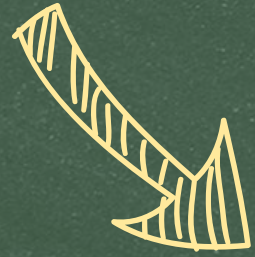
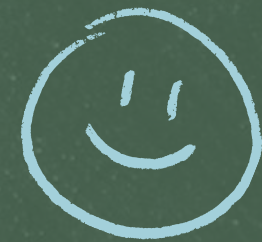

쉽고 재미있게 배우는 코딩 시간



AI·파이썬 기초 체험형



교육 봉사 교안



SWAT 10기 임세은

교육개요

주제: 파이썬을 활용한 나만의 챗봇 만들어보기

목표: 파이썬의 기본문법을 학습 후, 직접 챗봇 만들어보기, AI를 활용한 챗봇 업그레이드 하기

대상 및 시간: 중학교 및 고등학교(2시간)

주요 교육 내용:

- a. 파이썬에 대한 기초적인 문법 학습(변수, 조건문, 출력문 등) → 1시간
- b. 나만의 간단한 python 챗봇 만들기
- c. 인공지능을 활용한 대화형, 이미지 인식형 챗봇 만들어보기

교육환경: 전산실 PC 및 개인 PC

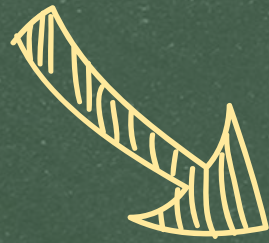
사전 준비물: 웹캠 및 마이크

기대효과:

- a. 대학 진학 전 파이썬을 직접 경험하며 코딩에 대한 이해도를 높임
- b. 인공지능 및 프로그래밍 분야에 대한 흥미 유도



오늘의 수업내용은?



1. 파이썬 기초 다지기

2. AI 체험

- 음성인식 기술 체험
- 이미지 인식 데모

오늘의 학습 목표

1 파이썬 기본 문법 체험

print, input, if/else, while 등 핵심 문법을 직접 사용해보며 프로그래밍의 기초를 다집니다.

2 챗봇 제작 실습

간단한 텍스트 챗봇을 만들어보며 내가 만든 프로그램이 실제로 동작하는 경험을 합니다.

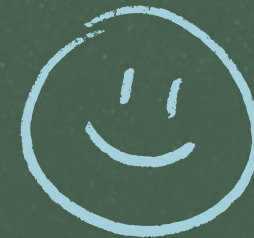
3 AI 기술 동작 원리 이해

실제 AI 기술이 어떻게 동작하는지 체험하며 인공지능에 대한 이해의 폭을 넓힙니다

파이썬 환경구성



설치방법





python



AI 모드 전체 이미지 쇼핑 동영상 뉴스 짧은 동영상 더보기 도구



Python.org

<https://www.python.org>

Welcome to Python.org

Python is a programming language that lets you work quickly more effectively. [Learn More](#)

파이썬

프로그래밍 언어



Donate



Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

All releases

Source code

Windows

macOS

Android

Other Platforms

License

Alternative Implementations

Download for Windows

Python install manager

Or get the standalone installer for Python 3.14.2

Not the OS you are looking for? Python can be used on many operating systems and environments.

[View the full list of downloads.](#)

```
# Python
```

```
>>> def fib(n):
```

```
>>>     a, b = 0, 1
```

```
>>>     while n > 0:
```

```
>>>         print(a)
```

```
>>>         a, b = b, a + b
```

```
>>>         n -= 1
```

```
>>>     fib(10)
```

```
0 1 1 2 3
```

