

# 파이썬 프로그래밍 설치와 실행

---

컴퓨터소프트웨어공학과  
성낙준

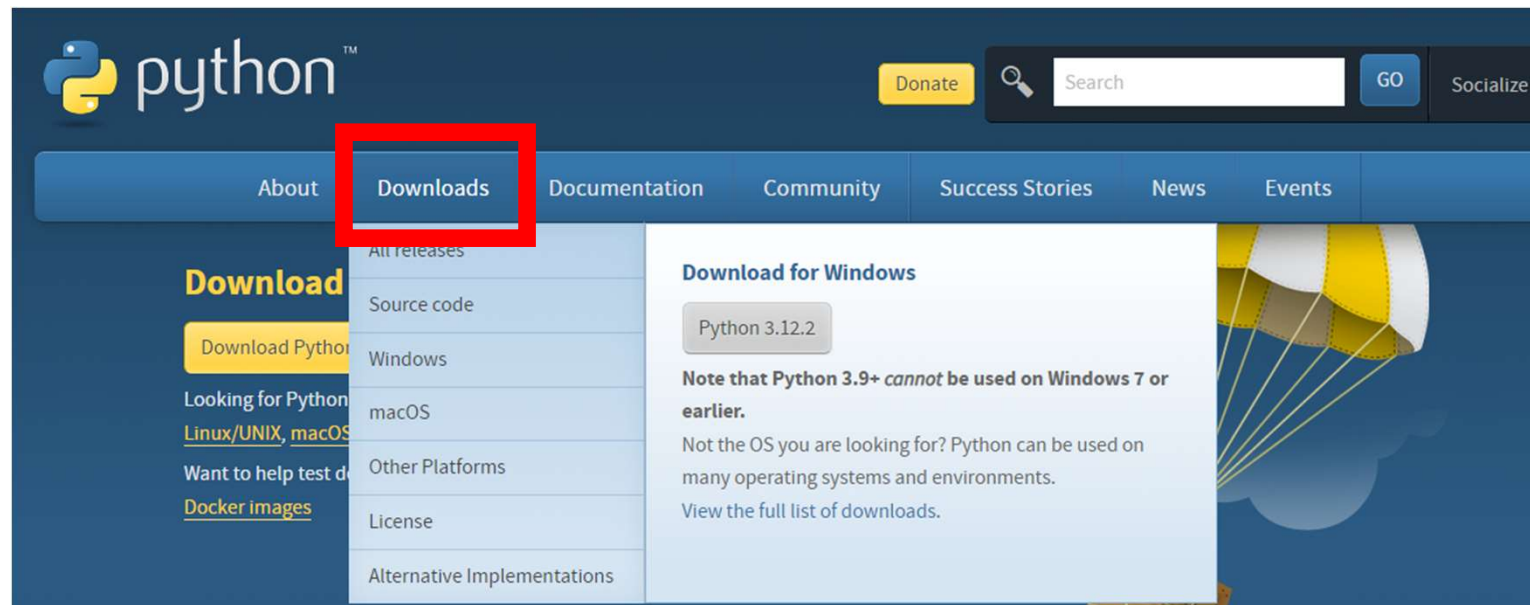


- 파이썬 프로그래밍을 수행할 수 있는 개발 환경 세팅하기
- 이 자료는 두 가지 개발 환경을 구축해보는 내용을 담고 있음
  - 1) 로컬 환경에서 개발하기(vscode + python)
  - 2) 클라우드 환경에서 개발하기(google colab)



# 파이썬 설치하기

- Python 설치(홈페이지에서 타고 들어가기)
  - <https://www.python.org/>



## Active Python Releases

For more information visit the [Python Developer's Guide](#).

| Python version | Maintenance status | First released       | End of support | Release schedule |
|----------------|--------------------|----------------------|----------------|------------------|
| 3.13           | prerelease         | 2024-10-01 (planned) | 2029-10        | PEP 719          |
| 3.12           | bugfix             | 2023-10-02           | 2028-10        | PEP 693          |
| 3.11           | bugfix             | 2022-10-24           | 2027-10        | PEP 664          |
| 3.10           | security           | 2021-10-04           | 2026-10        | PEP 619          |
| 3.9            | security           | 2020-10-05           | 2025-10        | PEP 596          |
| 3.8            | security           | 2019-10-14           | 2024-10        | PEP 569          |



- Python 설치(다운로드 링크 바로가기)
  - <https://www.python.org/downloads/windows/>

■ No files for this release.

■ [Python 3.9.17 - June 6, 2023](#)  
**Note that Python 3.9.17 cannot be used on Windows 7 or earlier.**

■ [Python 3.10.11 - April 5, 2023](#)  
**Note that Python 3.10.11 cannot be used on Windows 7 or earlier.**

- Download [Windows installer \(64-bit\)](#)
- Download [Windows help file](#)
- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows installer \(32-bit\)](#)

■ [Python 3.11.3 - April 5, 2023](#)  
**Note that Python 3.11.3 cannot be used on Windows 7 or earlier.**

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

■ [Python 3.10.10 - Feb. 8, 2023](#)  
**Note that Python 3.10.10 cannot be used on Windows 7 or earlier.**

- Download [Windows installer \(64-bit\)](#)
- Download [Windows help file](#)
- Download [Windows embeddable package \(64-bit\)](#)

■ [Download Windows embeddable package \(32-bit\)](#)

■ [Download Windows embeddable package \(ARM64\)](#)

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

■ [Download Windows embeddable package \(32-bit\)](#)

■ [Download Windows embeddable package \(ARM64\)](#)

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

■ [Download Windows embeddable package \(64-bit\)](#)

■ [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 \(ARM64\)](#)

■ [Python 3.12.0a5 - Feb. 7, 2023](#)

- [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 \(ARM64\)](#)

■ [Python 3.12.0a4 - Jan. 10, 2023](#)

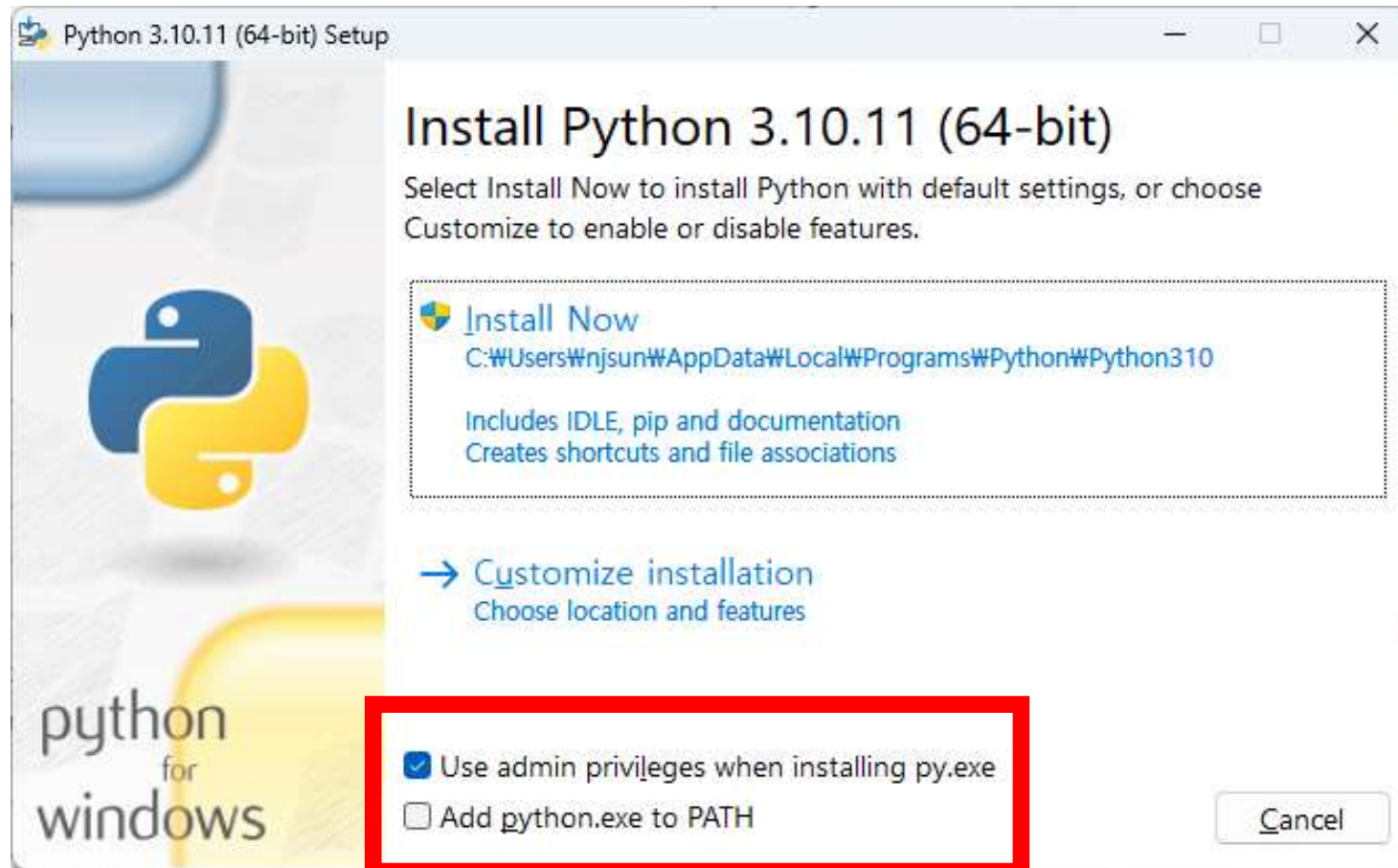
- [Download Windows installer \(64-bit\)](#)
- [Download Windows installer \(32-bit\)](#)
- [Download Windows installer \(ARM64\)](#)
- [Download Windows embeddable package \(64-bit\)](#)
- [Download Windows embeddable package \(32-bit\)](#)
- [Download Windows embeddable package \(ARM64\)](#)

■ [Python 3.12.0a3 - Dec. 6, 2022](#)

