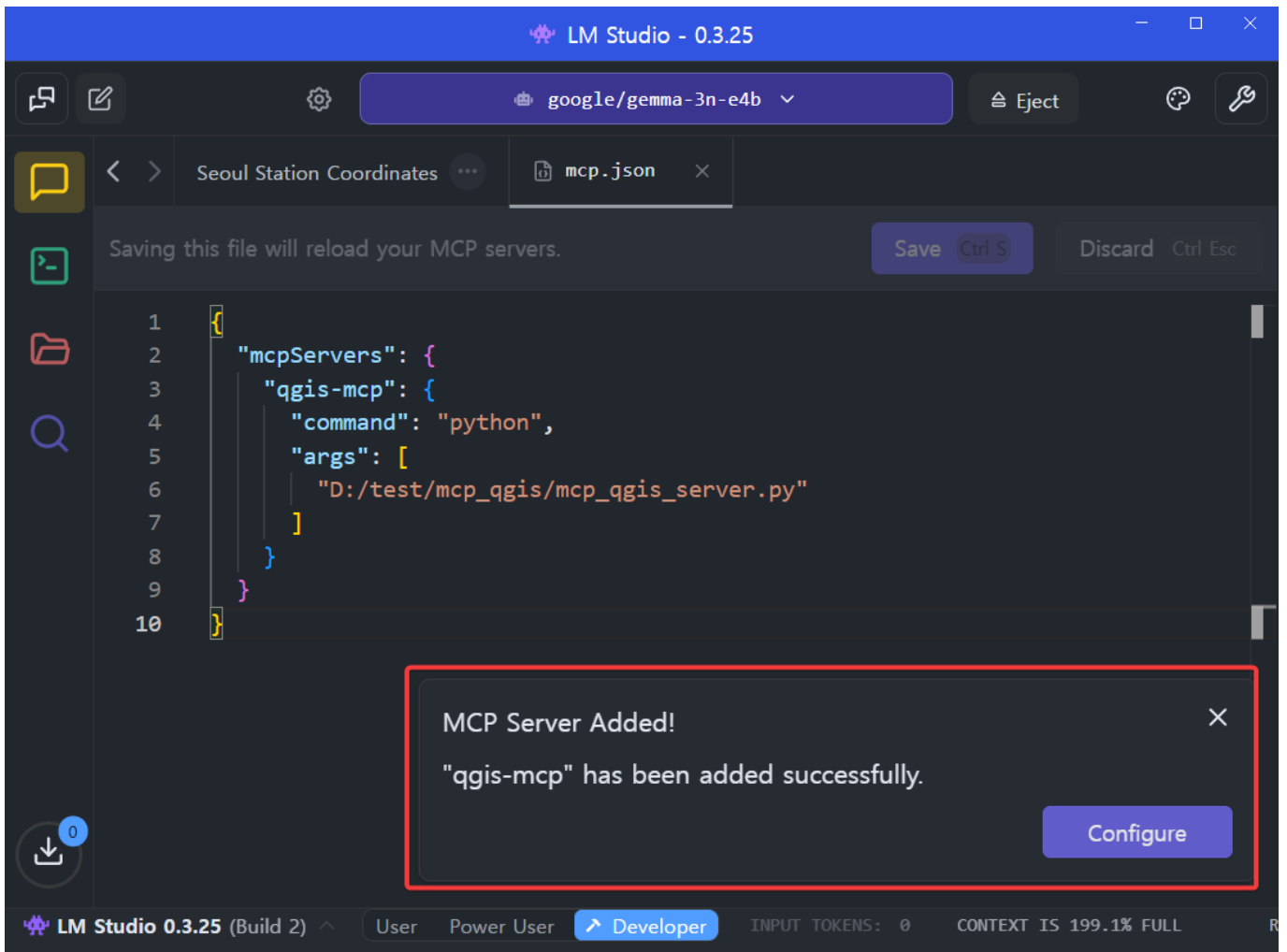
[Edit mcp.json] 버튼을 누르면 mcp.json 파일을 설정할 수 있도록 한다.