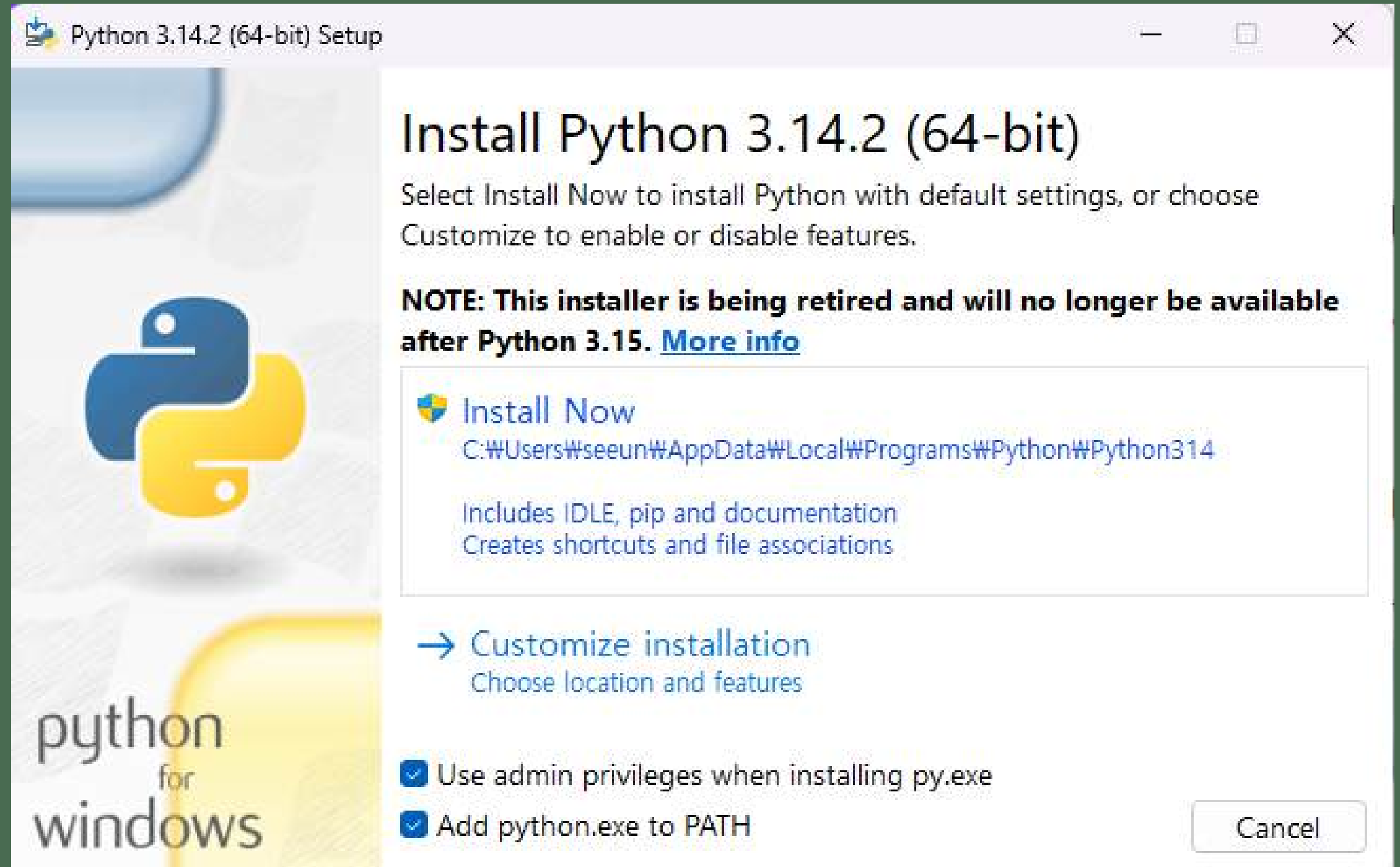
```
5 8 13
```

```
21 34 55
```

```
610 987
```

Stable Releases

- [Python 3.14.2 - Dec. 5, 2025](#)
Download using the [Python install manager](#).
 - 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 install manager 25.2 - Dec. 5, 2025](#)
 - Download [Installer \(MSIX\)](#)
 - Download [MSI package](#)
- [Python 3.13.11 - Dec. 5, 2025](#)
Note that Python 3.13.11 cannot be used on Windows 7 or




Python 3.14.2 (64-bit) Setup

Install Python 3.14.2 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

NOTE: This installer is being retired and will no longer be available after Python 3.15. [More info](#)

 **Install Now**
C:#Users#seeun#AppData#Local#Programs#Python#Python314

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ **Customize installation**
Choose location and features

