

1장 파이썬 프로그램

목차

1. 파이썬 소개
2. 파이썬의 특징
3. 파이썬 활용
4. 파이썬 개발 환경 설정
5. 파이썬 개발 툴 소개



1. 파이썬 소개

▶ 파이썬(Python)

- ▶ 1990년 암스테르담의 귀도 반 로섬(Guido Van Rossum)이 개발한 인터프리터 언어
- ▶ 파이썬의 사전적 의미
 - ▶ 고대 신화에 나오는 파르나소스 산의 동굴에 살던 큰 뱀
- ▶ 구글에서 만든 소프트웨어의 50% 이상이 파이썬으로 작성됨
- ▶ 인스타그램(Instagram), 넷플릭스(Netflix), 아마존(Amazon) 등에서 사용
- ▶ 공동 작업과 유지 보수가 매우 쉽고 편리함



2. 파이썬 특징

▶ 파이썬은 인간다운 언어이다

- ▶ 파이썬은 사람이 생각하는 방식을 그대로 표현할 수 있는 언어

```
if 4 in [1, 2, 3, 4]: print("4가 있습니다")
```



만약 4가 1, 2, 3, 4 중에 있으면 '4가 있습니다'를 출력한다.



2. 파이썬 특징

▶ 문법이 쉬워 빠르게 배울 수 있다

- ▶ 문법 자체가 아주 쉽고 간결하며 사람의 사고 체계와 매우 닮아 있음
- ▶ 유명한 프로그래머인 에릭 레이먼드(Eric Raymond)는 공부한 지 단 하루 만에 자신이 원하는 프로그램을 작성!

▶ 무료이지만 강력하다

- ▶ 오픈 소스 → 무료로 언제 어디서든 파이썬을 다운로드하여 사용 가능
- ▶ 파이썬과 C는 찰떡 궁합
 - ▶ 프로그램의 전반적인 뼈대는 파이썬으로 만들고,
빠른 실행 속도가 필요한 부분은 C로 만들어서 파이썬 프로그램 안에 포함



2. 파이썬의 특징

▶ 간결하다

펄(Perl)	파이썬
100가지 방법으로 1가지 일 처리	가장 좋은 방법 1가지만 사용

프로그램이 실행되게 하려면
줄(들여쓰기)을 반드시 맞추어야 함
가독성 ↑

```
languages = ['python', 'perl', 'c', 'java']

for lang in languages:
    if lang in ['python', 'perl']:
        print("%6s need interpreter" % lang)
    elif lang in ['c', 'java']:
        print("%6s need compiler" % lang)
    else:
        print("should not reach here")
```



2. 파이썬의 특징

- ▶ **프로그래밍을 즐기게 해 준다**

- ▶ 프로그래머가 만들고자 하는 기능에만 집중할 수 있음

- ▶ **개발 속도가 빠르다**

“Life is too short, You need Python”

인생은 너무 짧으니 파이썬이 필요해.

- ▶ 파이썬의 엄청나게 빠른 개발 속도를 두고 유행처럼 퍼진 말



2. 파이썬의 특징

▶ 플랫폼 독립적인 언어

- ▶ 어떤 운영체제든 상관없이 사용할 수 있는 언어.

▶ 인터프리터 언어

- ▶ 컴파일러 언어와 달리, 소스코드 자체가 바로 실행되는 특징이 있는 언어. 속도는 느리지만, 굉장히 간편하게 사용할 수 있다.

구분	컴파일러	인터프리터
작동 방식	소스코드를 기계어로 먼저 번역하고, 해당 플랫폼에 최적화되어 프로그램을 실행함	별도의 번역 과정 없이 소스코드를 실행 시점에 해석하여 컴퓨터가 처리할 수 있도록 함
장점	실행 속도가 빠름	간단히 작성, 메모리가 적게 필요
단점	한 번에 많은 기억 장소가 필요함	실행 속도가 느림
주요 언어	C, 자바(Java), C++, C#	파이썬, 스칼라

2. 파이썬의 특징

▶ 객체 지향 언어

- ▶ 해당 프로그램이 해결해야 할 문제의 구성요소를 요소별로 정의한 뒤, 각 요소의 기능(메서드)과 정보(속성)를 정의하여 요소들을 결합하고, 프로그램을 작성하는 방식.











▶ 동적 타이핑 언어

- ▶ 프로그램의 실행 시점에서 각 프로그램 변수의 타입을 결정하는 언어.



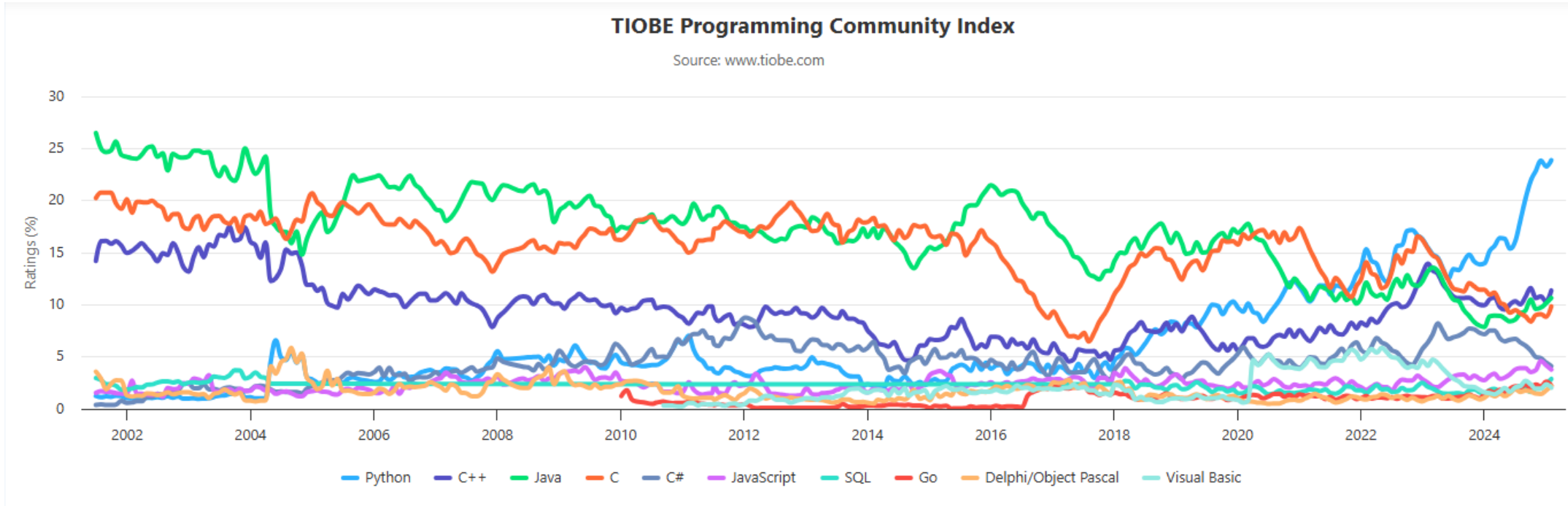
3. 파이썬 활용

▶ 프로그래밍언어 사용 순위(TIOBE Index for Feb 2025)

Feb 2025	Feb 2024	Change	Programming Language		Ratings	Change
1	1			Python	23.88%	+8.72%
2	3	▲		C++	11.37%	+0.84%
3	4	▲		Java	10.66%	+1.79%
4	2	▼		C	9.84%	-1.14%
5	5			C#	4.12%	-3.41%
6	6			JavaScript	3.78%	+0.61%
7	7			SQL	2.87%	+1.04%
8	8			Go	2.26%	+0.53%
9	12	▲		Delphi/Object Pascal	2.18%	+0.78%
10	9	▼		Visual Basic	2.04%	+0.52%

3. 파이썬 활용

▶ 프로그래밍 언어 사용 트렌드



3. 파이썬 활용

▶ 파이썬으로 할 수 있는 일

▶ 웹 프로그래밍

- ▶ 파이썬은 웹 프로그램을 만들기에 매우 적합한 도구
- ▶ 파이썬으로 제작한 웹 사이트는 셀 수 없이 많을 정도

▶ 인공지능과 머신러닝

- ▶ 자연어 처리, 음성 인식, 이미지 인식과 같은 인공지능 기술 구현
- ▶ 인공지능과 머신러닝 프로그래밍을 쉽게 할 수 있도록
scikit-learn, TensorFlow, PyTorch, Keras 등과 같은 다양한 라이브러리 제공

▶ 수치 연산 프로그래밍

- ▶ C로 작성된 넘파이(Numpy) 수치 연산 모듈을 통해 빠른 수치 연산 가능



3. 파이썬 활용

▶ 파이썬으로 할 수 있는 일

▶ 데이터 분석

- ▶ NumPy, Pandas, Matplotlib 등의 라이브러리를 활용해 데이터 처리, 통계 분석, 시각화를 손쉽게 수행
- ▶ 판다스가 등장한 이래 데이터 분석에 R보다 파이썬을 사용하는 사례 증가



▶ 데이터베이스 프로그래밍

- ▶ Sybase, Infomix, Oracle, MySQL, PostgreSQL 등의 데이터베이스에 접근하기 위한 도구 제공
- ▶ 자료를 변형 없이 그대로 파일에 저장하고 불러오는 파이썬 모듈 피클(pickle)

▶ 시스템 유틸리티 제작하기

- ▶ 운영체제(윈도우, 리눅스 등)의 시스템 명령어를 사용하는 도구를 통한 시스템 유틸리티 제작



3. 파이썬 활용

▶ 파이썬으로 할 수 있는 일

▶ GUI(Graphic User Interface) 프로그래밍

- ▶ 화면에 윈도우 창을 만들고 프로그램을 동작시킬 수 있는 메뉴나 버튼, 그림 등을 추가하는 것
- ▶ GUI 프로그래밍을 위한 도구들을 갖추고 있어, GUI 프로그램을 만들기 쉬움 (예 - Tkinter(티케이인터))



