

# 비전공자를 위한 환경 설정 가이드

## 현업 수준의 파이썬 개발 환경 구축 A to Z

본 자료는 입문자가 실무 환경을 완벽히 준비할 수 있도록 단계별 상세 지침을 제공합니다.

## 1. 코드 에디터: VS Code 설치

프로그래밍을 시작하기 위해 전 세계에서 가장 널리 쓰이는 도구인 **Visual Studio Code**를 설치합니다.

### 과정 01. 공식 배포처 접속 및 다운로드

- 웹사이트 접속: [code.visualstudio.com](#)
- 중앙의 '**Download for Windows**' 버튼을 클릭하여 설치 파일(.exe)을 내려받습니다.

### 과정 02. 설치 시 필수 체크 사항 (중요)

설치 프로그램을 실행한 후, '추가 작업 선택' 화면에서 다음 항목을 반드시 선택하십시오.

- [V] PATH에 추가 (이 옵션은 터미널에서 에디터를 호출하기 위해 필수적입니다)
- [V] Code로 열기 메뉴 추가 (폴더나 파일을 우클릭하여 즉시 편집할 수 있게 합니다)

## 2. 사용자 친화적인 환경 설정

에디터를 처음 실행하면 영어로 되어 있어 생소할 수 있습니다. 한글화와 편리한 기능을 설정합니다.

### 과정 03. 한글 언어 팩 적용

- 왼쪽 사이드바의 **Extensions 아이콘**(네모난 블록 모양)을 클릭합니다.
- 검색창에 **Korean**을 입력합니다.
- '**Korean Language Pack**' 항목의 [Install] 버튼을 클릭합니다.
- 우측 하단 알림창에서 [Change Language and Restart]를 선택하여 재시작합니다.

## 3. 필수 확장 프로그램(Extensions)

VS Code는 다양한 플러그인을 통해 기능을 확장합니다. 파이썬 개발에 반드시 필요한 3가지를 설치합니다.

### 권장 설치 목록

- Python (Microsoft)**: 파이썬 언어를 이해하고 오류를 잡아주는 핵심 도구입니다.
- GitHub Copilot**: 인공지능이 코드를 함께 작성해 주는 최신 기술입니다. (GitHub 계정 연동 필요)
- Jupyter**: 데이터 분석 결과를 시각적으로 확인하기 위한 필수 도구입니다.

※ 설치 방법: 사이드바의 Extensions 메뉴에서 이름을 검색한 뒤 [설치]를 클릭하십시오.

## 4-1. 패키지 관리자: Miniconda (Windows)

---

프로젝트 별로 독립된 실험실(가상환경)을 만들기 위해 **Miniconda**를 사용합니다.

Windows용 설치 지침

- **직접 링크:** [Miniconda3 최신 버전 다운로드](#)
  - **설치 주의사항:**
    - 설치 중 "**Add Miniconda3 to my PATH**" 체크박스가 나타나면 반드시 체크해 주십시오.
    - 체크하면 별도의 프로그램 실행 없이 VS Code 터미널에서 즉시 파이썬을 사용할 수 있습니다.
- 

### 4-1-1. PowerShell 실행 정책 설정 (Windows 필수)

---

Conda가 정상적으로 작동하려면 **PowerShell의 실행 정책**을 변경해야 합니다.

과정 04. PowerShell 관리자 권한으로 실행

1. Windows 검색창에서 **PowerShell**을 입력합니다.
  2. **Windows PowerShell**을 우클릭한 후 \*\*'관리자 권한으로 실행'\*\*을 선택합니다.
- 

과정 05. 실행 정책 변경

관리자 권한 PowerShell 창에서 다음 명령어를 입력합니다.

```
# 실행 정책을 RemoteSigned로 변경  
Set-ExecutionPolicy RemoteSigned  
  
# 변경 확인 메시지가 나오면 'Y' 입력 후 Enter
```

과정 06. Conda 초기화

PowerShell에서 Conda를 사용할 수 있도록 초기화합니다.

```
# Conda를 PowerShell에서 사용할 수 있도록 초기화  
conda init powershell  
  
# PowerShell을 재시작하거나 VS Code를 재시작합니다.
```

※ 명령어 실행 후 **VS Code**를 완전히 종료했다가 다시 실행해야 정상 작동합니다.

---

## 4-2. 패키지 관리자: Miniconda (macOS)

---

Apple Mac 사용자는 하드웨어 사양에 맞는 버전을 선택하여 설치하셔야 합니다.

시스템 확인 및 다운로드

화면 왼쪽 위 [애플 로고] > [이 Mac에 관하여]에서 프로세서 종류를 먼저 확인하십시오.

- Apple M1/M2/M3 칩 (Apple Silicon): [arm64 \(.pkg\) 버전 다운로드](#)
- Intel 칩: [x86\\_64 \(.pkg\) 버전 다운로드](#)

---

#### 보안 경고 발생 시 조치 방법

macOS 보안 설정으로 인해 설치가 차단될 경우, 다운로드한 파일을 우클릭한 후 '열기'(또는 '설치 프로그램' 관련 기본 메뉴)를 선택하여 강제 실행해 주십시오.

#### 설치 완료 확인

터미널 앱을 열고 `conda --version`을 입력하여 버전 정보가 나오면 성공입니다.

---

## 5. 가상환경 구축 및 관리

---

여러 프로젝트를 동시에 진행할 때, 라이브러리 간 충돌을 방지하기 위해 독립된 환경을 만듭니다.

터미널은 왼쪽 위 또는 `Ctrl + ~` 키를 눌러 열 수 있습니다.

```
# 1. 'my_env'라는 이름의 새로운 가상환경을 생성합니다.  
conda create -n my_env python=3.10  
  
# 2. 생성한 환경을 활성화(입장)합니다. (프롬프트 앞 괄호 확인)  
conda activate my_env
```

---

#### 추가 패키지 설치 (선택 사항)

데이터 분석과 시각화에 필요한 주요 라이브러리들을 설치합니다.

```
# 3. 데이터 분석에 필수적인 패키지들을 설치합니다.  
pip install pandas numpy matplotlib seaborn scikit-learn  
  
# 4. 현재 생성되어 있는 가상환경들의 목록을 확인합니다.  
conda env list
```

---

## 6. VS Code 인터프리터 연결

---

에디터가 내가 만든 가상환경을 사용하도록 직접 지정해 주어야 합니다.

#### 연결 단계 설명

1. VS Code 내에서 `Ctrl + Shift + P`를 눌러 명령 창을 엽니다.
2. `Select Interpreter`를 입력하고 실행합니다.
3. 목록에서 만든 환경의 이름으로 표시된 항목을 선택합니다.

---

## 성공 여부 확인

에디터 우측 하단 상태 표시줄에 \*\*'my\_env'\*\*라는 이름이 표시되는지 확인하십시오.

이제 모든 코드는 이 안전한 환경 내에서 실행됩니다.

---

## 환경 구축을 축하드립니다! 🎉

---

이제 효율적이고 전문적인 개발을 위한 모든 준비가 끝났습니다.

실제 강의 예제로 돌아가서 코딩을 시작해 보세요.

---