위와 같이 MCP 설정 창이 나올텐데 이 부분을 지우고 아래의 코드로 바꾼다.

"D:/test/mcp_qgis/mcp_qgis_server.py" 이 부분은 파일이 저장된 폴더에 맞게 바꾼다.

```
{  
  "mcpServers": {  
    "qgis-mcp": {  
      "command": "python",  
      "args": [  
        "D:/test/mcp_qgis/mcp_qgis_server.py"  
      ]  
    }  
  }  
}
```



이런 메시지가 나오면 MCP 서버 등록에 성공한거다.

qgis용 플러그인 소스코드

다음 위치에서 파일 3개를 모두 다운로드 받는다.

https://github.com/jjsantos01/qgis_mcp/tree/main/qgis_mcp_plugin

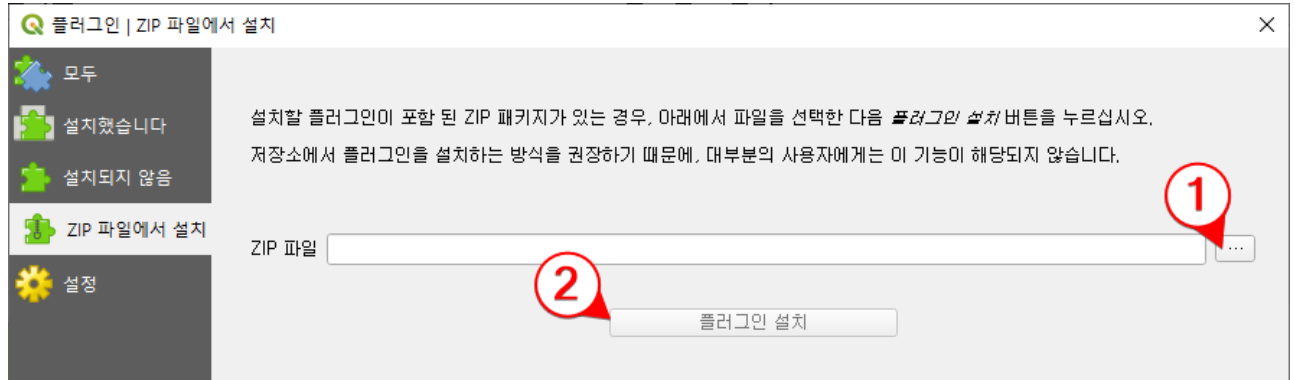
- **init.py**
- metadata.txt
- qgis_mcp_plugin.py

이렇게 다운로드 받은 3개 파일을 Qgis plugin으로 설치한다.