Use admin privileges when installing py.exe
 Add python.exe to PATH

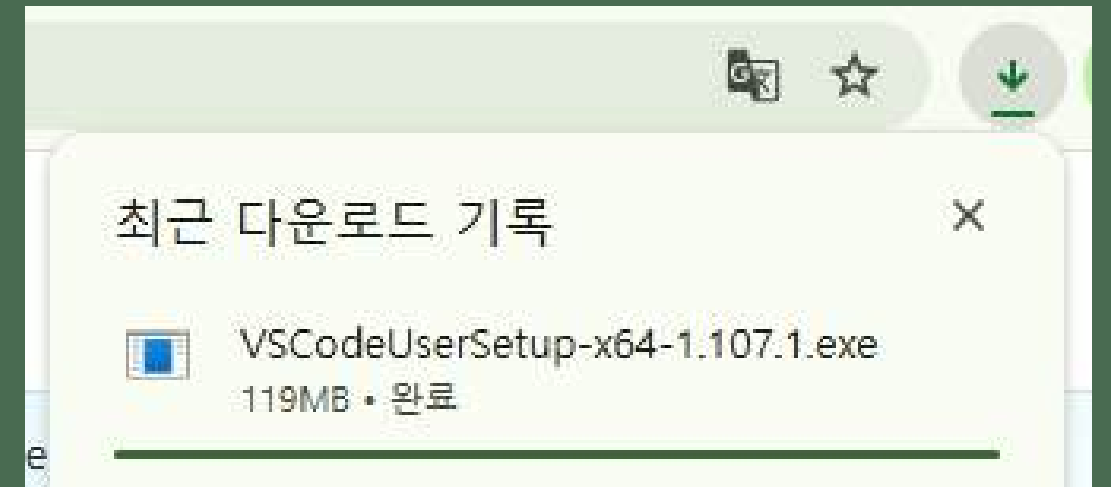
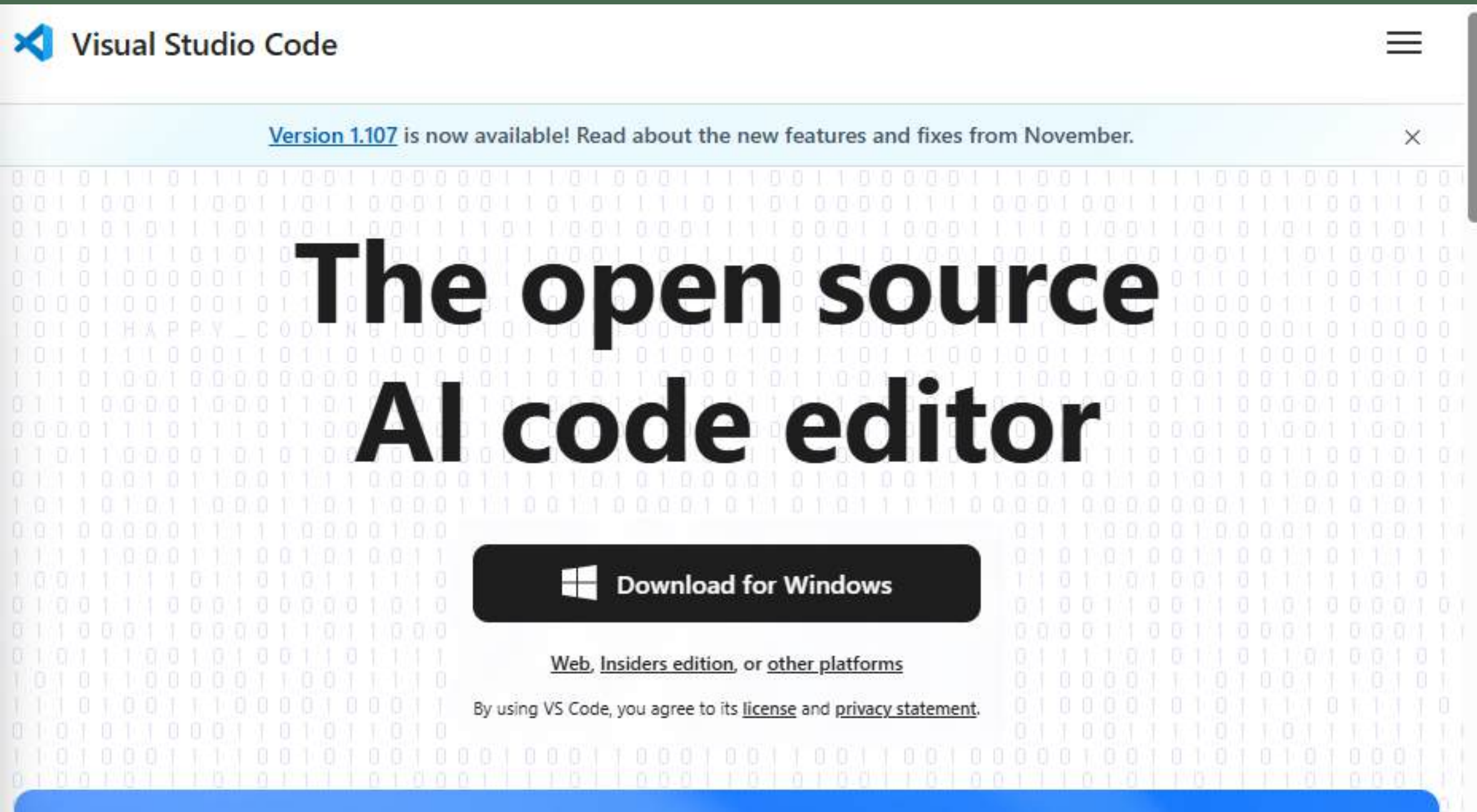
Cancel

