

02

클로드 코드 및 필수 프로그램 설치

클로드 앱 · 파이썬 · Git 설치 · 파이썬과 Git이란

01 설치

클로드 앱 · 파이썬 · Git 설치와 확인

설치

설치할 것 세 가지와 순서

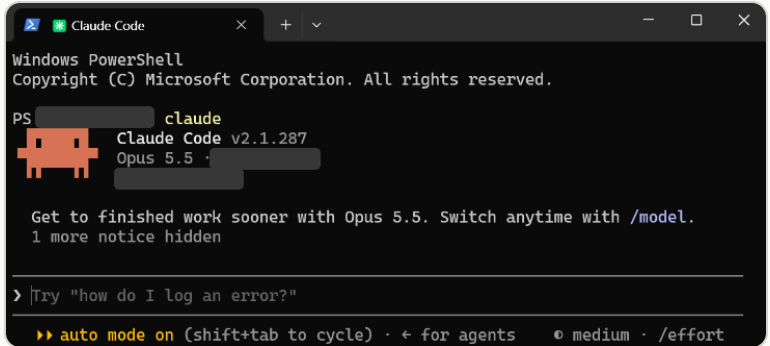
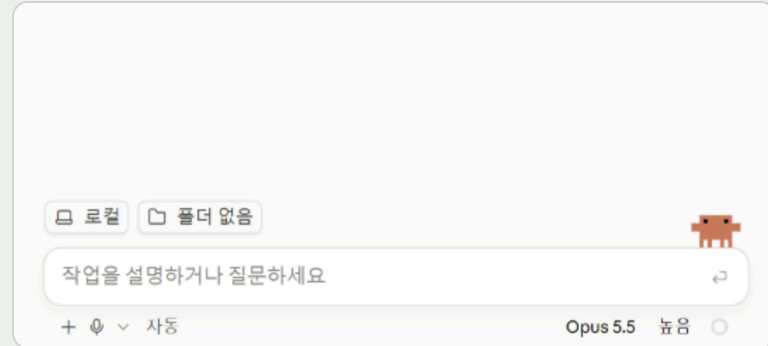
순서	프로그램	하는 일	받는 곳
1	클로드 코드 클로드 데스크톱 앱의 Code 탭	내 PC 폴더의 파일을 읽고, 코드를 쓰고, 실행한다	claude.com/download
2	파이썬 3.13 프로그래밍 언어 · 실행 프로그램	클로드 코드가 쓴 코드(.py)를 실행한다	python.org
3	Git 윈도우에서 설치 권장	파일이 바뀐 기록을 남긴다 · 클로드 코드가 명령을 실행할 때 쓴다	git-scm.com

세 가지 모두 공식 사이트 설치 파일로, 이 순서대로 설치한다 - 설치 전 · 후 터미널에서 확인



참고 · 설치

클로드 코드는 두 가지 모습 - 터미널과 데스크톱 앱

구분	터미널	데스크톱 앱 (이 강의)
화면		
설치	설치 명령 한 줄로 따로	앱에 들어 있다
새로 깐 프로그램 반영	터미널 창을 새로 연다	앱을 완전히 껐다 켜다
이쪽에서만 되거나 더 쉬운 것	대화 없이 한 번에 실행(<code>claude -p</code>) · 에이전트 팀 · 터미널 안 설정 화면	끌어다 놓기 · 바뀐 곳을 색으로 확인 · 예약 작업 버튼
같이 쓰는 것	CLAUDE.md · 스킬 · MCP · 설정 파일 - 같은 엔진이라 그대로 쓴다	

