AI Agent 전문가 양성과정

미디어 산업의 디지털 전환을 선도할 인공지능 에이전트 개발자 양성

LLM 기초

AI Agent 프로그래밍

심화 프로젝트

2025년 12월 ●일

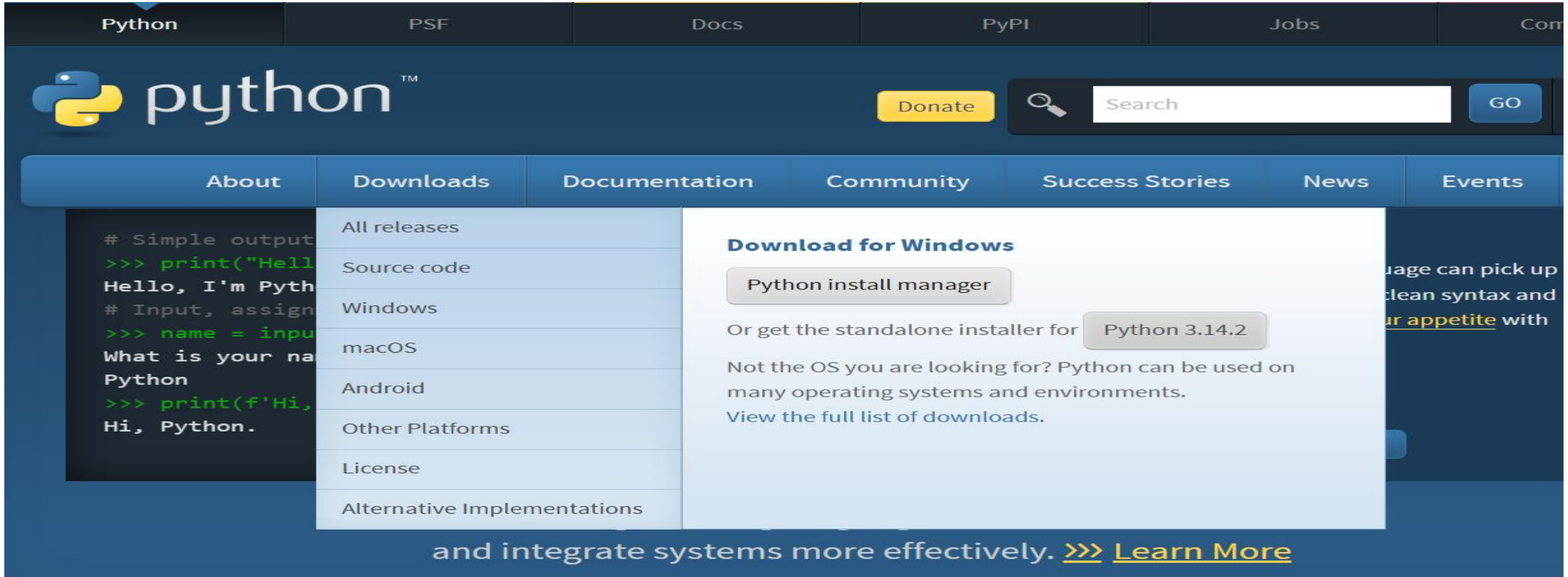
1. 프로그래밍 역량 강화 파이썬 설치

파이썬 다운로드

웹 브라우저에서 파이썬 공식 사이트에 접속한다.

https://www.python.org/downloads/

화면 상단의 Download 항목에서 운영체제에 맞는 설치 파일을 선택한다.
최신 기능 중심의 버전은 라이브러리 충돌이 발생할 수 있으므로, 실습에서는 Python 3.12.x 버전을 사용하는 것이 좋습니다. 이유는 안정화 버전으로 강력 추천합니다.