아래 2개 꼭 체크해야 합니다!!!

```
명령 프롬프트 - python
Microsoft Windows [Version 10.0.26100.7462]
(c) Microsoft Corporation. All rights reserved.

C:\Users\seeun>python
Python 3.14.2 (tags/v3.14.2:df79316, Dec 5 2025,
Type "help", "copyright", "credits" or "license" f
>>>
```

CMD 창을 열어
PYTHON이 설치되었는지 확인합니다.



가운데 검정 Download를 하여 설치합니다.

File Edit Selection View Go ... Search

Walkthrough: Setup VS Code X

Welcome

Get started with VS Code

Customize your editor, learn the basics, and start coding

Use AI features with Copilot for free

You can use Copilot to generate code across multiple files, fix errors, ask questions about your code, and much more using natural language.

[Use AI Features](#)

Choose your theme

Watch video tutorials

Mark Done

[Next Section →](#)

Code collects usage data. Read our [privacy statement](#) and learn how to [opt out](#).

CHAT

Build with Agent

AI responses may be inaccurate.

[Generate Agent Instructions](#) to onboard onto your codebase.

SUGGESTED ACTIONS

[Ask @vscode](#) [Create Project](#)

[Add Context...](#)

Describe what to build next

Agent Auto

EXTENSIONS

Search Extensions in Marketplace

⚠️ Deprecated extensions detected. Review them and migrate to alternatives. [Show](#)

INSTALLED 17

- IntelliCode**
AI-assisted development
Microsoft [Migrate](#) ⚠️
- IntelliCode API Usage Examples**
See relevant code examples from Gi...
Microsoft [Migrate](#) ⚠️
- Python** 160ms
Python language support with exte...
Microsoft [Restart Extensions](#)
- Python Debugger**
Python Debugger extension using d...
Microsoft [Restart Extensions](#)
- Python Environments** 45ms
Provides a unified python environm...
Microsoft [Restart Extensions](#)
- autoDocstring - Python Docstrin...**
Generates python docstrings autom...
Nils Werner
- Django**
Beautiful syntax and scoped snippet...
Baptiste Darthenay
- Jinja**

RECOMMENDED 5

MCP SERVERS



The image shows the Visual Studio Code interface with the Extensions Marketplace open. The left sidebar shows a list of installed extensions, with the 'Python' extension by Microsoft selected. The main panel displays the details for the 'Python' extension, including its logo, name, publisher (Microsoft), and download count (198,592,045). Below the extension name, there are buttons for 'Disable', 'Uninstall', and 'Switch to Pre-Release Version'. The 'Installation' section on the right provides technical details about the extension's identifier, version, and size.

File Edit Selection View Go ... Search

EXTENSIONS

Search Extensions in Marketplace

⚠️ Deprecated extensions detected. Review them and migrate to alternatives. [Show](#)

INSTALLED 17

- IntelliCode**
AI-assisted development
Microsoft [Migrate](#) ⚠️ ⚙️
- IntelliCode API Usage Examples**
See relevant code examples from Gi...
Microsoft [Migrate](#) ⚠️ ⚙️
- Python**
Python language support with exte...
Microsoft ⚙️
- Python Debugger**

Extension: Python X

Python

Microsoft [microsoft.com](#) | 198,592,045

Python language support with extension access p...

[Disable](#) [Uninstall](#) [Switch to Pre-Release Version](#) ✓

DETAILS FEATURES CHANGELOG EXTENSION PACK

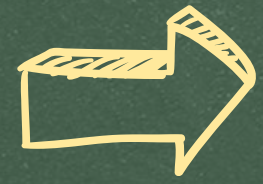
Python extension for Visual Studio Code

A Visual Studio Code extension with rich support for the Python language (for all actively supported Python versions) providing

Installation

Identifier	ms-python.python
Version	2025.20.1
Last Updated	7 minutes ago
Size	29.61MB

PYTHON을 설치해 줍니다.