같은 엔진이지만 결과를 화면으로 보고 확인하며 일하는 실무자에게는 데스크톱 앱이 더 낫다

먼저 확인 - 이미 있으면 그 설치 건너뛰다

- 1 작업 표시줄의 시작 버튼(윈도우 로고) 클릭 → powershell 입력

윈도우 10도 같다

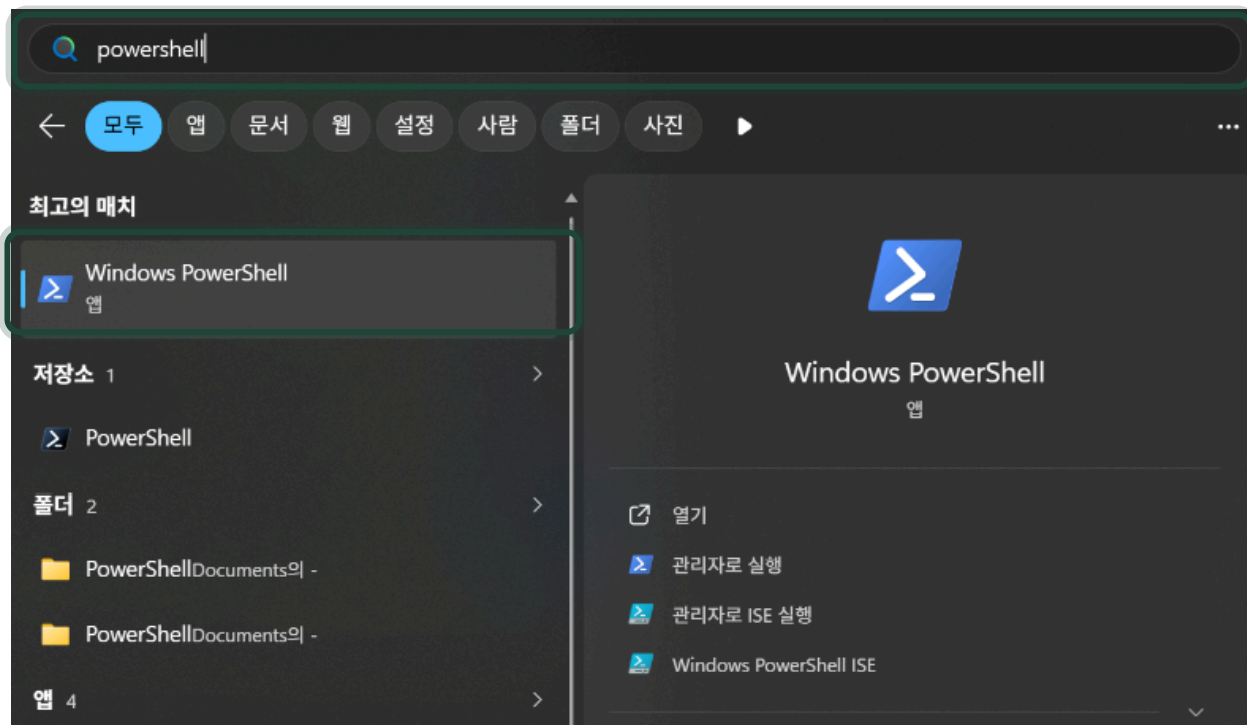
- 2 맨 위 Windows PowerShell 클릭

- 3 검은 창이 뜨면 준비 끝
깜박이는 자리에 입력한다

- 4 아래 두 줄을 한 줄씩 입력 → 줄마다 Enter

```
python --version  
git --version
```

- 5 버전 숫자가 나오면 그 설치 건너뛰다
파이썬은 3.11 ~ 3.14면 건너뛴 · 클로드 앱은 로그인까지
했으면 건너뛴



시작 → powershell 입력

먼저 확인 - 이미 있으면 그 설치는 건너뛰다

- 1 작업 표시줄의 시작 버튼(윈도우 로고) 클릭 → powershell 입력
윈도우 10도 같다

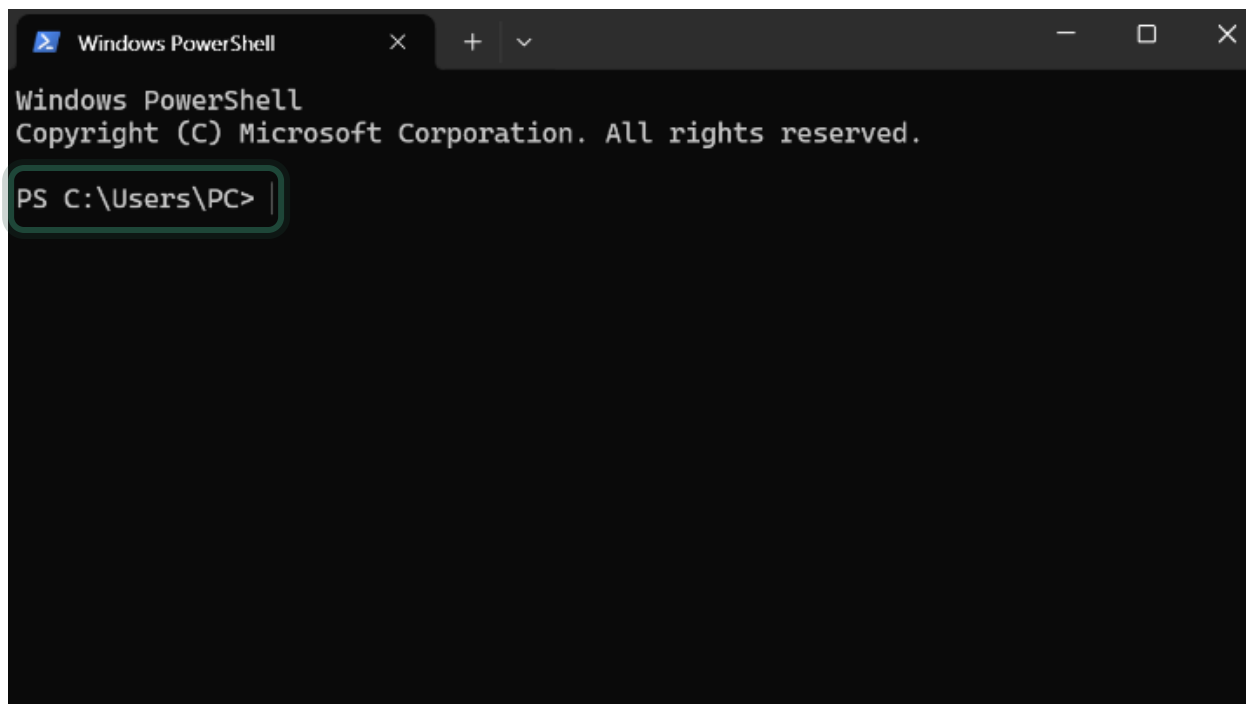
- 2 맨 위 Windows PowerShell 클릭

- 3 검은 창이 뜨면 준비 끝
깜박이는 자리에 입력한다

- 4 아래 두 줄을 한 줄씩 입력 → 줄마다 Enter

```
python --version  
git --version
```

- 5 버전 숫자가 나오면 그 설치의 건너뛰다
파이썬은 3.11 ~ 3.14면 건너뛰 · 클로드 앱은 로그인까지
했으면 건너뛰



PowerShell을 연 화면

먼저 확인 - 이미 있으면 그 설치는 건너뛰다

- 1 작업 표시줄의 시작 버튼(윈도우 로고) 클릭 →
powershell 입력
윈도우 10도 같다

- 2 맨 위 Windows PowerShell 클릭

- 3 검은 창이 뜨면 준비 끝
깜박이는 자리에 입력한다

- 4 아래 두 줄을 한 줄씩 입력 → 줄마다 Enter

```
python --version  
git --version
```

- 5 버전 숫자가 나오면 그 설치의 건너뛰다
파이썬은 3.11 ~ 3.14면 건너뛰 · 클로드 앱은 로그인까지
했으면 건너뛰

버전 숫자가 나온다 → 이미 있다 · 건너뛰다

```
Windows PowerShell  ×  +  v  
  
PS C:\Users\<이름>> python --version  
Python 3.11.15  
PS C:\Users\<이름>> git --version  
git version 2.54.0.windows.1
```

이런 문장이 나온다 → 없다 · 설치한다

```
Windows PowerShell  ×  +  v  
  
PS C:\Users\<이름>> python --version  
Python was not found; run without arguments to install from the  
Microsoft Store, or disable this shortcut from Settings > Manage App  
Execution Aliases.  
PS C:\Users\<이름>> git --version  
git : 'git' 용어가 cmdlet, 함수, 스크립트 파일 또는 실행할 수 있는 프로그램 이름  
으로 인식되지 않습니다. ...
```

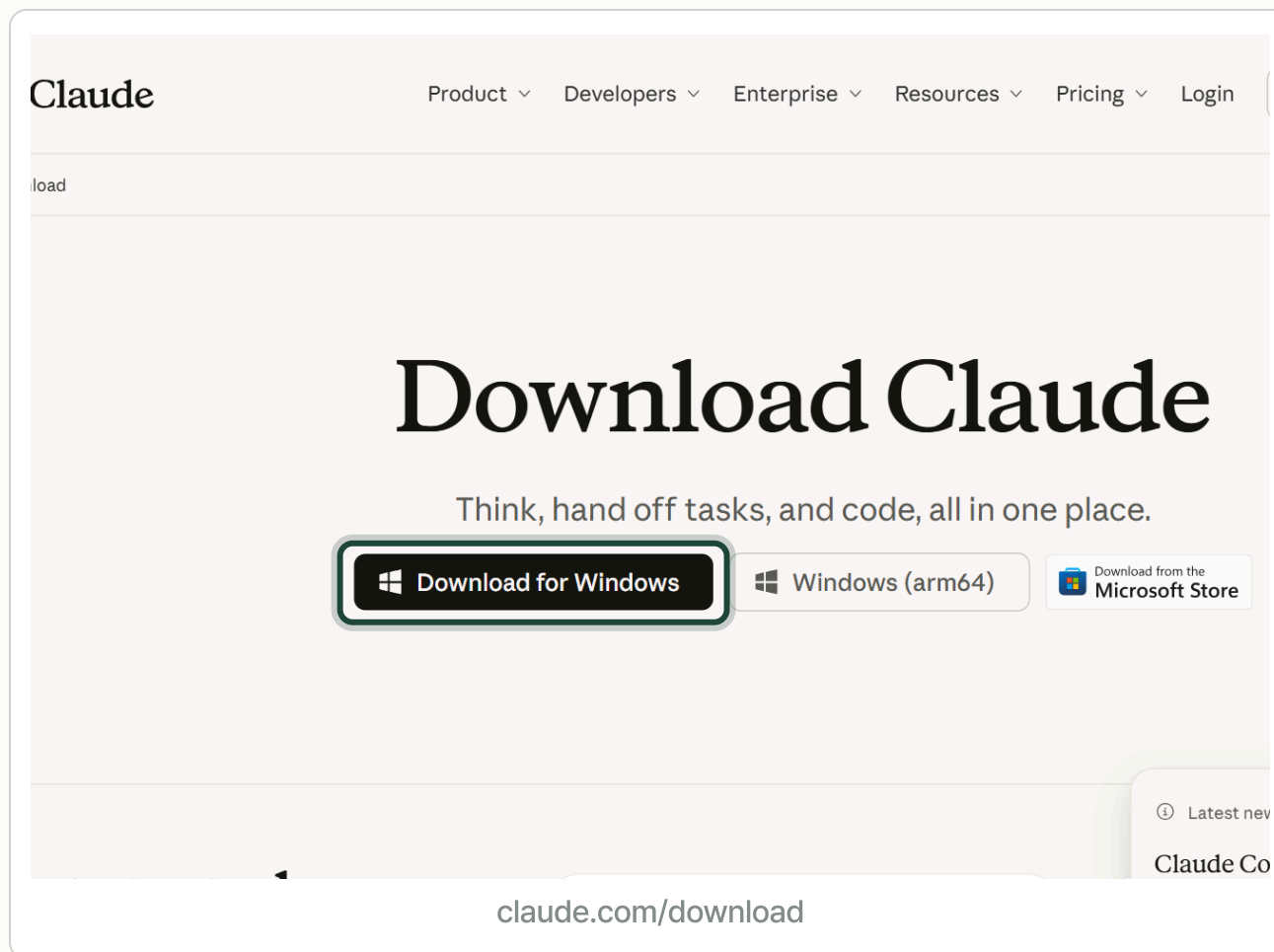
클로드 데스크톱 앱 설치

1 claude.com/download → Download for Windows

퀄컴 칩 노트북은 Windows (arm64)

2 받은 설치 파일 실행 → 로그인 → 위쪽에 Code 탭(</> 아이콘)이 보이면 끝

유료 계정(Pro 이상) · Code 탭을 여는 것은 기본 사용법
장에서



설치 1

설치 1 · 클로드 코드 · 2/2

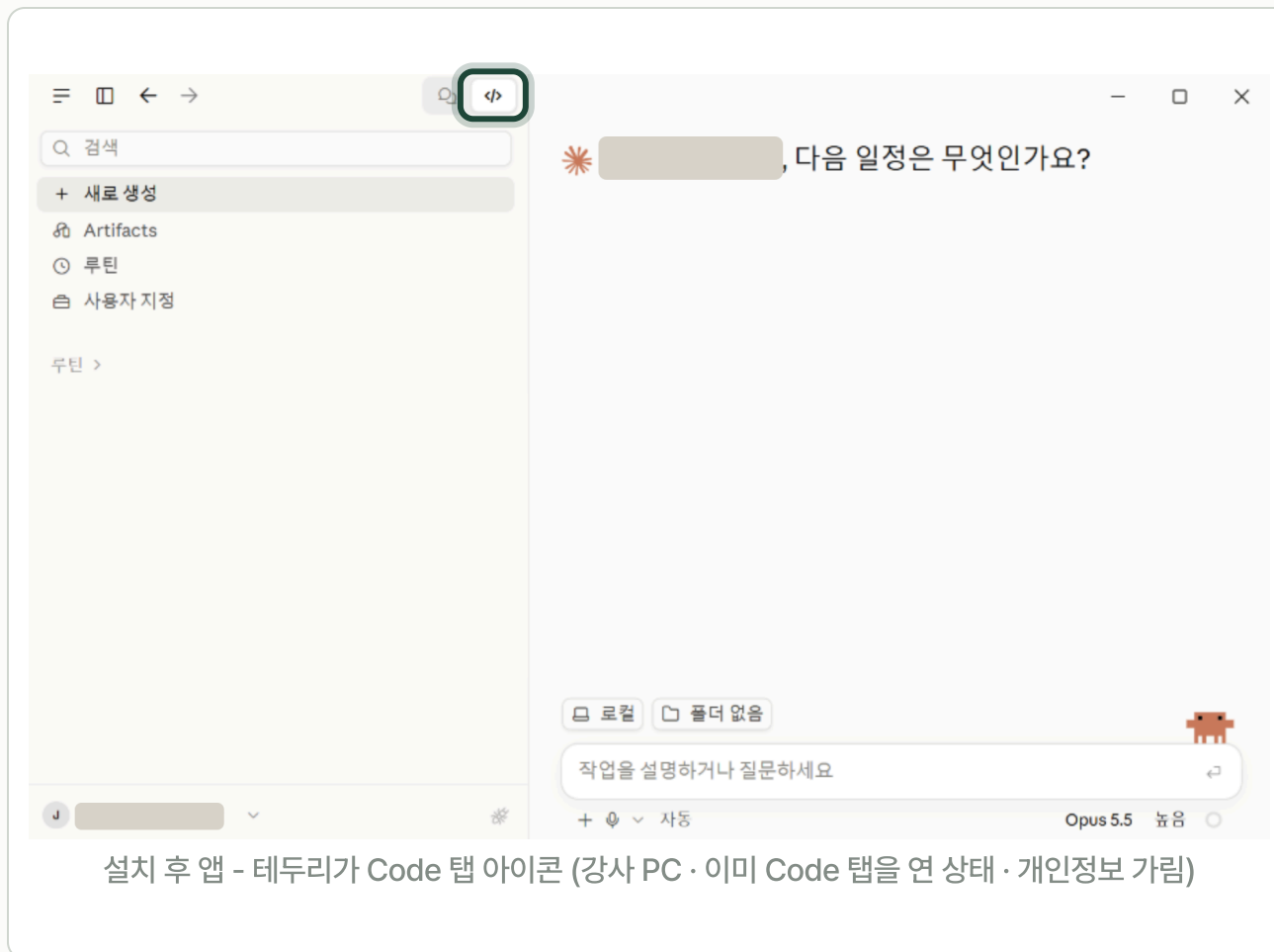
클로드 데스크톱 앱 설치

- 1 claude.com/download → Download for Windows

퀄컴 칩 노트북은 Windows (arm64)

