

## Windows 와 macOS 에서 필수 패키지 설치 가이드

Python 패키지를 설치할 때, 특히 **C++ 확장 모듈이 포함된 패키지 (예: TensorFlow, PyTorch, OpenCV, Scipy, Numpy, LightGBM)** 를 사용할 경우 **C++ 컴파일러 및 관련 도구**가 필요합니다. 이 과정은 **Windows 와 macOS 에서 다르게 진행되므로**, 각각의 방법을 설명합니다.

### ▪ Windows 에서 설치 과정

#### 1. Visual C++ Build Tools 설치

Python 의 일부 패키지는 C++ 코드로 작성되어 있으며, 이를 Windows 에서 사용하려면 **Microsoft C++ 빌드 도구**가 필요합니다.

#### ※ 설치 방법

1. [Microsoft Visual Studio Build Tools 다운로드](https://visualstudio.microsoft.com/ko/visual-cpp-build-tools/) 페이지에 접속합니다.

<https://visualstudio.microsoft.com/ko/visual-cpp-build-tools/>

2. "Build Tools for Visual Studio 2022" 를 다운로드합니다.
3. 다운로드한 vs\_BuildTools.exe 파일을 실행합니다.
4. 설치 화면에서 다음과 같은 필수 컴포넌트를 선택합니다:

| 항목  | 설명  |
|---|---|
| <input checked="" type="checkbox"/> MSVC v143 - VS 2022 C++       | 가장 최신 C++ 컴파일러 (Windows 10/11 모두 사용 가능)     |
| <input checked="" type="checkbox"/> Windows 10 SDK (10.0.20348.0) | Windows 10 용 API, 헤더, 라이브러리 (가장 최신 버전)      |
| <input checked="" type="checkbox"/> Windows 용 C++ CMake 도구        | CMake 기반 빌드 지원 (vcpkg, cross-platform 에 중요) |
| <input checked="" type="checkbox"/> 테스트 도구 핵심 기능 - 빌드 도구          | 테스트 프레임워크를 위한 구성                            |
| <input checked="" type="checkbox"/> C++ AddressSanitizer          | 런타임 메모리 오류 탐지에 유용                           |
| <input checked="" type="checkbox"/> vcpkg 패키지 관리자                 | 외부 라이브러리 설치용 필수 도구                          |

#### ▼ 포함됨

- ✓ C++ 빌드 도구 핵심 기능
- ✓ C++ 2022 재배포 가능 업데이트
- ✓ C++ 핵심 데스크톱 기능

#### ▼ 선택 사항

- ☒ MSVC v143 - VS 2022 C++ x64/x86 빌드 도...
- ☐ Windows 11 SDK (10.0.22H2.3916)
- ☒ Windows용 C++ CMake 도구
- ☒ 테스트 도구 핵심 기능 - 빌드 도구
- ☒ C++ AddressSanitizer
- ☒ vcpkg 패키지 관리자
- ☐ 최신 v143 빌드 도구용 C++ ATL(x86 및 x64)
- ☐ 최신 v143 빌드 도구용 C++ MFC(x86 및 x64)
- ☐ v143 빌드 도구에 대한 C++/CLI 지원(최신)
- ☐ v143 빌드 도구용 C++ 모듈(x64/x86 - 실험...
- ☐ Windows용 C++ Clang 도구(19.1.5 - x64/x...
- ☐ Windows 11 SDK(10.0.22H2.10)
- ☐ Windows 11 SDK(10.0.22H2.0)
- ☒ Windows 10 SDK(10.0.20348.0)
- ☐ Windows 10 SDK(10.0.19041.0)

- ✓ **C++ 빌드 도구 (C++ CMake Tools for Windows)**
- ✓ **MSVC v143 - VS 2022 C++ x64/x86 build tools**
- ✓ 본인 OS 에 따라서 **Windows 10 SDK** (최신 버전 1 개만 설치) 또는 **Windows 11 SDK** (최신 버전 1 개만 설치)

5. "설치" 버튼을 클릭하여 설치를 진행합니다. (상기 예시는 PC 가 윈도우 10 인 경우 선택)

※ 설치 후 설정 반영을 위해 반드시 PC 를 재부팅하세요.

---

## 2. pip 및 Python 가상환경 확인

설치 후, Python 환경에서 제대로 적용되었는지 확인합니다.

> python -m pip install --upgrade pip setuptools wheel

이 명령어를 실행하면 pip, setuptools, wheel 을 최신 버전으로 업데이트합니다.