1부 수업 목표



파이썬의 기본
구조 이해

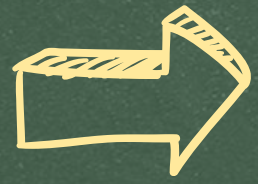


컴퓨터의 판단과
반복 방식 이해



내가 만든
프로그램 경험

- 변수, 출력, 입력 등 프로그래밍의 핵심 개념을 이해합니다
- 조건문과 반복문을 통해 컴퓨터가 생각하는 방식을 배웁니다
- 직접 만든 챗봇을 동작시켜 봅니다.
→ 직접코드를 작성하면서 프로그램의 기본 개념을 익힙니다.



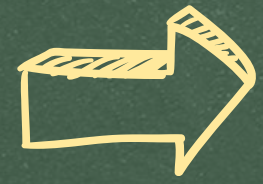
파이썬이란?

배우기 쉬운
언어

AI 분야
필수 언어

빠른
결과 확인

- 영어 문장처럼 읽히는 간단한 문법으로 프로그래밍 초보자에게 최적화되어 있습니다. 복잡한 기호나 규칙이 적어 학습 부담이 적습니다.
- 인공지능, 머신러닝, 데이터 분석 분야에서 가장 많이 사용됩니다. 구글, 넷플릭스 등 세계적인 기업들도 파이썬을 활용합니다.
- 코드를 작성하면 즉시 결과를 확인할 수 있어 학습 동기가 높아집니다. 초보자도 첫날부터 동작하는 프로그램을 만들 수 있습니다.



파이썬의 변수는 무엇이 있을까?

INT
(정수형)

- 정수형은 소수점이 없는 숫자이다.
- 1, 2, 3 은 정수형 이다.

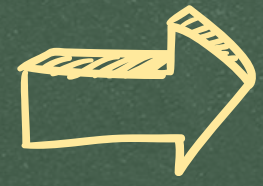
FLOAT
(실수형)

- 실수형은 소수점이 있는 숫자이다.
- 10.05, 12.2, 5.7 은 실수형이다.

STR
(문자열)

- 문자열은 글자나 문장을 저장할 때 사용한다.

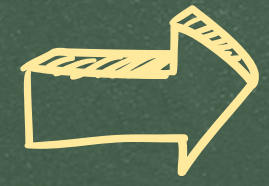
파이썬은 자료형을 미리 선언하지 않고 값으로 자동결정을 하는 언어이다



각 변수는 어떤 자료형일까?

```
1 x = 10
2 y = 10.5
3 z = "hello"
4
5 print(type(x))
6 print(type(y))
7 print(type(z))
8
```

```
<class 'int'>
<class 'float'>
<class 'str'>
```



python의 연산기

1. + 연산자

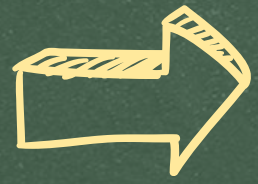
4. /연산자

2. -연산자

5. %연산자

3. * 연산자

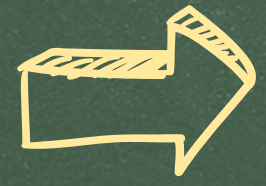
6. 비교 연산자



python의 연산기

```
PS C:\Users\seeun\OneDrive\Desktop\SWAT 봉사단> & C:/Users,
10+3의 연산결과는 13 입니다
10-3의 연산결과는 7 입니다
10*3의 연산결과는 30 입니다
10/3의 연산결과는 3.3333333333333335 입니다
10%3의 연산결과는 1 입니다
10>3의 연산결과는 True 입니다
10==3의 연산결과는 False 입니다
PS C:\Users\seeun\OneDrive\Desktop\SWAT 봉사단> █
```

```
test.py  교안.zip
test.py > ...
1 a = 10
2 b = 3
3
4 print(str(a) + "+" + str(b) + "의 연산결과는", a + b, "입니다")
5 print(str(a) + "-" + str(b) + "의 연산결과는", a - b, "입니다")
6 print(str(a) + "*" + str(b) + "의 연산결과는", a * b, "입니다")
7 print(str(a) + "/" + str(b) + "의 연산결과는", a / b, "입니다")
8 print(str(a) + "%" + str(b) + "의 연산결과는", a % b, "입니다")
9
10 print(str(a) + ">" + str(b) + "의 연산결과는", a > b, "입니다")
11 print(str(a) + "==" + str(b) + "의 연산결과는", a == b, "입니다")
12
13
14
```