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

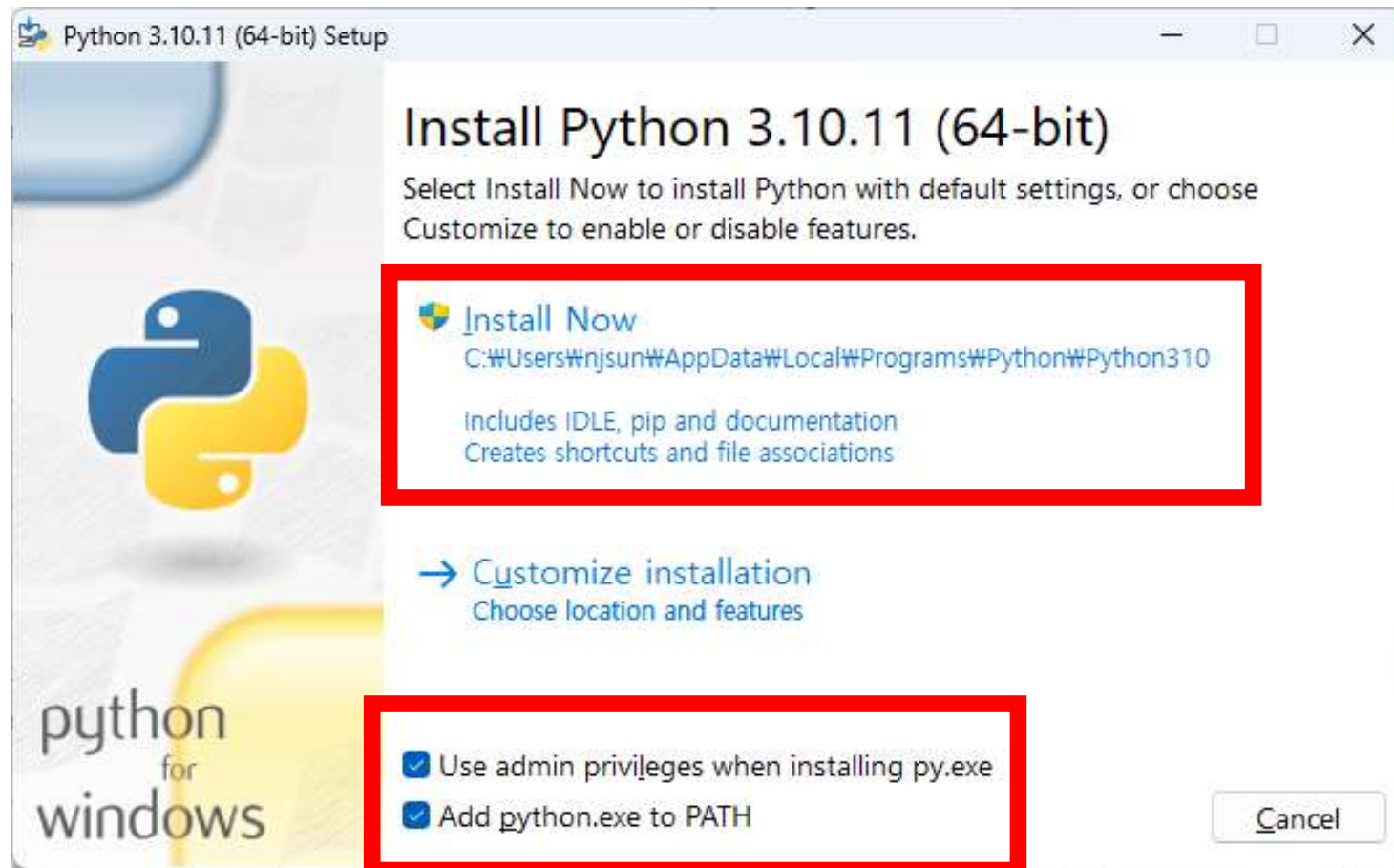


- Python 설치
  - 설치파일 실행 결과



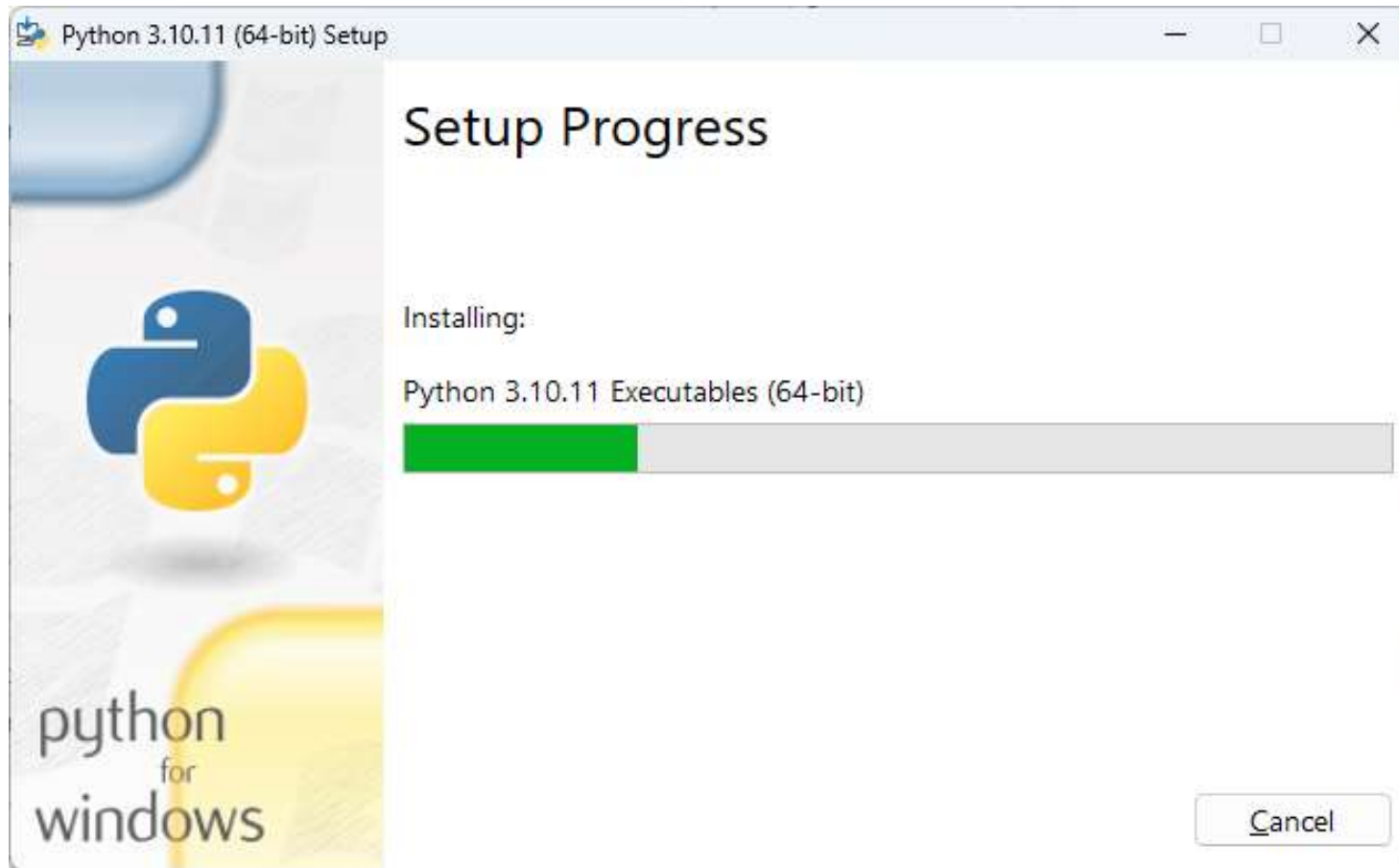


- Python 설치
  - 시스템 환경 변수에 자동으로 등록해주는 기능 체크





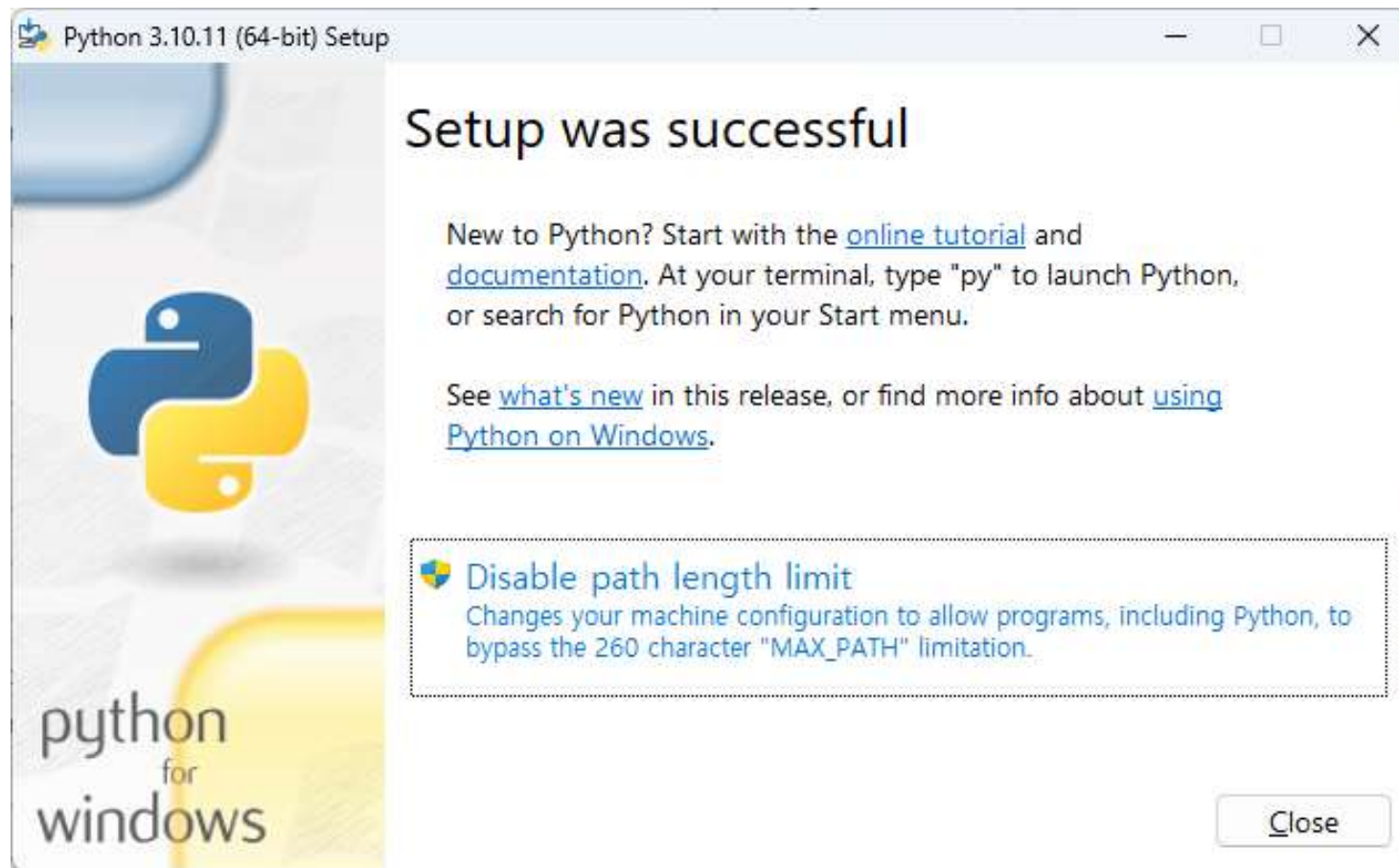
- Python 설치
  - 설치 중







- Python 설치
  - 설치 완료 확인 -> Disable path length limit 클릭
    - 파일 경로와 이름의 길이가 260자로 제한이 되어있음
    - 260자를 넘더라도 문제 없이 파이썬을 이용할 수 있도록 해제하기







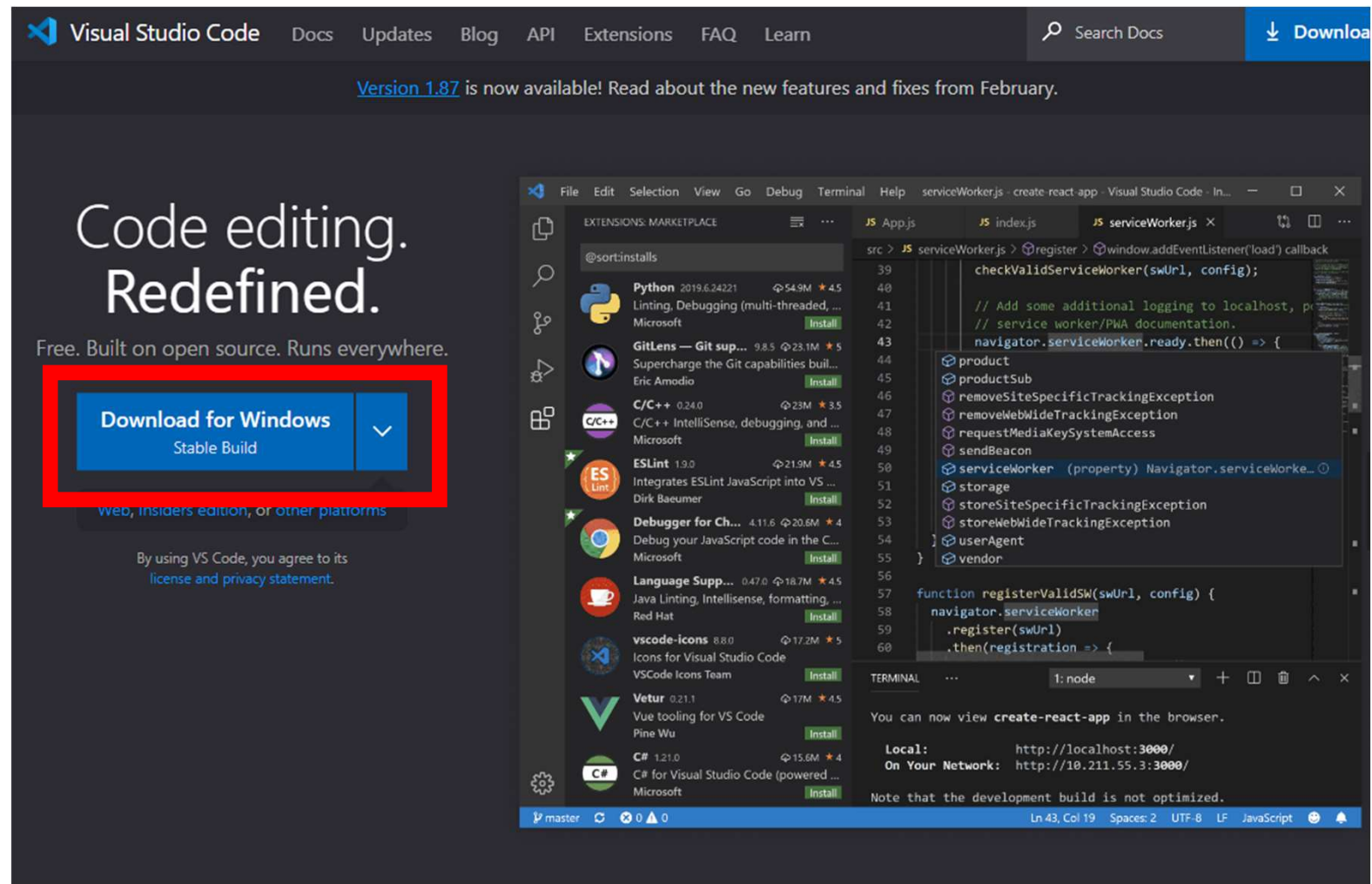
- Python 설치 확인하기
  - 1) cmd 실행하기
  - 2) Powershell 실행하기
  - python --version 명령어 작성하기