2. 프로그래밍 역량 강화 파이썬 설치

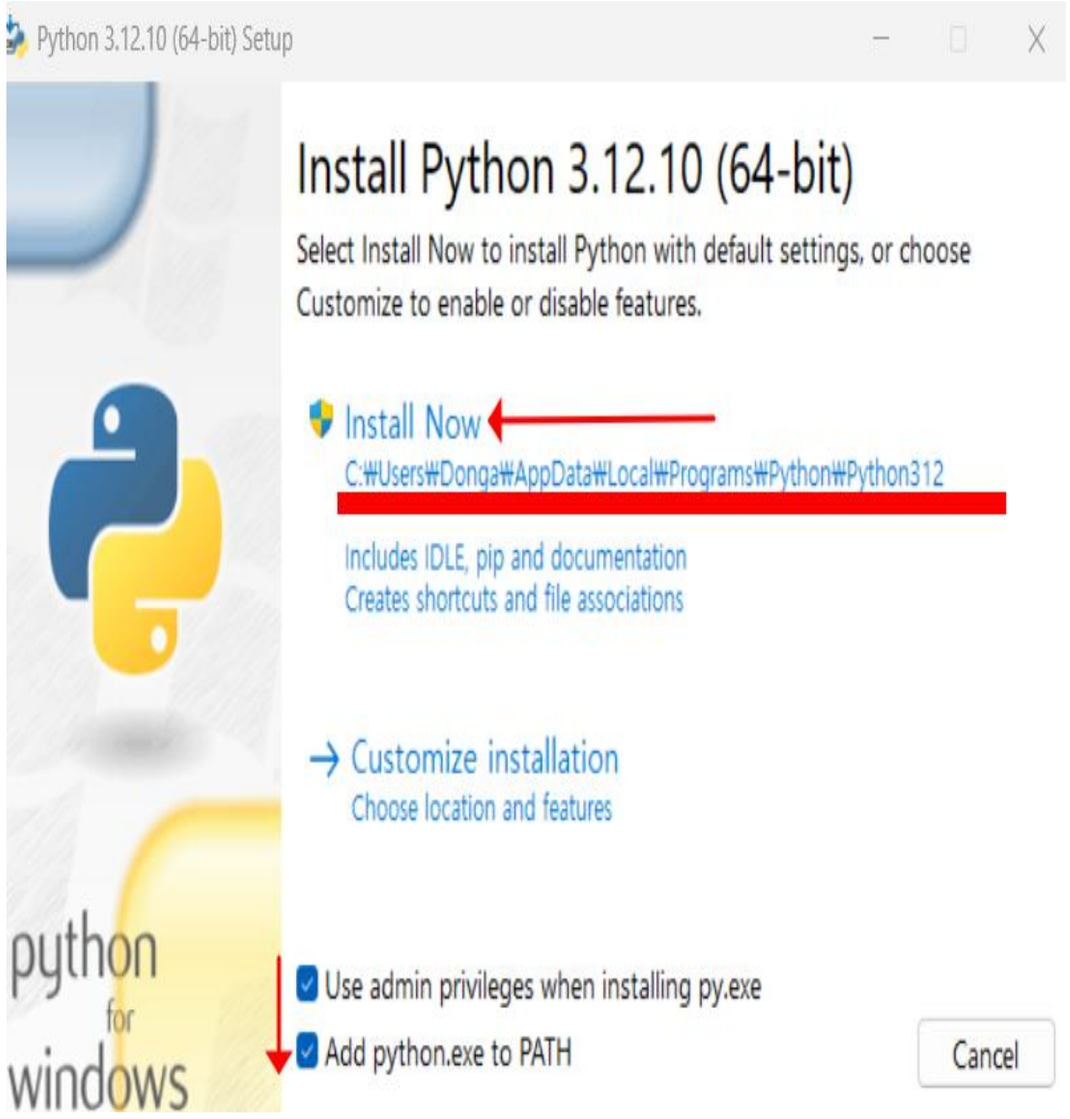
- Python 3.12.10 - April 8, 2025

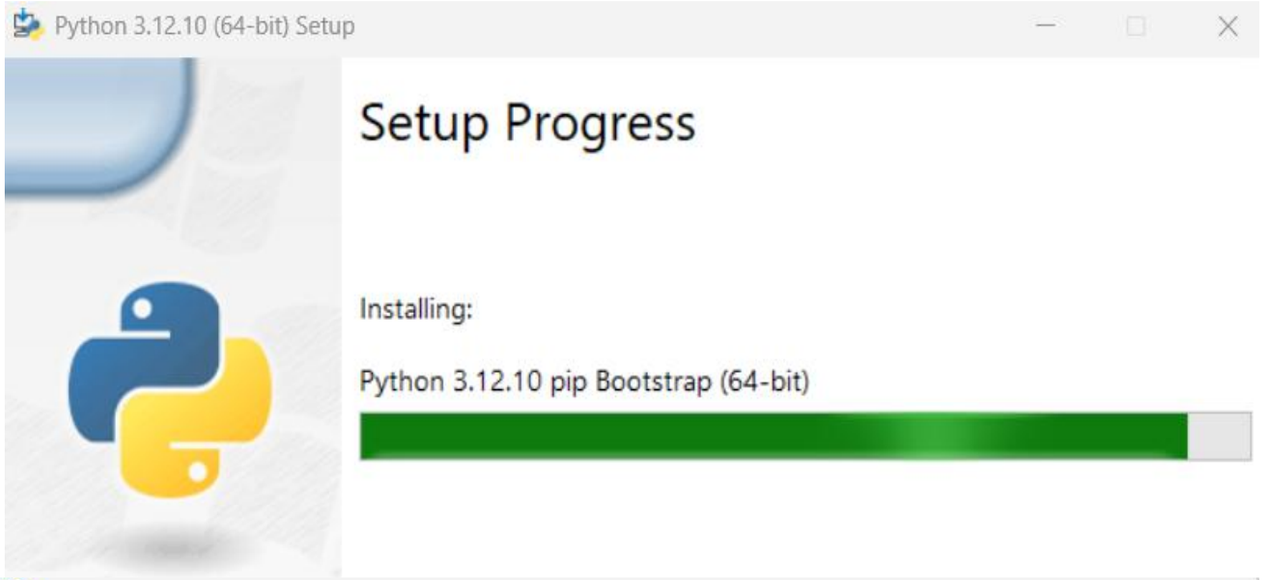
  Note that Python 3.12.10 *cannot* be used on Windows 7 or earlier.

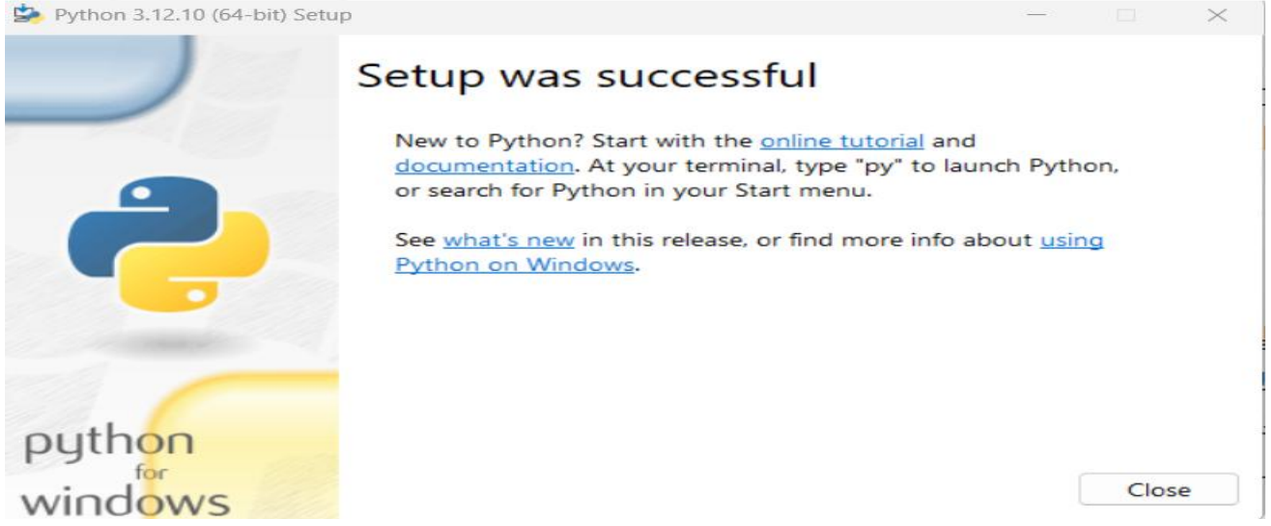
  - Download Windows installer (64-bit)
  - Download Windows installer (32-bit)
  - Download Windows installer (ARM64)
  - Download Windows embeddable package (64-bit)
  - Download Windows embeddable package (32-bit)
  - Download Windows embeddable package (ARM64)

> TMI
> 설치 파일을 대부분 64비트로 다운로드할 텐데, 파일 이름이 현재 날짜 기준으로 python-3. -amd64.exe 으로 되어있다. 끝에 amd64를 보고 "나는 인텔 CPU인데 왜 amd 파일이 다운되지?" 라는 의문이 생겨서 찾아보니 64비트 구조체를 amd에서 개발하여 amd64로 명명했다고 한다. 당황하지 말자(나만 당황했을 수도 있다.)

