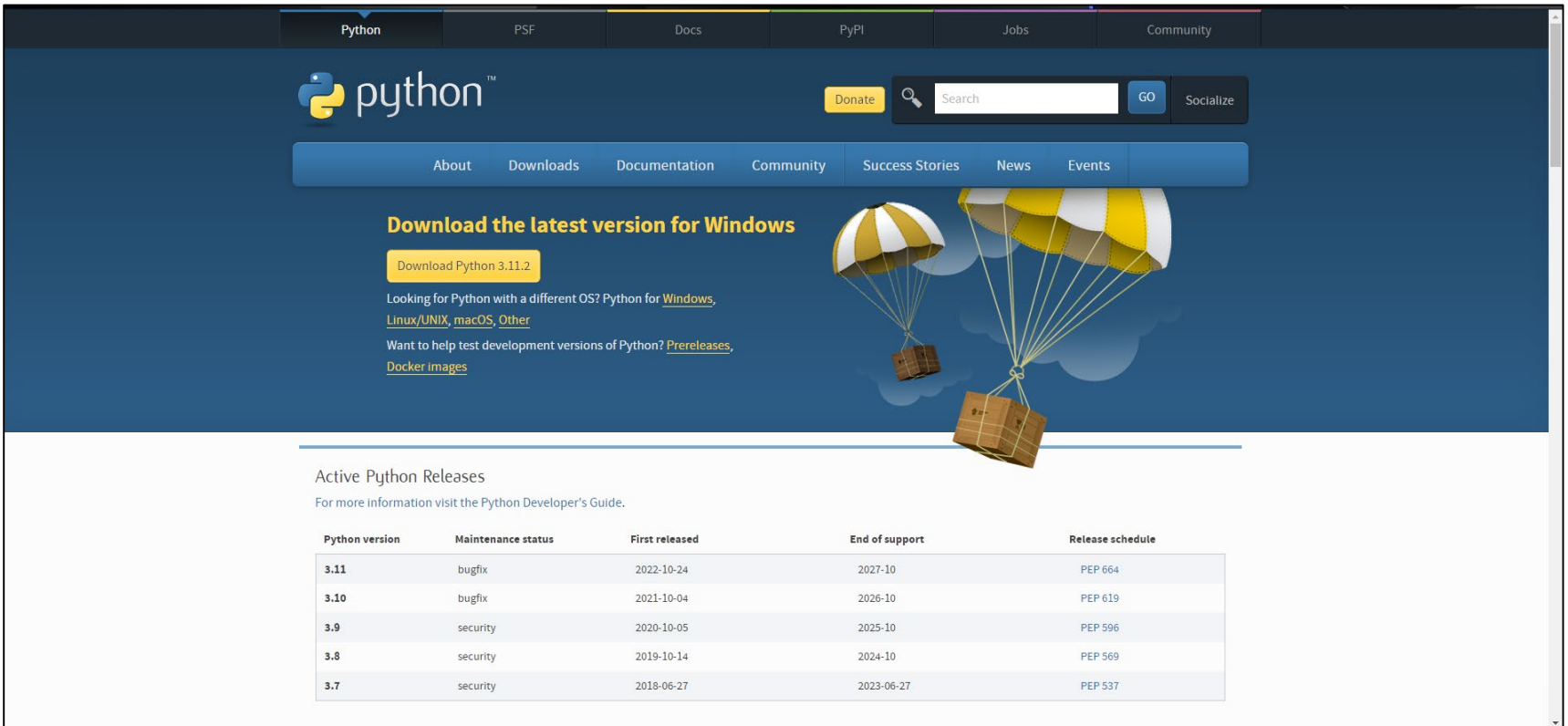


Python_intro

Python 설치

- Google에서 python 검색
- 공식 사이트 접속: <https://www.python.org/>

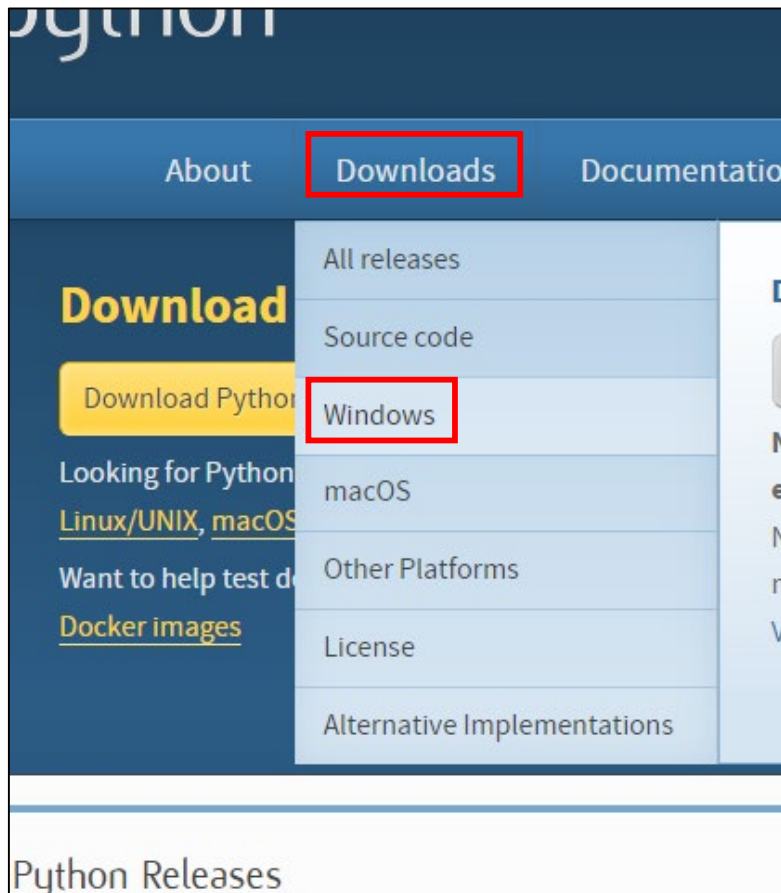


The screenshot shows the Python.org website. The top navigation bar includes links for Python, PSF, Docs, PyPI, Jobs, and Community. Below this is a search bar and a 'Donate' button. The main content area features a large banner with the text 'Download the latest version for Windows' and a button to 'Download Python 3.11.2'. To the right of the banner is an illustration of two parachutes carrying boxes. Below the banner, there is a section titled 'Active Python