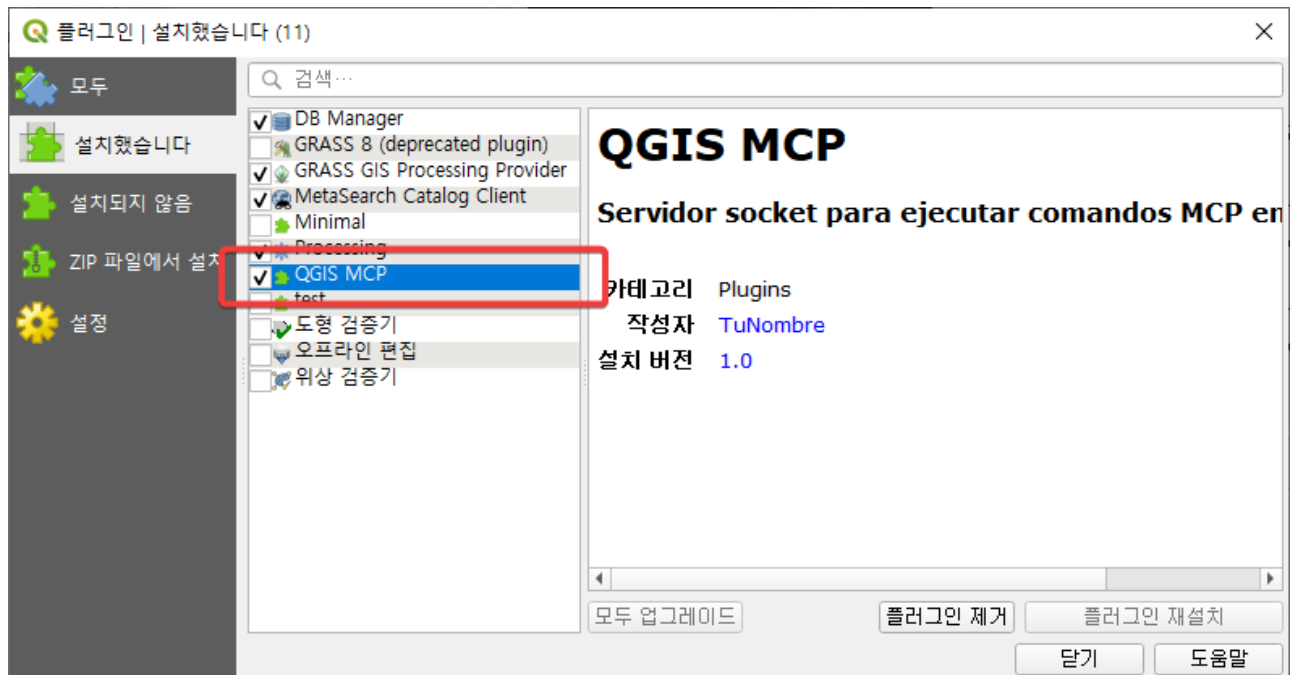
설치

1. QGIS를 종료한다.

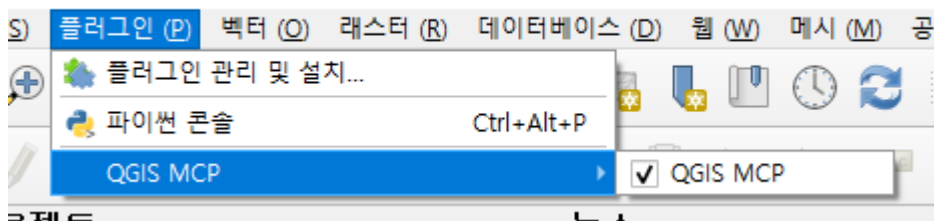
- 다운로드 받은 3개의 파일을 폴더째 압축한다. (파일 3개 선택한 다음 오른쪽 클릭해서 압축하면 QGIS에서 플러그인으로 설치 안됨. 반드시 3개의 파일이 들어있는 폴더를 압축해서 zip 파일을 만들어야 한다.)
- QGIS를 실행한다.
- QGIS에서 **플러그인 > 플러그인 관리 및 설치** 메뉴를 실행한다.
- Zip 파일에서 설치 메뉴를 실행해 위 2에서 만든 zip 파일을 읽어들이고 설치한다.



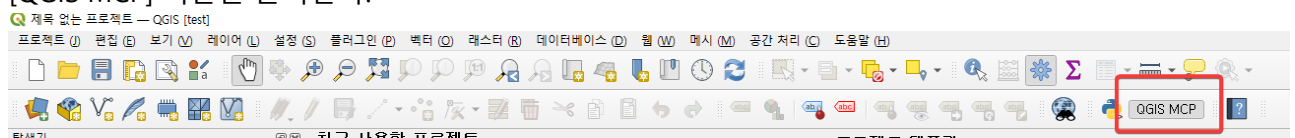
- QGIS_MCP 플러그인이 체크표시되어 있는지 확인한다.



- QGIS_MCP 플러그인을 활성화한다.