The image shows two screenshots of Windows command shells. The top screenshot is a Command Prompt window with the title bar 'C:\Users\Wnjsun\AppData\WR...'. It displays the text: 'Microsoft Windows [Version 10.0.22621.1992] (c) Microsoft Corporation. All rights reserved. C:\Users\njsun>python --version Python 3.10.11'. The bottom screenshot is a PowerShell window with the title bar 'C:\Users\Wnjsun\AppData\WR...'. It displays the text: 'Windows PowerShell Copyright (C) Microsoft Corporation. All rights reserved. 새로운 기능 및 개선 사항에 대한 최신 PowerShell PS C:\Users\njsun> python --version Python 3.10.11'.



# Visual Studio Code 설치하기

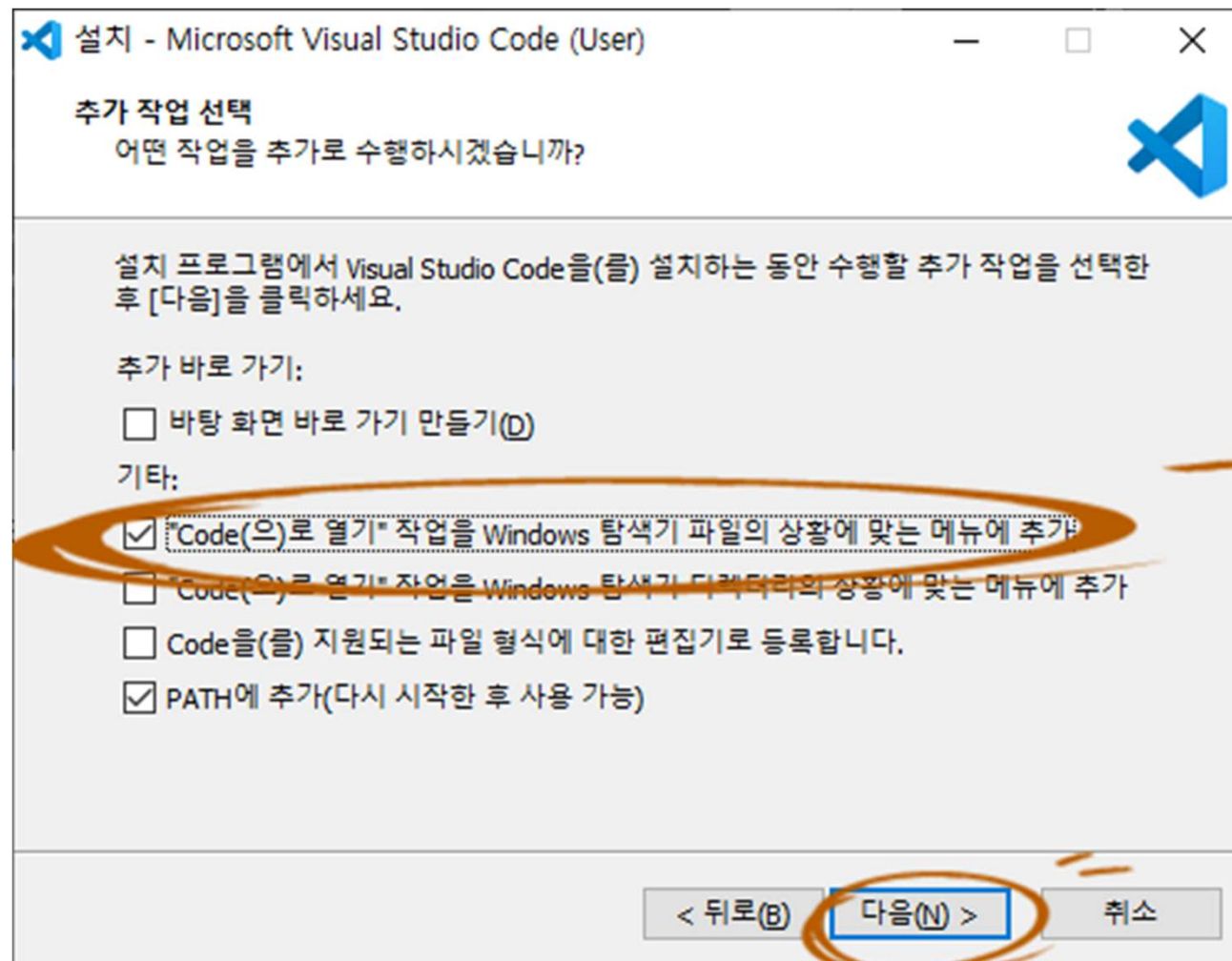
- Visual Studio Code 설치
  - <https://code.visualstudio.com/>





