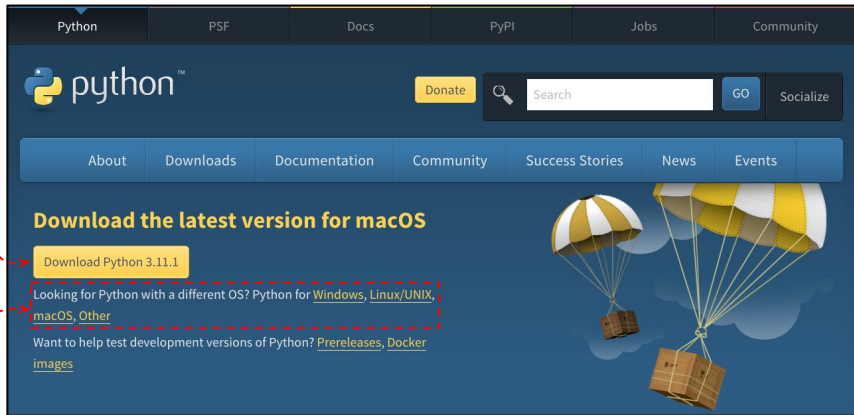


실습 환경 구축

Python 및 Jupyter Notebook 설치

Python 설치 파일 다운로드

- 크롬에서 아래 링크로 접속하여 Python 설치 파일을 내려받습니다.
 - URL: <https://www.python.org/downloads/>
 - 최신 버전의 설치 파일을 다운로드하려면 노란색 버튼을 클릭합니다.
 - 사용 중인 컴퓨터 운영체제에 맞는 설치 파일을 다운로드합니다.
 - 이전 버전의 설치 파일을 다운로드하려면 사용 중인 컴퓨터 운영체제 버튼을 클릭하면 안정 배포 버전을 포함하는 페이지로 이동합니다.



[참고] 이전 버전 설치 파일 선택

Python >>> Downloads >>> macOS

Python Releases for macOS

- [Latest Python 3 Release - Python 3.11.1](#)

Stable Releases # 안정 배포 버전에서 선택합니다.

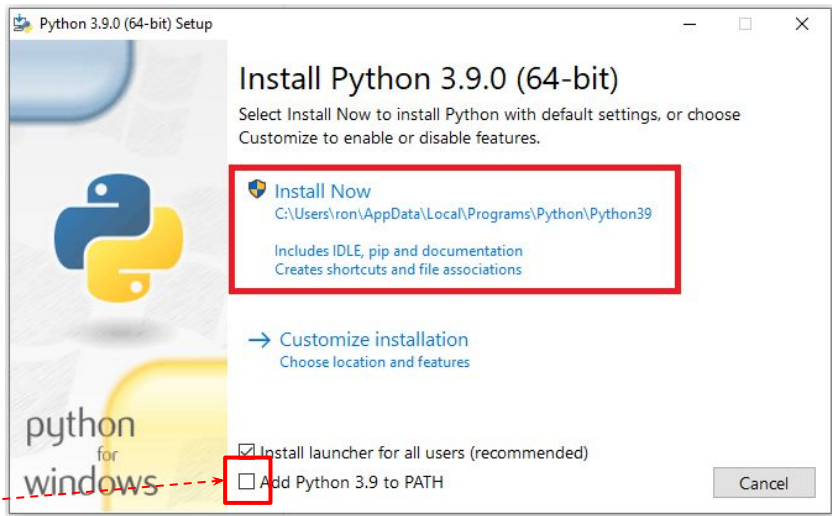
- [Python 3.11.1 - Dec. 6, 2022](#)
 - Download [macOS 64-bit universal2 installer](#)
- [Python 3.10.9 - Dec. 6, 2022](#)
 - Download [macOS 64-bit universal2 installer](#)
- [Python 3.9.16 - Dec. 6, 2022](#)
 - No files for this release.

Pre-releases

- [Python 3.12.0a4 - Jan. 10, 2023](#)
 - Download [macOS 64-bit universal2 installer](#)
- [Python 3.12.0a3 - Dec. 6, 2022](#)
 - Download [macOS 64-bit universal2 installer](#)
- [Python 3.12.0a2 - Nov. 15, 2022](#)
 - Download [macOS 64-bit universal2 installer](#)

Python 설치

- 내려받은 설치 파일을 더블 클릭하면 Python 설치 팝업 메뉴가 뜹니다.
- MacOS 사용자는 '계속' 버튼만 클릭하면 됩니다.
- Windows 사용자는 설치 화면 하단에 있는 'Add Python 3.x to PATH' 앞을 체크해야 합니다.(매우 중요!)
- Python 설치를 완료했으면 Window는 CMD, MacOS는 Terminal을 열고 Python을 실행합니다.