---

## 3. C++ 빌드 도구 확인

설치가 잘 되었는지 터미널(CMD 또는 PowerShell) 에서 아래 명령어를 입력합니다.

> cl

출력 결과가 아래와 같이 나오면 성공적으로 설치된 것입니다.

> Microsoft (R) C/C++ Optimizing Compiler Version XX.XX.XXXXX for x64

또는, 아래 명령어를 실행하여 CMake 가 정상적으로 설치되었는지 확인합니다.

> cmake --version

출력이 나온다면 정상적으로 설치된 것입니다.

---

## 4. 필요한 패키지 설치

이제 필요한 패키지를 설치할 수 있습니다.

> pip install numpy scipy pandas lightgbm opencv-python

---

## 5. PowerShell 에서 배치 스크립트 동작 허용

Windows 의 VS Code 터미널에서 PowerShell 로 파이썬 가상화 동작 등 배치

스크립트(PowerShell 스크립트)를 실행하려면, 실행 정책(Execution Policy)을 해제 또는 완화해야 한다. 기본 설정은 Restricted 상태라서 스크립트 실행이 차단이다.

※ 실행 정책 목록 확인

> Get-ExecutionPolicy -List

※ 실행 정책 완화 적용

> Set-ExecutionPolicy RemoteSigned -Scope CurrentUser

※ 실행 정책 적용 확인

> Get-ExecutionPolicy -Scope CurrentUser

## ▪ macOS 에서 설치 과정

macOS 에서는 Windows 와 달리 Visual Studio Build Tools 가 필요하지 않지만, **Apple 의 Command Line Tools** 및 **Homebrew** 가 필요합니다.

### 1. Xcode Command Line Tools 설치

macOS 에서는 기본적으로 C++ 컴파일러(Clang)가 포함되어 있지만, 최신 패키지를 설치하려면 **Xcode Command Line Tools** 가 필요합니다.

#### ※ 설치 방법

```
> xcode-select --install
```

설치 완료 후, 아래 명령어로 정상 설치되었는지 확인합니다.

```
> xcode-select -p
```

출력이 /Library/Developer/CommandLineTools 라면 정상적으로 설치된 것입니다.

---

### 2. Homebrew 설치

Homebrew 는 macOS 에서 **CMake, GCC** 등의 패키지 관리를 쉽게 해줍니다.

#### ※ 설치 방법

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

설치 후, 아래 명령어로 Homebrew 가 정상 작동하는지 확인합니다.

```
> brew --version
```

출력 예시:

```
> Homebrew 4.0.11
```

---

### 3. 필수 개발 도구 설치

#### ※ Homebrew 를 이용하여 필요한 도구 설치

```
> brew install cmake
```

```
> brew install gcc
```

```
> brew install llvm
```

설치가 완료되었는지 확인하려면 다음 명령어를 실행하세요.

```
> cmake --version
```

```
> gcc --version
```

```
> llvm-config --version
```

---

### 4. pip 및 Python 가상환경 확인

```
> python3 -m pip install --upgrade pip setuptools wheel
```

---

### 5. 필요한 패키지 설치

이제 필요한 Python 패키지를 설치합니다.

```
> pip install numpy scipy pandas lightgbm opencv-python
```

▪ Windows vs macOS 비교

| 항목         | Windows   | macOS                                  |
|------------|---|--|
| 필수 툴       | Visual Studio C++ Build Tools                         | Xcode Command Line Tools               |
| 설치 방법      | Visual Studio 설치 후, C++ 컴파일러 & SDK 추가                 | xcode-select --install                 |
| 패키지 관리자    | 기본 제공   | Homebrew (brew install cmake gcc llvm) |
| pip 패키지 설치 | pip install numpy scipy pandas lightgbm opencv-python | 동일                                     |

※ 요약

1. Windows → Visual Studio C++ Build Tools 설치 필수!
2. macOS → Xcode Command Line Tools + Homebrew 설치 필요!
3. 설치 후 pip install 을 사용하여 원하는 패키지를 설치하면 정상 작동합니다.

※ Poetry 를 사용한다면?

- 추후 여러 프로젝트를 관리하려면 Poetry 를 설치하여 가상환경 관리를 쉽게 할 수 있습니다.
- 설치 방법:  
 > curl -sSL https://install.python-poetry.org | python3 -  
 이후 poetry 명령어를 활용하여 프로젝트별 환경을 관리하면 됩니다. 이는 FastAPI 를 이용한 Server Local 상용 서비스 구성 시 별도 안내합니다.