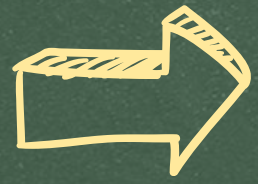


코드 실행



기본적인 문법

- 출력과 입력(print, input)
 - 조건문 (if/else)
 - 반복문(while)
-



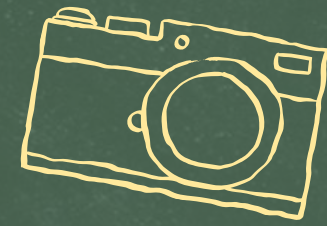
출력과 입력(input, print)

text

```
이름을 입력하세요: 민수  
안녕하세요, 민수!
```

python

```
name = input("이름을 입력하세요: ")  
print("안녕하세요,", name)
```

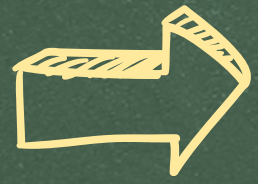


Print

프로그램이 사용자에게 메시지를 전달하는 명령어입니다. 화면에 텍스트, 숫자, 계산 결과 등을 보여줄 수 있습니다.

input

사용자가 키보드로 입력한 내용을 프로그램이 받아오는 명령어입니다. 사용자와 대화하는 프로그램을 만들 수 있습니다.



조건문(if/else)

컴퓨터가 상황을 판단하고 다르게 행동하도록 만드는 핵심 문법입니다. 마치 사람이 "만약~라면, ~해라"라고 생각하는 것

나이를 입력하세요: 17
미성년자입니다.

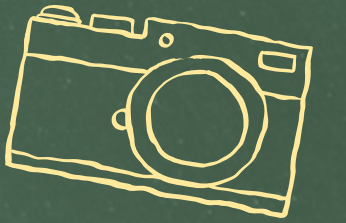
text

나이를 입력하세요: 22
성인입니다.

```
python
age = int(input("나이를 입력하세요:
if age >= 20:
    print("성인입니다.")
else:
    print("미성년자입니다.")
```

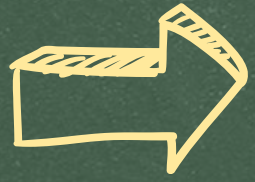
조건이 맞으면 실행

if 문으로 조건을 확인하고, 조건이 참이면 해당 코드를 실행합니다



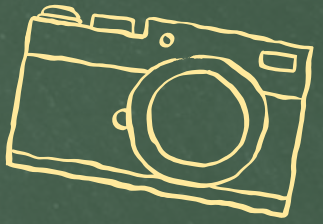
아니면 다른 동작 수행

else 문으로 조건이 거짓일 때의 동작을 정의합니다



반복문(while)

같은 동작을 계속 반복하도록 만드는 문법



text

너: 안녕
챗봇: 안녕하세요!
너: 이름
챗봇: 저는 파이썬 챗봇입니다.
너: 종료
챗봇: 대화를 종료합니다.

python

```
while True:  
    q = input("너: ")  
  
    if q == "종료":  
        print("챗봇: 대화를 종료합니다.")  
        break  
    else:  
        print("챗봇: 안녕하세요!")
```

특정 조건이 만족되는 동안 반복

특정 조건이 만족되는 동안 반복

while 뒤에 조건을 작성하면, 그 조건이 참인 동안 계속 코드를 실행합니다. 조건이 거짓이 되면 반복을 멈춥니다.

대화가 끊기지 않고 이어지도록 함

챗봇처럼 사용자가 종료를 원할 때까지 계속 대화를 이어가는 프로그램을 만들 수 있습니다.

나만의 챗봇 만들기

필수 조건

1. 입력한 문장에 따라 적절한 답변을 제공
2. “종료” 입력시 프로그램이 종료
3. 코드를 수정해 나만의 챗봇으로 만들기

추가 조건

1. 질문을 할때마다 “현재 질문은 n번째 질문입니다.” 출력하기.
2. 욕설/금지어 입력시 → “올바른 표현을 사용해주세요” 출력하기

나만의 챗봇 만들기



```
너: 안녕
현재 질문은 1 번째 질문입니다.
챗봇: 안녕하세요!
너: 이름
현재 질문은 2 번째 질문입니다.
챗봇: 저는 나만의 챗봇입니다.
너: 반가워
현재 질문은 3 번째 질문입니다.
챗봇: 잘 이해하지 못했어요.
너: 너는 바보야
현재 질문은 4 번째 질문입니다.
올바른 표현을 사용해주세요.
너: 종료
챗봇을 종료합니다.
```

```
count = 0
ban_words = ["욕", "바보"]

while True:
    user = input("너: ")

    if user == "종료":
        print("챗봇을 종료합니다.")
        break

    count += 1
    print("현재 질문은", count, "번째 질문입니다.")

    banned = False
    for word in ban_words:
        if word in user:
            print("올바른 표현을 사용해주세요.")
            banned = True
            break

    if banned:
        continue

    if "안녕" in user:
        print("챗봇: 안녕하세요!")
    elif "이름" in user:
        print("챗봇: 저는 나만의 챗봇입니다.")
    else:
        print("챗봇: 잘 이해하지 못했어요.")
```