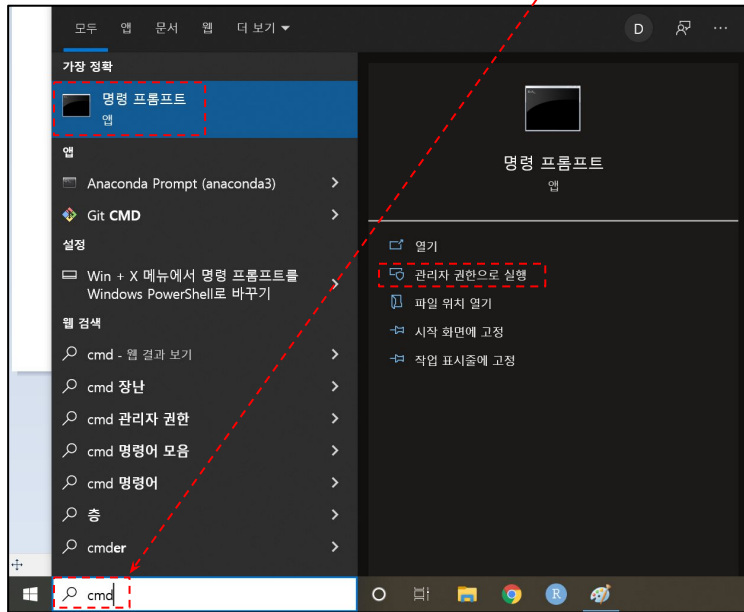


출처: <https://datatofish.com/install-python/>
[참고] 현재 이미지는 체크 안된 상태입니다.

Windows에서 CMD 실행

- Windows 작업 표시줄에 있는 돋보기 메뉴에서 **cmd**를 입력하고 실행합니다.

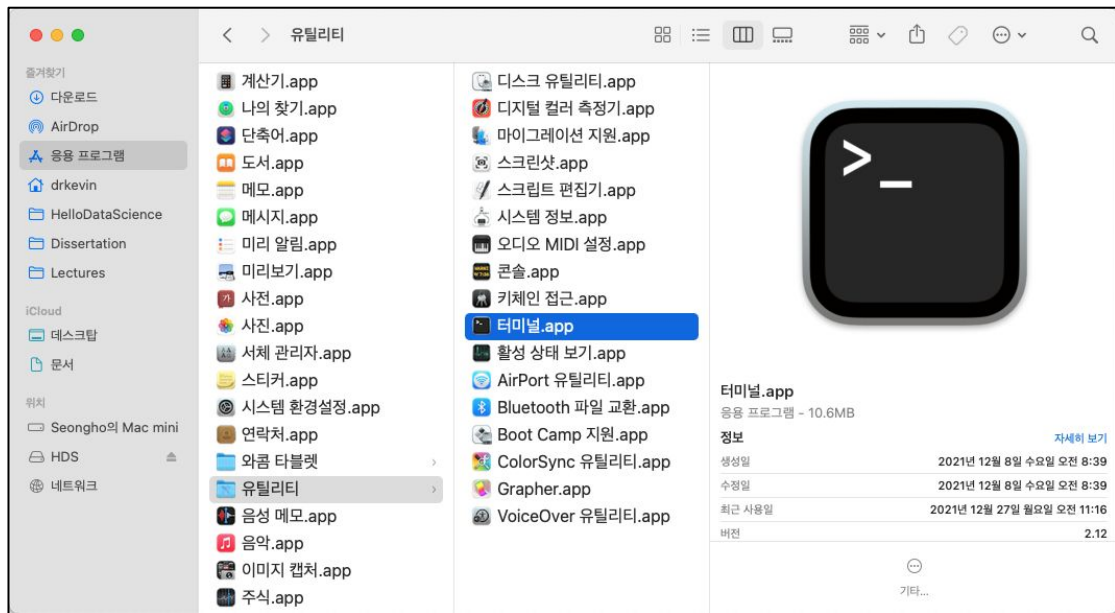
명령 프롬프트에서 마우스
오른쪽 버튼을 클릭합니다.