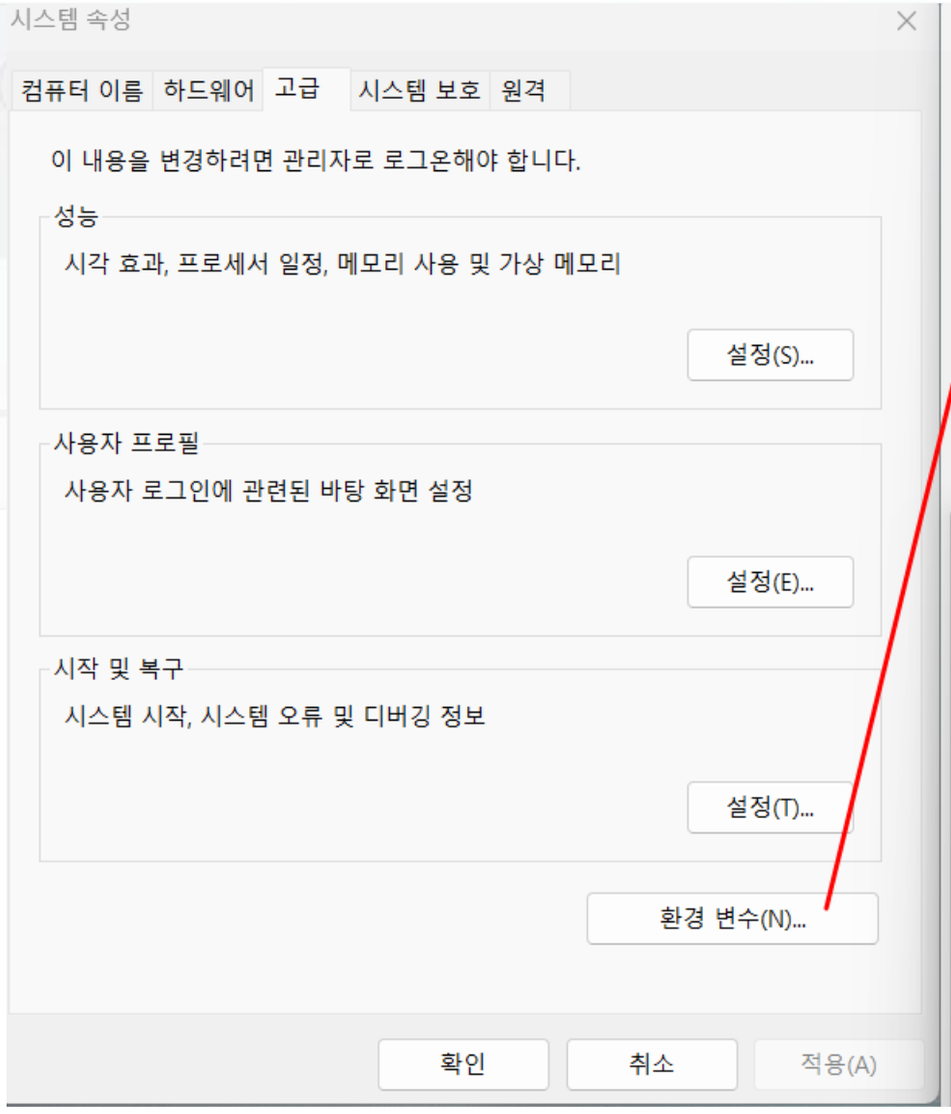
3. 프로그래밍 역량 강화 파이썬 설치

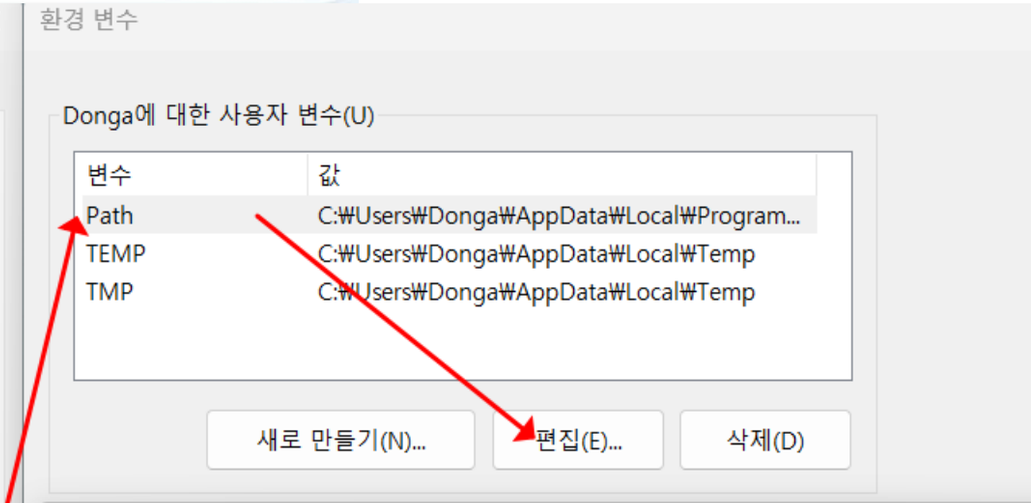


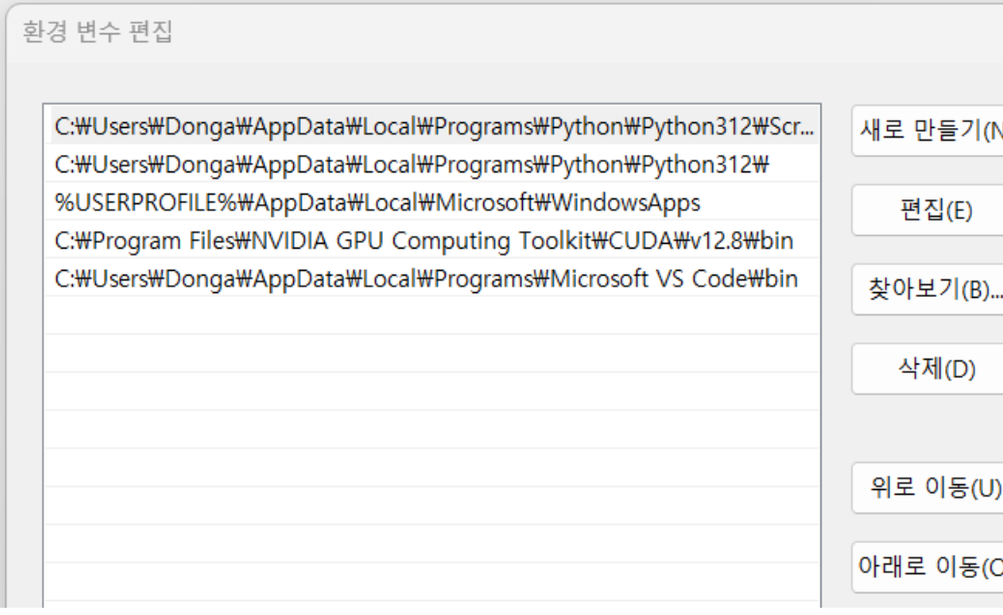




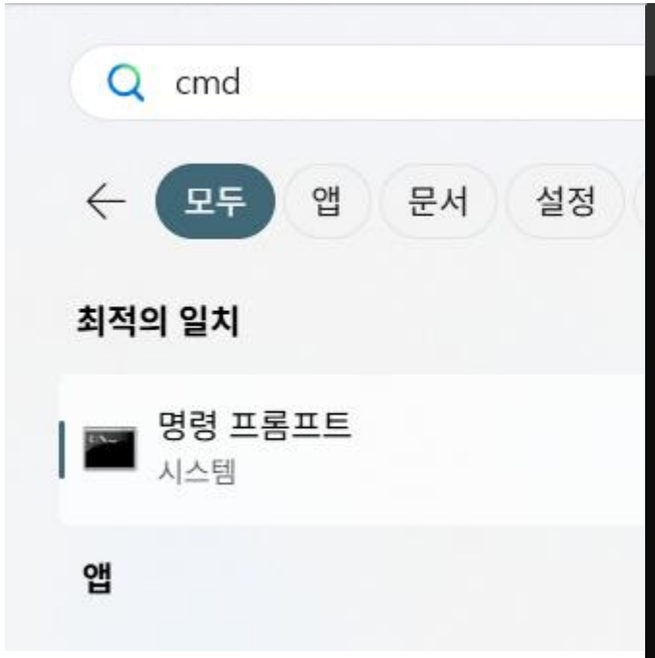
4. 프로그래밍 역량 강화 파이썬 설치







5. 프로그래밍 역량 강화 파이썬 설치



명령 프롬프트 - python

```
C:\Users\Donga>
C:\Users\Donga> python --version
Python 3.12.10

C:\Users\Donga>
C:\Users\Donga>python
Python 3.12.10 (tags/v3.12.10:0cc8128, Apr  8 2025, 12:21:36) [MSC v.1943 64 b
it (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> print(2+3)
5
>>> print("blue")
blue
>>>
>>> print('sky')
sky
>>>
>>> exit( ) #종료
```

6. 개발환경 VS code 환경구축

AI 프로그래밍을 시작하려면 먼저 코드를 작성하고 실행할 수 있는 환경을 준비해야 한다. 개발환경이란 단순히 파이썬을 설치하는 것이 아니라, 코드 편집기, 패키지 관리, 실행 콘솔이 함께 작동하는 작업 공간을 말한다. 이 절에서는 가장 많이 사용하는 두 환경인 VS Code와 Google Colab을 다룬다.

VS Code는 내 컴퓨터에서 프로젝트를 직접 관리하는 개발 도구로, 확장 기능을 추가하면 디버깅, 버전 관리, 가상환경 설정까지 가능해 실무에서 널리 쓰인다. 반면 Google Colab은 설치 없이 웹 브라우저에서 바로 파이썬을 실행하는 클라우드 환경으로, GPU 지원 덕분에 데이터 분석과 AI 실습에 특히 편리하다.