# Visual Studio Code 설치하기

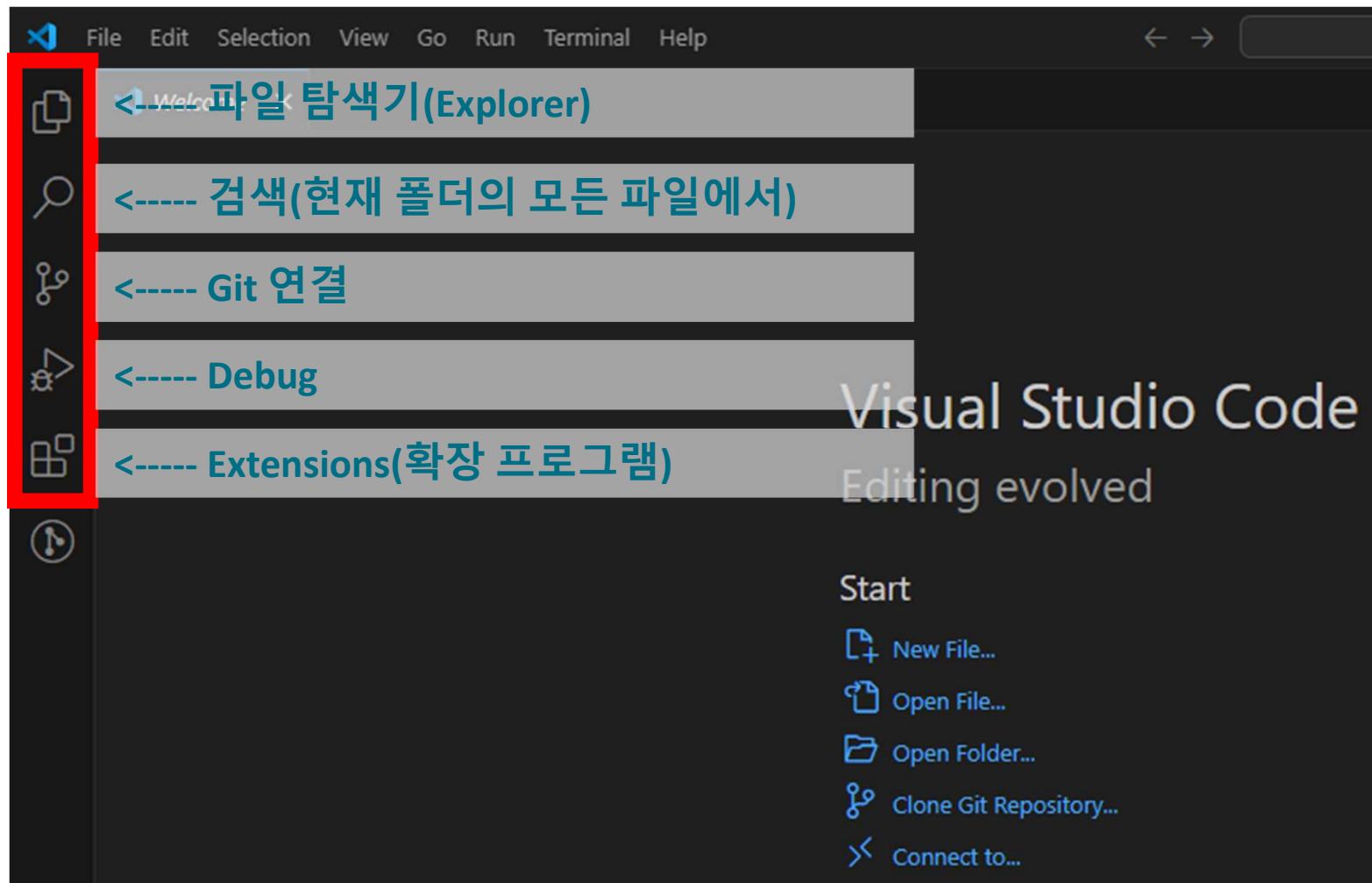
- Visual Studio Code 설치
  - 설치파일 실행 후 기타의 아래 내용 선택





# Visual Studio Code 설치하기

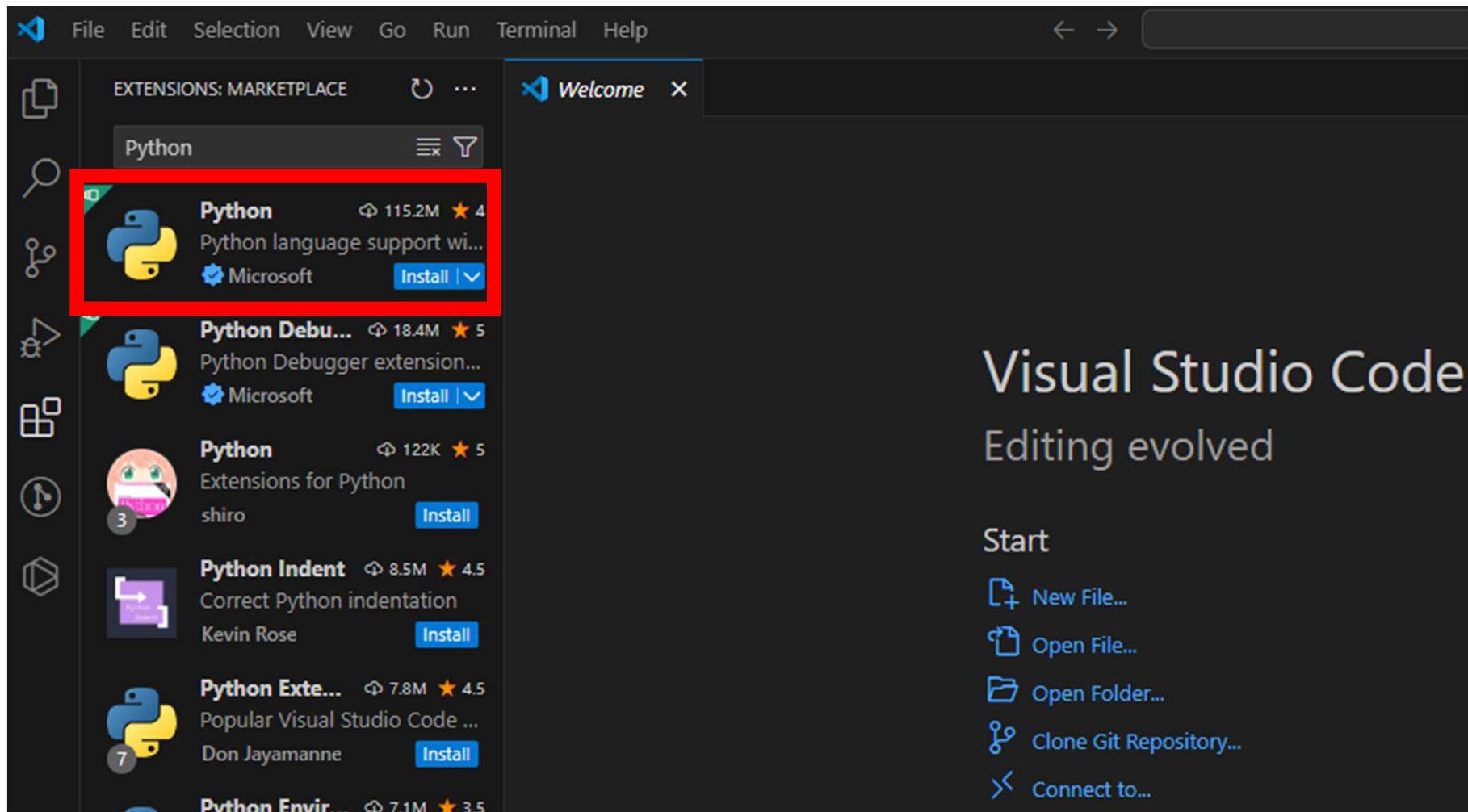
- Visual Studio Code 설치
  - 실행 후 좌측 메뉴 확인 / Extensions 선택