Releases' with a link to the 'Python Developer's Guide'. A table lists the active releases, including version, maintenance status, first release date, end of support date, and release schedule.

Python version	Maintenance status	First released	End of support	Release schedule
3.11	bugfix	2022-10-24	2027-10	PEP 664
3.10	bugfix	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
3.7	security	2018-06-27	2023-06-27	PEP 537

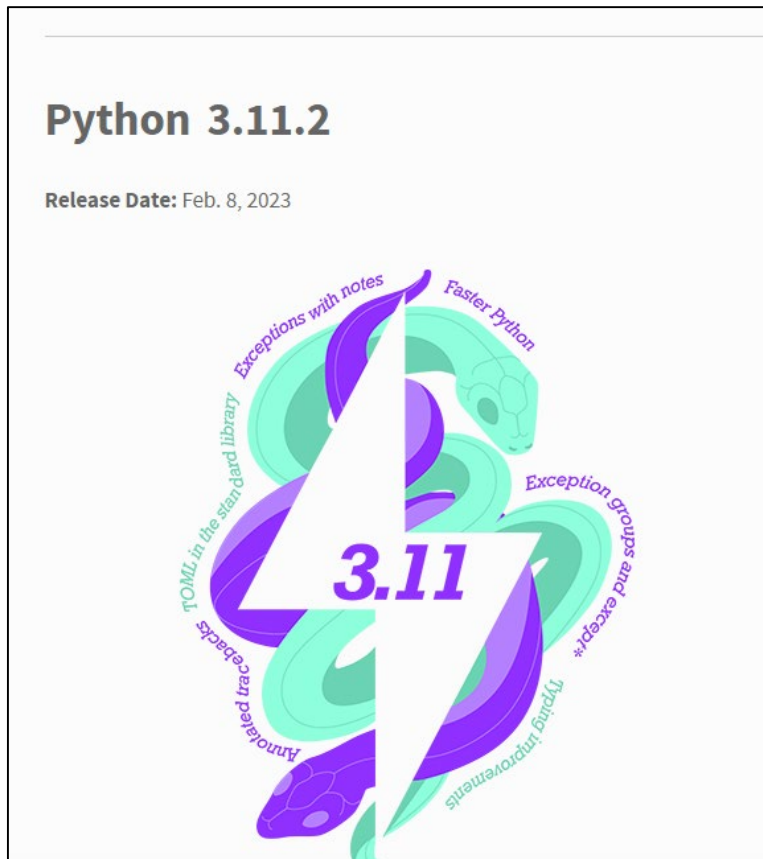
Python 설치

- 다운로드



Python 설치

- 본인의 운영체제에 맞게 설치

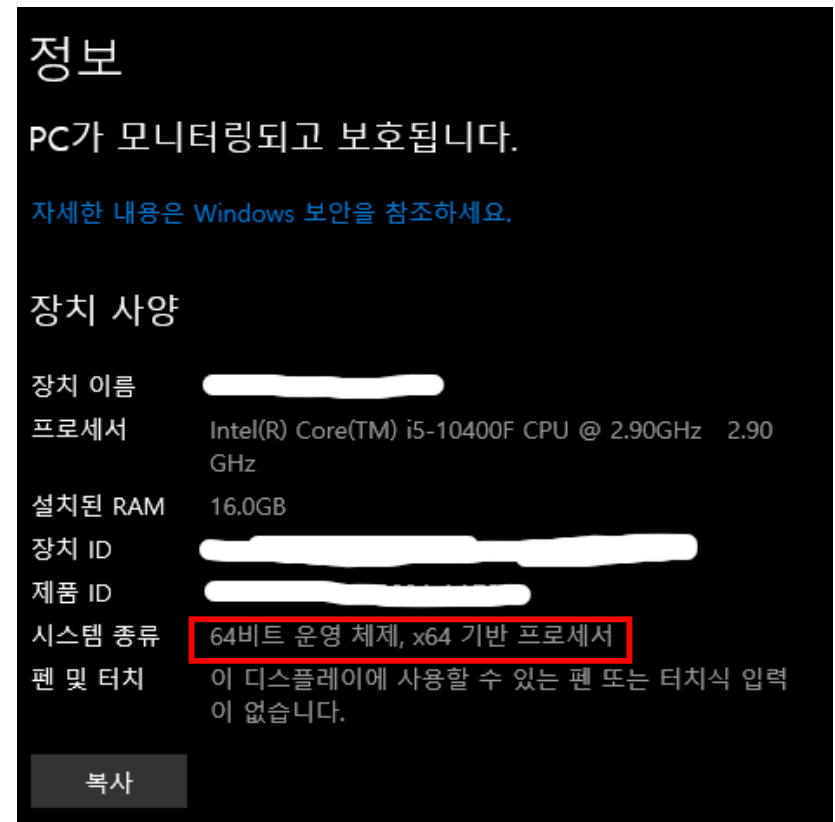
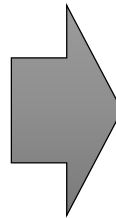
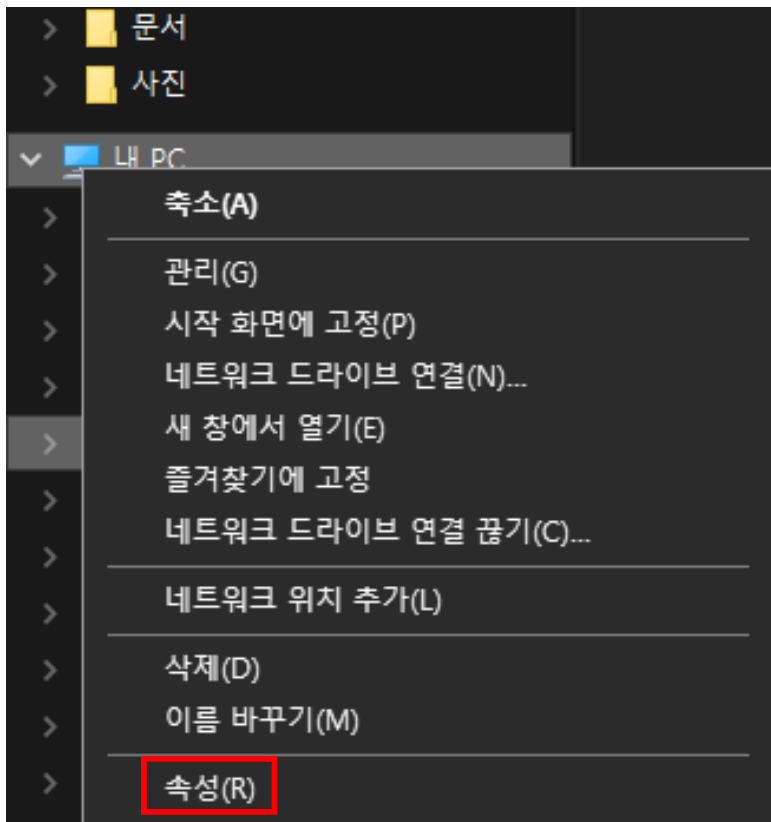