관리자 권한으로 실행을
선택합니다.

MacOS에서 Terminal 실행

- 파인더 > 응용 프로그램 > 유틸리티 폴더에서 터미널.app을 실행합니다.



Python pip 업데이트

- Python 라이브러리를 설치할 때 pip를 이용하면 좋습니다.
 - pip는 Python 라이브러리 설치 및 관리 시스템이며, 3.4 버전에서 추가되었습니다.
 - Windows의 CMD, MacOS의 Terminal을 열고 아래 명령어를 입력하여 pip를 실행합니다.
- 현재 설치된 Python 라이브러리 목록을 확인합니다. *# % 기호는 프롬프트이므로 아래 명령어에서 % 기호를 제외한 코드를 입력하세요.*
% pip list # 왼쪽 명령어를 실행했을 때 맨 아래에 노란색 경고를 확인했다면 pip를 업그레이드해야 합니다.
[참고] Python을 설치할 때 환경변수에 추가하지 않았다면 맨 앞에 python -m을 추가해야 합니다.
- pip 버전을 업그레이드하고, 설치된 버전을 확인합니다.
% pip install --upgrade pip # [참고] 왼쪽 명령어를 실행했을 때 CMD를 관리자 권한으로 실행하는 것이 아니라면 에러가 발생할 수 있으므로 마지막에 --user를 추가합니다.
% pip --version # pip 현재 버전을 확인합니다.

Python 라이브러리 설치(for Basic Course)

- Basic 과정에 필요한 Python 라이브러리를 설치합니다. # [참고] Python 라이브러리 탐색 위치 <https://pypi.org/>

```
% pip install jupyter
```

 # **jupyter** 라이브러리를 설치합니다. 의존성 라이브러리를 함께 설치합니다.
[참고] JupyterLab을 실행하려면 **jupyterlab** 라이브러리를 추가 설치해야 합니다.

```
% pip install pandas
```

 # **pytz, numpy, pandas** 라이브러리를 설치합니다.

```
% pip install openpyxl
```

 # **et-xmlfile, openpyxl** 라이브러리를 설치합니다.

```
% pip install xlrd
```

 # **xlrd** 라이브러리를 설치합니다.

```
% pip install seaborn
```

 # **pyparsing, pillow, kiwisolver, fonttools, cyciler, contourpy, matplotlib, seaborn** 라이브러리를 설치합니다.

```
% pip install chardet
```

 # **chardet** 라이브러리를 설치합니다.

```
% pip install joblib
```

 # **joblib** 라이브러리를 설치합니다.

[참고] Python 라이브러리 설치 오류 해결

- 외부 라이브러리를 설치할 때 신뢰할 수 없다는 이유로 라이브러리를 설치하지 못하는 오류가 발생할 수 있습니다.

```
Could not fetch URL https://pypi.org/simple/pip/: There was a problem confirming the ssl certificate  
: HTTPSConnectionPool(host='pypi.org', port=443): Max retries exceeded with url: /simple/pip/ (Cause  
d by SSLError<SSLError[1, 'SSL: CERTIFICATE_VERIFY_FAILED'] certificate verify failed (<_ssl.c:852>')  
,>> - skipping
```

- 이와 같은 오류를 해결하기 위해 pip 명령어에 아래 코드를 추가합니다.

```
% pip install charset-normalizer --trusted-host pypi.org
```

[참고] 위 명령어로 해결되지 않으면 *pypi.org* 대신 *pypi.python.org* 또는 *files.pythonhosted.org*로 바꿔서 실행해 보시기 바랍니다.

Jupyter Notebook 실행

- CMD(Windows) 또는 Terminal(MacOS)에서 아래 코드를 실행하면 기본 브라우저에서 새 탭이 열리고, Jupyter Notebook을 실행합니다.
[참고] Jupyter Notebook을 크롬에서 실행하는 것을 추천합니다.

```

명령 프롬프트
Microsoft Windows [Version 10.0.19043.1415]
(c) Microsoft Corporation. All rights reserved.
C:\Users\User>jupyter notebook
  
```

```

hdsceokevin — zsh — 80x24
(Venv) hdsceokevin@Seonghoui-Macmini ~ %jupyter lab
# [참고] 위 코드는 JupyterLab을 실행합니다.
  