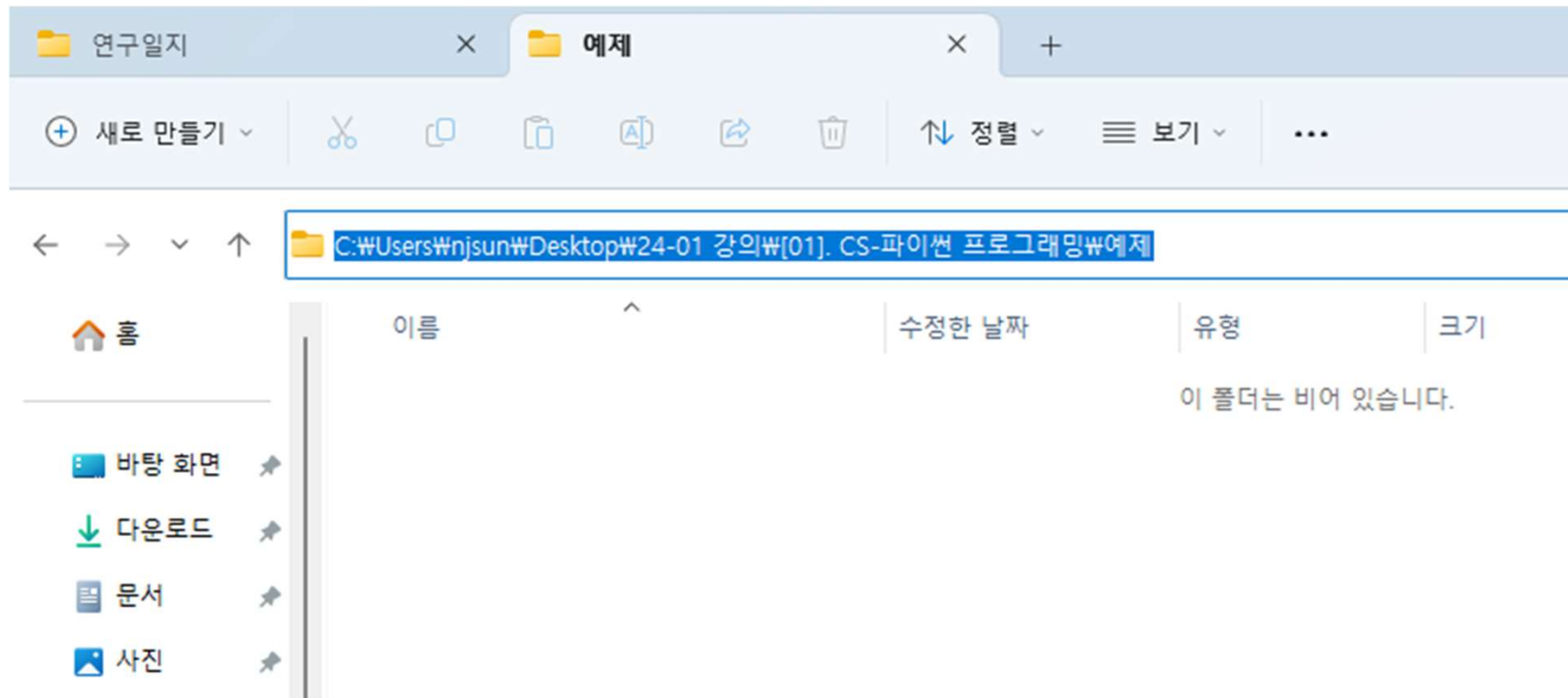
# Visual Studio Code 설치하기

- Visual Studio Code 설치
  - Python 검색 후 가장 위에 나오는 Python Install 버튼 클릭





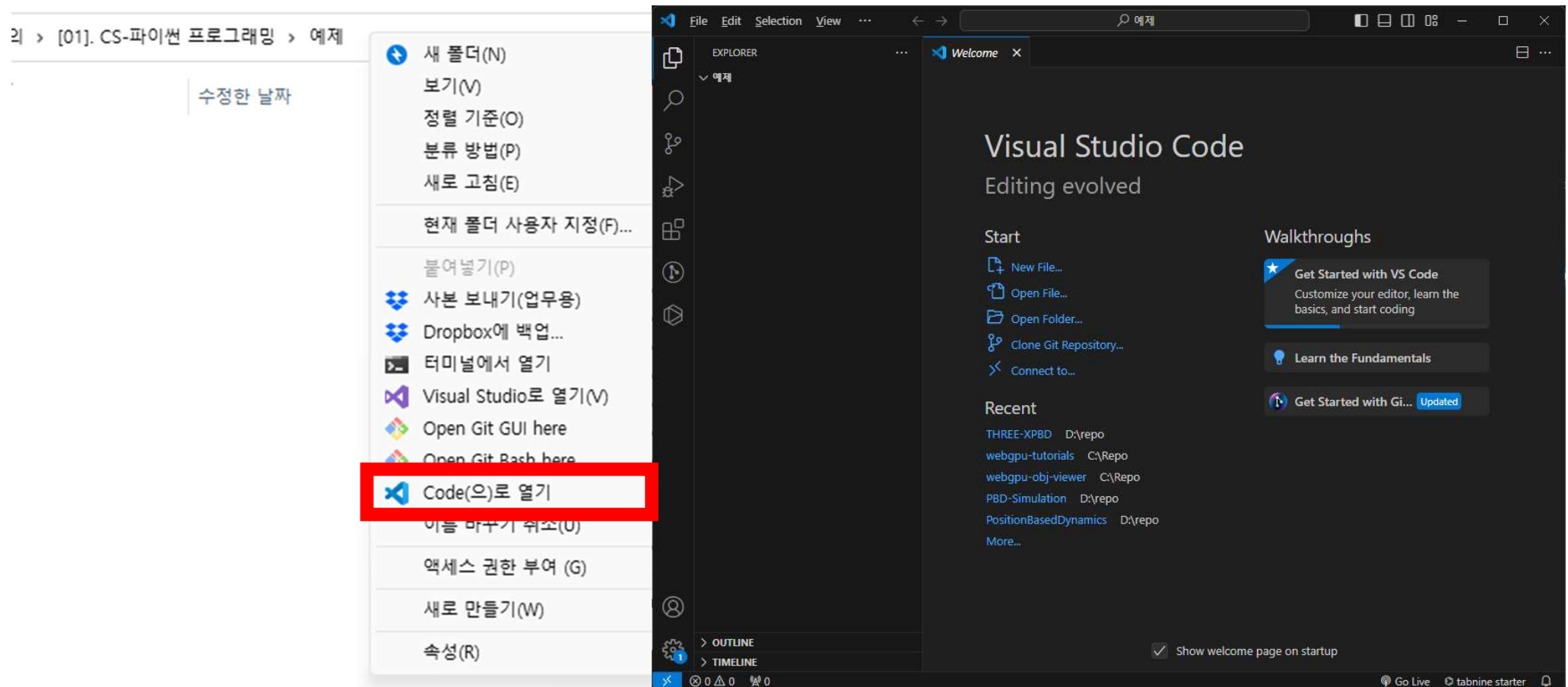
- Python 프로젝트를 저장할 폴더를 새로 생성
  - 예시) C:\Users\wnjsun\Desktop\24-01 강의\01]. CS-파이썬 프로그래밍\예제
  - 예시) D:\Python\Projects





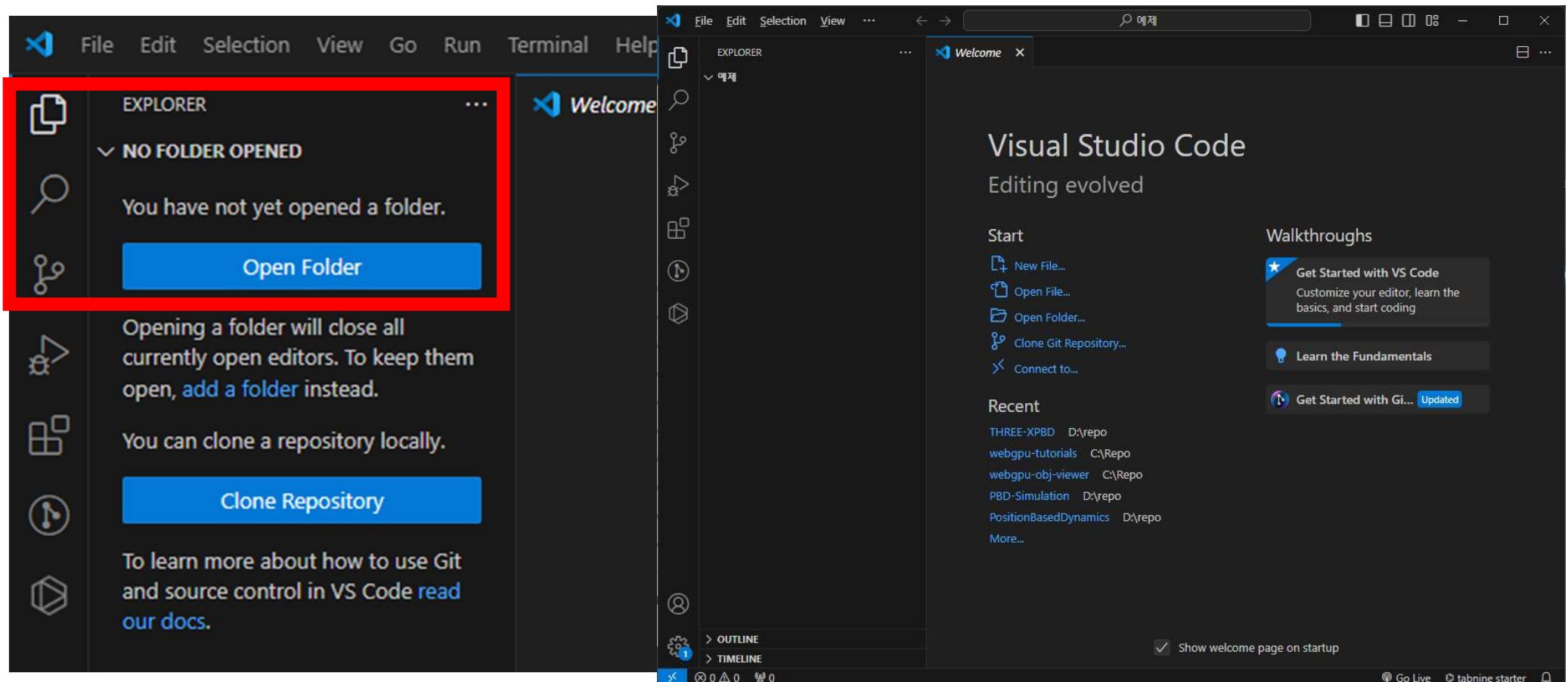


- vs code에서 프로젝트 폴더 열기
  - 방법 1
    - 생성한 폴더의 빈 공간에서 마우스 우클릭
    - Code로 열기 선택



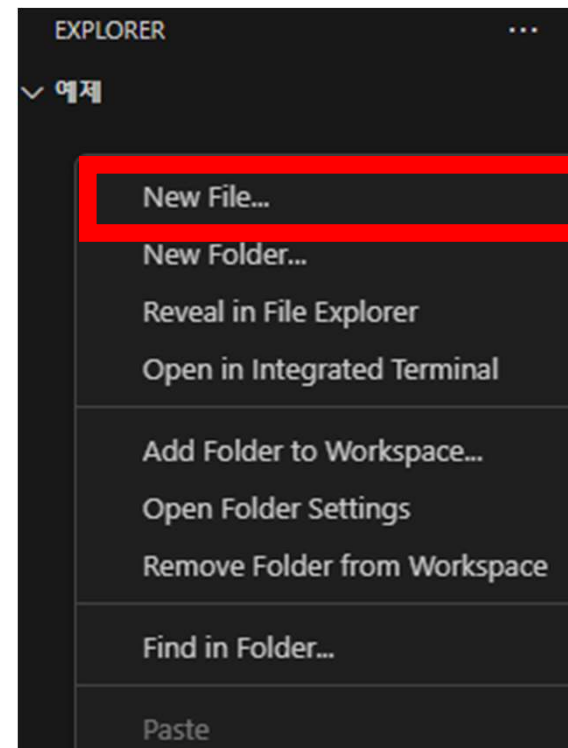
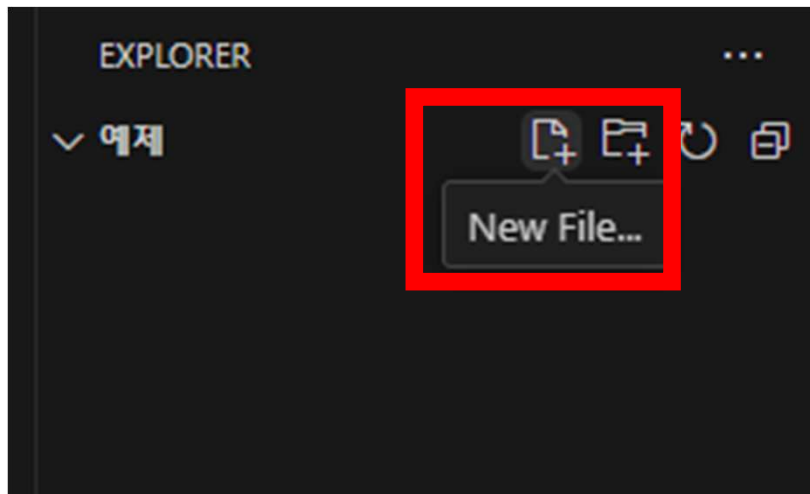