Files

Version	Operating System
Gzipped source tarball	Source release
XZ compressed source tarball	Source release
macOS 64-bit universal2 installer	macOS
Windows embeddable package (32-bit)	Windows
Windows embeddable package (64-bit)	Windows
Windows embeddable package (ARM64)	Windows
Windows installer (32-bit)	Windows
Windows installer (64-bit)	Windows
Windows installer (ARM64)	Windows

Python 설치

- 자신의 PC의 운영체제 bit 확인
 - [내 PC] 오른쪽 클릭 -> 속성



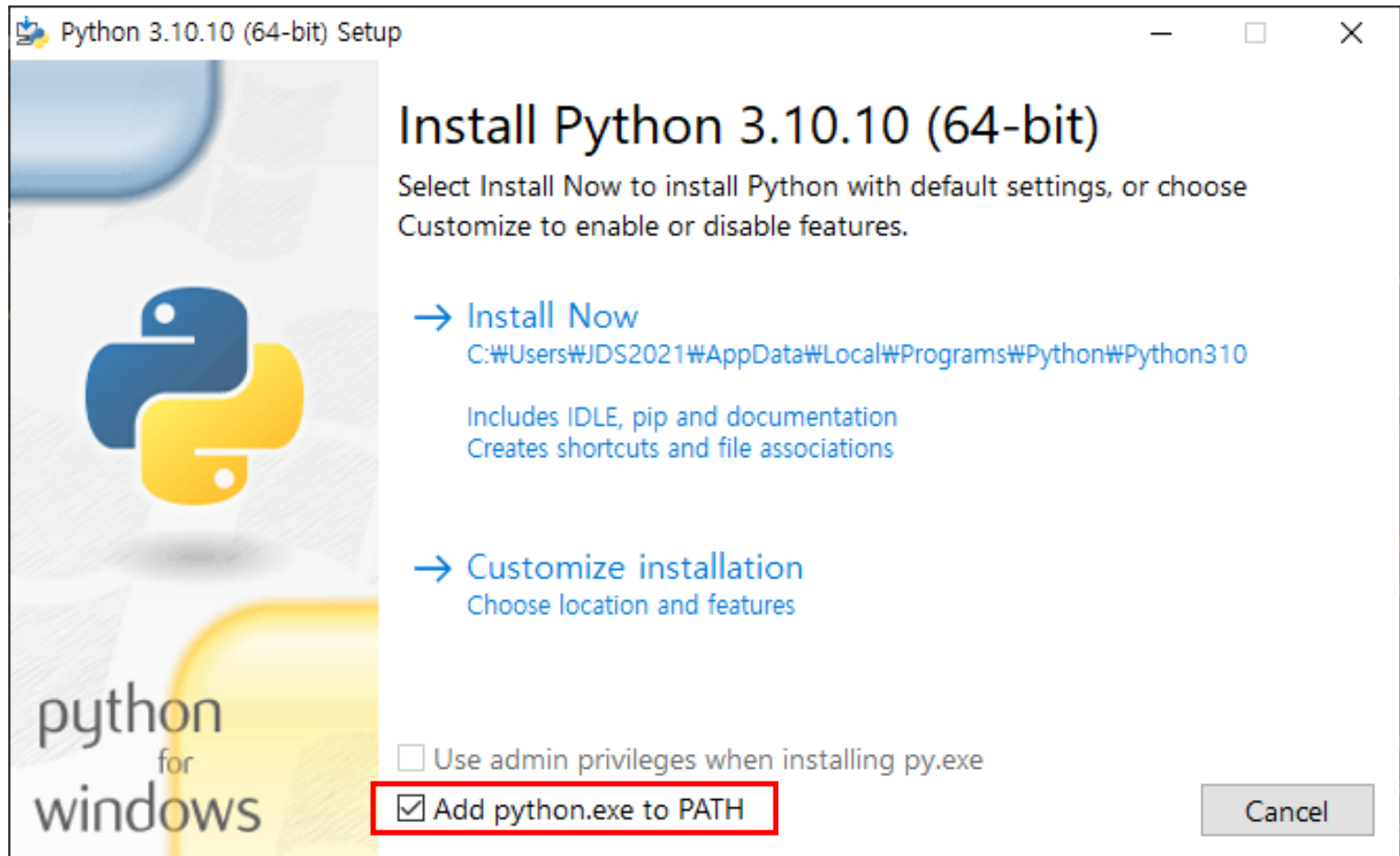
Python 설치

- 현재 Python은 크게 2.x버전과 3.x버전으로 나뉘어짐
- 크게 다르기 때문에 구분해서 사용
- 3.x 버전으로 실습



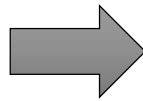
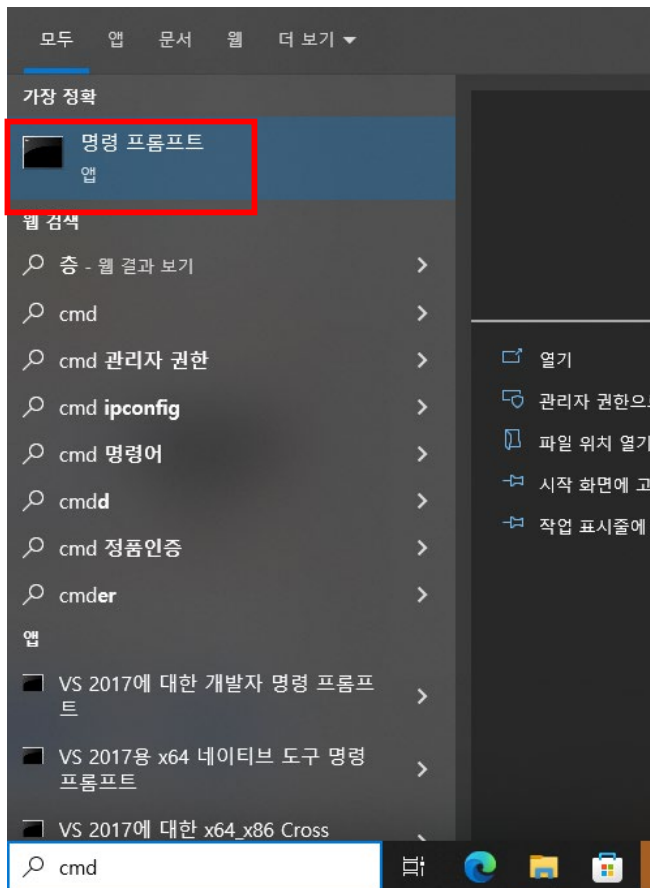
Python 설치