- 2 받은 설치 파일 실행 → 로그인 → 위쪽에 Code 탭(</> 아이콘)이 보이면 끝

유료 계정(Pro 이상) · Code 탭을 여는 것은 기본 사용법
장에서



설치 후 앱 - 테두리가 Code 탭 아이콘 (강사 PC · 이미 Code 탭을 연 상태 · 개인정보 가림)

파이썬 설치

1 python.org/downloads/windows 접속 → Python 3.13.16 아래 Windows installer (64-bit)

퀄컴(ARM) 노트북도 64-bit · 이미 3.11 ~ 3.14가 있으면 건너뛰

2 내려받은 python-3.13.16-amd64.exe 실행

브라우저 오른쪽 위 다운로드 → 파일 열기

3 첫 화면 맨 아래 Add python.exe to PATH 체크 → Install Now

관리자 권한이 없는 PC는 Use admin privileges 체크 해제

4 Setup was successful → Close

Disable path length limit은 누르지 않아도 된다

- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows embeddable package \(ARM64\)](#)
- Download [Windows release manifest](#)

▪ [Python 3.13.16 - Sept. 30, 2026](#)

Note that Python 3.13.16 cannot be used on Windows 7 or earlier.

- Download [Windows installer \(64-bit\)](#)
- Download [Windows installer \(32-bit\)](#)
- Download [Windows installer \(ARM64\)](#)
- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows embeddable package \(ARM64\)](#)
- Download [Windows release manifest](#)

▪ [Python 3.13.15 - Aug. 5, 2026](#)

Note that Python 3.13.15 cannot be used on Windows 7 or earlier.

- Download [Windows installer \(64-bit\)](#)
- Download [Windows installer \(32-bit\)](#)

python.org/downloads/windows

파이썬 설치

- 1 python.org/downloads/windows 접속 → Python 3.13.16 아래 Windows installer (64-bit)

퀄컴(ARM) 노트북도 64-bit · 이미 3.11 ~ 3.14가 있으면 건너뛰

- 2 내려받은 `python-3.13.16-amd64.exe` 실행

브라우저 오른쪽 위 다운로드 → 파일 열기

- 3 첫 화면 맨 아래 Add python.exe to PATH 체크 → Install Now

관리자 권한이 없는 PC는 Use admin privileges 체크 해제

- 4 Setup was successful → Close

Disable path length limit은 누르지 않아도 된다

다운로드



python-3.13.16-amd64.exe

[파일 열기](#)

엣지 오른쪽 위 다운로드 목록 · 안 보이면 탐색기 > 다운로드 폴더

파이썬 설치

- 1 python.org/downloads/windows 접속 → Python 3.13.16 아래 Windows installer (64-bit)

퀄컴(ARM) 노트북도 64-bit · 이미 3.11 ~ 3.14가 있으면 건너뛰

- 2 내려받은 `python-3.13.16-amd64.exe` 실행

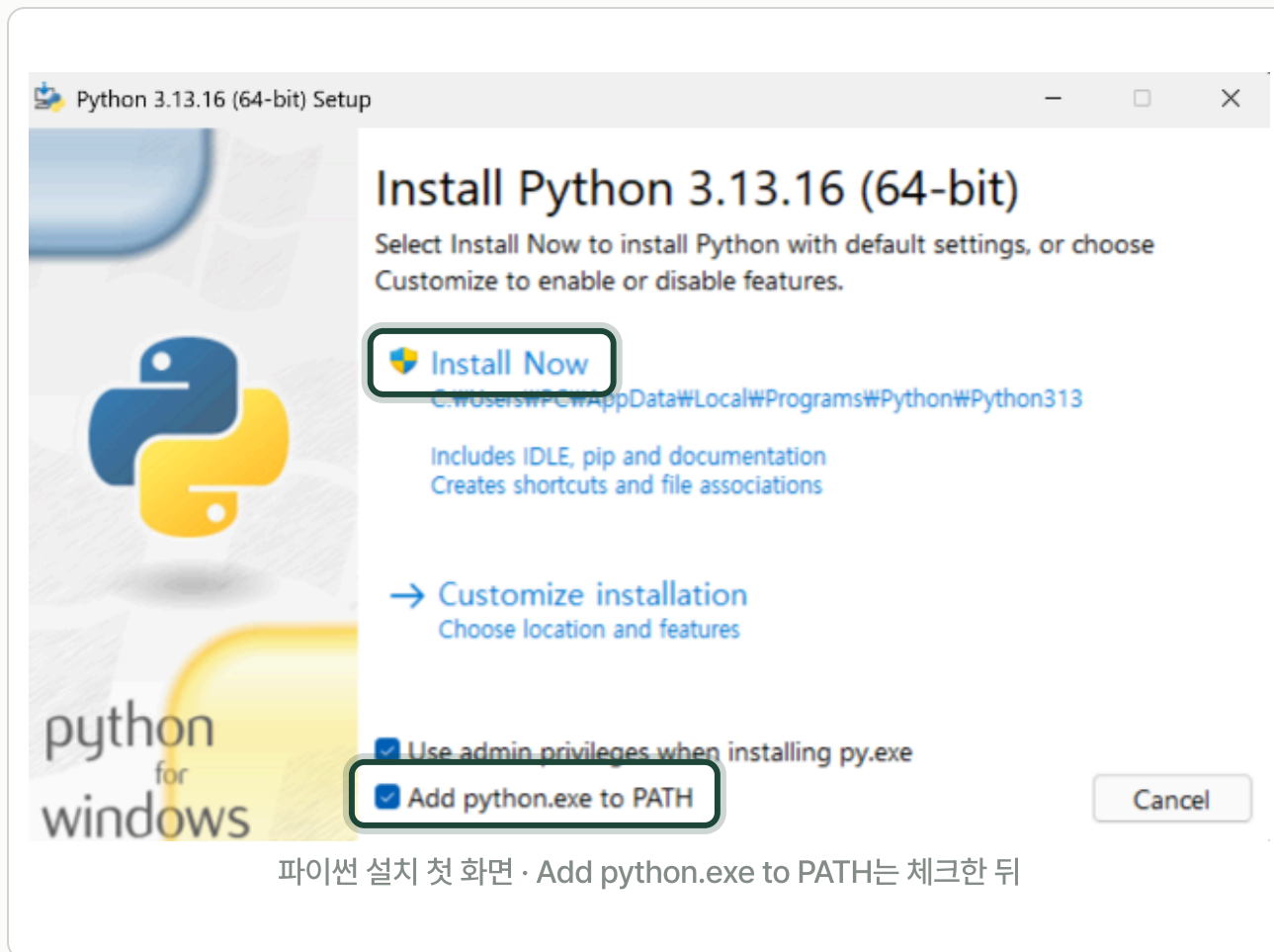
브라우저 오른쪽 위 다운로드 → 파일 열기

- 3 첫 화면 맨 아래 Add python.exe to PATH 체크 → Install Now

관리자 권한이 없는 PC는 Use admin privileges 체크 해제

- 4 Setup was successful → Close

Disable path length limit은 누르지 않아도 된다



파이썬 설치 첫 화면 · Add python.exe to PATH는 체크한 뒤

파이썬 설치

- 1 python.org/downloads/windows 접속 → Python 3.13.16 아래 Windows installer (64-bit)