▶ C/C++와 결합하기

- ▶ C나 C++로 만든 프로그램을 파이썬에서, 파이썬으로 만든 프로그램을 C나 C++에서 사용 가능

▶ 사물 인터넷

- ▶ 라즈베리파이를 제어하며 사물 인터넷 구현



3. 파이썬 활용

▶ 파이썬으로 할 수 없는 일

▶ 시스템과 밀접한 프로그래밍 영역

- ▶ 운영체제, 엄청난 횟수의 반복과 연산이 필요한 프로그램 등
매우 빠른 속도를 요구하거나 하드웨어를 직접 건드릴어야 하는 프로그램에는 적합하지 않음

▶ 모바일 프로그래밍

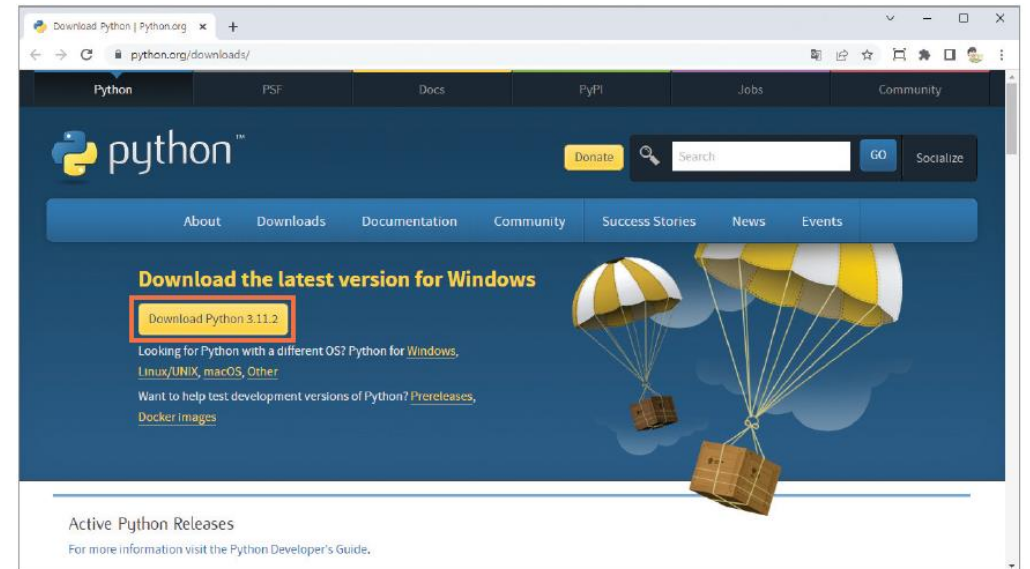
- ▶ 안드로이드 네이티브 앱(android native app) 개발에는 아직 역부족
- ▶ 아이폰 앱 개발은 할 수 없음



4. 파이썬 개발 환경 설정

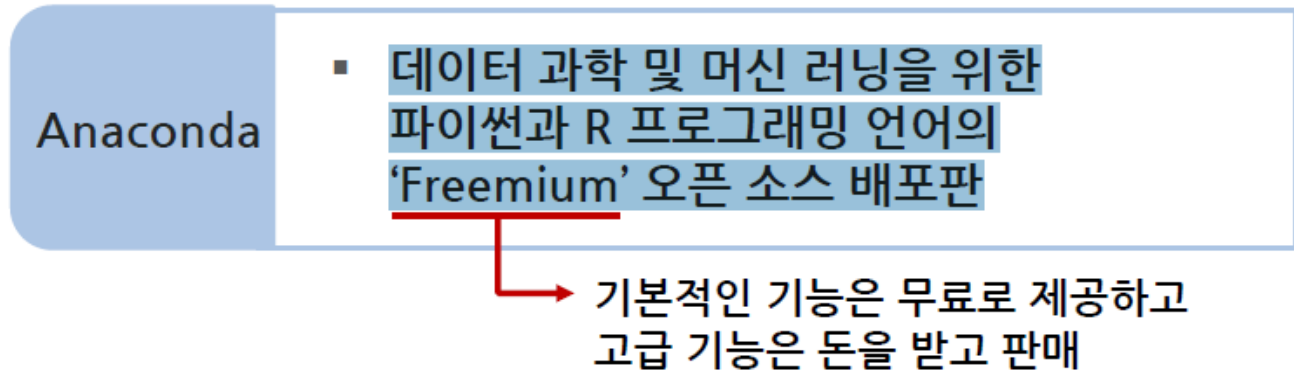
▶ 다운로드 사이트

- 아래사이트에서 원하는 버전의 파이썬 설치 프로그램을 다운로드하여 설치 후 **필요한 라이브러리와 개발 툴 추가 설치**
 - <https://www.python.org/downloads/>
 - 일반적인 Python , 기본 모듈 포함
- 아래사이트에서 Anaconda Distribution 설치
 - <https://www.anaconda.com/download>
 - 다양한 과학 계산용 모듈을 묶어 패키지로 제공



2. 파이썬 개발 환경 설정


▶ 파이썬 인터프리터 설치 : Anaconda



- 가장 유명한 파이썬 데이터 과학 플랫폼
 - 모든 데이터 과학패키지의 쉬운설치, 패키지·종속성 및 환경관리 가능
 - CONDA, 파이썬 및 150개가 넘는 과학 패키지와 그 종속성과 함께 제공되는파이썬 배포판
- ❖ CONDA : 패키지 및 환경관리자

2. 파이썬 개발 환경 설정

▶ 파이썬 인터프리터 설치 : Anaconda

 [Products](#) [Solutions](#) [Resources](#) [Partners](#) [Company](#)

Free Download

Sign Up

Sign In

Distribution

Register to get everything you need to get started on your workstation

- ✓ Distribution installation on Windows, MacOS, or Linux
- ✓ Easily search and install thousands of data science, machine learning, and AI packages
- ✓ Manage packages and environments from a desktop application or work from the command line
- ✓ Deploy across hardware and software platforms

Commercial use at a company of more than 200 employees requires a Business or Enterprise license. [See Pricing](#)

Provide email to download Distribution

Don't miss out! Get access to: Cloud Notebooks, Anaconda Assistant, easy application deployment, learning resources, and updates from Anaconda.

Email Address:

☐ I agree to receive communication from Anaconda regarding relevant content, products, and services. I understand that I can revoke this consent [here](#) at any time.

By continuing, I agree to Anaconda's [Privacy Policy](#) and [Terms of Service](#).

Submit >

Skip registration

2. 파이썬 개발 환경 설정

▶ 파이썬 인터프리터 설치 : Anaconda

Anaconda Installers



Windows

Python 3.12

📄 64-Bit Graphical Installer (912.3M)



Mac

Python 3.12

- 📄 64-Bit (Apple silicon) Graphical Installer (704.7M)
- 📄 64-Bit (Apple silicon) Command Line Installer (707.3M)
- 📄 64-Bit (Intel chip) Graphical Installer (734.7M)
- 📄 64-Bit (Intel chip) Command Line Installer (731.2M)



Linux

Python 3.12

- 📄 64-Bit (x86) Installer (1007.9M)
- 📄 64-Bit (AWS Graviton2 / ARM64) Installer (800.6M)
- 📄 64-bit (Linux on IBM Z & LinuxONE) Installer (425.8M)

2. 파이썬 개발 환경 설정

▶ 파이썬 인터프리터 설치 : Anaconda

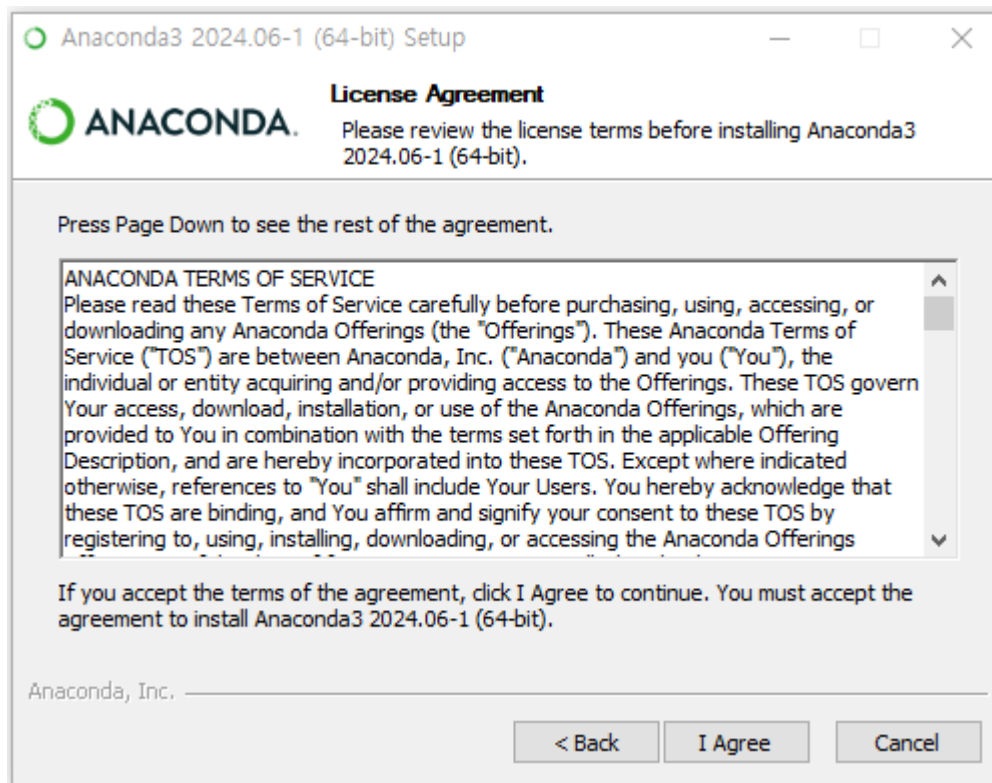
- ▶ 파이썬 개발환경과 수학 ·과학 ·데이터분석분야에서 필요한 거의 모든 패키지(NumPy, SciPy, Pandas, Matplotlib 등) 포함