- 환경변수 설정: Add python.exe to PATH

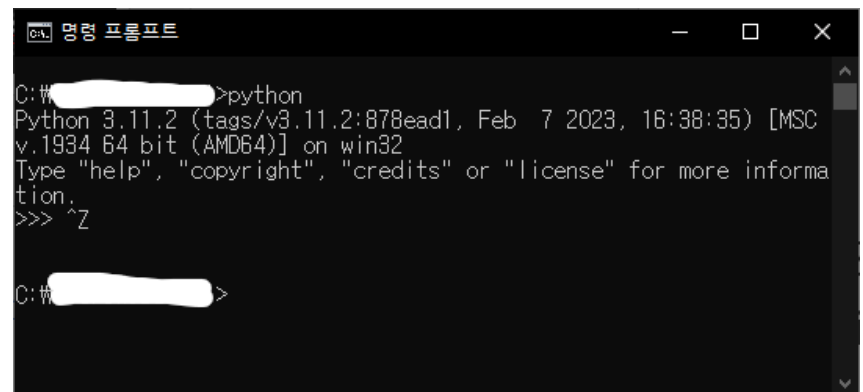


Python 설치

- TEST(python 인터프리터)
 - 시작 메뉴의 검색창에 cmd를 입력 -> cmd.exe 실행

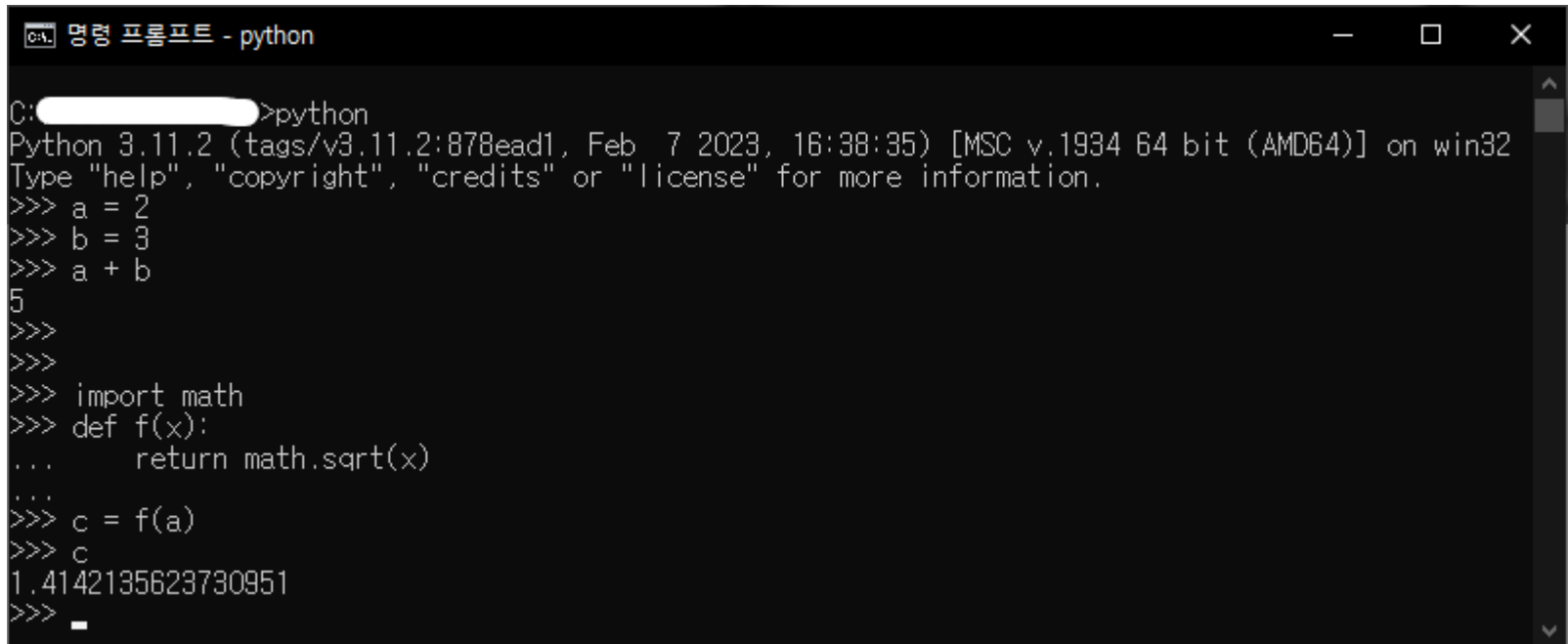


Python을 입력시 위의 메뉴가 뜨면 설치 성공
(Python 인터프리터 환경)
종료는 [Ctrl]+[z]키를 누른후 [Enter]



Python