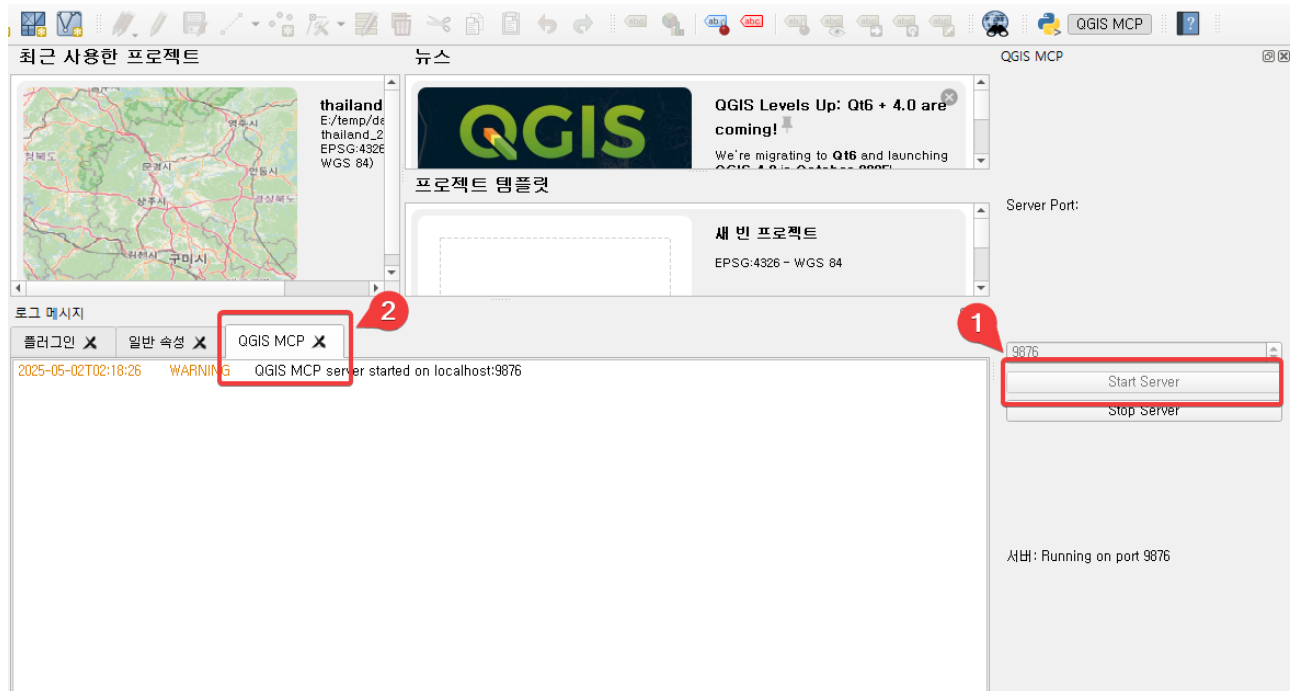
- [QGIS MCP] 버튼을 클릭한다.



9. 로그를 표시한다

보기 > 패널 > 로그 메시지 를 체크

10. 플러그인이 활성화되면 오른쪽에 서버 관련 패널이 나타난다.



11. **Start Server** 버튼을 눌러 플러그인 서버를 실행한다. 이제부터 플러그인 서버를 통해 명령을 전달할 수 있는 상태가 되었다. 플러그인 관련 메시지는 2번으로 표시된 곳에 나타난다.