2. 파이썬 개발 환경 설정

▶ Anaconda 설치

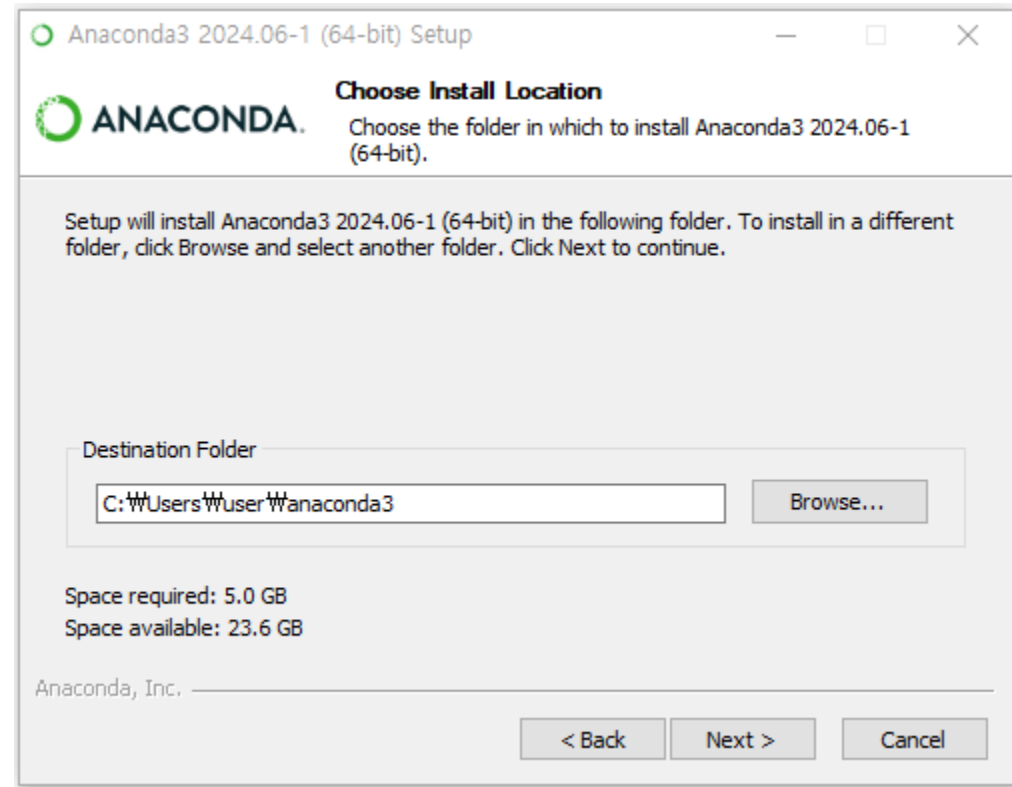
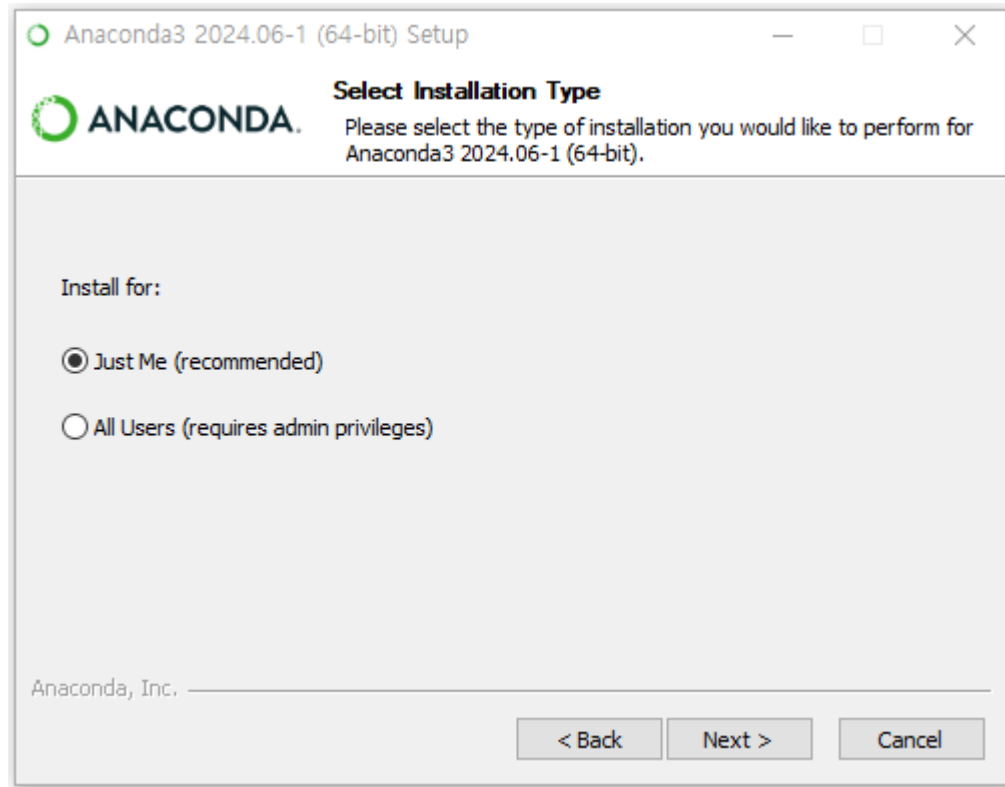
- ▶ 다운로드한 인스톨러를 실행하고 [Next]를 클릭한 후, 다음 화면에서 [I Agree]를 클릭한다



2. 파이썬 개발 환경 설정

▶ Anaconda 설치

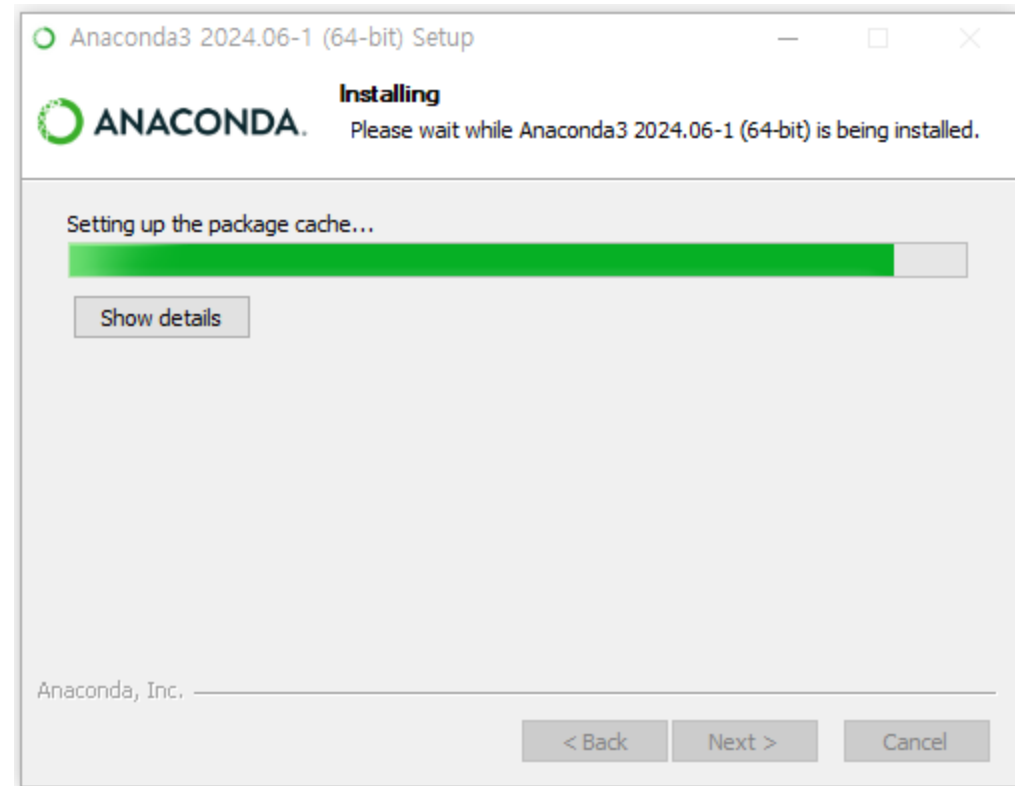
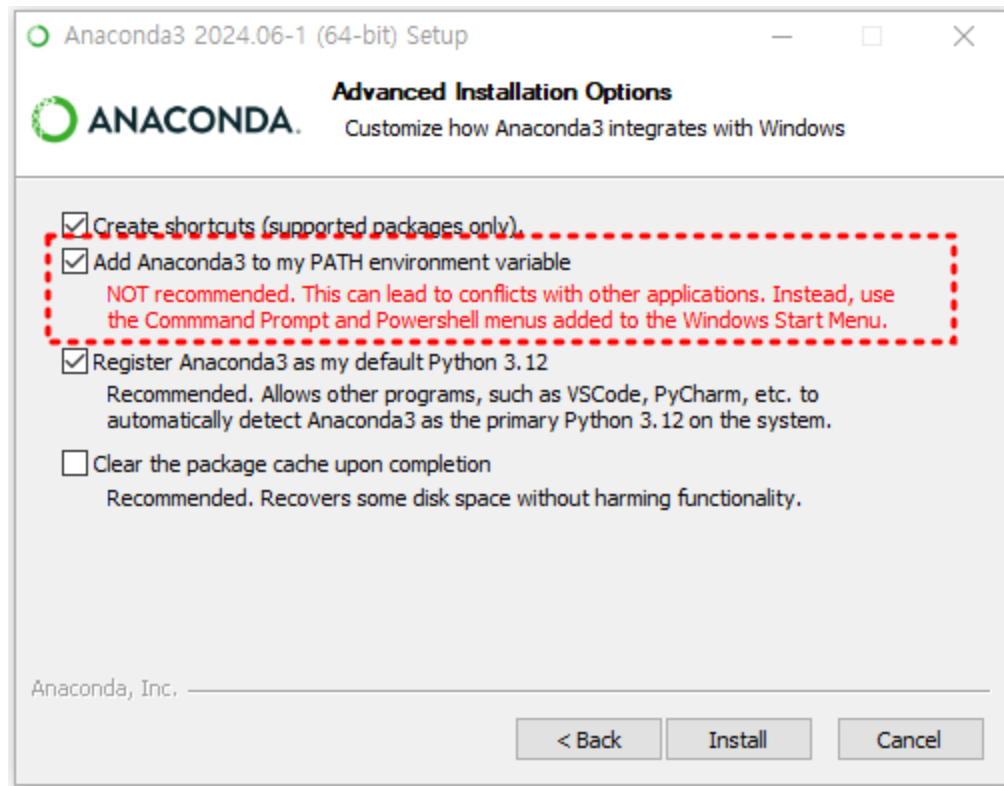
- ▶ 인스톨 타입 설정 창에서 'Just Me'를 선택하고, [Next]를 클릭한다.
- ▶ 인스톨 경로 설정 창에서 [Browse]를 클릭하여 적절한 경로를 지정하고, [Next]를 클릭한다.