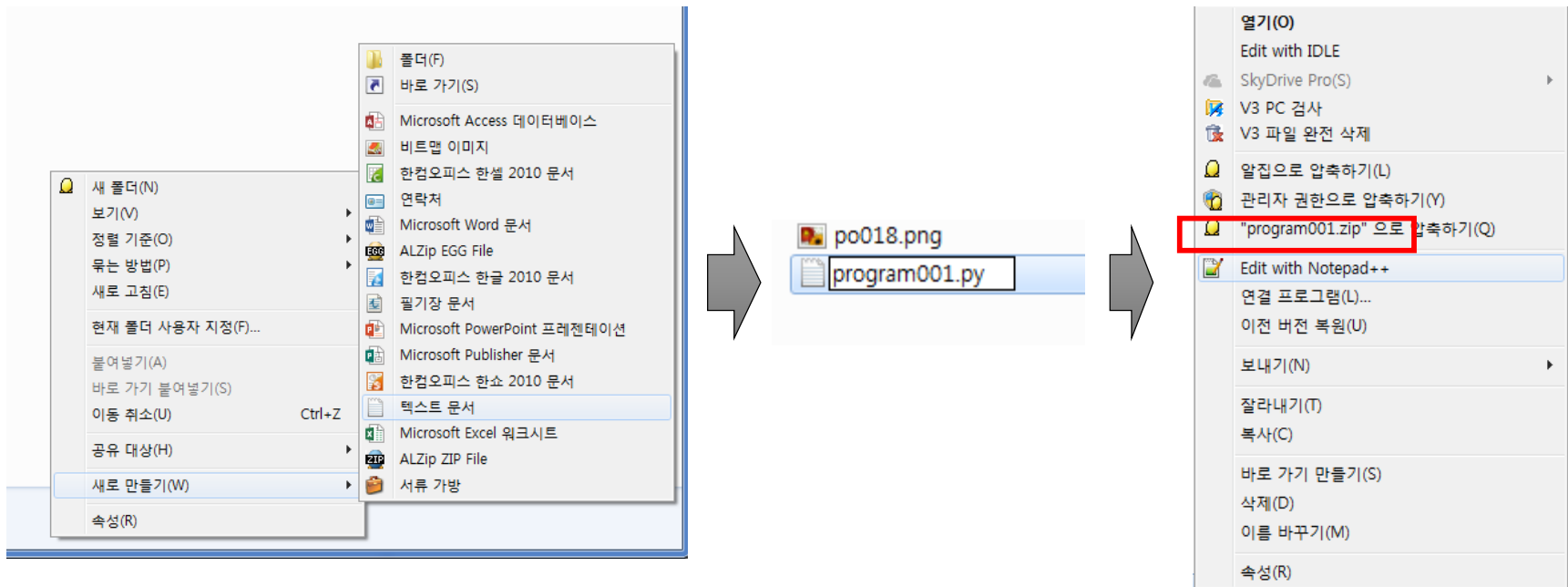
- Python 인터프리터



```
C:\>python
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 2
>>> b = 3
>>> a + b
5
>>>
>>>
>>> import math
>>> def f(x):
...     return math.sqrt(x)
...
>>> c = f(a)
>>> c
1.4142135623730951
>>> _
```