나만의 챗봇의 한계점

☆ 미리 정해진 말만 이해 가능

☆ 새로운 질문에는 대응 불가

☆ 스스로 학습하지 못함

☆ 인공지능(AI)가 필요함 ☆

규칙 기반 vs AI

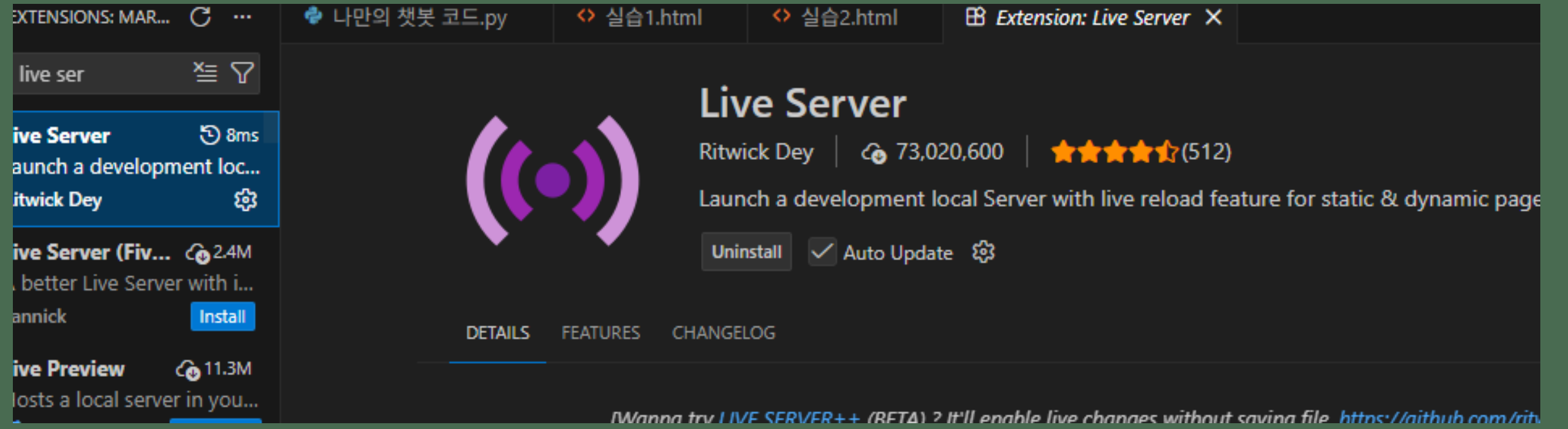
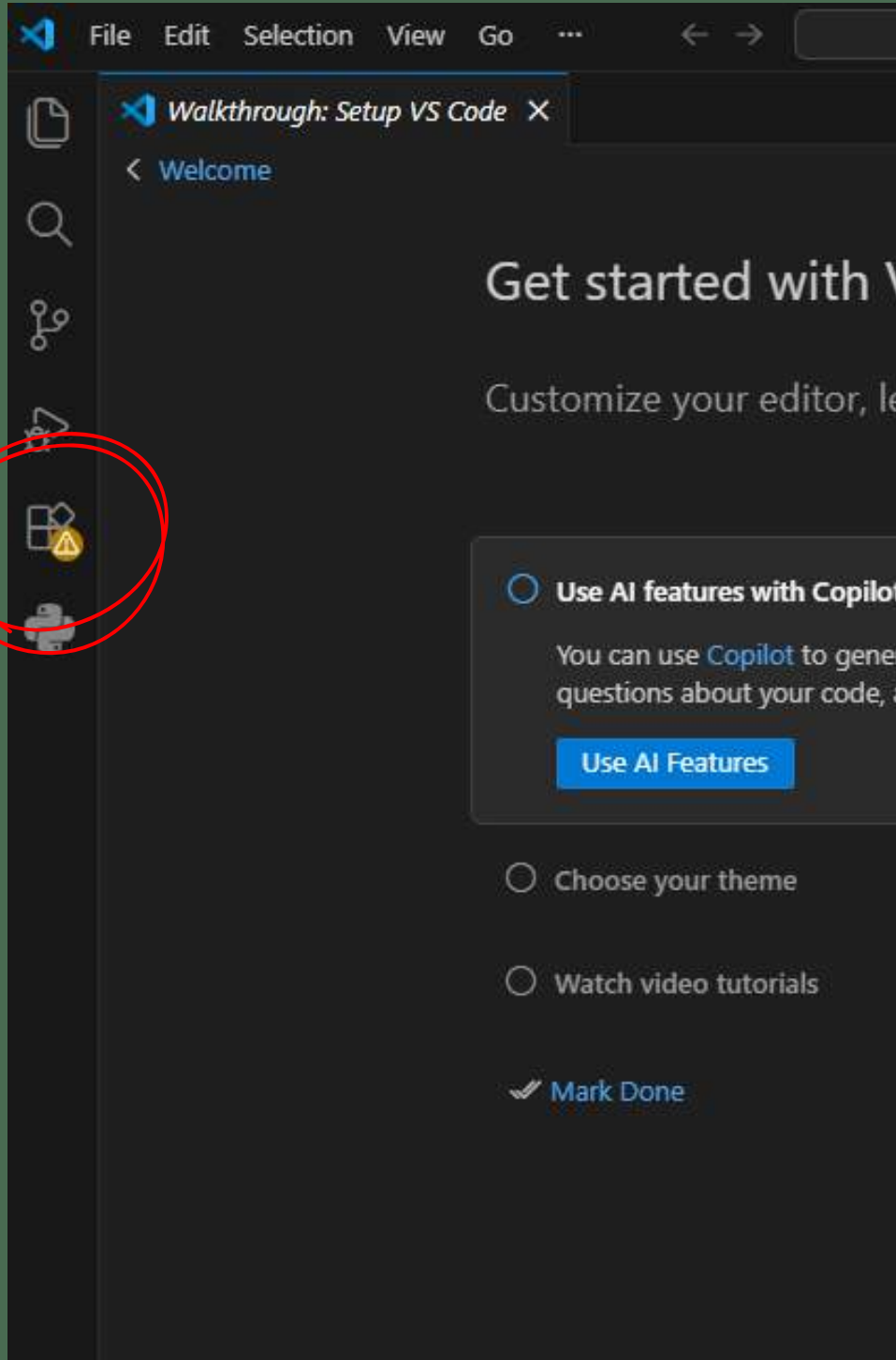
규칙 기반 프로그램