사용, 그리고 허가

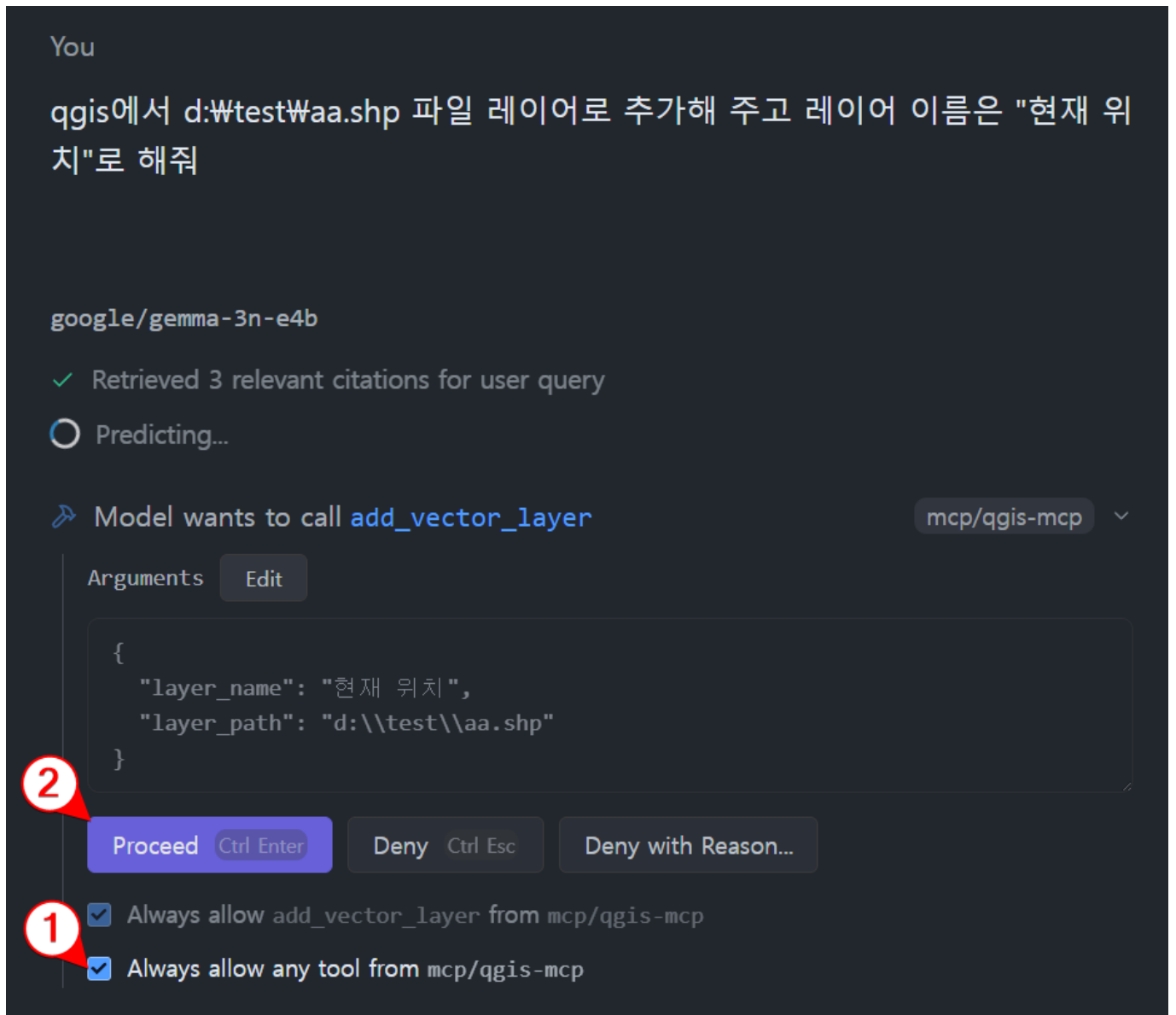
QGIS에서 새로운 프로젝트를 하나 생성하자.

그리고 QGIS의 MCP 플러그인 서버를 멈췄다가([Stop] 버튼) 다시 실행([Start] 버튼)하자. 서버는 오랜 시간 접속이 없으면 소리소문없이 동작이 중지되기도 하기 때문.

이제 LM Studio 창에서,

"qgis에서 d:\test\aa.shp 파일 레이어로 추가해 주고 레이어 이름은 "현재 위치"로 해줘"

라고 명령을 내려보자.



MCP 설정하고 맨 처음 LM Studio를 통해 실행하면 사용 허가를 묻는데 위와 같이 **Always allow any tool from mcp/qgis-mcp**를 체크하고 [Proceed] 버튼을 누른다.

테스트

- qgis에 새 벡터 레이어를 추가해 줘. 벡터 파일 경로는 "e:\temp\data\aa.shp"이고, 레이어 이름은 "test"야.
- qgis 프로젝트를 "e:\temp\qgis\test.qgz"에 저장해줘
- qgis 프로젝트의 내용을 "e:\temp\qgis\test.png"에 렌더링 해줘
- qgis 프로젝트의 내용을 "e:\temp\qgis\test.png"에 렌더링 해줘 가로는 2000, 세로는 1500이야 <- 이거 잘 안됨.