Python_Natepad

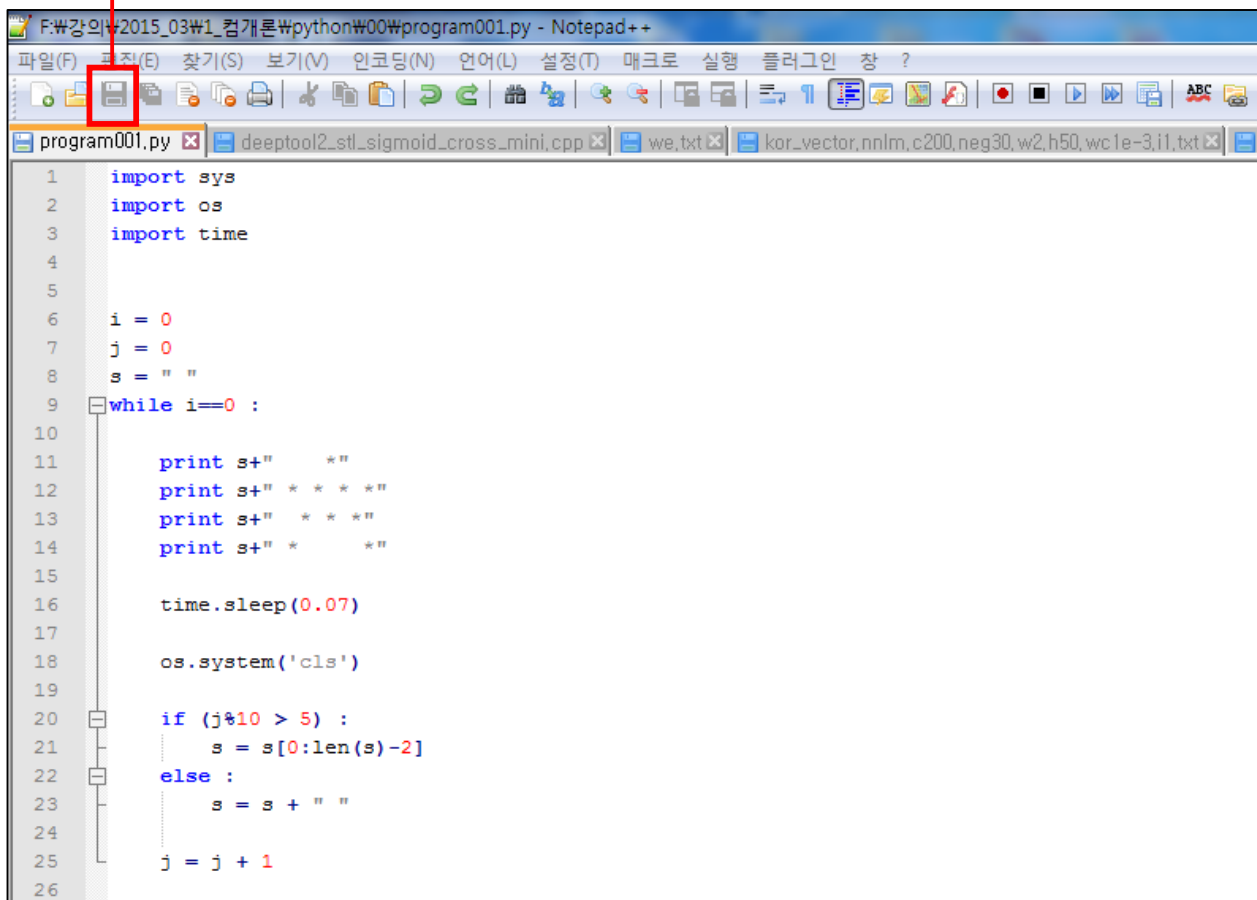
- 작업폴더 생성
- 소스코드(text파일) 생성
- 파일명 변경(이때 확장자는 .py)
- 해당 파일 오른쪽 클릭후 Notepad++(메모장)로 실행



Python_Natepad

- 기본 작업 환경

저장 필수!