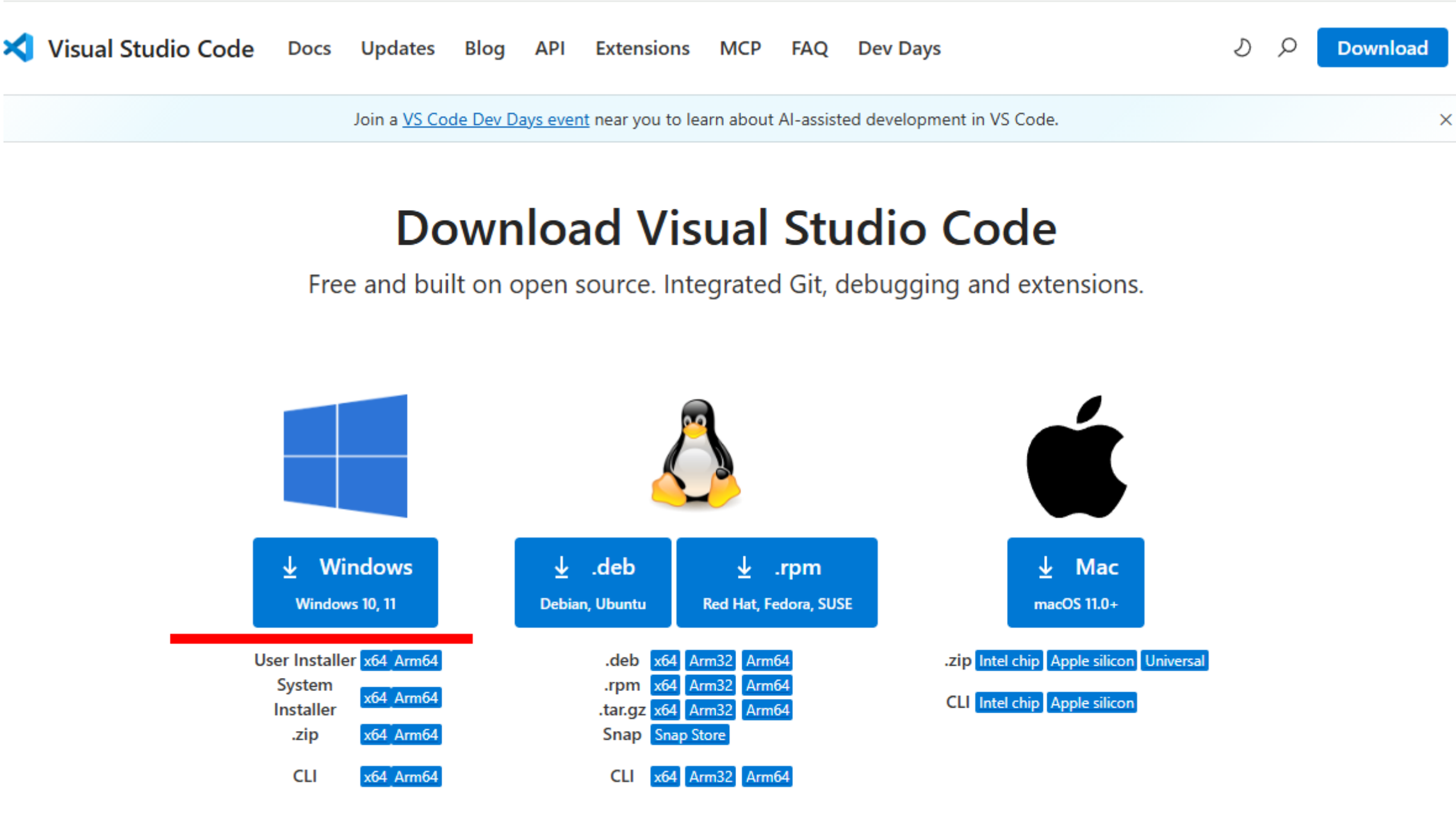
Colab에서도 간단한 코드를 실행해보면 더 효과적, VS Code로 프로젝트 구조를 만들며 실습해 나가는 것이 좋습니다. 두 환경을 모두 익히면 어떤 컴퓨터에서도 파이썬과 AI 실습을 안정적으로 수행할 수 있는 기본기가 완성된다.

특히 VS Code(Visual Studio Code)는 마이크로소프트에서 개발한 무료 코드 편집기이다. 가볍고 확장성이 뛰어나며 파이썬뿐 아니라 다양한 언어를 지원한다. AI 프로그래밍을 위해서는 기본 편집기만으로는 부족하므로 파이썬 관련 확장 기능을 추가해야 한다.