퀄컴(ARM) 노트북도 64-bit · 이미 3.11 ~ 3.14가 있으면 건너뛰

- 2 내려받은 `python-3.13.16-amd64.exe` 실행

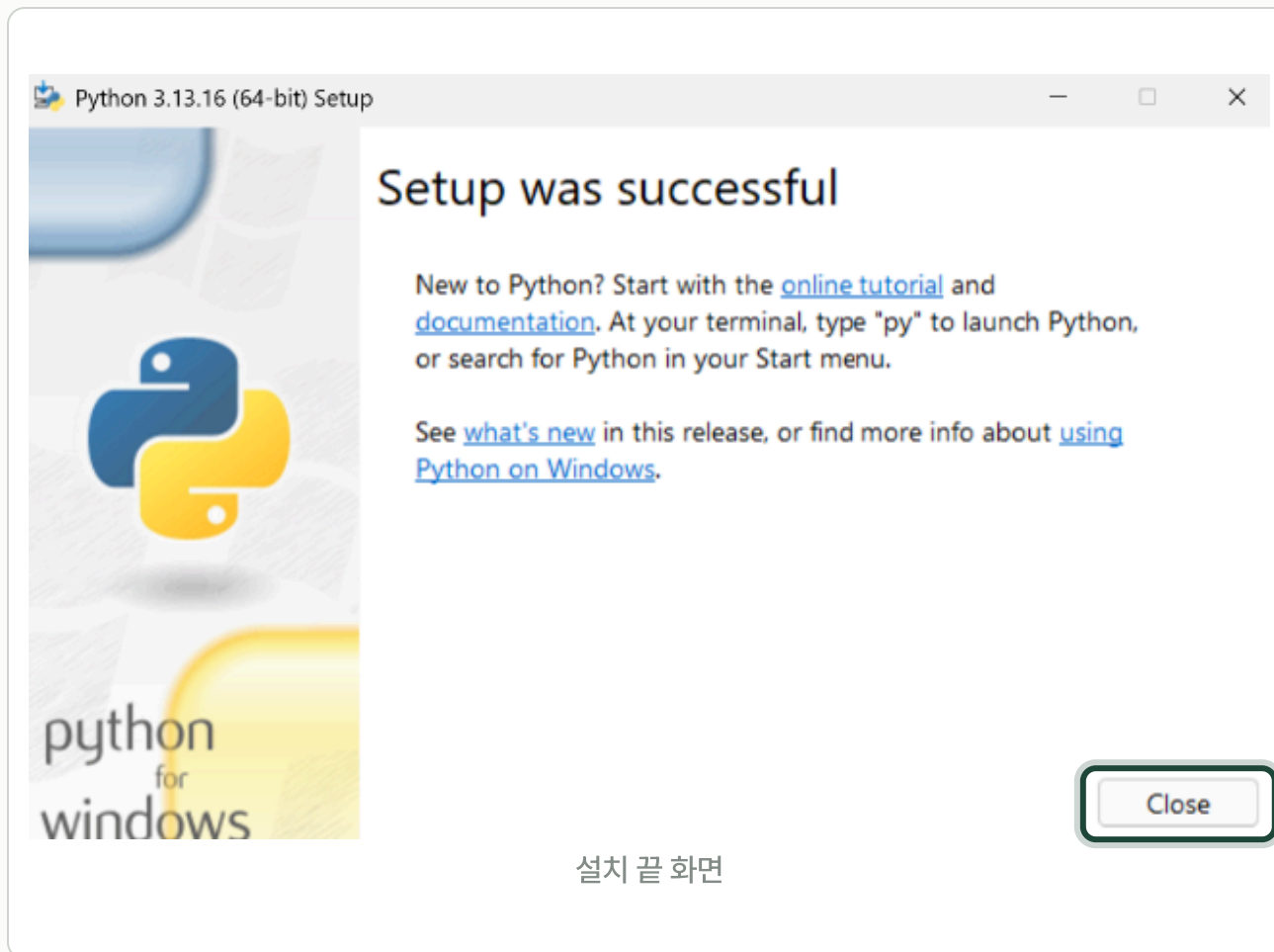
브라우저 오른쪽 위 다운로드 → 파일 열기

- 3 첫 화면 맨 아래 Add python.exe to PATH 체크 → Install Now

관리자 권한이 없는 PC는 Use admin privileges 체크 해제

- 4 Setup was successful → Close

Disable path length limit은 누르지 않아도 된다



Git 설치

- 1 git-scm.com/install/windows → Click here to download
- 2 내려받은 **Git-2.56.0-64-bit.exe** 실행 → 확인
창 예 → 처음 4개 화면 그대로 Next
라이선스 · 설치 위치 · 구성 요소 · 시작 메뉴 폴더
- 3 편집기 화면 → 목록에서 **Use Notepad as Git's default editor** → Next
처음엔 Vim · 다음 화면(브랜치 이름)은 그대로 Next
- 4 PATH 화면 → 가운데 **(Recommended)** 그대로 → Next
- 5 나머지 7개 화면 그대로 Next → 마지막에 **Install**
- 6 마지막 화면 → 체크 두 개 모두 해제 → **Finish**

Install

Latest version: 2.56.0 ([Release Notes](#))

Windows

macOS

Linux

Build from Source

[Click here to download](#) the latest (2.56.0) x64 version of **Git for Windows**. This is the most recent [maintained build](#). It was released **6 days ago**, on 2026-09-28.

Other Git for Windows downloads

Standalone Installer

[Git for Windows/x64 Setup.](#)[Git for Windows/ARM64 Setup.](#)

Portable ("thumbdrive edition")

[Git for Windows/x64 Portable.](#)[Git for Windows/ARM64 Portable.](#)

Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

```
winget install --id Git.Git -e --source winget
```

The current source code release is version **2.56.0**. If you want the newer version, you

git-scm.com/install/windows

Git 설치

1 git-scm.com/install/windows → Click here to download

2 내려받은 **Git-2.56.0-64-bit.exe** 실행 → 확인 창 예 → 처음 4개 화면 그대로 Next

라이선스 · 설치 위치 · 구성 요소 · 시작 메뉴 폴더

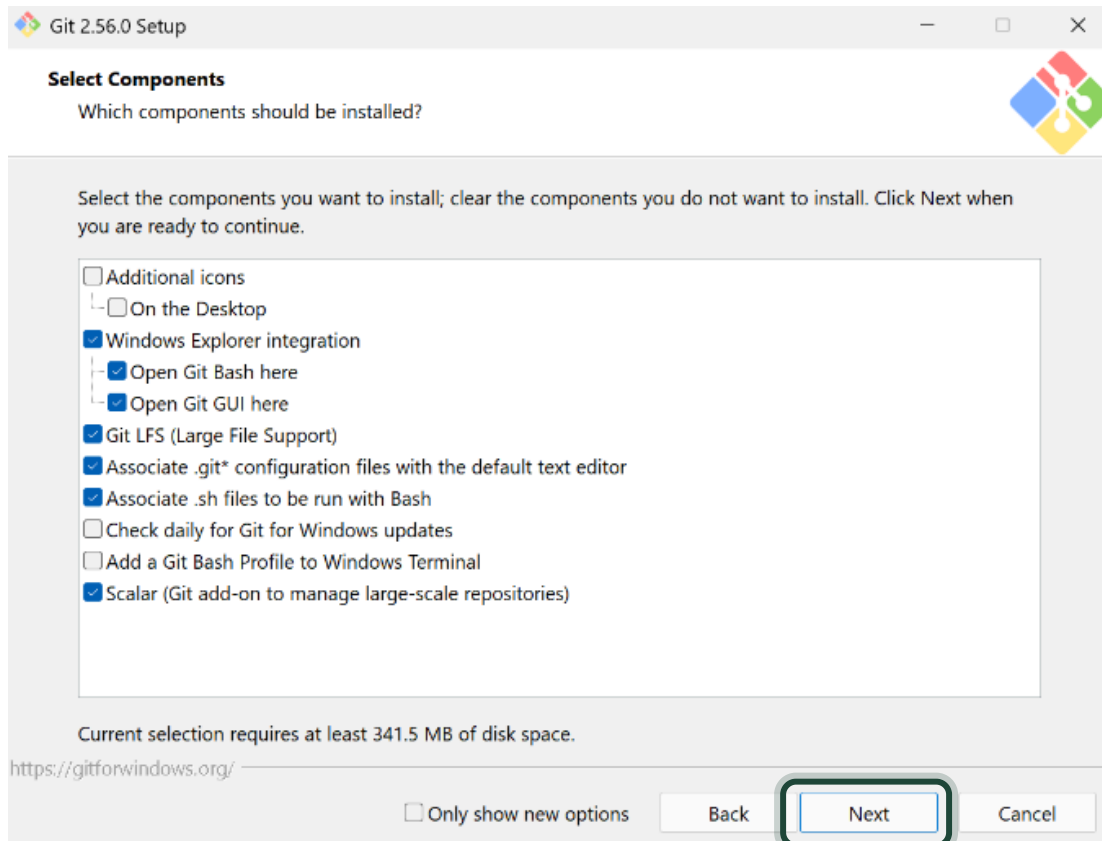
3 편집기 화면 → 목록에서 **Use Notepad as Git's default editor** → Next