- vs code에서 프로젝트 폴더 열기
  - 방법 2
    - Vscode의 파일 탐색기 메뉴에서 Open Folder 선택
    - 생성한 폴더를 찾아가서 선택 완료



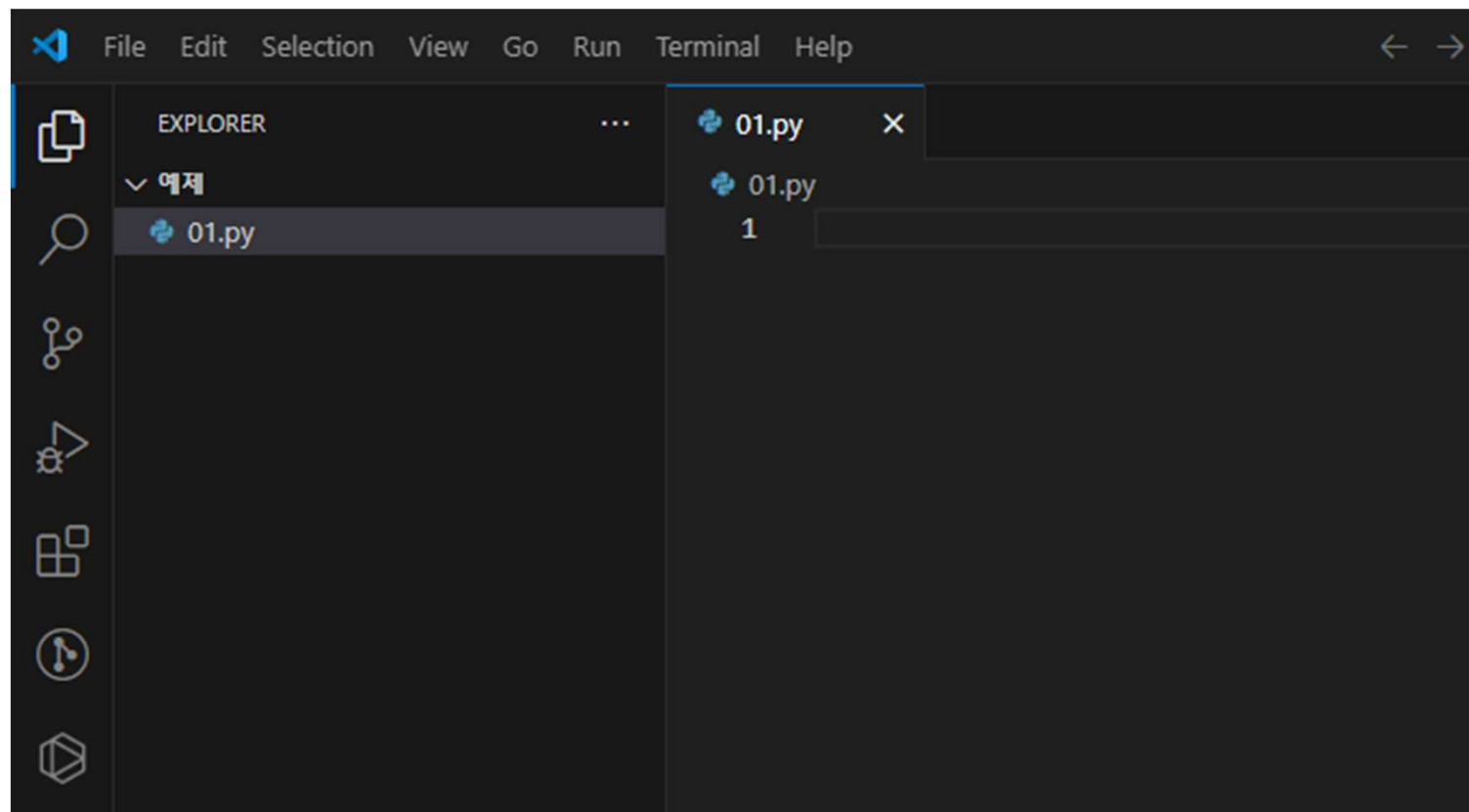


- Python 파일 생성하기
  - 방법 1) Explorer에서 New File 버튼 클릭하기
  - 방법 2) Explorer의 빈 공간에서 우클릭 후 New File 버튼 클릭하기



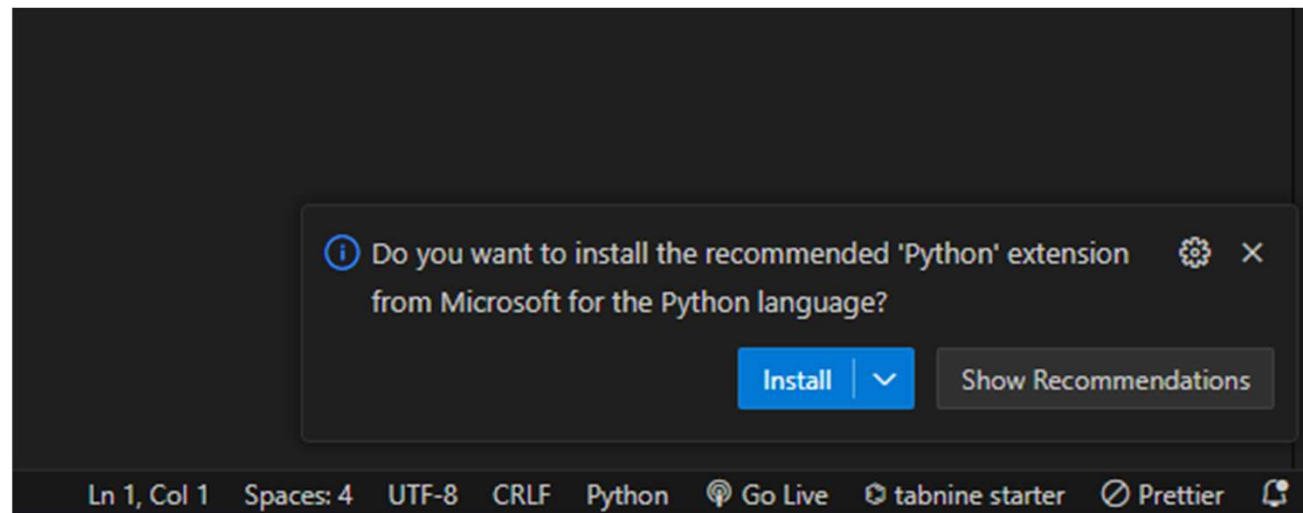


- Python 파일 생성하기
  - 파일 이름을 01.py 로 작성 후 생성 완료



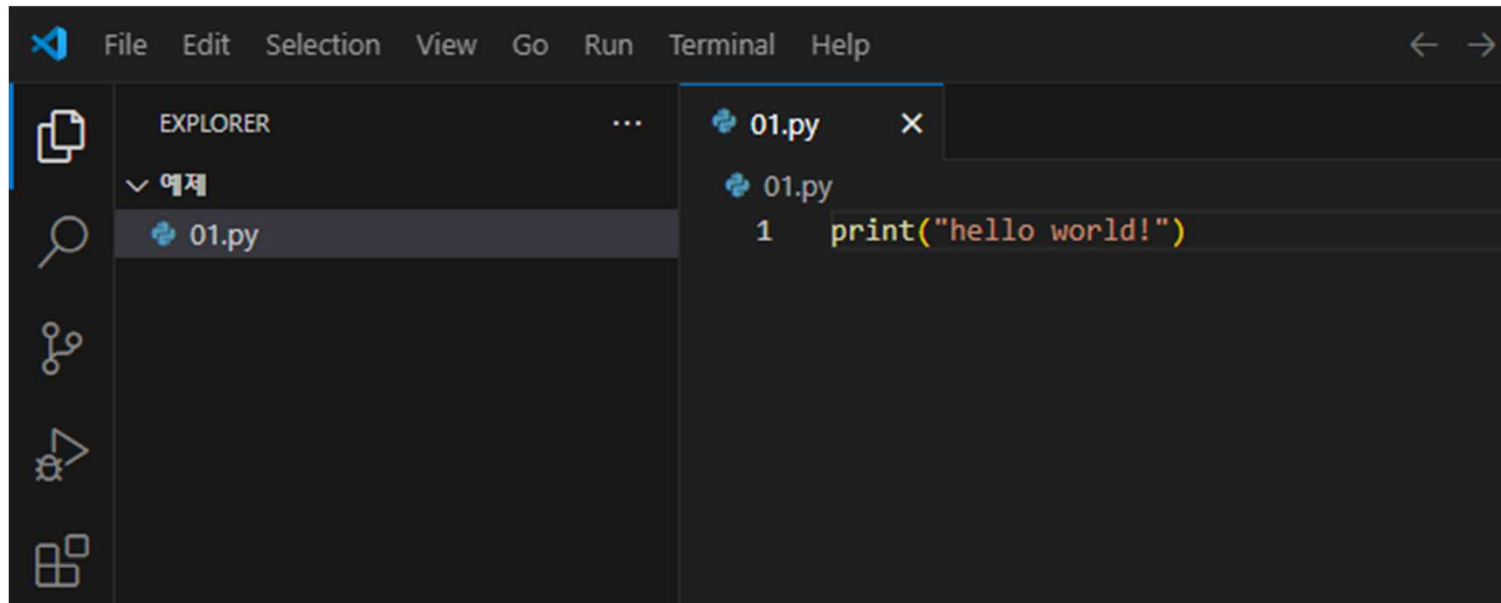


- 혹시?
  - 이런 문구가 오른쪽 아래에 나온다면 Python extension이 설치되지 않은 것이므로 설치하기(Install) 버튼 클릭





- Python 파일 생성하기
  - 생성한 01.py 파일에 가장 간단한 프로그래밍의 시작 코드 `print("hello world!")` 입력