```

[참고] Jupyter Notebook 실행 경로를 아래와 같이 추가할 수 있습니다.
% jupyter notebook --notebook-dir=경로명

Jupyter Notebook 메인화면

jupyter

Quit Logout

Files Running Clusters

Select items to perform actions on them.

Jupyter Notebook의 시작 위치는 User 폴더입니다.

Python 3

R

Other:

Text File

Folder

Terminal

새 Python Jupyter Notebook을 열려면 New → Python 3를 차례대로 선택합니다.

[참고] Jupyter Notebook을 열었을 때 500 : Internal Server Error가 발생하면 아래 코드를 실행하세요.
% pip install --upgrade jupyter

Python 라이브러리 설치(for Advanced Course)

- Advanced 과정에 필요한 Python 라이브러리를 추가 설치합니다.

```
% pip install scikit_learn # threadpoolctl, scipy, scikit-learn 라이브러리를 설치합니다.
```

```
% pip install statsmodels # pasty, statsmodels 라이브러리를 설치합니다.
```

```
% pip install graphviz # graphviz 라이브러리를 설치합니다.
```

```
% pip install scikit_posthocs # scikit_posthocs 라이브러리를 설치합니다.
```

```
% pip install pingouin # littleutils, charset-normalizer, urllib3, tabulate, lazy_loader, certifi, requests, xarray, outdated, pandas_flavor, pingouin 라이브러리를 설치합니다.
```

```
# [참고] 아나콘다 사용자는 터미널에서 아래 코드를 실행합니다.  
% conda install -c conda-forge pingouin
```

```
# [참고] Python 라이브러리 설치에 관한 튜토리얼은 아래 링크를 참고하세요.  
https://packaging.python.org/en/latest/tutorials/installing-packages/
```

[참고] Python 라이브러리 업그레이드

- 현재 설치된 Python 라이브러리 중 업그레이드가 필요한 목록을 출력합니다.

```
% pip list --outdated # 라이브러리로 현재 버전과 업그레이드 버전을 함께 출력합니다.
```

- 라이브러리를 하나씩 업그레이드합니다.

```
% pip install --upgrade 라이브러리 # 위 결과로 확인한 라이브러리를 하나씩 업그레이드합니다.
```

- [참고] 업그레이드가 필요한 라이브러리 목록을 txt 파일로 생성합니다.

```
% pip list --outdated --format=freeze > upgradeList.txt # 'upgradeList.txt' 파일을 열고  
# '='를 '>='으로 변경합니다.
```

- [참고] txt 파일을 읽고 여러 라이브러리를 한 번에 업그레이드합니다.

```
% pip install -r upgradeList.txt --upgrade
```

[참고] Jupyter Notebook 확장 프로그램 설치

- Jupyter Notebook에서 확장 프로그램을 사용하려면 아래 코드를 실행합니다.

```
% pip install jupyter_contrib_nbextensions
```

Jupyter Notebook 확장 라이브러리를 설치합니다.
[참고] JupyterLab은 설치할 필요 없습니다.

```
% jupyter contrib nbextension install --user
```

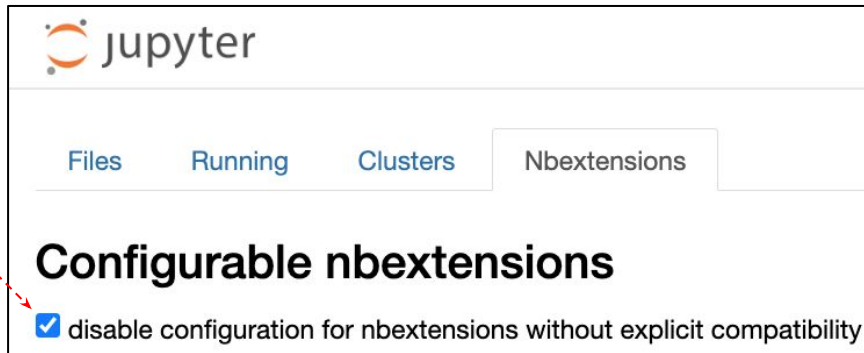
Jupyter Notebook 확장 프로그램을 사용하려면
왼쪽 코드를 실행합니다.