처음엔 Vim · 다음 화면(브랜치 이름)은 그대로 Next

4 PATH 화면 → 가운데 (**Recommended**) 그대로 → Next

5 나머지 7개 화면 그대로 Next → 마지막에 Install

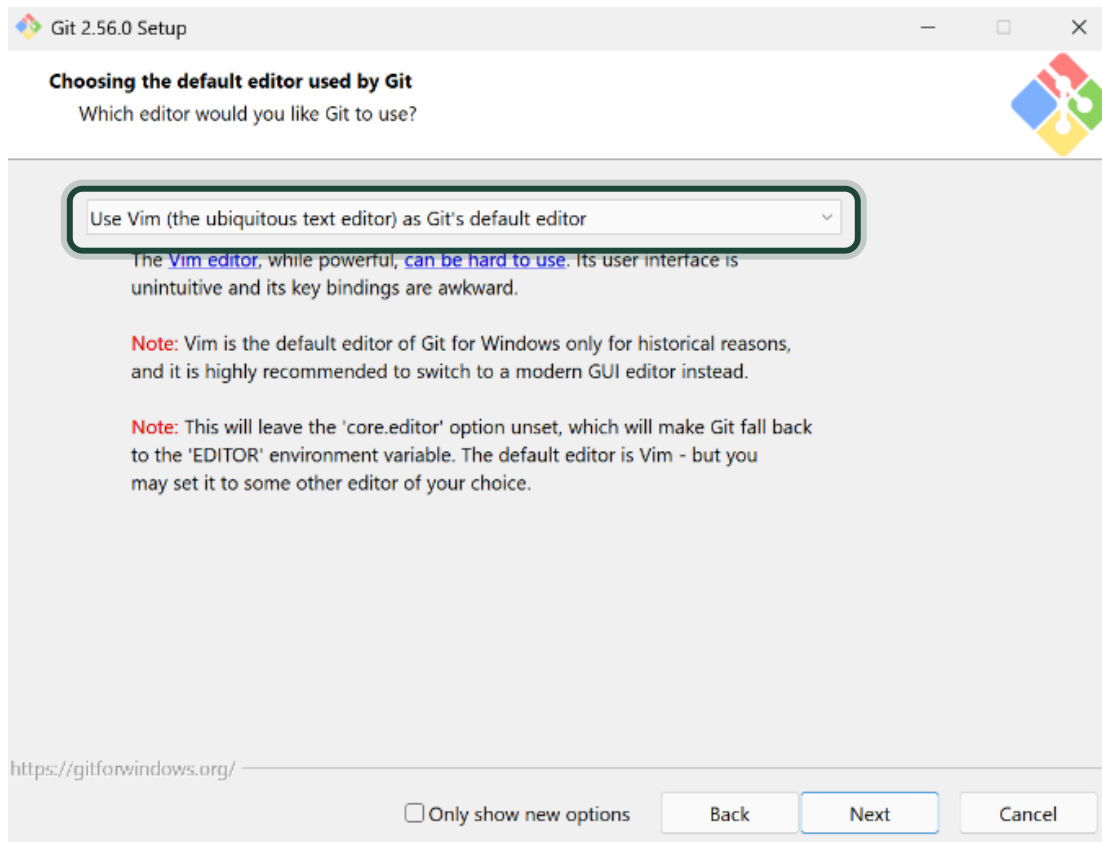
6 마지막 화면 → 체크 두 개 모두 해제 → Finish



처음 4개 화면 중 구성 요소 화면

Git 설치

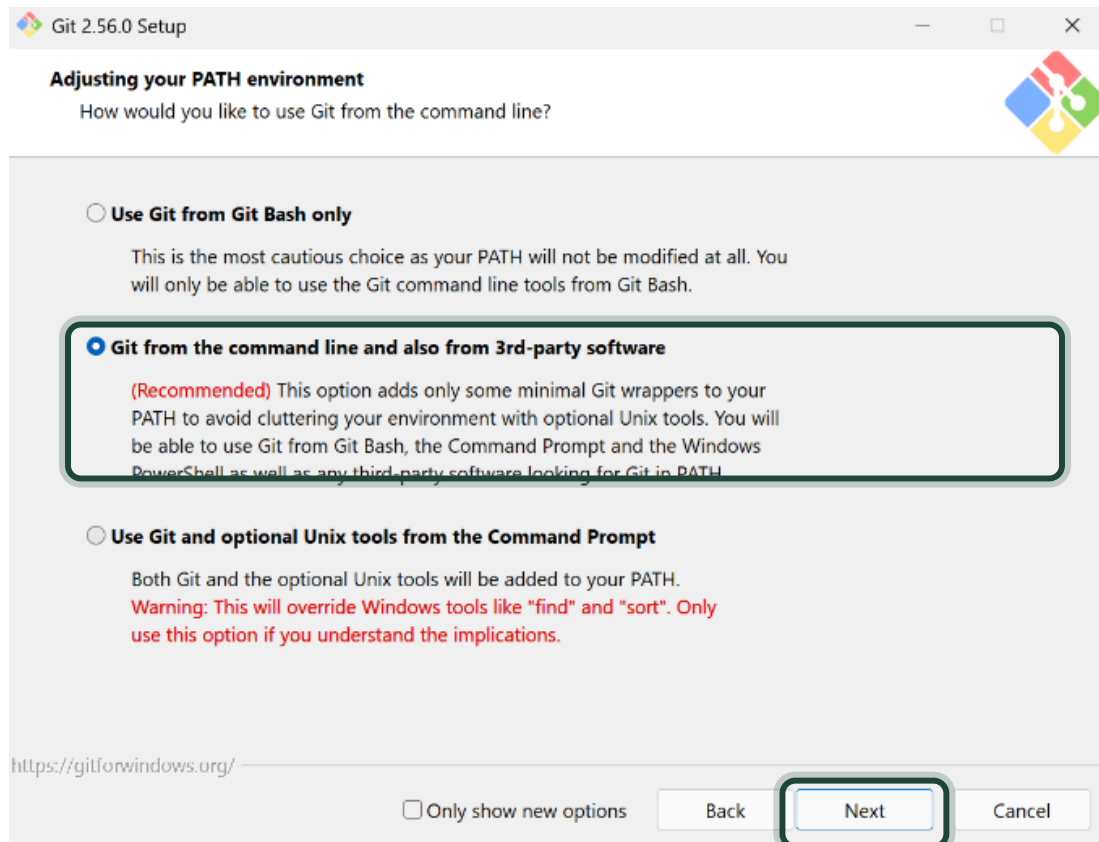
- 1 git-scm.com/install/windows → Click here to download
- 2 내려받은 **Git-2.56.0-64-bit.exe** 실행 → 확인 창 예 → 처음 4개 화면 그대로 Next
라이선스 · 설치 위치 · 구성 요소 · 시작 메뉴 폴더
- 3 편집기 화면 → 목록에서 **Use Notepad as Git's default editor** → Next
처음엔 Vim · 다음 화면(브랜치 이름)은 그대로 Next
- 4 PATH 화면 → 가운데 (Recommended) 그대로 → Next
- 5 나머지 7개 화면 그대로 Next → 마지막에 Install
- 6 마지막 화면 → 체크 두 개 모두 해제 → Finish



편집기 화면 · 처음엔 Vim

Git 설치

- 1 git-scm.com/install/windows → Click here to download
- 2 내려받은 **Git-2.56.0-64-bit.exe** 실행 → 확인 창 예 → 처음 4개 화면 그대로 Next
라이선스 · 설치 위치 · 구성 요소 · 시작 메뉴 폴더
- 3 편집기 화면 → 목록에서 **Use Notepad as Git's default editor** → Next
처음엔 Vim · 다음 화면(브랜치 이름)은 그대로 Next
- 4 **PATH 화면** → 가운데 **(Recommended)** 그대로 → Next
- 5 나머지 7개 화면 그대로 Next → 마지막에 Install
- 6 마지막 화면 → 체크 두 개 모두 해제 → Finish



PATH 화면 · 가운데가 처음부터 골라져 있다

Git 설치

1 git-scm.com/install/windows → Click here to download

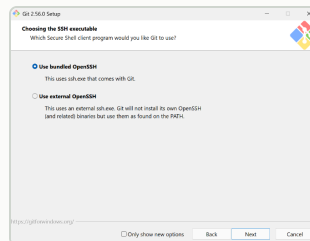
2 내려받은 **Git-2.56.0-64-bit.exe** 실행 → 확인 창 예 → 처음 4개 화면 그대로 Next
라이선스 · 설치 위치 · 구성 요소 · 시작 메뉴 폴더

3 편집기 화면 → 목록에서 **Use Notepad as Git's default editor** → Next
처음엔 Vim · 다음 화면(브랜치 이름)은 그대로 Next

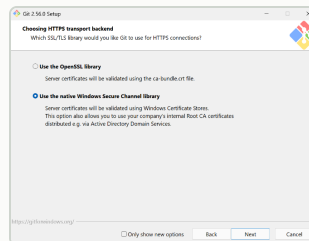
4 PATH 화면 → 가운데 **(Recommended)** 그대로 → Next