2. 파이썬 개발 환경 설정

▶ Anaconda 설치

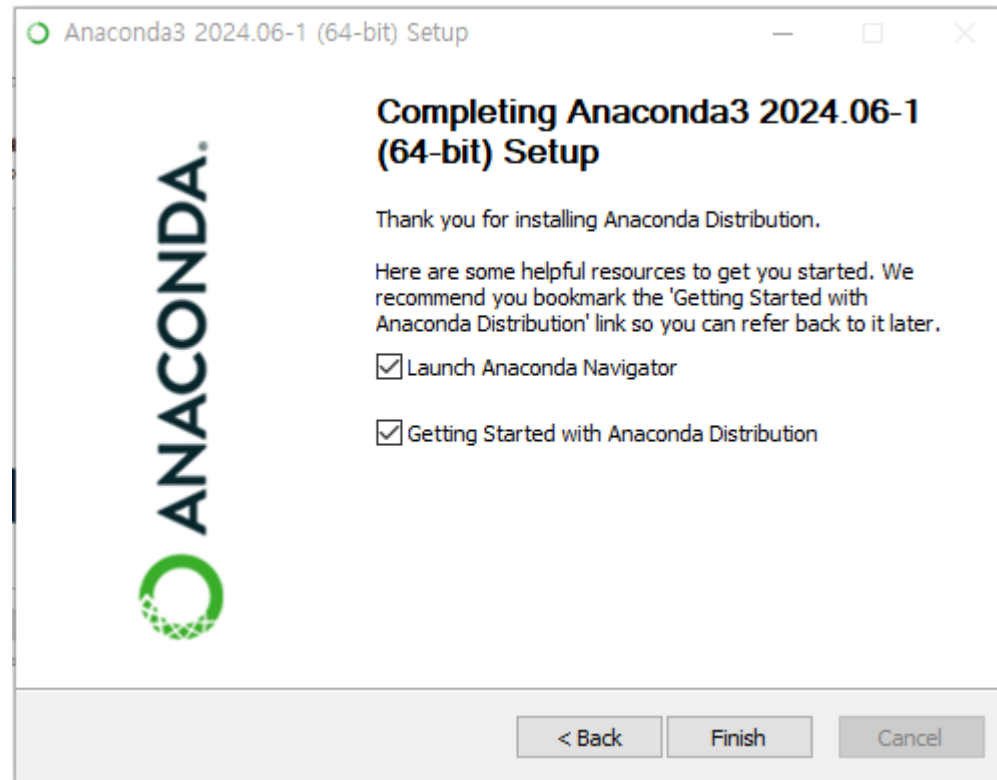
- ▶ 인스톨 옵션 설정 창에서 [Add Anaconda3 to the system PATH environment variable], [Register Anaconda3 as the system Python3.8] 체크박스를 모두 체크한 후, [Install]을 클릭한다.



2. 파이썬 개발 환경 설정

▶ Anaconda 설치

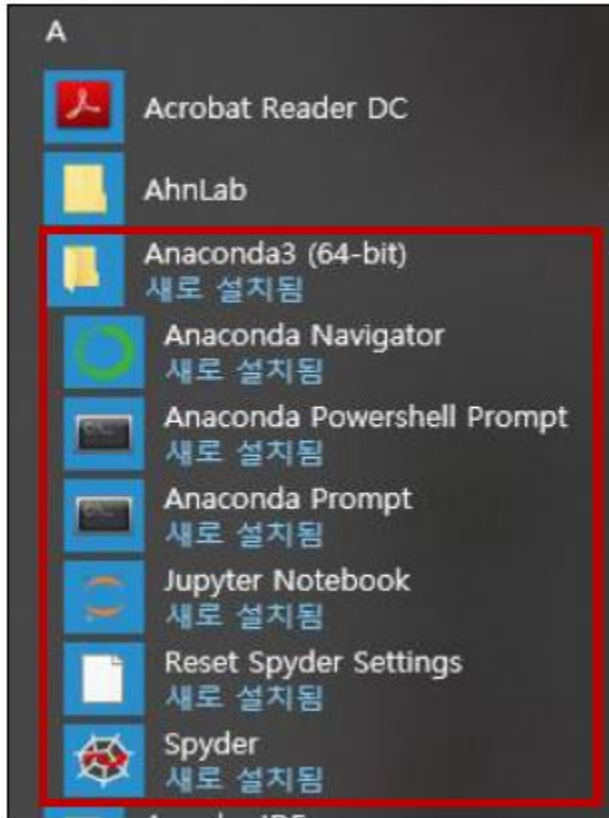
- ▶ 다음화면에서 각각 Next, Finish 버튼을 클릭하여 설치를 완료한다.



2. 파이썬 개발 환경 설정

▶ Anaconda 설치

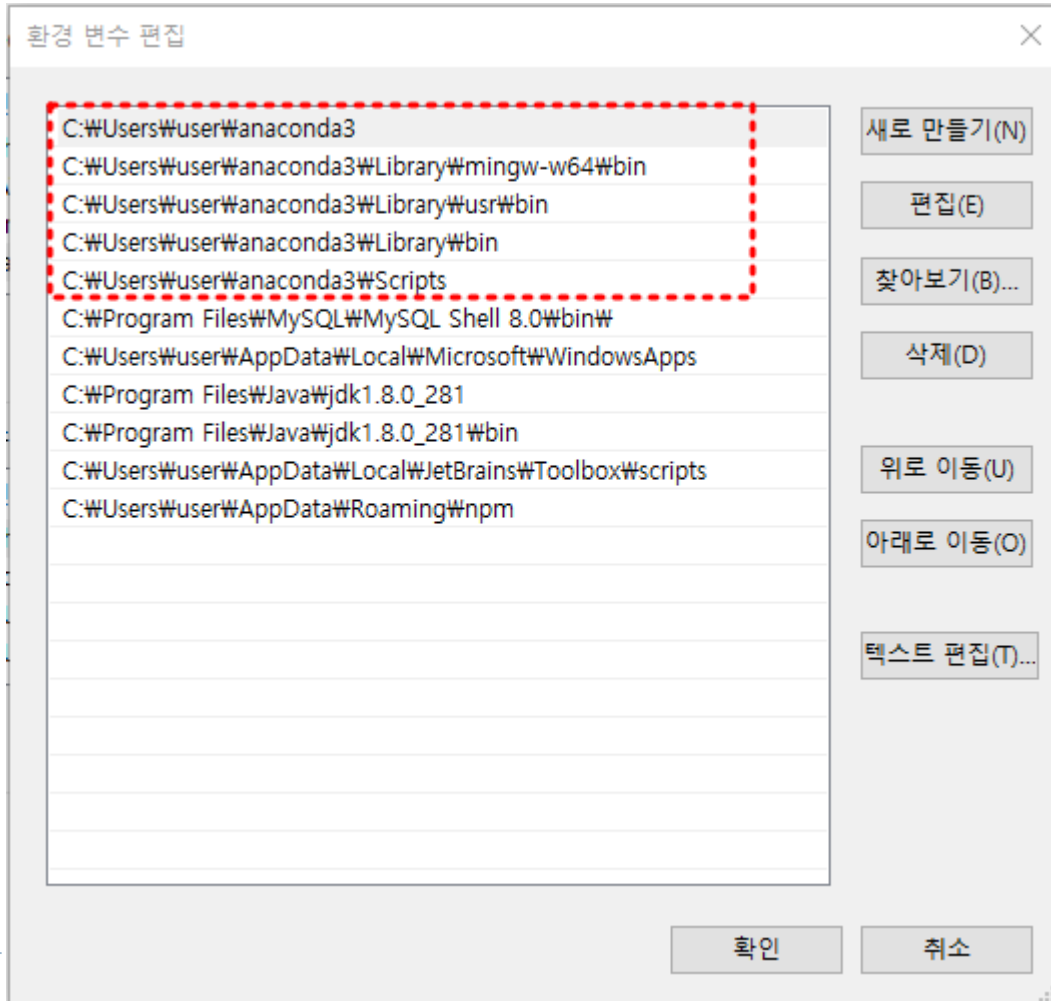
- ▶ Anaconda 설치를 모두 끝내면, 윈도우 운영체제의 '시작' 메뉴를 선택하여 Anaconda의 설치상태 확인