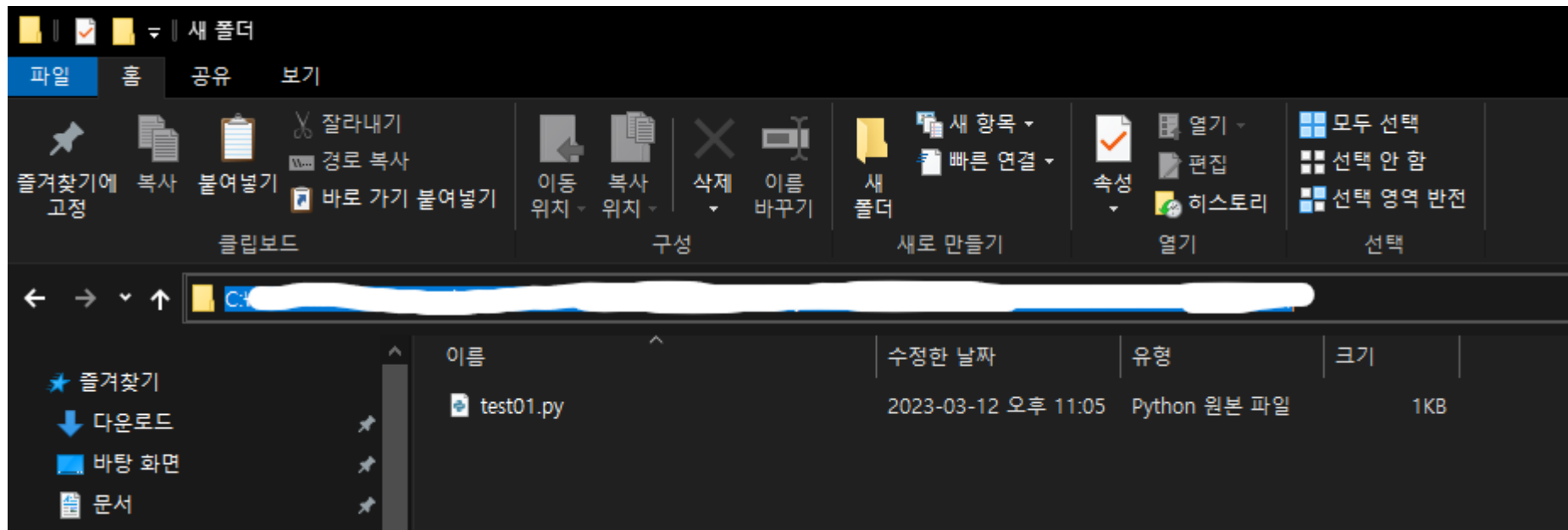


The screenshot shows the Notepad++ application window. The title bar reads "F:\강의\2015_03\1_컴개론\python\00\program001.py - Notepad++". The menu bar includes "파일(F)", "편집(E)", "찾기(S)", "보기(V)", "인코딩(N)", "언어(L)", "설정(T)", "매크로", "실행", "플러그인", and "창?". The toolbar contains various icons, with the "Save" icon (a floppy disk) highlighted by a red rectangle. A red arrow originates from this icon and points towards the text "저장 필수!". The main text area displays a Python script with the following code:

```
1  import sys
2  import os
3  import time
4
5
6  i = 0
7  j = 0
8  s = " "
9  while i==0 :
10
11      print s+"    *"
12      print s+" * * * *"
13      print s+" * * * *"
14      print s+" *      *"
15
16      time.sleep(0.07)
17
18      os.system('cls')
19
20      if (j%10 > 5) :
21          s = s[0:len(s)-2]
22      else :
23          s = s + " "
24
25      j = j + 1
26
```

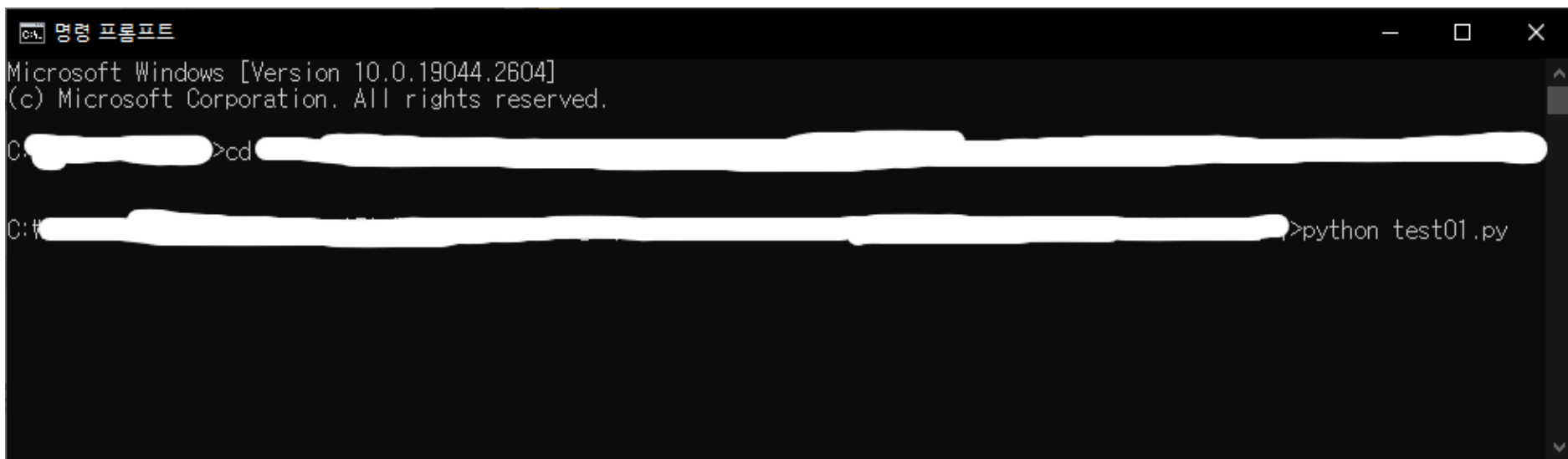
Python_Natepad

- 현재 작업폴더 경로 복사
- cmd창에 경로 적용
 - >> cd [경로]
 - 붙이기: 오른쪽 마우스 클릭



Python_Natepad

- Python 파일 실행
 - >>python [파일명]
- 종료는 [Ctrl]+[c]



A screenshot of a Windows Command Prompt window titled "명령 프롬프트". The window shows the following text:

```
Microsoft Windows [Version 10.0.19044.2604]
(c) Microsoft Corporation. All rights reserved.

C:\>cd [redacted]

C:\>python test01.py
```

The paths are redacted with white bars. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

Python_ANACONDA



- a **distribution** of the **Python** (or other language) programming languages for scientific computing

 - Pros
 - **one-stop destination** for important data science and programming tools such as Jupyter.
 - supports multiple **environments**
 - install multiple libraries in Python easily
- ```
>> conda install numpy
```

# Python\_ANACONDA

## Anaconda Navigator

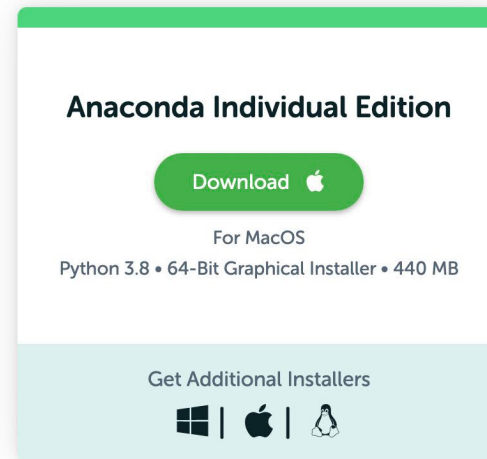
- Download [[link](#)]




Individual Edition

## Your data science toolkit

With over 25 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries.







