

MAC 사용자를 위한 프롬프트 관리자 도구 설치 가이드

이 가이드는 깃허브에서 프롬프트 매니저 프로젝트를 내려받고 실행하는 방법을 단계별로 안내합니다.

필요한 프로그램

1. **IDE**: 코드를 편집하고 터미널에서 명령어를 실행할 수 있는 개발 환경 (VSCode 또는 Cursor.ai)
2. **Homebrew**: MAC용 패키지 관리자
3. **Miniconda**: Python 가상환경 관리 도구
4. **Git**: 깃허브에서 코드를 내려받기 위한 프로그램
5. **Node.js**: 프론트엔드 실행에 필요한 자바스크립트 실행 환경

1. 준비하기

IDE 설치하기

1. VSCode 또는 Cursor.ai를 다운로드하여 설치합니다.
2. 설치 후 프로그램을 실행합니다.

터미널 실행하기

1. IDE에서 새 터미널을 엽니다:
 - VSCode: 상단 메뉴 **터미널 > 새 터미널**
 - Cursor.ai: 상단 메뉴 **터미널 > 새 터미널**

필요한 도구 설치 확인하기

설치되어 있지 않다면 2번 항목을 참조해 주세요.

1. Homebrew 설치 확인:

```
brew --version
```

2. Git 설치 확인:

```
git --version
```

3. Miniconda 설치 확인:

```
conda --version
```

1. Node.js 설치 확인:

```
node --version  
npm --version
```

Tip. Node.js는 Python과 달리 가상환경 내에 별도로 설치하지 않습니다. Node.js는 시스템 전역에 설치되며, npm(Node Package Manager)으로 관리하는 패키지들은 프로젝트 폴더 내의 `node_modules` 디렉토리에 저장됩니다.

2. 필요한 프로그램 설치하기 (설치되어 있지 않은 경우)

Homebrew 설치하기

```
/bin/bash -c "$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh>)"
```

설치 후에 표시되는 안내에 따라 추가 설정을 완료하세요.

Git 설치하기

```
brew install git
```

Miniconda 설치하기

```
brew install --cask miniconda
```

설치 후 초기화:

```
conda init zsh
```

초기화 후에는 터미널을 다시 열어야 합니다.

Node.js 설치하기

```
brew install node
```

3. 프로젝트 내려받기

1. 터미널에서 프로젝트를 저장할 폴더로 이동합니다:

```
cd 원하는/폴더/경로
```

Tip. 터미널을 사용한 경로 설정이 어려울 경우 좌측 상단 메뉴 **파일 > 폴더 열기** 를 통해 프로젝트를 저장할 폴더를 선택할 수 있습니다.

2. 프로젝트를 내려받을 새 폴더를 만들고 이동합니다:

```
mkdir prompt-manager  
cd prompt-manager
```

3. 깃허브에서 프로젝트를 내려받습니다:

```
git clone <https://github.com/mainhyosuk/prompt-manager.git> .
```

(마지막의 점은 현재 폴더에 내려받겠다는 의미입니다)

4. 백엔드 설정하기

Python 가상환경 설정

1. 프로젝트를 위한 가상환경 만들기:

```
conda create -n prompt-manager python=3.11 -y
```

2. 만든 가상환경 활성화하기:

```
conda activate prompt-manager
```

(터미널에서 앞에 (prompt-manager)가 표시되면 활성화된 것입니다)

3. 필요한 파이썬 패키지 설치하기:

```
pip install -r requirements.txt
```

5. 프론트엔드 설정하기

1. 프로젝트 폴더에서 프론트엔드 패키지 설치하기:

```
npm install
```

(이 단계는 시간이 다소 걸릴 수 있습니다)

6. 프로젝트 실행하기

두 개의 터미널 창이 필요합니다.

첫 번째 터미널 (백엔드 실행)

1. IDE에서 새 터미널을 열거나 기존 터미널을 사용합니다.
2. 가상환경이 활성화되어 있는지 확인합니다. 활성화되어 있지 않다면:

```
conda activate prompt-manager
```

1. 백엔드 서버 실행하기:

```
python python/api/app.py
```

"프롬프트 관리 도구 API"와 같은 메시지가 표시되면 정상적으로 실행된 것입니다.

두 번째 터미널 (프론트엔드 실행)

1. IDE에서 새 터미널을 엽니다.
2. 프로젝트 폴더에 있는지 확인합니다. 아니라면:

```
cd 원하는/폴더/경로/prompt-manager
```

1. 프론트엔드 개발 서버 실행하기:

```
npm run dev
```

1. 브라우저에서 접속하기:
 - 터미널에 표시된 주소(일반적으로 <http://localhost:5173>)를 브라우저에 입력합니다.

주의사항

- 백엔드와 프론트엔드 모두 실행해야 애플리케이션이 정상적으로 작동합니다.
- 프로그램을 종료하려면 각 터미널에서 Ctrl+C를 누르세요.
- 다음에 다시 실행할 때는 처음부터 할 필요 없이 6번 단계부터 시작하면 됩니다.

문제 해결

- "command not found" 오류가 발생하면 해당 프로그램이 설치되지 않은 것입니다. 위 안내에 따라 필요한 프로그램을 설치하세요.
- 포트가 이미 사용 중이라는 오류가 나타나면 해당 포트를 사용하는 다른 프로그램을 종료하거나, 소스 코드에서 포트 번호를 변경하세요.
- 패키지 설치 중 오류가 발생하면 인터넷 연결을 확인하고 다시 시도하세요.
- Homebrew 설치 후 'brew command not found'가 발생하면 PATH 설정이 필요할 수 있습니다. Homebrew 설치 과정에서 표시된 안내를 확인하세요.

이 가이드를 따라하면 MAC에서 프롬프트 매니저를 성공적으로 설치하고 실행할 수 있습니다.