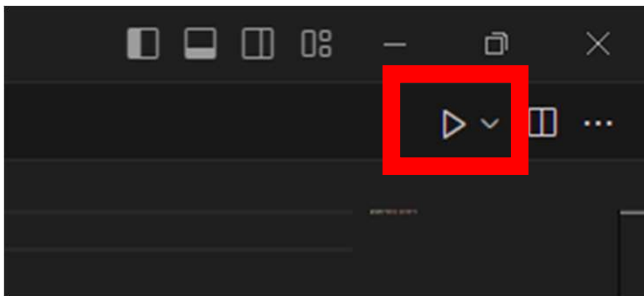






## ■ Python 파일 실행하기

- 작성이 완료된 01.py를 실행하려면 vscode 오른쪽 상단에 있는 ▶ 버튼을 찾아 클릭하면 됨
- 정상적으로 실행될 경우 vscode 하단의 TERMINAL 메뉴에서 여러분들의 hello world! 가 출력될 것



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS Python + v [icon] [icon] ... ^ X
PS C:\Windows\System32\WindowsPowerShell\v1.0> & C:/Users/njsun/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/njsun/Desktop/24-01 강의/[01]. CS-파이썬 프로그래밍/예제/01.py"
hello world!
PS C:\Windows\System32\WindowsPowerShell\v1.0> [ ]
```



- Google Colab(Colaboratory) 접속
  - <https://colab.research.google.com/>

Colaboratory에 오신 것을 환영합니다  
파일 수정 보기 삽입 런타임 도구 도움말

공유 설정 로그인

목차

- 시작하기
- 데이터 과학
- 머신러닝
- 추가 리소스
- 주천 예시

+ 섹션

Colab 시작 페이지

(신규) Gemini API를 사용해 보세요

- [Generate a Gemini API key](#)
- [Talk to Gemini with the Speech-to-Text API](#)
- [Gemini API: Quickstart with Python](#)
- [Gemini API code sample](#)
- [Compare Gemini with ChatGPT](#)
- [More notebooks](#)

Colab에 이미 익숙하다면 이 동영상을 통해 양방향 테이블, 코드 실행 기록 보기, 명령어 팔레트에 관해 알아보세요.

3 Cool Google Colab Features

[ ]

Colab이란?

Colaboratory(줄여서 'Colab'이라고 함)을 통해 브라우저 내에서 Python 스크립트를 작성하고 실행할 수 있습니다.

- 구성이 필요하지 않음
- 무료로 GPU 사용



- Google Colab(Colaboratory) 접속
  - 목차 내 항목에는 주제별 튜토리얼(예제)이 있음

The screenshot shows the Google Colab web interface. On the left, a sidebar menu is highlighted with a red box, containing options like '시작하기' (Get started), '데이터 과학' (Data science), '머신러닝' (Machine learning), '추가 리소스' (Additional resources), and '추천 예시' (Recommended examples). The main content area displays a tutorial titled '데이터 과학' (Data science). It includes a text block explaining that Colab can be used to analyze and visualize data using Python libraries like Numpy. Below the text is a code cell with Python code that generates a sample visualization. The code imports numpy, IPython.display, matplotlib.pyplot, io, and base64. It creates a random array 'ys' and a list 'x' of indices. It then plots 'x' against 'ys' with a green shaded area under the line. The plot is titled 'Sample Visualization' and is displayed as a markdown image. The resulting visualization is a line plot with a green shaded area underneath, titled 'Sample Visualization'.

```
[ ] import numpy as np
import IPython.display as display
from matplotlib import pyplot as plt
import io
import base64

ys = 200 + np.random.randn(100)
x = [x for x in range(len(ys))]

fig = plt.figure(figsize=(4, 3), facecolor='w')
plt.plot(x, ys, '-')
plt.fill_between(x, ys, 195, where=(ys > 195), facecolor='g', alpha=0.6)
plt.title("Sample Visualization", fontsize=10)