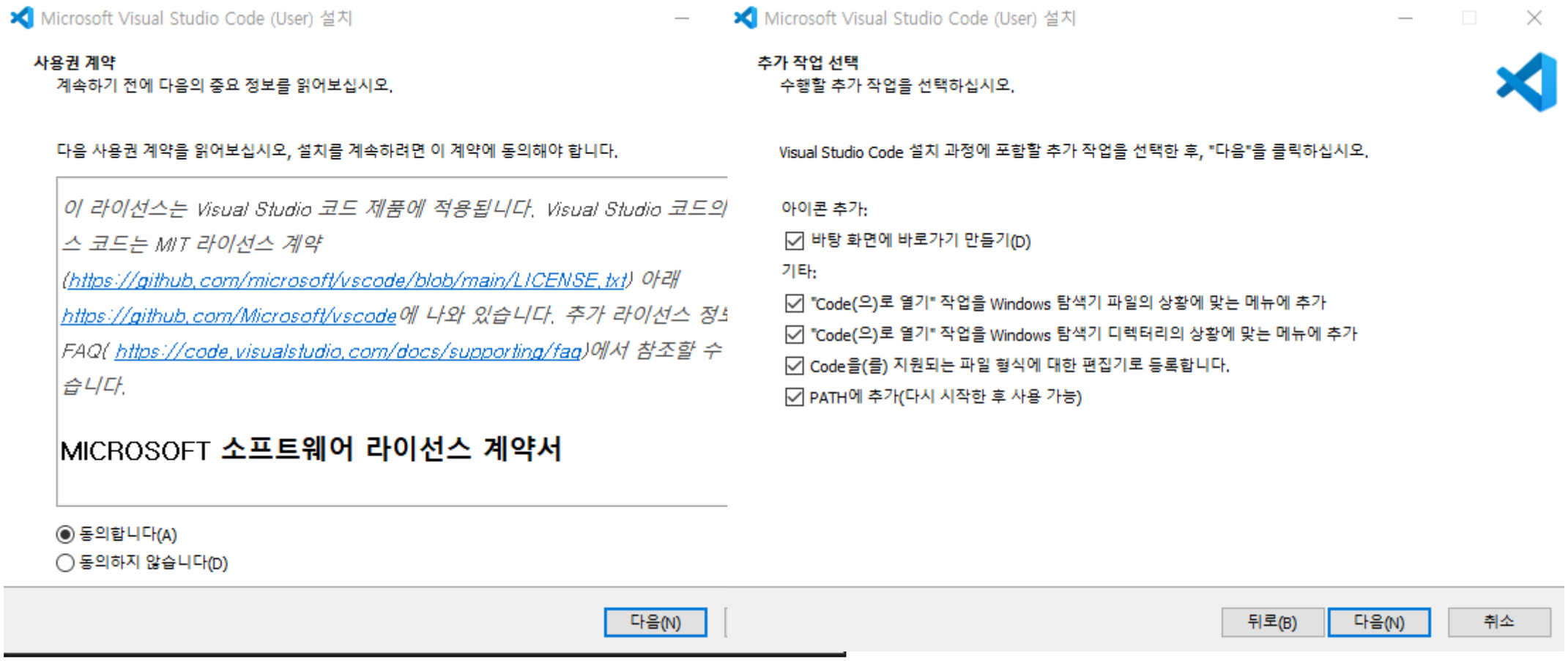
웹브라우저에서 VS code 공식 페이지 접속한다

https://code.visualstudio.com/

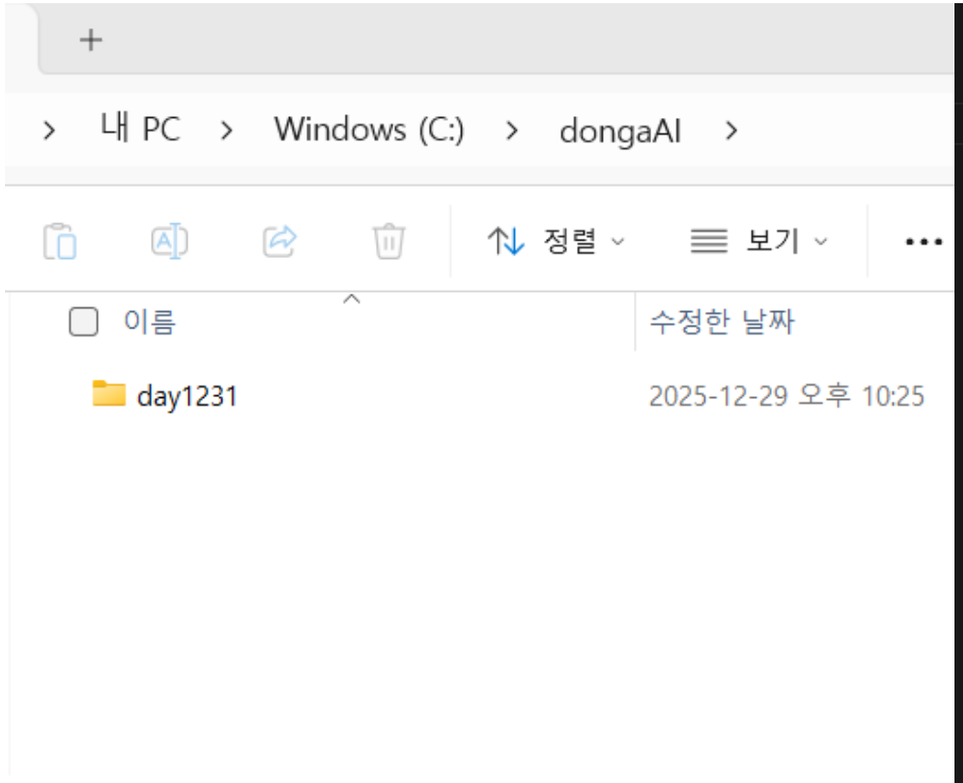
7. 프로그래밍 역량 강화 VS code 설치

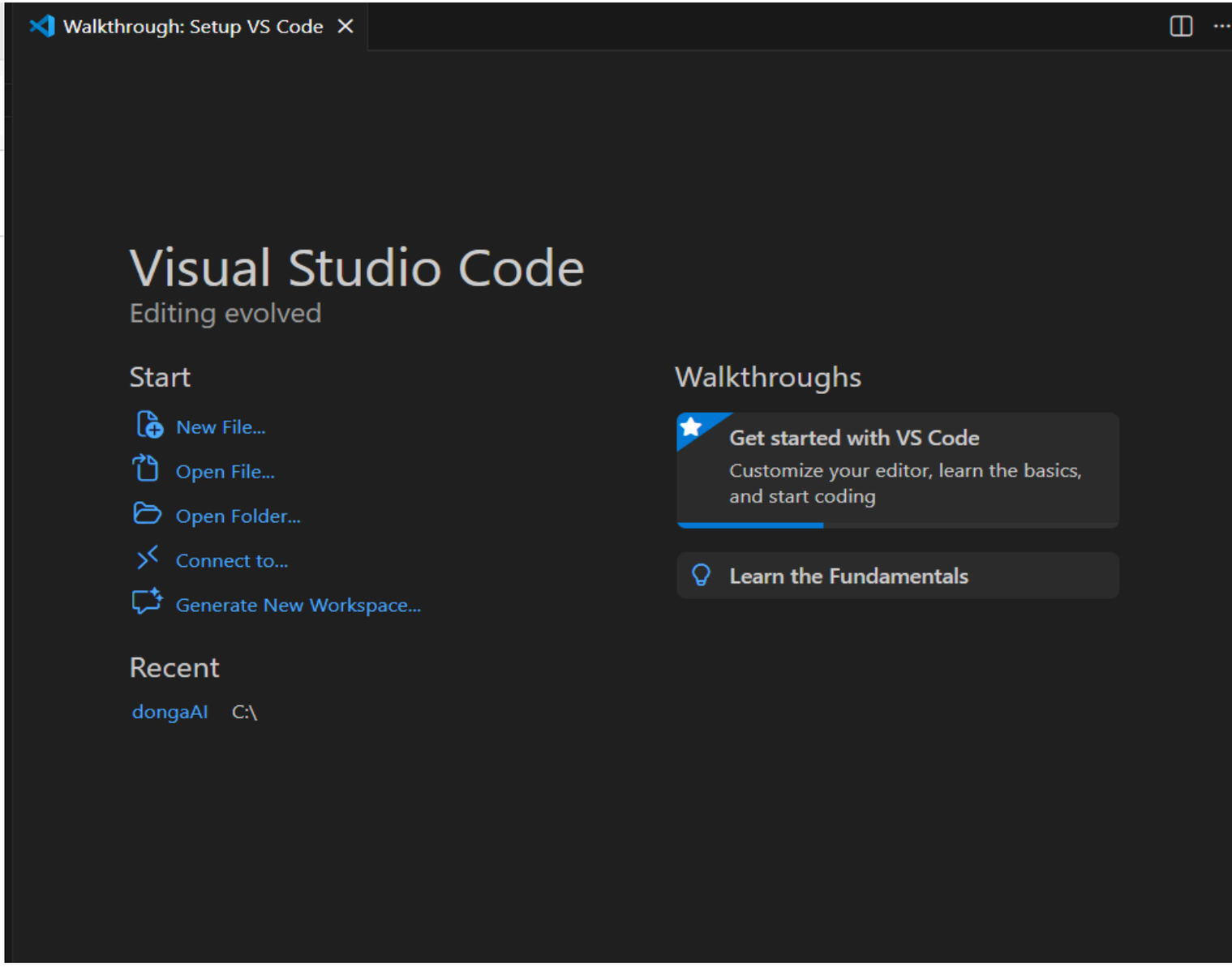


8. 프로그래밍 역량 강화 VS code 설치

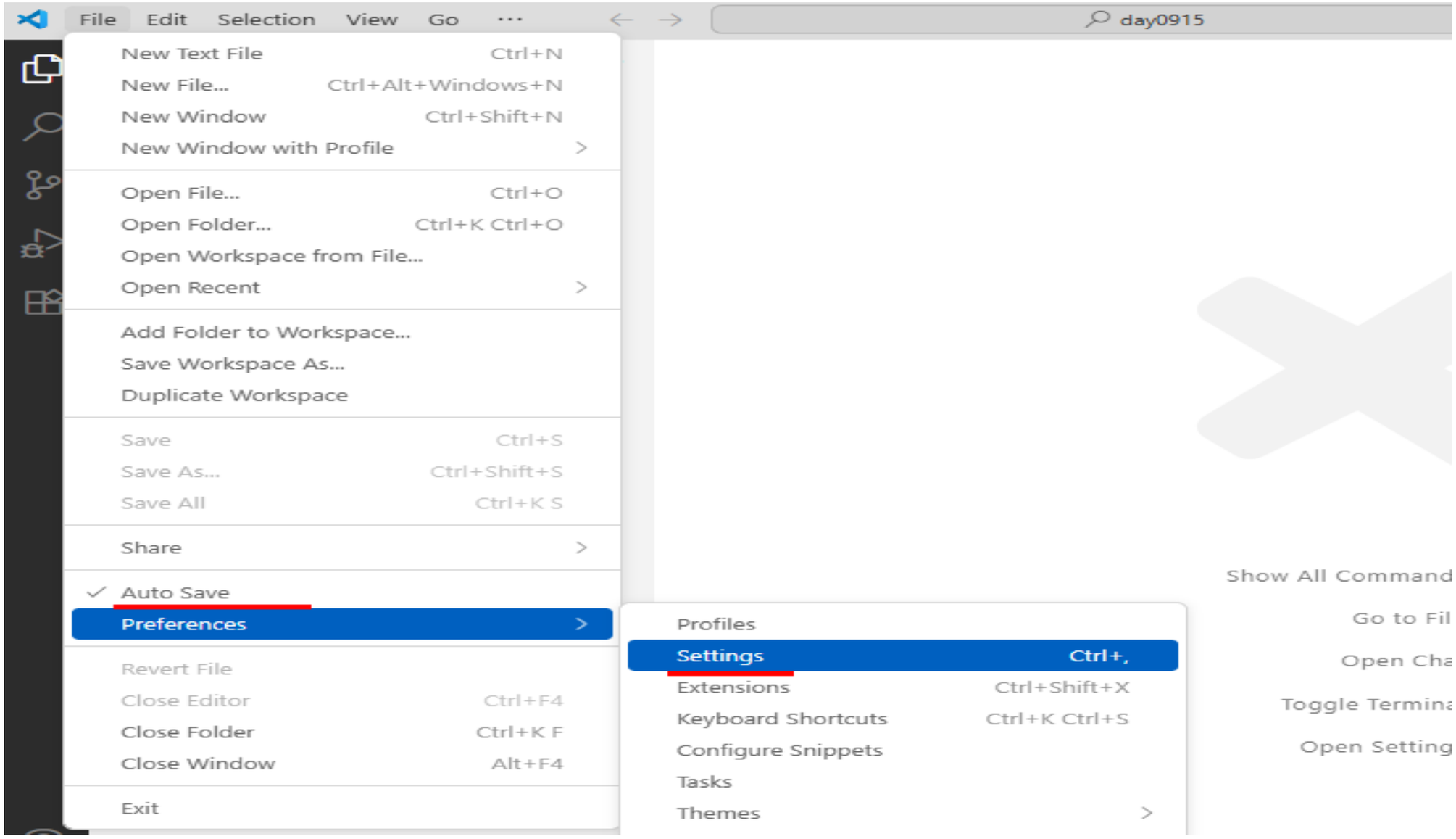


9. 프로그래밍 역량 강화 VS code 설치





10. 프로그래밍 역량 강화 VS code 설치



11. 프로그래밍 역량 강화 VS code 확장팩 설치

VS Code의 기본 상태에서는 단순한 텍스트 편집만 가능하므로 다음 세 가지 확장팩을 설치해야 한다.

확장팩 이름	주요 기능
Python	파이썬 코드 실행, 디버깅, 터미널 실행 지원
Jupyter	노트북(.ipynb) 형식의 셀 단위 실행 지원
Pylance	코드 자동 완성, 오류 탐지, 타입 검사 기능 제공 (Python extensions pack에 포함 됨)

① 왼쪽 사이드바에서 Extensions(확장기능) 아이콘을 클릭한다.

② 검색창에 각 확장팩 이름을 입력하고 Install을 누른다.

③ 설치가 완료되면 VS Code가 자동으로 확장 기능을 활성화한다.

12. 프로그래밍 역량 강화 파이썬 인터프리터 설정

인터프리터 설정

확장팩 설치 후에는 파이썬이 실행될 인터프리터(Interpreter)를 지정해야 한다. 인터프리터는 VS Code가 어떤 파이썬 환경(venv, 시스템 파이썬 등)을 사용할지를 결정한다. 이 과정을 완료해야 VS Code가 올바른 파이썬 실행 경로를 인식한다.

View메뉴
Command Palette (단축키 Ctrl + Shift + P)선택.
“Python: Select Interpreter”를 입력한다.
목록에서 사용할 환경을 선택한다. (python 3.12.10선택 추천)
상태바에 선택된 인터프리터가 표시되면 설정이 완료된다.