2. 파이썬 개발 환경 설정

▶ Anaconda 설치

- ▶ 윈도우 운영체제의 사용자 환경변수 path 정보를 출력한 화면으로 Anaconda의 패스 설정 확인



3. 파이썬 개발 툴 소개

▶ IDLE

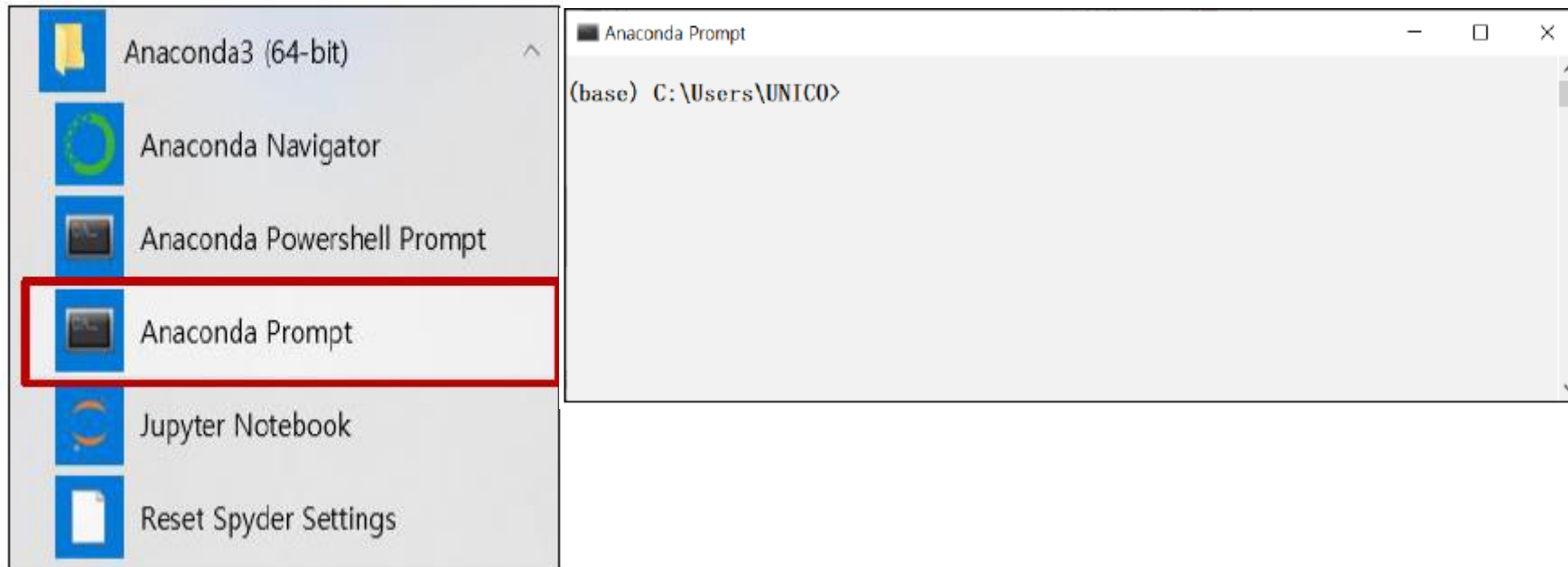


3. 파이썬 개발 툴 소개

▶ IDLE

▶ Anaconda Prompt 창

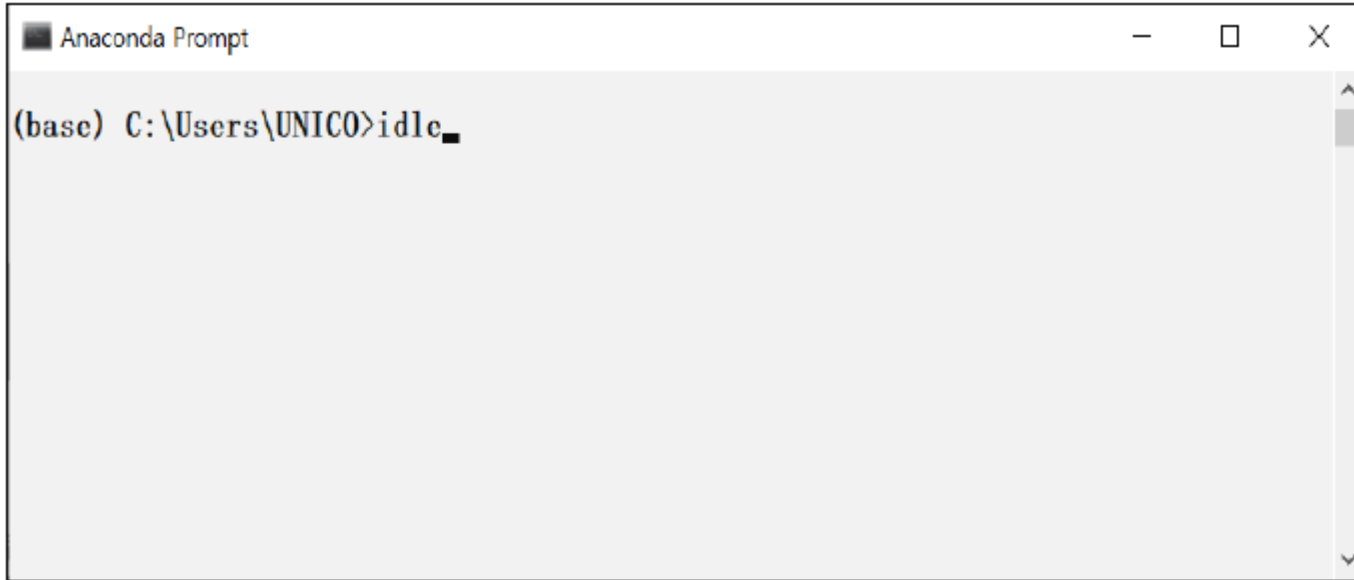
- ▶ 시작메뉴 -> Anaconda3 메뉴 -> Anaconda Prompt 서버메뉴 클릭



3. 파이썬 개발 툴 소개

▶ IDLE

▶ Anaconda Prompt 창에서 'idle' 명령어 입력



```
Anaconda Prompt
(base) C:\Users\UNICO>idle
```

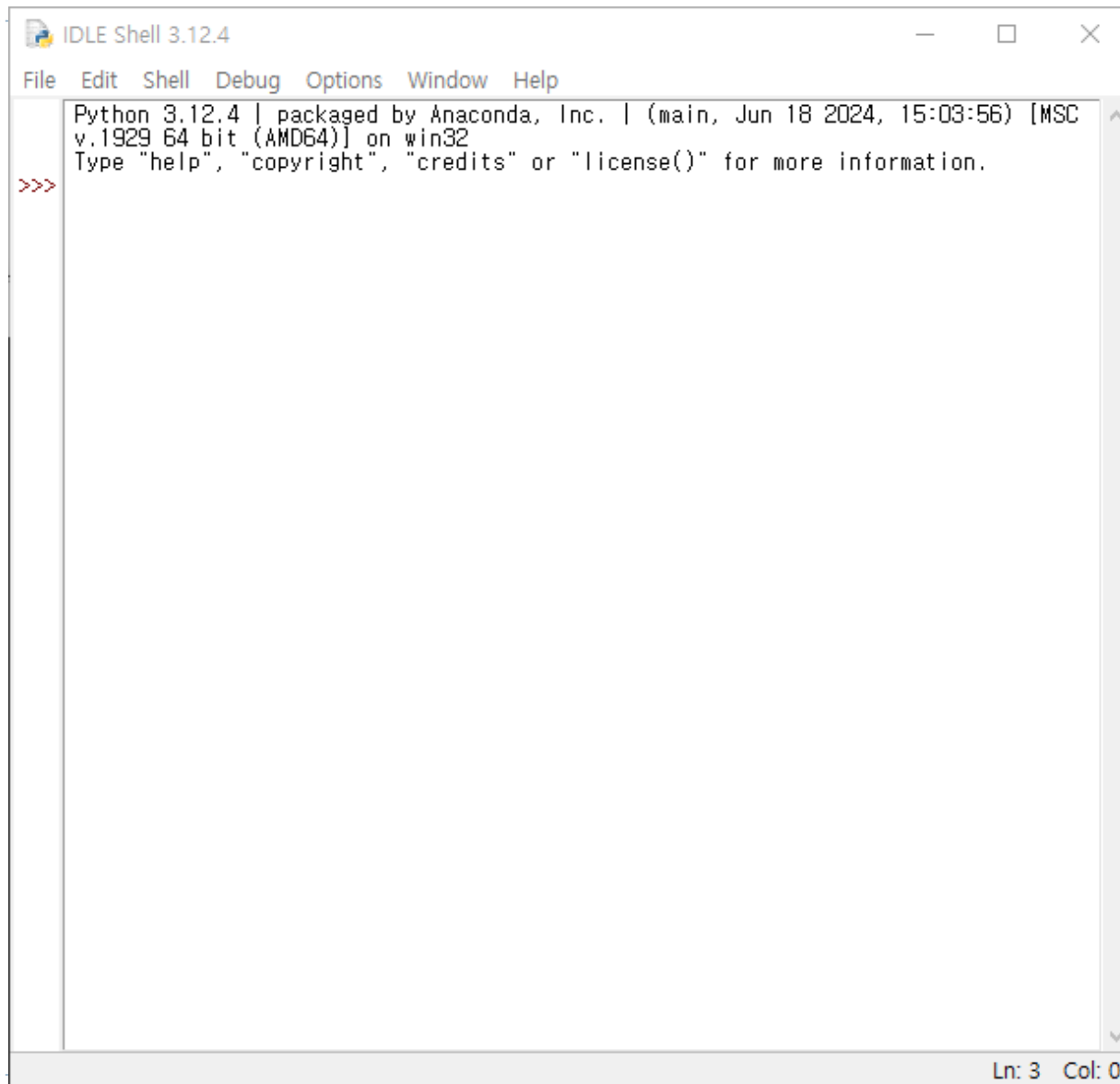
The image shows a screenshot of the Anaconda Prompt terminal window. The window title is 'Anaconda Prompt'. The prompt shows the current directory as 'C:\Users\UNICO' and the command 'idle' has been entered, followed by a cursor. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