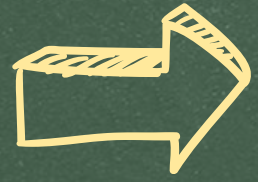
1. 사람이 모든 규칙을 직접 작성
2. 정해진 입력에만 반응
3. 규칙 밖 질문에는 대응 불가

AI 프로그램

1. 데이터를 통해 판단 기준을 학습
2. 비슷한 질문에도 유연하게 대응
3. 상황에 따라 다른 답변 가능

← 같은 챗봇이라도 동작 방식이 다름





음성인식 AI 체험

- 소리로 숫자로 변환되어 처리된다.

학습형 음성인식 챗봇

Chrome + VS Code Live Server 권장

- ▶ 음성 → 텍스트 → 의미 유사도 기반 응답
- ▶ 학습되지 않은 질문은 "아직 준비 중"으로 응답

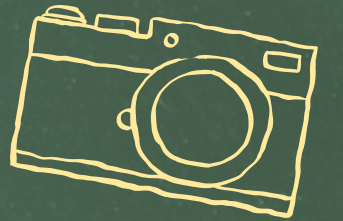
🗨 시작

■ 중지

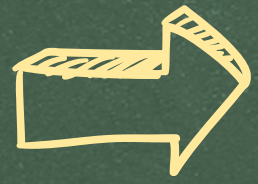
상태
중지됨

인식된 문장
안녕

챗봇 응답
안녕하세요!



- 사람이 말한 소리를 인식
- 음성을 글자로 변환
- 챗봇과 연결해 명령 수행



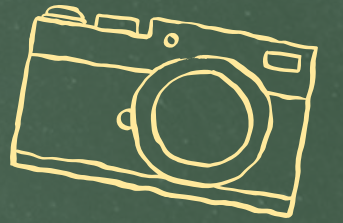
영상 인식시 체험

- 이미지도 숫자 데이터로 처리됨

일상 의사표현 제스처 → 챗봇 응답



👋 안녕하세요



- 웹캠으로 얼굴·손 동작 인식
- 특정 동작에 반응하는 프로그램
- 인식 성공과 실패 비교



AI의 활용 사례

- 음성 비서
- 얼굴인식
- 자율 주행
- 추천 시스템
- 챗봇

다음 시간에 만나요! 