5 나머지 7개 화면 그대로 Next → 마지막에 **Install**

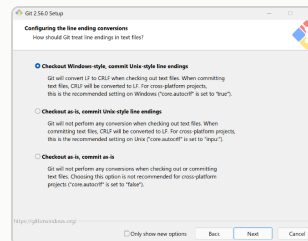
6 마지막 화면 → 체크 두 개 모두 해제 → **Finish**



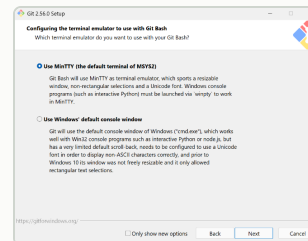
1 SSH



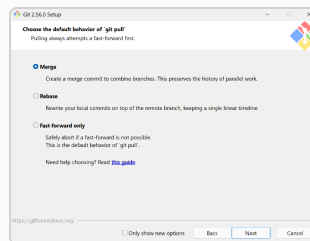
2 HTTPS



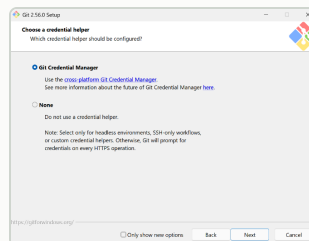
3 줄바꿈



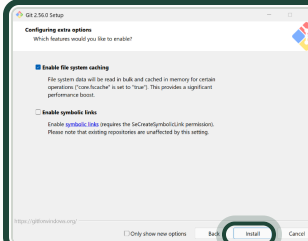
4 터미널



5 git pull



6 로그인 정보

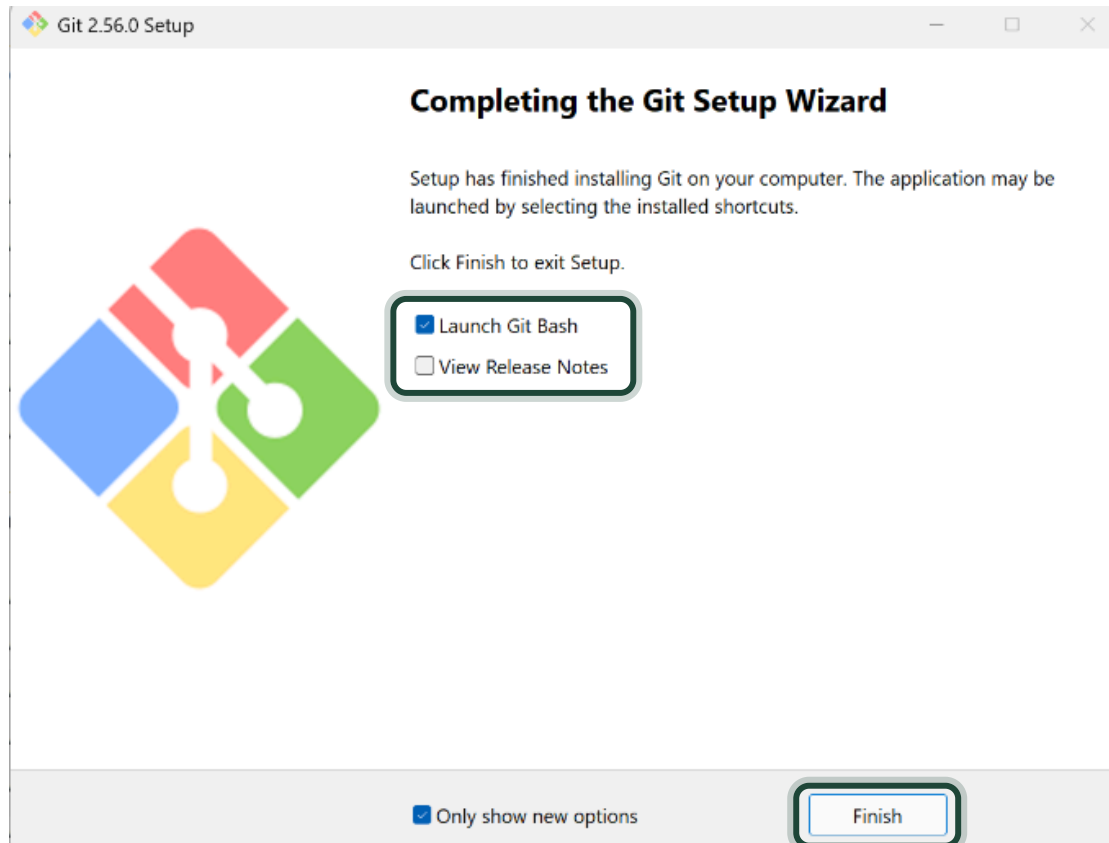


7 추가 옵션 → Install

고르는 것 없이
모두 그대로 Next
7번째에서 Install

Git 설치

- 1 git-scm.com/install/windows → Click here to download
- 2 내려받은 **Git-2.56.0-64-bit.exe** 실행 → 확인
창 예 → 처음 4개 화면 그대로 Next
라이선스 · 설치 위치 · 구성 요소 · 시작 메뉴 폴더
- 3 편집기 화면 → 목록에서 **Use Notepad as Git's default editor** → Next
처음엔 Vim · 다음 화면(브랜치 이름)은 그대로 Next
- 4 PATH 화면 → 가운데 (**Recommended**) 그대로 → Next
- 5 나머지 7개 화면 그대로 Next → 마지막에 **Install**
- 6 마지막 화면 → 체크 두 개 모두 해제 → **Finish**



Git 마지막 화면 · 체크 상태는 PC마다 다를 수 있다

명령 한 줄로 설치하기 - winget

1 PowerShell을 연다

winget은 따로 여는 프로그램이 아니라 이 창에 입력하는 명령.
여는 법은 먼저 확인과 같다

2 파이썬 - 아래 한 줄을 그대로 입력 → Enter

```
winget install --id Python.Python.3.13  
-e --source winget --scope user
```

3 Successfully installed(설치 성공)가 나올 때까지 기다린다

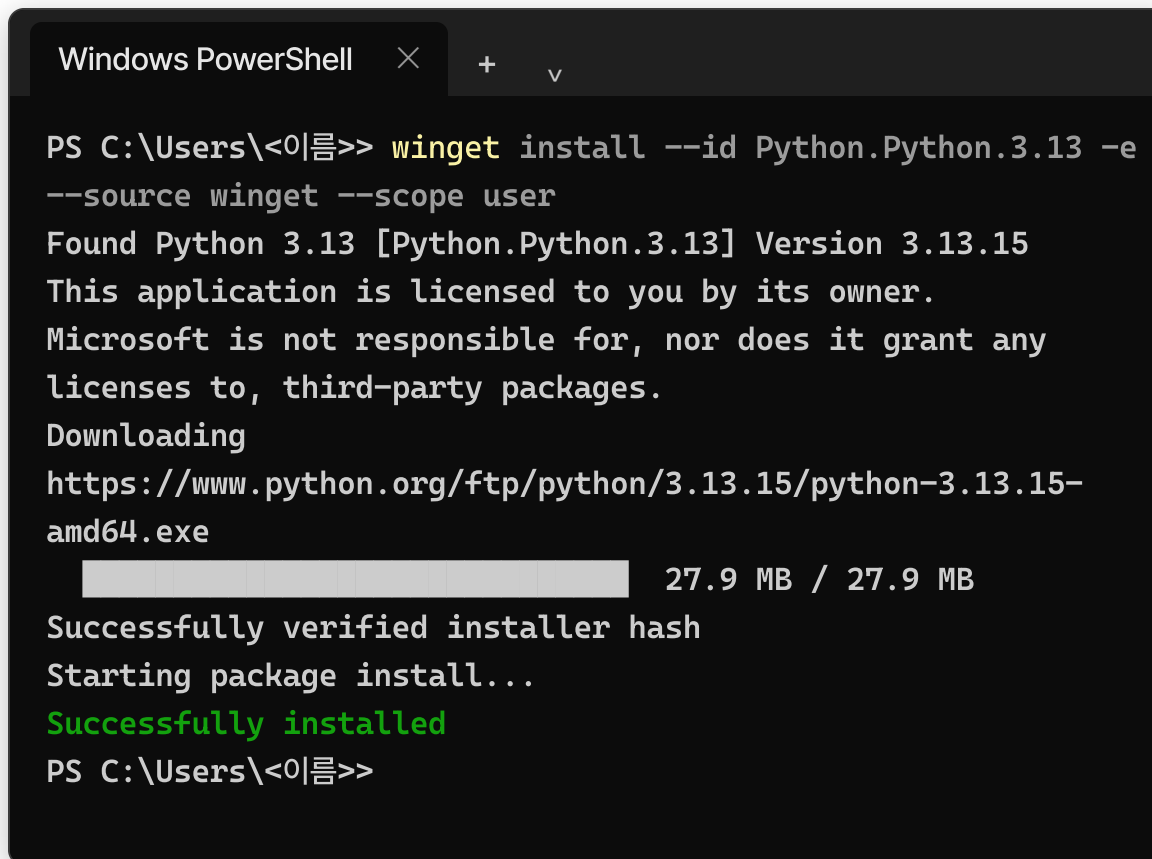
1 ~ 2분 · 그동안 창을 닫지 않는다

4 Git - 아래 한 줄을 입력 → Enter

```
winget install --id Git.Git -e --source  
winget
```

"앱이 디바이스를 변경하도록 허용" 창이 뜨면 예

5 다시 Successfully installed → 끝 · 창을 닫는다



```
Windows PowerShell  ×  +  v  
  
PS C:\Users\<이름> winget install --id Python.Python.3.13 -e  
--source winget --scope user  
Found Python 3.13 [Python.Python.3.13] Version 3.13.15  
This application is licensed to you by its owner.  
Microsoft is not responsible for, nor does it grant any  
licenses to, third-party packages.  
Downloading  
https://www.python.org/ftp/python/3.13.15/python-3.13.15-  
amd64.exe  
27.9 MB / 27.9 MB  
Successfully verified installer hash  
Starting package install...  
Successfully installed  
PS C:\Users\<이름>
```

영어 표시 기준(한국어로 나올 수 있다) · 버전은 winget 기준(2026.10.4)

명령 한 줄로 설치하기 - winget

1 PowerShell을 연다

winget은 따로 여는 프로그램이 아니라 이 창에 입력하는 명령.
여는 법은 먼저 확인과 같다

2 파이썬 - 아래 한 줄을 그대로 입력 → Enter

```
winget install --id Python.Python.3.13
-e --source winget --scope user
```

3 Successfully installed(설치 성공)가 나올 때까지 기다린다

1 ~ 2분 · 그동안 창을 닫지 않는다

4 Git - 아래 한 줄을 입력 → Enter

```
winget install --id Git.Git -e --source  
winget
```

"앱이 디바이스를 변경하도록 허용" 창이 뜨면 예

5 다시 Successfully installed → 끝 · 창을 닫는다

```
Windows PowerShell  X + v

PS C:\Users\<>이름>> winget install --id Git.Git -e --source
winget

Found Git [Git.Git] Version 2.55.0.5

This application is licensed to you by its owner.
Microsoft is not responsible for, nor does it grant any
licenses to, third-party packages.

Downloading https://github.com/git-for-
windows/git/releases/download/v2.55.0.windows.5/Git-
2.55.0.5-64-bit.exe

68.4 MB / 68.4 MB

Successfully verified installer hash
Starting package install...

Successfully installed

PS C:\Users\<>이름>>
```

영어 표시 기준(한국어로 나올 수 있다) · 버전은 winget 기준(2026.10.4)