3. 파이썬 개발 툴 소개

▶ IDLE

▶ IDLE 윈도우 창 출력

- ▶ '>>>'는 프롬프트(Prompt)라고함
- ▶ 프롬프트에 파이썬 문법에 맞는 코드를 작성하고 엔터키를 누르면 파이썬 인터프리터가 입력된 코드를 해석해 실제 동작 수행



```
IDLE Shell 3.12.4
File Edit Shell Debug Options Window Help
Python 3.12.4 | packaged by Anaconda, Inc. | (main, Jun 18 2024, 15:03:56) [MSC
v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

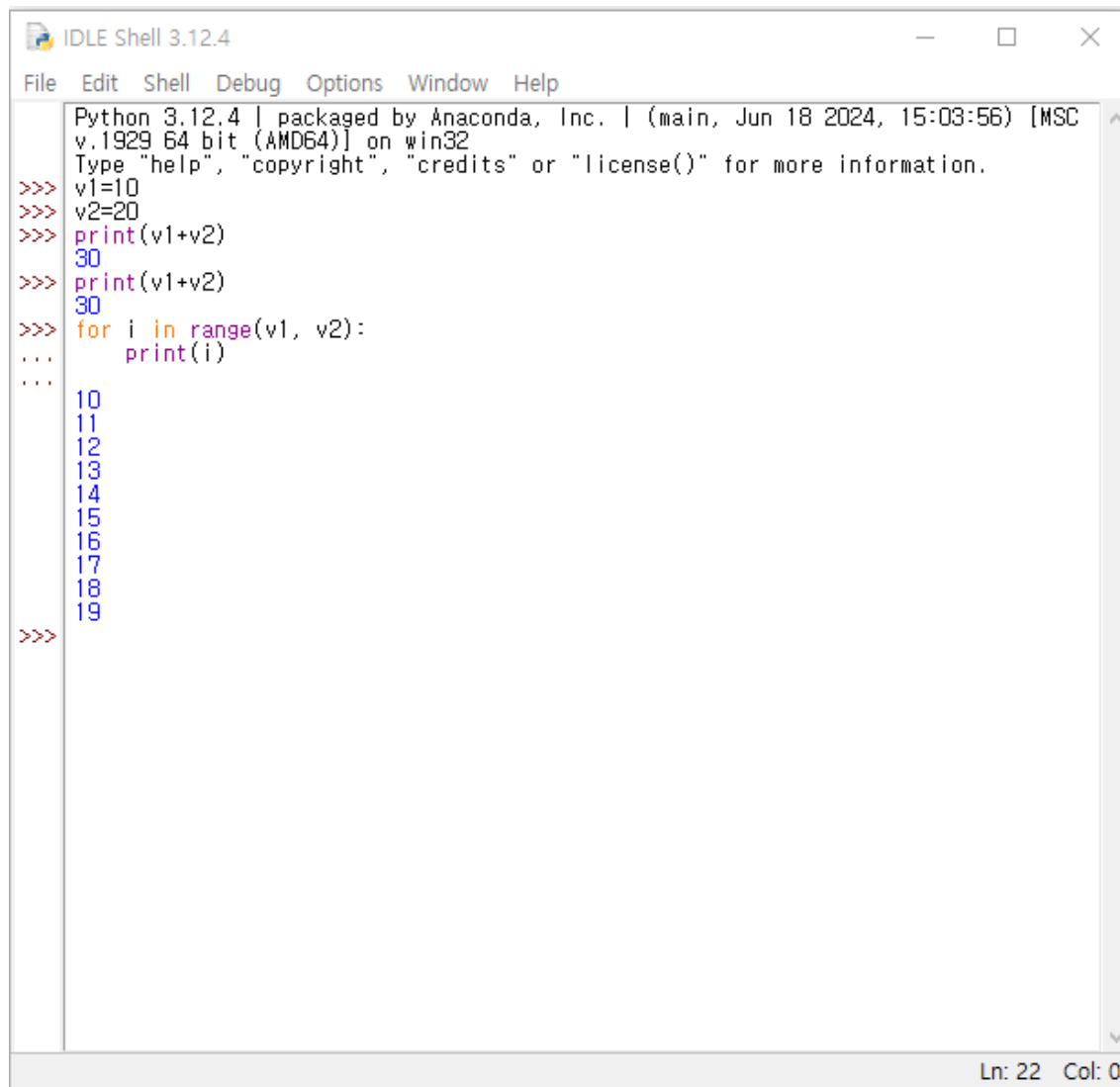
Ln: 3 Col: 0

3. 파이썬 개발 툴 소개

▶ IDLE

▶ IDLE 창에서 파이썬 코드 실행

- ▶ 코드를 작성하고 엔터키를 누를 때마다 해당 코드를 수행 시키는 방식으로 대화식 프로그래밍 가능



The screenshot shows the IDLE Shell 3.12.4 window. The title bar reads 'IDLE Shell 3.12.4'. The menu bar includes 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area displays the following content:

```
Python 3.12.4 | packaged by Anaconda, Inc. | (main, Jun 18 2024, 15:03:56) [MSC
v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> v1=10
>>> v2=20
>>> print(v1+v2)
30
>>> print(v1+v2)
30
>>> for i in range(v1, v2):
...     print(i)
...
10
11
12
13
14
15
16
17
18
19
>>>
```

The status bar at the bottom right indicates 'Ln: 22 Col: 0'.

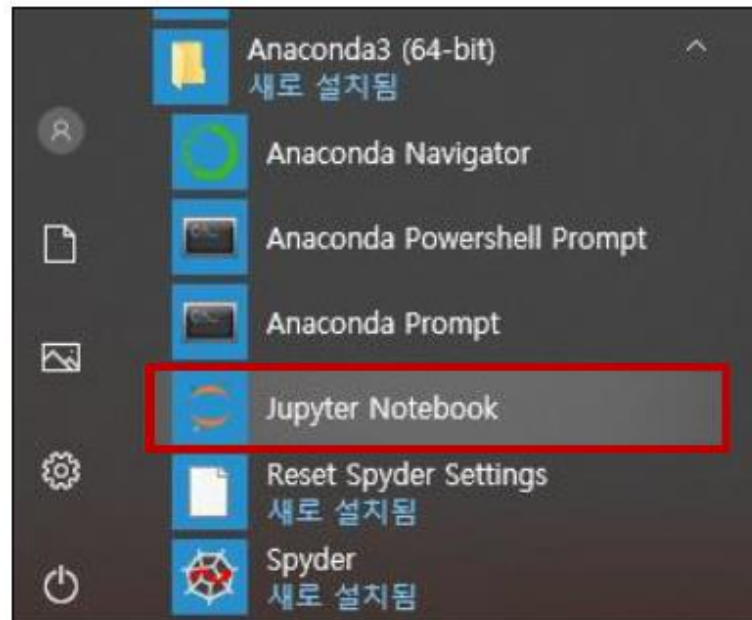
3. 파이썬 개발 툴 소개

▶ Jupyter Notebook

- ▶ 크롬과 같은 웹브라우저에서 코드를 작성하고 실행이 가능한 툴

▶ Jupyter Notebook 실행

- ▶ Anaconda에는 Jupyter Notebook이 포함되어 있으므로 '시작' 메뉴를 클릭하여 수행 시작
- ▶ Jupyter Notebook 실행을 시작하는 윈도우 출력



```
Jupyter Notebook
[2024-07-03 17:53:41.432 LabApp] JupyterLab extension loaded from C:\Users\User\Anaconda3\Lib\site-packages\jupyterlab
[2024-07-03 17:53:41.432 LabApp] JupyterLab application directory is C:\Users\User\Anaconda3\share\jupyterlab
[2024-07-03 17:53:41.433 LabApp] Extension Manager is 'pypi'.
[2024-07-03 17:53:41.435 ServerApp] jupyterlab | extension was successfully loaded.
[2024-07-03 17:53:41.444 ServerApp] notebook | extension was successfully loaded.
[2024-07-03 17:53:41.445 ServerApp] panel.io.jupyter_server_extension | extension was successfully loaded.
[2024-07-03 17:53:41.446 ServerApp] Serving notebooks from local directory: C:\Users\User
[2024-07-03 17:53:41.446 ServerApp] Jupyter Server 2.14.1 is running at:
[2024-07-03 17:53:41.447 ServerApp] http://localhost:8888/tree?token=678b4a3d044a9e7773575fe6cb750fca676de1eb2db05039
[2024-07-03 17:53:41.447 ServerApp] http://127.0.0.1:8888/tree?token=678b4a3d044a9e7773575fe6cb750fca676de1eb2db05039
[2024-07-03 17:53:41.447 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 2024-07-03 17:53:41.725 ServerApp]