인터프리터 설정까지 끝났다면 작업 폴더를 새로 열고(File > Open Folder) 다음 단계의 작업을 진행한다.
메뉴에서 **File → Open Folder(ex day0108폴더선택)**
원하는 작업폴더 선택 다시 메뉴에서 **File → New File**을 선택한다.
파일 이름을 test.py로 저장한다. (File > Save)
샘플코드 print(“ hello ai test”)

파이썬 실행 python 파일명.py
C:₩day0108> python test.py 명령어 입력하면 실행

13. 프로그래밍 역량 강화 파이썬 가상환경 생성및 실행

1) 명령 프롬프트 실행 cmd

2) 새 프로젝트 폴더 생성 mkdir myproject

3) 생성한 폴더로 이동 cd myproject

4) 가상환경 생성 (venv 모듈로 venv 이름의 가상환경 생성)
python -m venv venv # 가상환경 생성 명령

5) venv : 생성될 폴더 이름
(일반적으로 "디렉토리가 venv" 사용하지만 이름은 아무거나 사용 가능 myenv, myAI)
이 명령을 실행하면 myproject 폴더 안에 venv 생성되고, 그 안에 독립된 파이썬 실행 환경
이 준비됨. --->실습은 1월9일 금요일 1월12일 월요일 설명하고 실습test합니다

6) 가상환경 활성화 명령 (이해하기 어려워요)
 C:₩myproject> venv₩Scripts₩activate
(venv) C:₩myproject> python test.py

7) 가상환경 비활성화
(venv) C:₩myproject> deactivate
(venv) C:₩myproject> where python #가상환경 경로 확인

14. 프로그래밍 역량 강화 파이썬 패키지 설치

8) 패키지설치및 목록 확인
(venv) C:₩myproject> pip install pandas numpy matplotlib openai
(venv) C:₩myproject> pip list

9) 패키지 관리 파일 생성
(venv) C:₩myproject> pip freeze > requirements.txt
(venv) C:₩myproject> type requirements.txt

10) 패키지 관리 파일 설치
(venv) C:₩myproject> pip install –r requirements.txt

15. 미디어 trend & 파이썬 조건문 중요성 이해

1) AI가 미디어 산업을 바꾸는 흐름을 이해한다.
생성형 AI가 뉴스·영상·음악을 자동으로 만들며 기존 제작 방식을 어떻게 바꾸는지 살펴보고, ChatGPT ,image-fx, runwayml, mixboard, stitch, canva, gamma 같은 도구가 실무에서 어떻게 활용되는지를 실습 체험했습니다.

2) AI 시대의 미디어 리터러시를 익힌다.
AI가 만든 정보의 출처·신뢰성·윤리성을 스스로 확인하고 판단하는 능력을 기른다. 미디어 이용자가 단순 소비자가 아니라 직접 만들고 평가하는 **참여자**라는 점을 이해한다.

3) 조건문을 사용해 논리적 사고 방식을 익힌다.
if, elif, else를 사용해 상황에 따라 다른 결과를 만드는 프로그램 구조를 이해하고, 들여쓰기와 실행 순서를 통해 기본적인 논리 흐름을 익힌다.

4)비교·논리 연산으로 판단 로직을 만든다.
==, !=, >, <, and, or, not 연산을 활용해 값의 조건을 확인하고 결과를 자동으로 판정하는 간단한 로직을 구현한다.

A5)AI 미디어 예제로 조건문을 실제 적용한다.
AI 뉴스 추천, 점수 평가, 등급 판정 등 실생활 사례를 코드로 구현해 보며 조건문이 AI 기반 서비스의 기본 논리와 어떻게 연결되는지 이해한다.

16. 프로그래밍 역량 강화 파이썬 기본 문법

화면모니터 출력 print( )
키보드 입력 input( )
변수 데이터를 기억하는 임시공간
연산의 종류-사칙연산, 관계연산
제어문-if,else

```
#연산예제
A,b,total = 7,5,0
total = a + b
print(a, '+', b, '=', total)
print('%d + %d = %d' %(a,b,total))
print('{} + {} = {}'.format(a,b,total))
print(f'{a} + {b} = {total}')

#입력예제
name = input("이름입력 >>>")
age = int(input( "정수>>> " ))
fit = float(input("실수>>> "))
print("이름 : ", name)
print("나이 : ", age)
print("키값 : ", fit)
```

17. 프로그래밍 역량 강화 파이썬 기본 문법

화면모니터 출력 print( )
키보드 입력 input( )
변수 데이터를 기억하는 임시공간
연산의 종류-사칙연산, 관계연산
제어문-if,else

#if예제 홀짝구하기
a = int(input("정수 입력>>> "))
if a%2==0: print("짝수입니다")
else: print("홀수입니다")

#if~elif~else 예제
avg=int(78.9)
if avg>=90 : print(avg , " A학점")
elif avg>=80: print(avg , " B학점")
elif avg>=70: print(avg , " C학점")
elif avg>=60: print(avg , " D학점")
else: print(avg , " F학점")

18. 프로그래밍 역량 강화 파이썬 기본 문법

화면모니터 출력 print( )
키보드 입력 input( )
변수 데이터를 기억하는 임시공간
연산의 종류-사칙연산, 관계연산
제어문-if,else
예외처리-try~except

```
#try~except
try:
    num1 = int(input("수1 입력? ")).
    num2 = int(input("수2 입력? "))
    hap = num1+num2
    mok = num1/num2

    print("합=", hap)
    print("몫=", round(mok,2))
except Exception as ex:
    print(ex , " 에러가 발생했습니다")
```

19. 프로그래밍 역량 강화 파이썬 기본 문법

화면모니터 출력 print( )
키보드 입력 input( )
변수 데이터를 기억하는 임시공간
연산의 종류-사칙연산, 관계연산
제어문-if,else
예외처리-try~except
리스트-여러데이터나열

```
#list실습 print(lotto)  print(mydata) print(week) 출력
lotto = [ 23, 7, 19, 45, 31, 26] #int정수형구성 리스트
mydata = ['kim', 90, 85, 87.5, True, 'B', '축합격'] #혼합리스트
week = ['월요일','화요일','수요일','목요일','금요일','토요일','일요일' ] #문자열구성 리스트

#리스트 내용추가
dt=[5,7,33]
print(dt)

dt.append(9)
print(dt)

dt.insert(1, 46) #[5, 46, 7, 33, 9] insert(위치, 데이터)
print(dt)
```

https://donga-ai.factchat.bot/auth

로그인후 실습및 숙제 활용하면 좋아요

좋은 프롬프트를 만드는 것!

좋은 질문을 만들어내는 것!