data = io.BytesIO()
plt.savefig(data)
image = F"data:image/png;base64,{base64.b64encode(data.getvalue()).decode()}"
alt = "Sample Visualization"
display.display(display.Markdown(F"!![{alt}]({image})"))
plt.close(fig)
```

Sample Visualization

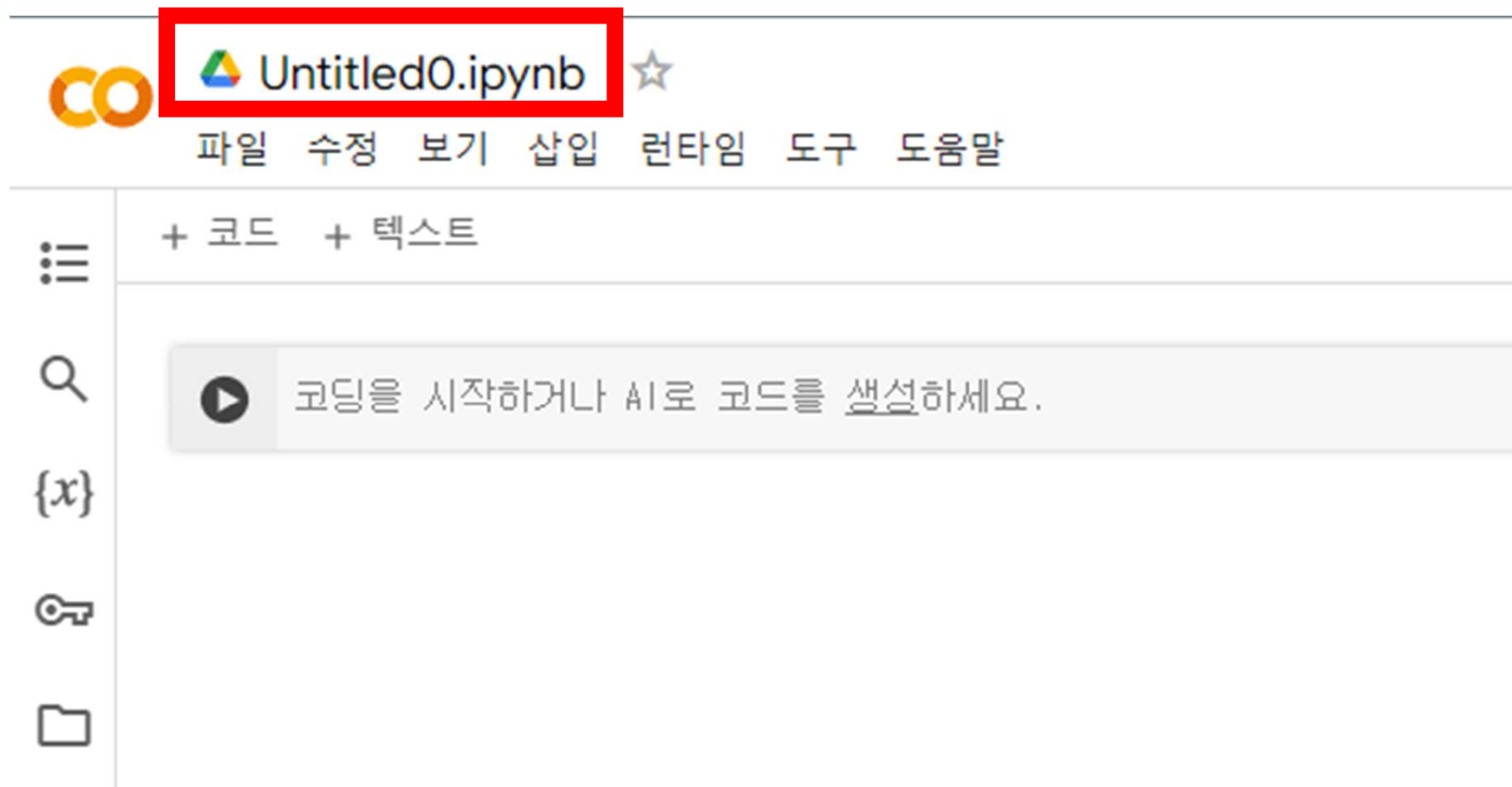


- Google Colab(Colaboratory) 접속
  - 상단 메뉴의 파일 -> 새 노트 선택



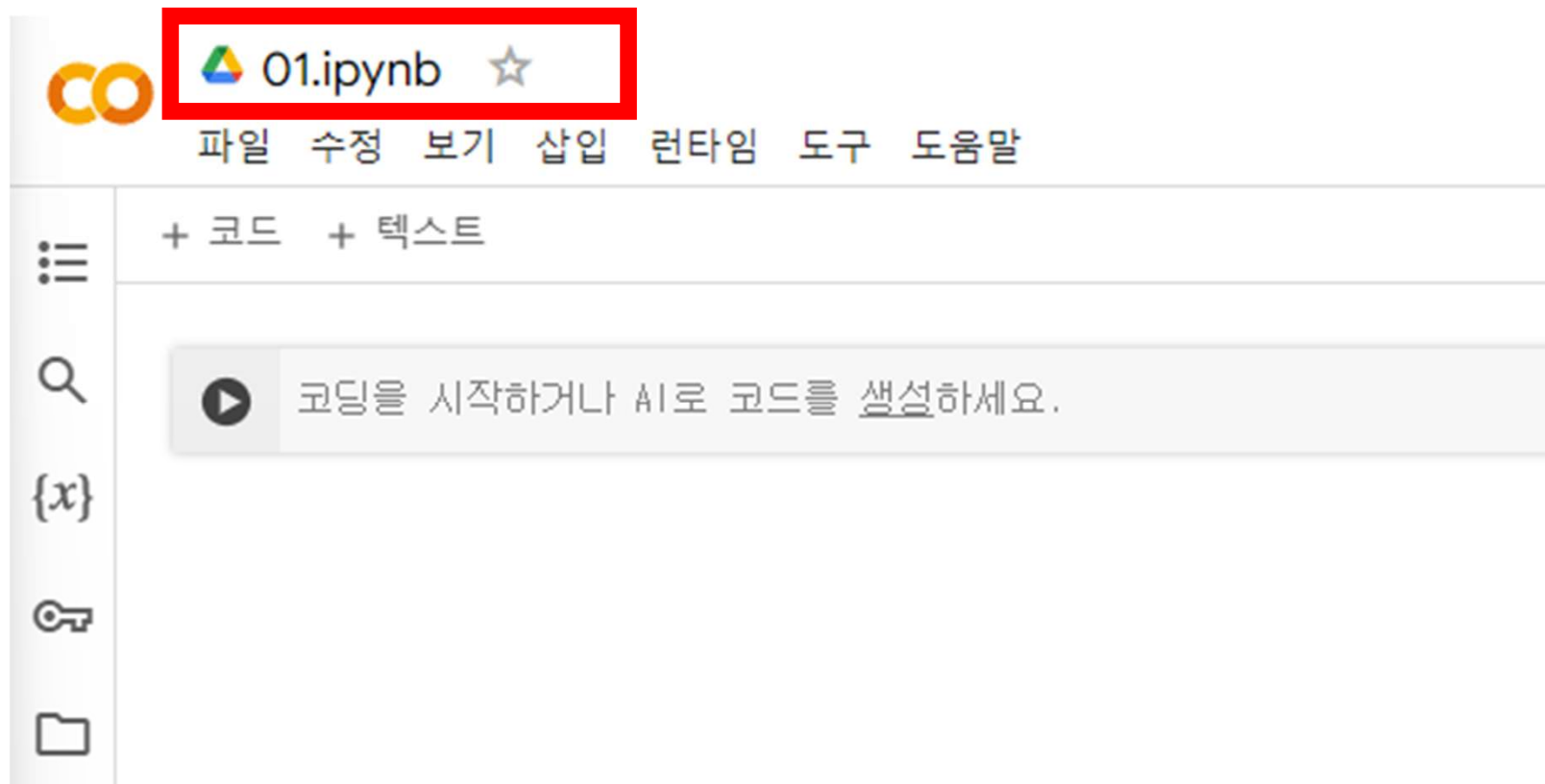


- Google Colab(Colaboratory) 접속
  - 제목 부분을 선택해서 원하는 파일 이름으로 변경





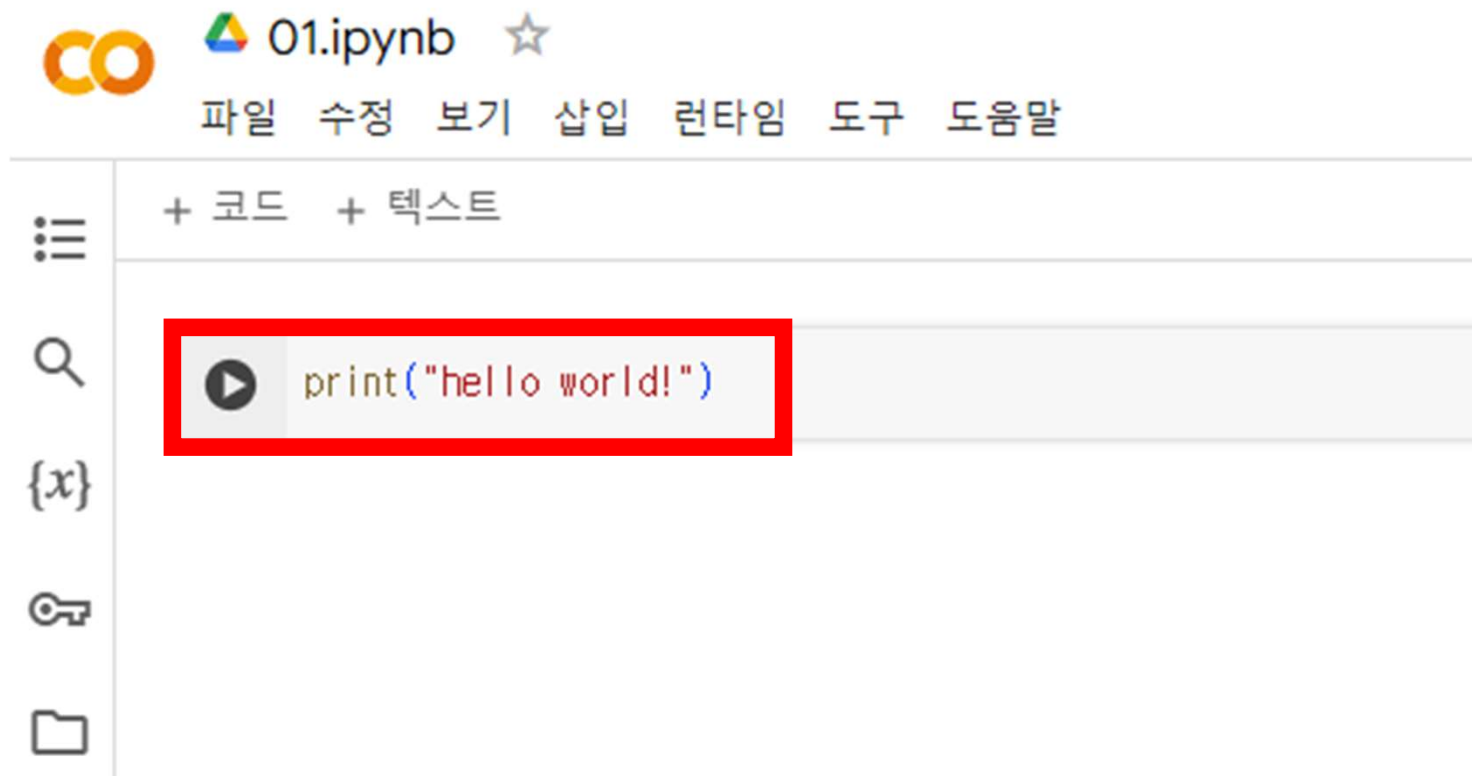
- Google Colab(Colaboratory) 접속
  - 제목 부분을 선택해서 원하는 파일 이름으로 변경





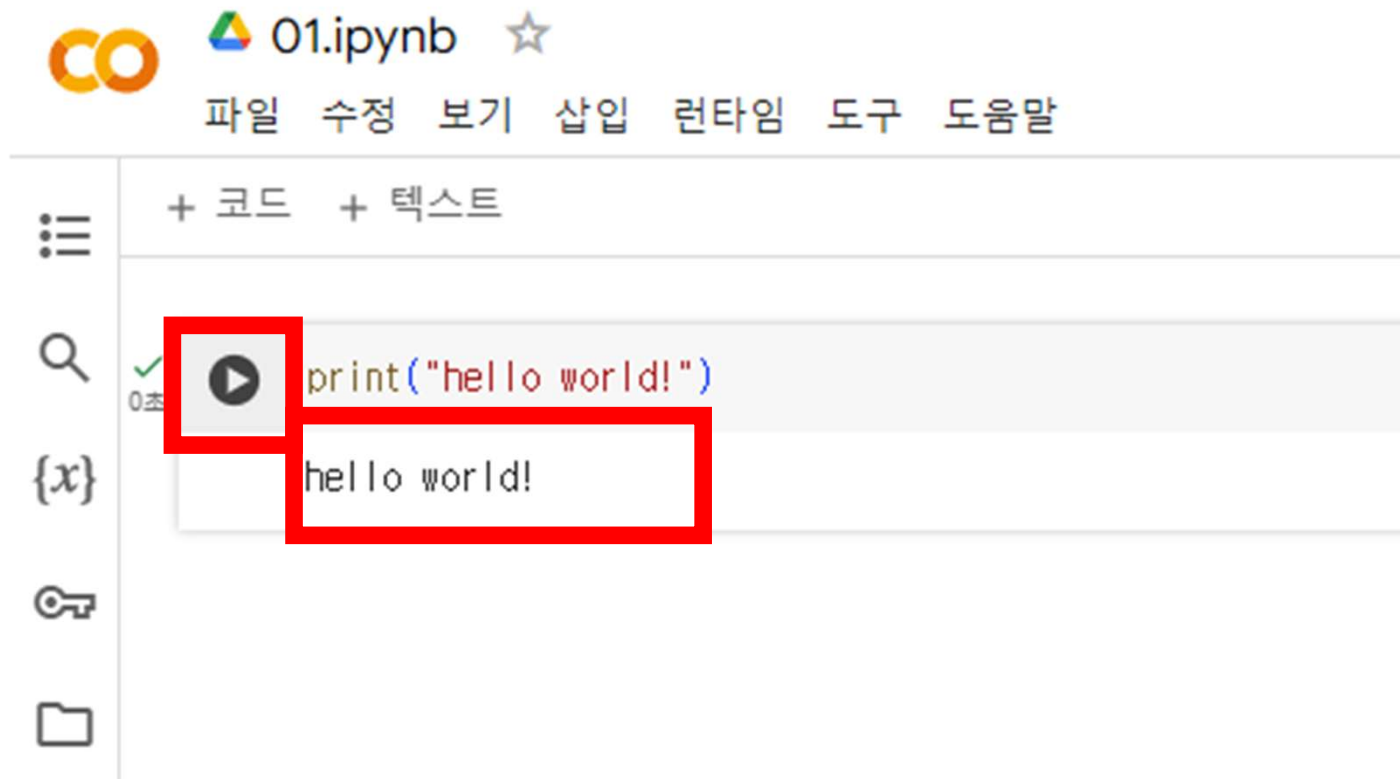


- Google Colab(Colaboratory) 접속
  - 코드 영역에 `print("hello world!")` 작성





- Google Colab(Colaboratory) 접속
  - 코드 부분 좌측의 ▶ 버튼을 클릭해서 실행해서 결과 확인



**THANK**  

---

**YOU**