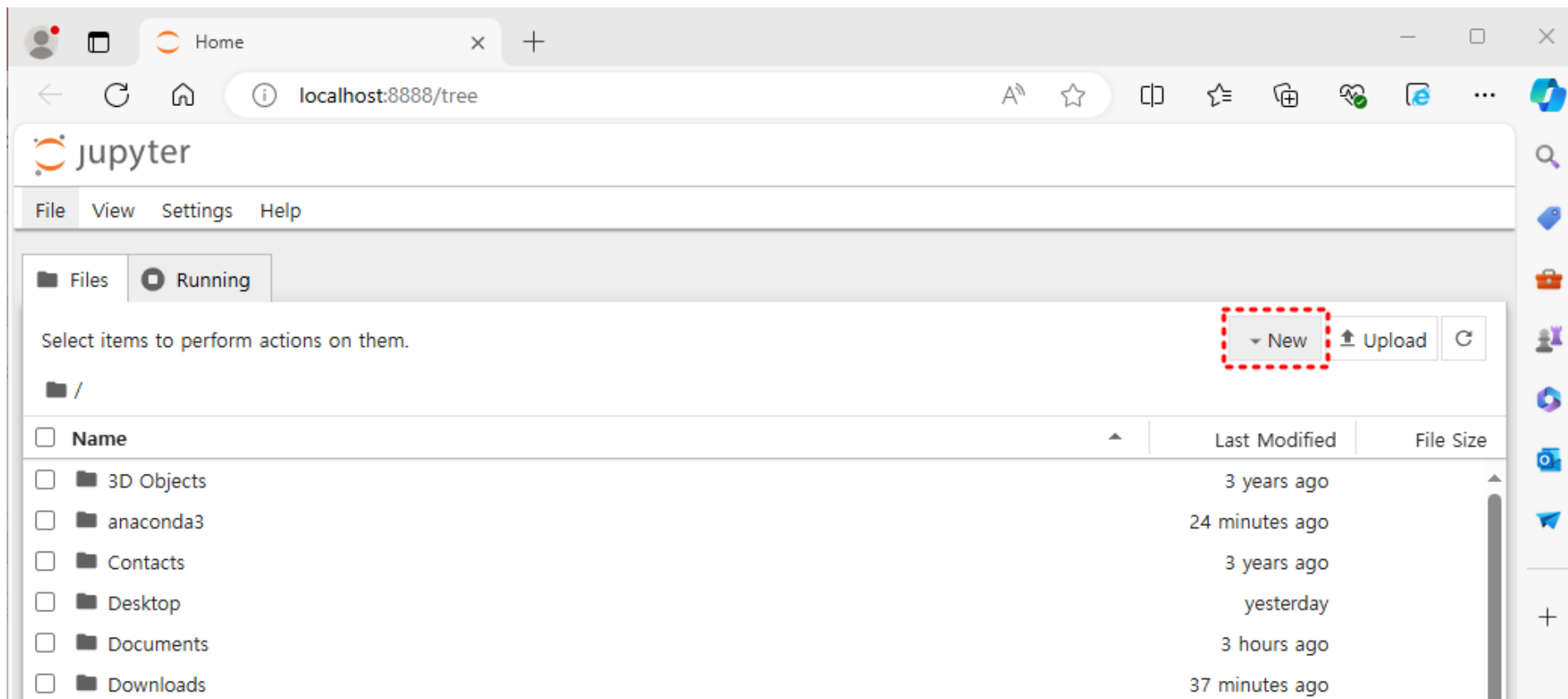
To access the server, open this file in a browser:
file:///C:/Users/user/AppData/Roaming/jupyter/runtime/jpserver-28888-open.html
Or copy and paste one of these URLs:
http://localhost:8888/tree?token=678b4a3d044a9e7773575fe6cb750fca676de1eb2db05039
http://127.0.0.1:8888/tree?token=678b4a3d044a9e7773575fe6cb750fca676de1eb2db05039
[W 2024-07-03 17:53:41.883 ServerApp] Could not determine npm prefix: [WinError 2] 지정된 파일을 찾을 수 없습니다
[2024-07-03 17:53:42.581 ServerApp] Skipped non-installed server(s): bash-language-server, dockerfile-language-server-
nodejs, javascript-typescript-langserver, jedi-language-server, julia-language-server, pyright, python-language-server,
r-languageserver, sql-language-server, texlab, typescript-language-server, unified-language-server, vscode-css-languages
server-bin, vscode-html-languageserver-bin, vscode-json-languageserver-bin, yaml-language-server
0.00s - Debugger warning: It seems that frozen modules are being used, which may
0.00s - make the debugger miss breakpoints. Please pass -Xfrozen_modules=off