설치 확인 - 터미널에서 버전이 나오면 끝

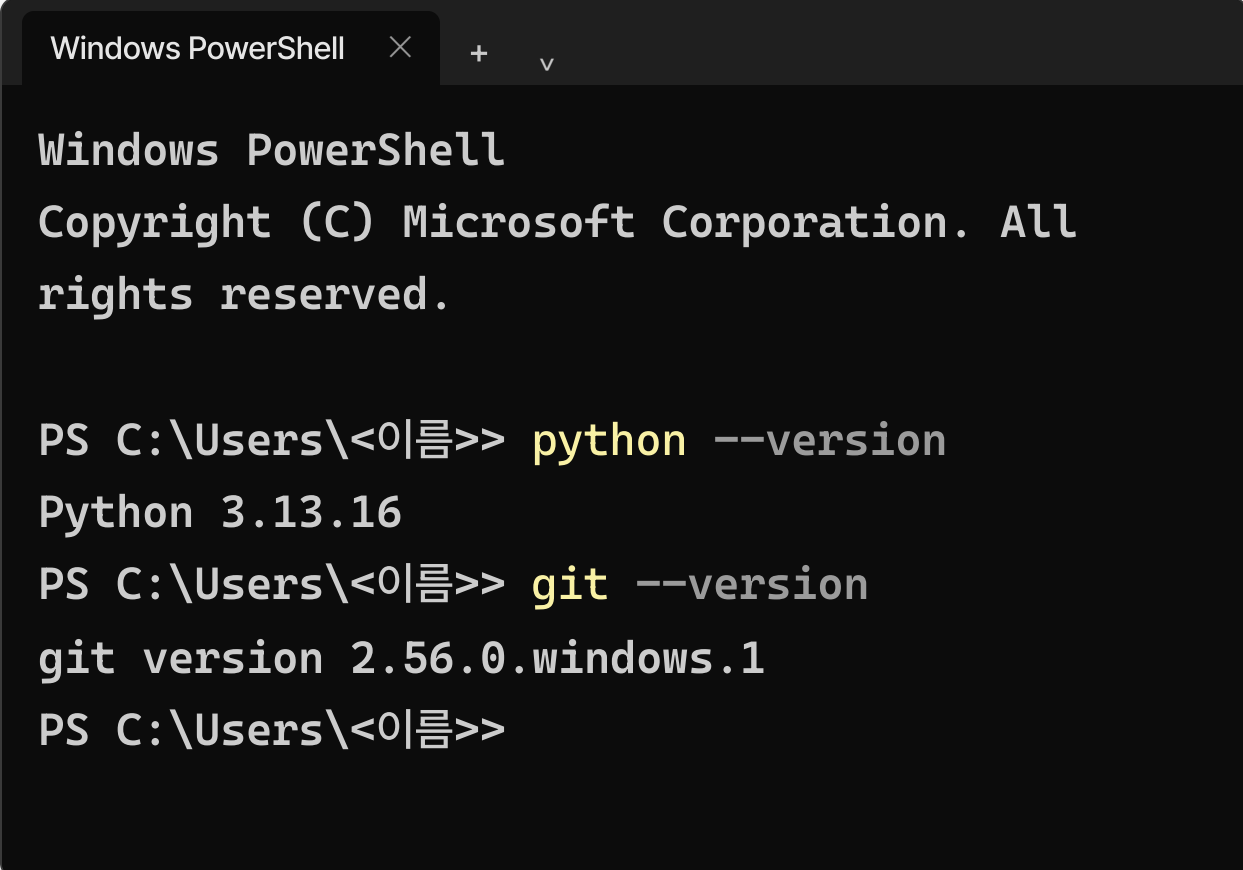
- 1 열려 있는 PowerShell 창을 모두 닫는다
창은 열릴 때 설치된 프로그램 목록을 읽는다 - 설치 전에 연 창은 새 프로그램을 모른다
- 2 시작 버튼 → powershell 입력 → Windows PowerShell로 새로 연다

- 3 아래 두 줄을 한 줄씩 입력 → 줄마다 Enter

```
python --version  
git --version
```

- 4 버전 숫자가 나오면 설치 완료
처음부터 있던 분은 먼저 확인 때와 같은 숫자

- 5 클로드 앱도 완전히 종료 → 다시 켜기
작업 표시줄 오른쪽 ^ → 클로드 아이콘 오른쪽 클릭 → 종료



```
Windows PowerShell  ×  +  v  
  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All  
rights reserved.  
  
PS C:\Users\<이름> python --version  
Python 3.13.16  
PS C:\Users\<이름> git --version  
git version 2.56.0.windows.1  
PS C:\Users\<이름>
```

설치 파일로 설치한 경우(winget이면 Python 3.13.15 · git version 2.55.0.windows.5)



참고 · 설치

앞으로 - 목적에 따라 더 필요해지는 것

바이브코딩

웹 화면 · 작은 프로그램

Node.js

웹 프로그램을 내 PC에서 실행하는 프로그램

GitHub

만든 코드 보관 · 공유

Vercel 등 배포 서비스

만든 화면을 인터넷에 올리기

VS Code

코드를 직접 열어 보기

콘텐츠 제작

동영상 · 이미지 · 슬라이드

FFmpeg

영상 자르기 · 붙이기 · 자막 · 변환

Whisper

음성을 자막 · 글로 받아쓰기

Pillow

이미지 크기 · 합치기 · 글자 넣기

Node.js

코드로 영상 · 슬라이드 만들 때

더 깊은 업무 자동화

PDF · 파일 · 외부 연결

pdfplumber · pypdf

PDF 표 뽑기 · 합치기 · 나누기

Tesseract OCR

스캔본 · 사진의 글자 읽기

xlrd · msoffcrypto-tool

구형 .xls · 암호 걸린 엑셀 열기

uv · Node.js

MCP 서버를 직접 붙일 때

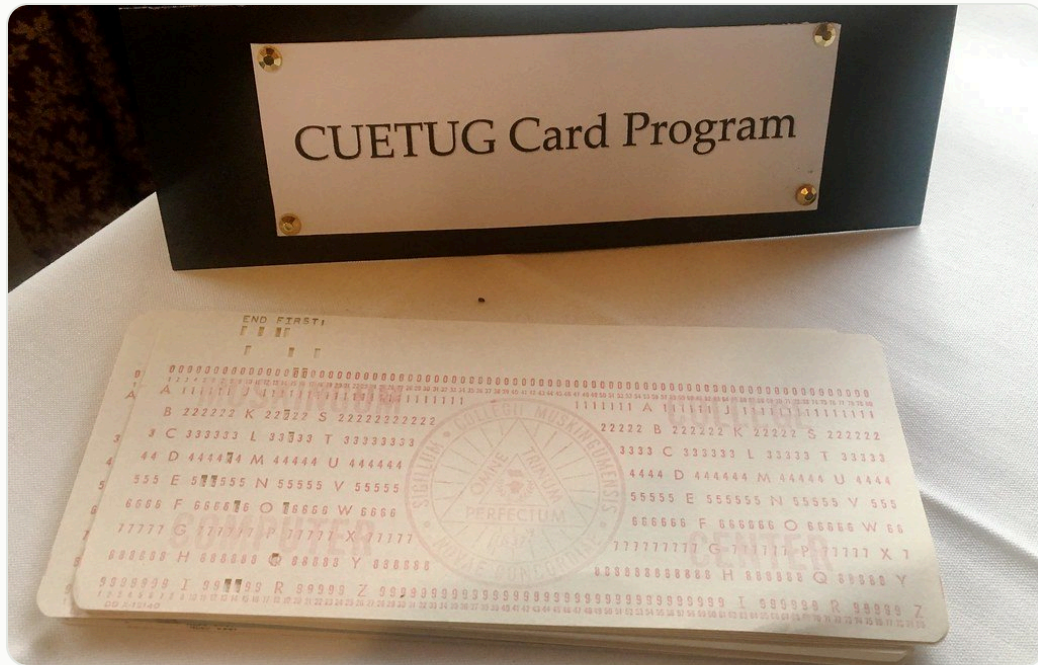
미리 깔 필요 없다 - 필요해지면 클로드가 먼저 설치를 제안한다

02

파이썬(Python)이란?

기계어 · 프로그래밍 언어 · 라이브러리

기계어 - 컴퓨터는 0과 1만 알아듣는다



▲ 천공 카드 - 초창기에는 구멍을 뚫어 명령을 넣었다

10110000 01100001

CPU에게 "이 칸에 97을 넣어라" - 명령 하나

기계어 = CPU가 실제로 실행하는 0과 1의 명령

사람은 읽을 수도, 쓸 수도 없다

사람이 기계어를 쓸 수는 없다 → 사람이 읽을 수 있는 명령문이 필요했다 = 프로그래밍 언어

프로그래밍 언어 - 사람이 쓴 명령을 기계어로 옮겨 실행한다

1 사람이 쓴다

프로그래밍 언어

```
print(1200 + 1350 + 830)
```

영어 단어와 기호의 명령문

사람이 읽을 수 있고, 뜻이 하나로 정해져 있다



2 옮긴다

번역 프로그램

파이썬 프로그램이 코드를 한 줄씩 읽어 CPU가 알아듣는 명령으로 바꾼다

방금 설치한 그 프로그램



3 실행한다

기계어 · CPU

```
10110000 01100001 ...
```

CPU가 0과 1의 명령을 실행

화면에 → **3380**

원래 개발자가 하던 일 = **프로그래밍 언어로 컴퓨터에게 명령하는 코드를 쓰는 것** - 이제는 그 코드를 AI가 쓴다

프로그래밍 언어의 종류 - 파이썬이 그 중 하나

지금 가장 많이 쓴다

Python

개발 · 데이터 · AI · 업무
자동화

C · C++

운영체제 · 게임 · 기계 제어

Java

안드로이드 앱

C#

윈도우 프로그램 · 게임

JavaScript

웹 페이지 화면

SQL

데이터베이스에서 자료 조회 -
ERP 조회 뒤에서 돈다

R

통계 분석

엑셀에서 본 그것

VBA

엑셀 매크로 편집 창 의 코드

가장 많이 쓴다 = TIOBE 지수 1위(2026년 9월)

지금 가장 많이 쓰이는 언어는 **파이썬** - 문법이 직관적이고 데이터·AI 생태계가 가장 풍부하기 때문

파이썬으로 할 수 있는 일

웹 개발

웹사이트 · 웹 서비스의
서버를 만든다

회원 가입 · 로그인 처리

주문 · 결제 처리

데이터 분석

대량의 데이터를 집계하고
그래프로 그린다

수백만 줄 거래 집계

추세 · 분포 그래프

머신러닝 · 딥러닝

데이터로 예측 모델 · AI
모델을 학습시킨다

매출 · 수요 예측

이미지 · 문장 인식

우리가 설치한 이유

엑셀 · 오피스 문서

엑셀 · 워드 · PPT · PDF
파일을 읽고 만든다

결산 파일 만들기 · 고치기

보고서 · 의견서 문서

분야마다 쓰는 패키지가 따로 있다 - 라이브러리

라이브러리 - 분야마다 쓰는 패키지가 따로 있다

라이브러리 = 남이 만들어 둔 기능 묶음
필요한 것을 불러다 쓴다

```
pip install openpyxl
```

처음 한 번 - 패키지를 내 PC에 설치

```
import openpyxl
```

코드 맨 위 - 불러와 쓰기

분야	대표 라이브러리
웹 개발	Django Flask
데이터 분석	pandas matplotlib
머신러닝 · 딥러닝	scikit-learn PyTorch TensorFlow
엑셀 · 오피스 문서	openpyxl xlwings python-docx python-pptx

03

Git이란?