# Python\_ANACONDA


 **ANACONDA**.NAVIGATOR


Upgrade Now


[Home](#)  
[Environments](#)  
[Learning](#)  
[Community](#)


  
Join Now  
Discover premium data science content  
Documentation  
Anaconda Blog  
  


Applications on base (root) Channels


  
Datalore  
Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.  
[Launch](#)


  
IBM Watson Studio Cloud  
IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.  
[Launch](#)


  
JupyterLab  
3.0.14  
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.  
[Launch](#)


  
Jupyter Notebook  
6.3.0  
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.  
[Launch](#)


  
Qt Console  
5.0.3  
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.  
[Launch](#)


  
Spyder  
4.2.5  
Scientific Python Development Environment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features  
[Launch](#)

  
VS Code  
1.57.0  
Streamlined code editor with support for development operations like debugging, task running and version control.  
[Launch](#)

  
Glueviz  
1.0.0  
Multidimensional data visualization across files. Explore relationships within and among related datasets.  
[Install](#)


  
Orange 3

  
PvCharm Professional

  
RStudio



# Python\_ANACONDA

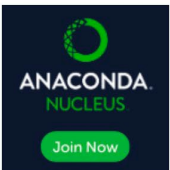
 Upgrade Now Sign in

Home

Environments

Learning




Community

  
Join Now

Discover premium data science content

Documentation

Anaconda Blog

Search Environments

base (root)

pytorch

Not installed

Channels

Update index...

pytorch

| Name                                     | Description                                                                   | Version |
|------------------------------------------|-------------------------------------------------------------------------------|---------|
| <input type="checkbox"/> _pytorch_select |                                                                               | 0.1     |
| <input type="checkbox"/> pytorch         | Pytorch is an optimized tensor library for deep learning using gpus and cpus. | 1.7.1   |

2 packages available matching "pytorch"

Create

Clone

Import

Remove

# Python\_Jupyter



- a web-based **interactive** computing platform.
  - Provides in-line code execution using blocks

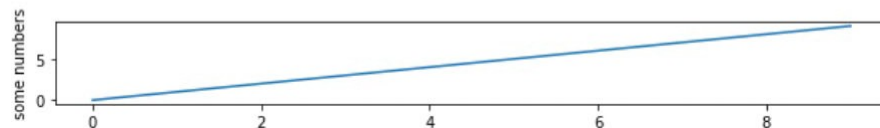
```
[2]: a = np.arange(10)
```

```
[3]: print(a)
```

```
[0 1 2 3 4 5 6 7 8 9]
```

- Provides in-line graphing support.

```
[5]: plt.figure(figsize=(10, 1))
plt.plot(a)
plt.ylabel('some numbers')
plt.show()
```



# Python\_Jupyter

File Edit View Run Kernel Tabs Settings Help

Filter files by name

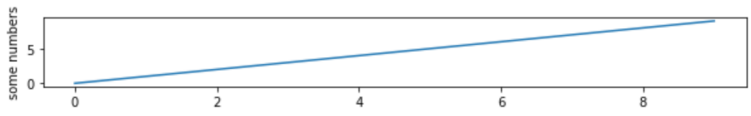
/ [Lecture] DAP /

| Name                          | Last Modified |
|-------------------------------|---------------|
| Lecture1.ip... 20 minutes ago |               |

Lecture1.ipynb

Python 3

```
[1]: import numpy as np
[2]: a = np.arange(10)
[3]: print(a)
[0 1 2 3 4 5 6 7 8 9]
[4]: import matplotlib.pyplot as plt
[5]: plt.figure(figsize=(10, 1))
plt.plot(a)
plt.ylabel('some numbers')
plt.show()
```



Simple 0 1 Python 3 | Idle Mode: Command Ln 1, Col 1 Lecture1.ipynb

# Python\_Colab

Colab



- allows you to write and execute Python in your **browser**
  - Zero configuration required
  - Free access to **GPUs**  
(include **Nvidia** K80s, T4s, P4s and P100s, but time limited)
  - Easy sharing
- pro version is also provided



## Faster GPUs

Priority access to faster GPUs and TPUs means you spend less time waiting while code is running.



## Longer runtimes

Longer running notebooks and fewer idle timeouts mean you disconnect less often.



## More memory

More RAM and more disk means more room for your data.

# Python 환경

Either one is fine!



**Local**

**VS.**



Colab

**Cloud**

*prefer!*

# 실습 0101

파이썬으로 계산기로 사용하여 다음 연산을 한다.

1.  $3 \times 2 - 8 \div 4$

2.  $25 \times 6 \div 3 + 17$

3.  $39021 - 276920 \div 12040$

4.  $2^6 - 10 \% 6$

위 식에서 %는 나머지를 구하는 연산이다.

# 실습평가 제출

- 1. 구글 클래스룸 과제에 제출: 코드(ipynb)
  - 파일명: [학번]\_[이름]\_[실습 번호].ipynb
  - e.g. 10701\_홍길동\_0101.ipynb
- 2. 구글 클래스룸 과제에 제출: 코드(py) + 결과 이미지(jpg, etc)
  - 파일명: [학번]\_[이름]\_[실습 번호].py + [학번]\_[이름]\_[실습 번호].jpg
  - e.g. 10701\_홍길동\_0101.py + 10701\_홍길동\_0101.jpg
  - 결과 이미지는 확장자 무관
- 지정된 형식(1 또는 2), 기한 준수 (특히 기간엄수)
- 구글 클래스룸 참여 차후 공지