0.00s - to python to disable frozen modules.
0.00s - Note: Debugging will proceed. Set PYDEVD_DISABLE_FILE_VALIDATION=1 to disable this validation.
```


3. 파이썬 개발 툴 소개

▶ Jupyter Notebook

▶ Jupyter Notebook의 작업 페이지

- ▶ 브라우저가 기동되고 Jupyter Notebook의 작업페이지 출력 -> New 클릭하여 새 문서 작성

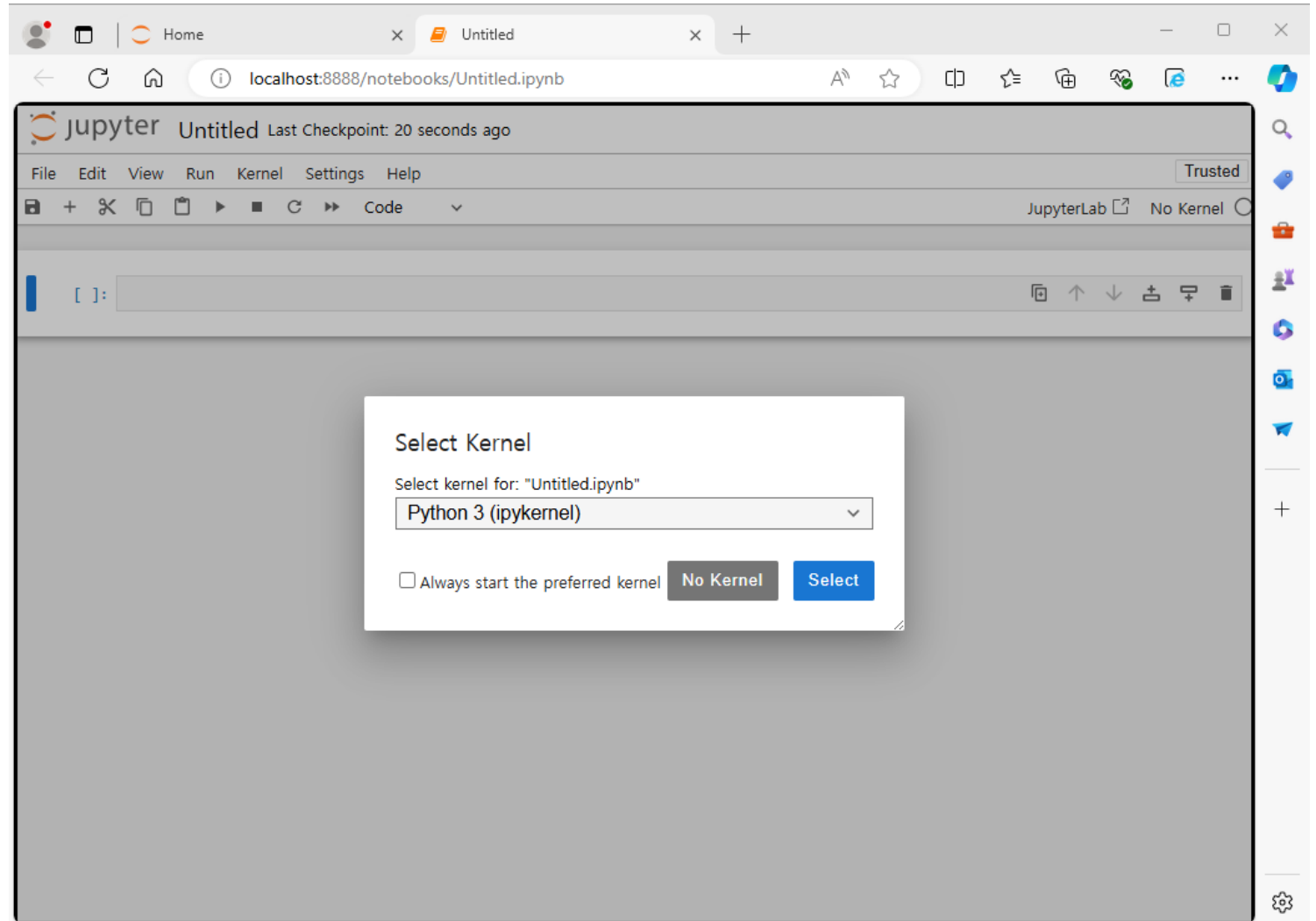


3. 파이썬 개발 툴 소개

▶ Jupyter Notebook

▶ 파이썬 코드 테스트

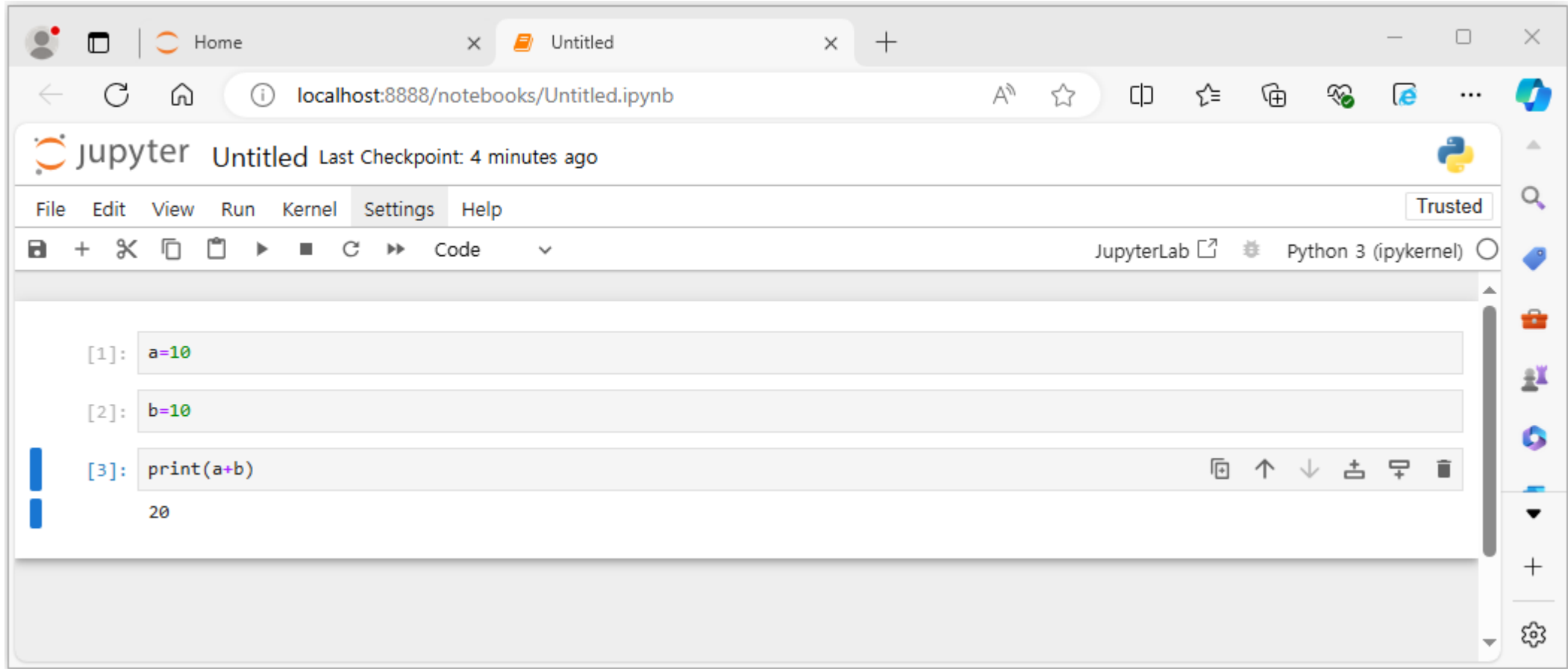
- ▶ 우측에 있는 'New'를 누르고
'Notebook -> Python 3(ipynb)
클릭하여 새문서 생성



3. 파이썬 개발 툴 소개

▶ Jupyter Notebook

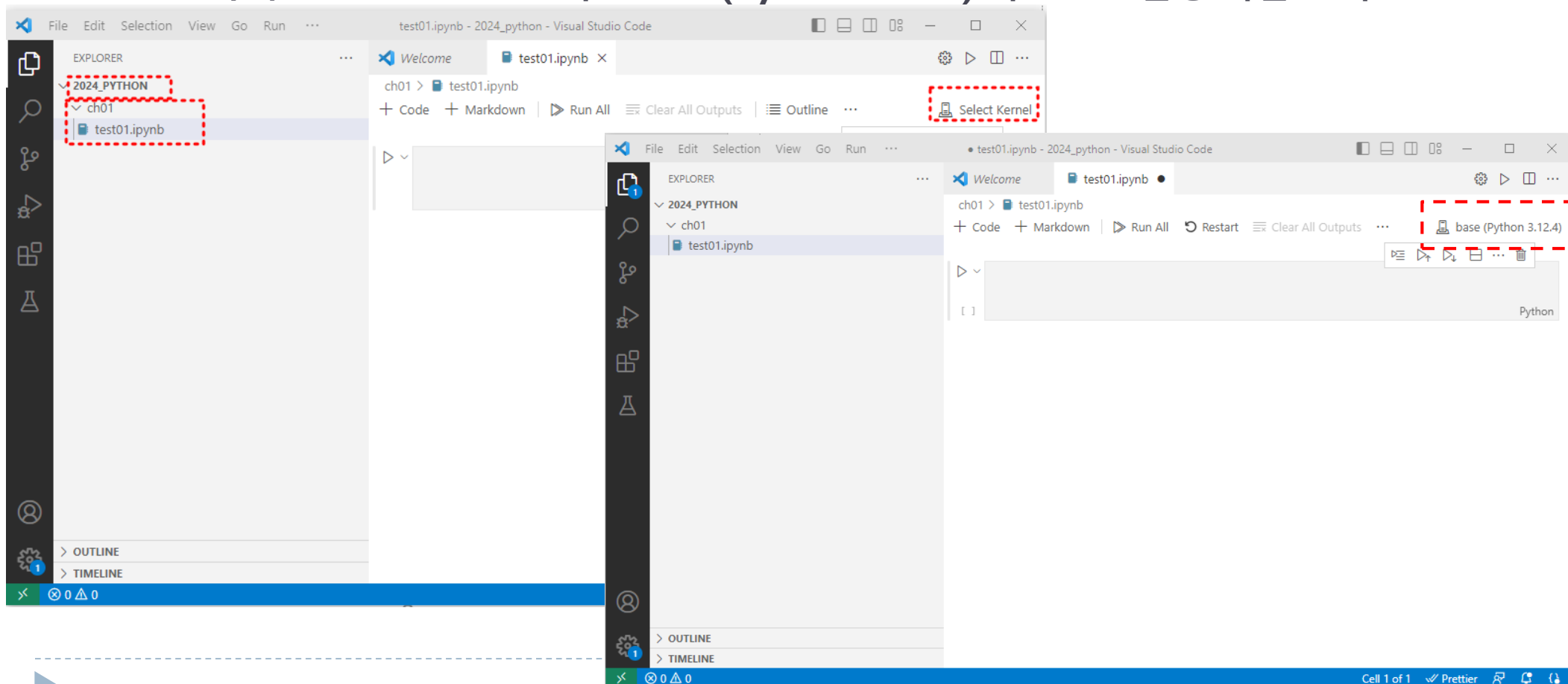
- ▶ 셀을 추가하여 다음과 같이 작성하고 셀에서 Ctrl+enter 키를 눌러 실행



3. 파이썬 개발 툴 소개

▶ VS Code

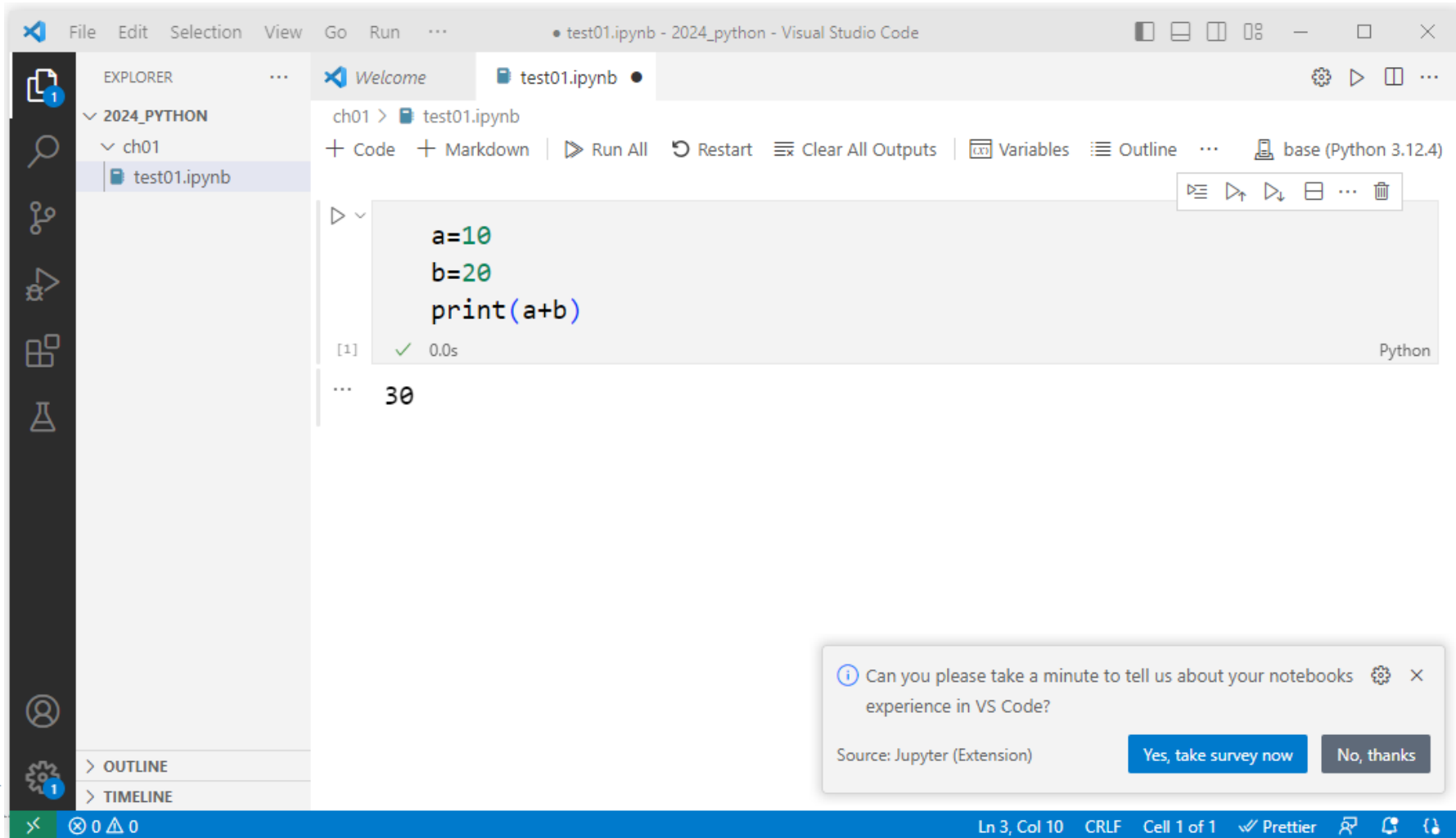
- ▶ VS Code 실행 후 작업폴더(2024_python) open -> ch01 폴더 작성 -> test01.ipynb 파일 작성
- ▶ 오른쪽에 Select Kernel 선택 -> base(Python 3.12.4) 버전으로 실행 커널 선택



3. 파이썬 개발 툴 소개

▶ VS Code

▶ 코드 입력 후 control+ Enter



3. 파이썬 개발 툴 소개

▶ VS Code : 테마변경

- ▶ VS Code 툴의 기본 컬러테마(Color Theme)는 Dark 이므로 검은 바탕으로 출력
- ▶ 흰바탕으로 변경 하고자 하면 File 메뉴의 'Preferences' 메뉴와 'Settings' 메뉴를 차례 대로 선택
- ▶ 세팅하려는 항목으로 **Color Theme**를 입력한 후 **[Workbench : Color Theme]** 부분에 **Default Light** 변경