시점 기록 · 커밋 · 푸시 · 브랜치

Git - 다른 이름으로 저장하는 대신, 시점을 기록한다

지금 - 고칠 때마다 다른 이름으로 저장

XLSX 결산표_v1.xlsx

XLSX 결산표_v2.xlsx

XLSX 결산표_v3.xlsx

XLSX 결산표_최종.xlsx

XLSX 결산표_진짜최종.xlsx

무엇이 바뀌었는지는 하나씩 열어 비교해야 안다



Git - 파일은 하나, Ctrl+S로 덮어써도 된다

XLSX 결산표.xlsx

- 초안 기록 1
- 대손충당금 검토 반영 기록 2
- 검토 의견 반영 기록 3

① 언제든지 그 시점으로 돌아간다

② 두 시점 사이에 무엇이 바뀌었는지 본다

남겨 두고 싶은 시점마다 기록 = 커밋 - 저장은 평소처럼 Ctrl+S

커밋하면 저장소에, 푸시하면 GitHub에

내 PC

작업 폴더

내가 파일을 고치는 곳

XLSX 결산표.xlsx

MD 결산메모.md

커밋
기록
→

내 PC

저장소 숨은 .git 폴더

커밋한 기록이 쌓인다 · 인터넷
없이

b9a4e77 검토 의견 반영

0909443 대손충당금 검토 반영

7ccde9d 초안

푸시
올리기
→

←
풀
불러오기

인터넷

GitHub

올려 둔 기록

b9a4e77 검토 의견 반영

0909443 대손충당금 검토
반영

7ccde9d 초안

백업 · 다른 PC · 동료와 공유

클로드에게 이렇게 시키면 - 저장소 · 커밋 · 푸시 · 풀

"이 폴더를 저장소로 만들어 줘" 저장소

클로드가 `git init` 으로 바꿔 실행 - 이 작업 폴더가 **Git**이 기록하는 폴더가 된다 · 숨은 `.git`이 생기고 그때부터 기록할 수 있다

작업 폴더마다 처음 한 번

"지금 상태를 커밋해 줘" 커밋

클로드가 `git add` · `git commit` 으로 바꿔 실행 - 지금 상태를 메모와 함께 **저장소**에 기록 · 그 시점으로 언제든지 돌아갈 수 있다

작업 한 단위가 끝났을 때 · 메모는 무슨 작업인지

"GitHub에 푸시해 줘" 푸시

클로드가 `git push` 로 바꿔 실행 - 저장소의 기록을 **GitHub**에 올린다 · 백업 · 다른 PC · 동료와 공유

GitHub 계정이 필요 · 고객 자료는 공개 저장소에 올리지 않는다 · 이 과정에서는 안 한다

"GitHub에서 풀해 줘" 풀

클로드가 `git pull` 로 바꿔 실행 - GitHub의 새 기록을 내 작업 폴더로 불러온다 · 다른 PC · 동료가 올린 것을 받는다

다른 곳에서 이어서 작업할 때 · 이 과정에서는 안 한다

Git 명령어는 몰라도 된다 - 말로 시키면 클로드 코드가 **Git 명령어로 바꿔 실행**한다 · 나는 언제 커밋할지 정한다 · 다음 장부터 실습 단위마다 커밋



참고 · Git이란

Git은 프로그램, GitHub는 사이트

Git

내 PC에 설치하는 프로그램

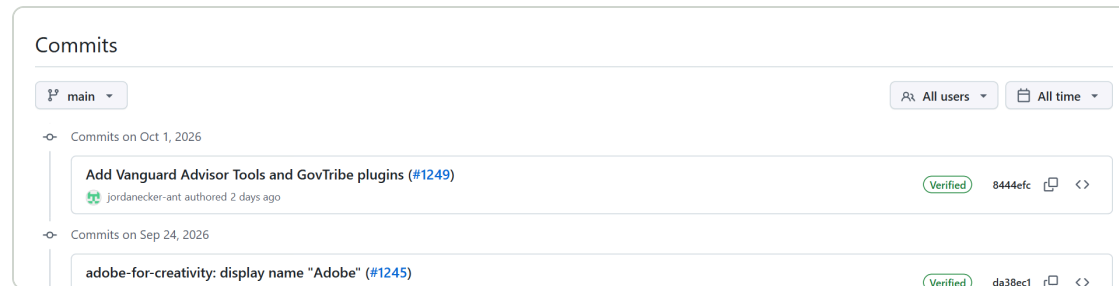
기록을 남기고 · 그 시점으로 돌아가고 · 바뀐 줄을 보여 준다

인터넷 없이도 된다

GitHub

Git 저장소를 올려 두는 사이트

백업 · 여러 PC · 동료와 공유 · 함께 고치기(PR)



▲ GitHub의 커밋 기록 - 한 줄이 커밋 하나

혼자 내 PC에서
Git만 - 커밋으로 기록

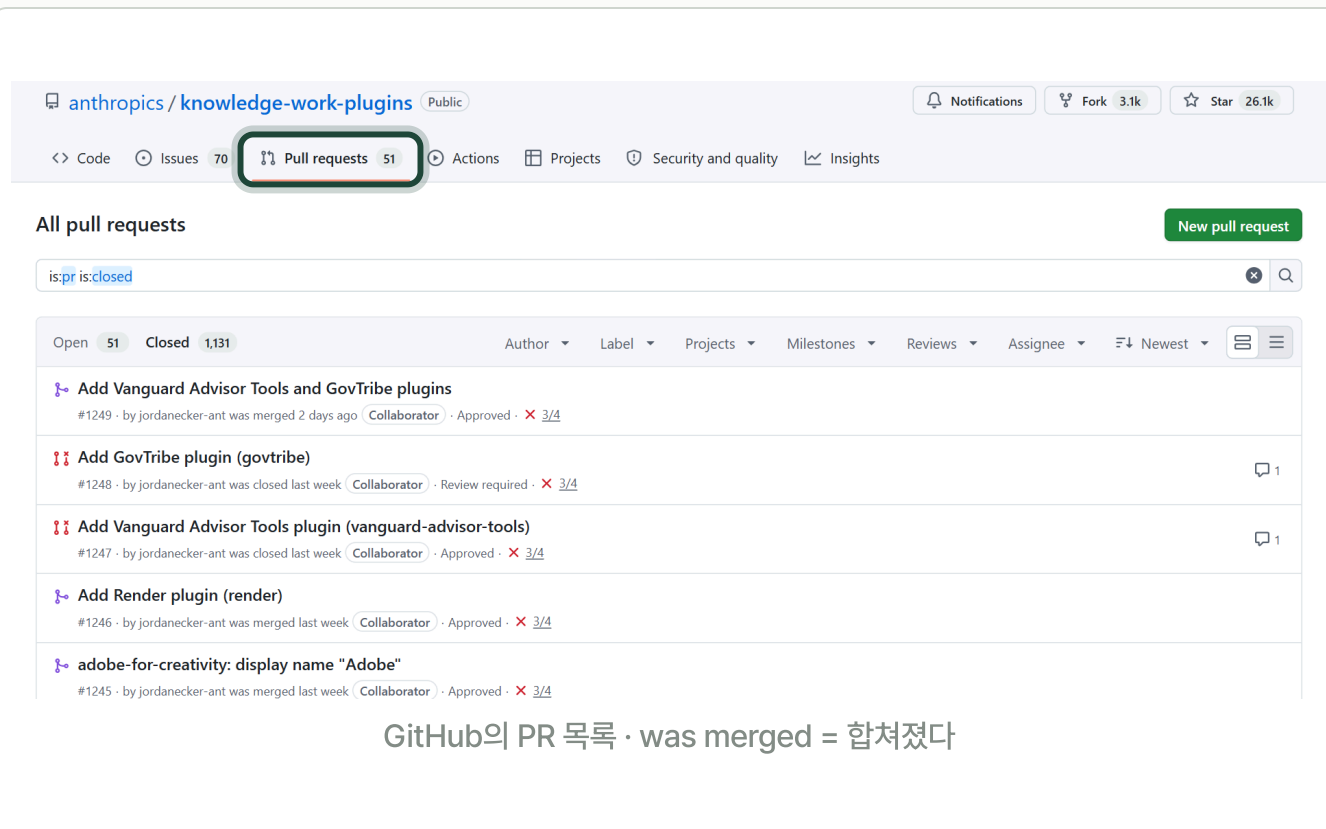
백업 · 집과 회사 PC
GitHub에 푸시 · 풀

여럿이 함께
GitHub + PR(다음 참고 쪽)

고객 자료가 든 폴더는 공개 저장소에 올리지 않는다 - 비공개 저장소 · 회사 정책 먼저

브랜치와 PR - 바꾼 내용을 합쳐 달라고 요청

- 1 브랜치 - 원본 줄기(main · master)는 그대로 두고 갈래를 만든다
- 2 갈래에서 고치고 커밋 → 푸시
- 3 PR(Pull Request) - "바꾼 내용을 main에 합쳐 주세요"
GitHub의 기능 · 무엇이 바뀌었는지 줄 단위로 보인다
- 4 검토 · 의견 → 합치기(merge)



지금은 몰라도 된다 - 혼자 쓸 때는 커밋만으로 충분 · 여럿이 함께 고칠 때 쓴다