- Jupyter Notebook을 열고 확장 프로그램을 사용할 수 있도록 설정합니다.

Nbextensions 탭을 열고 Configurable nbextensions 앞 체크를 해제합니다.

일부 유용한 도구를 추천합니다.

- Variable Inspector
- Table of Contents (2)
- ExecuteTime



JupyterLab Manual

JupyterLab 메인화면

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
/	
Applications	19 days ago
Desktop	35 minutes ago
Documents	11 minutes ago
Downloads	6 hours ago
Movies	4 days ago
Music	19 days ago
Pictures	17 days ago
Public	19 days ago

JupyterLab 시작 위치는 User 폴더입니다.

Launcher

Notebook

Python 3

새 Jupyter Notebook 파일을 열려면 왼쪽 아이콘을 클릭합니다.

Console

Python 3

Other

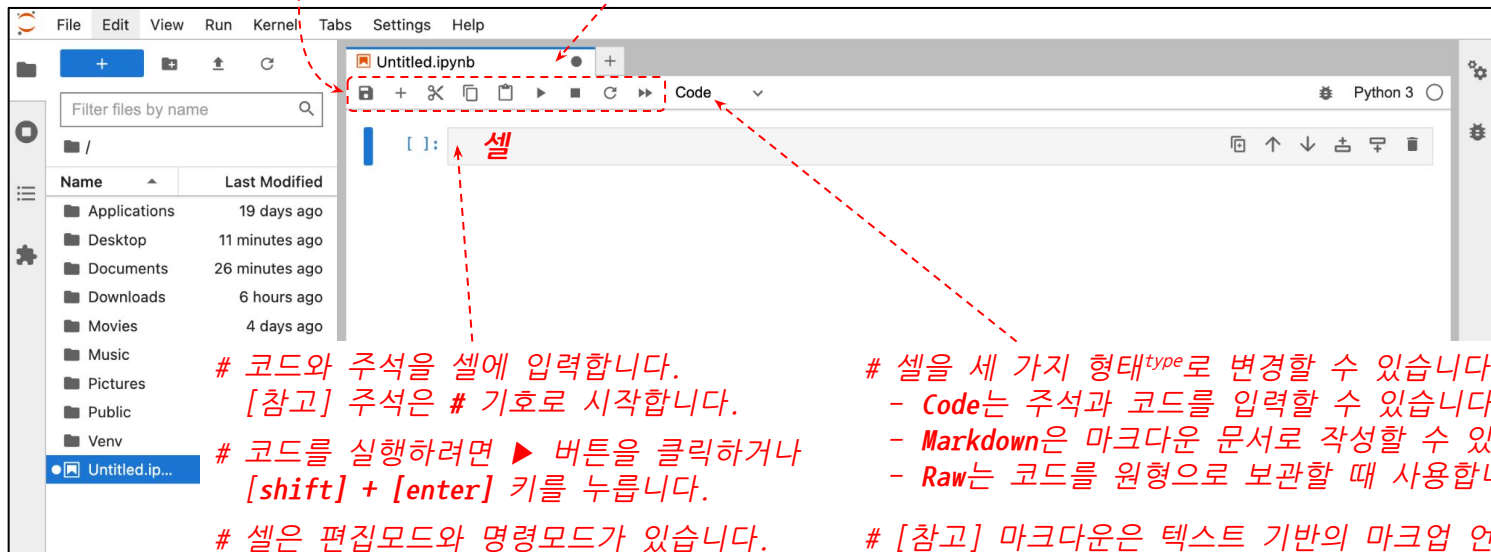
Terminal Text File Markdown File Python File Show Contextual Help

Simple 0 0 Launcher

JupyterLab에서 Notebook 메뉴

저장, 셀^{Cell} 추가, 잘라내기, 복사, 붙여넣기 및 코드 실행 메뉴입니다.

탭에서 마우스 오른쪽 버튼을 클릭하면 메뉴 팝업이 뜨는데 Rename Notebook을 선택하면 이름을 변경할 수 있습니다.



코드와 주석을 셀에 입력합니다.
[참고] 주석은 # 기호로 시작합니다.

코드를 실행하려면 ▶ 버튼을 클릭하거나 [shift] + [enter] 키를 누릅니다.

셀은 편집모드와 명령모드가 있습니다.
편집모드일 때 코드를 입력할 수 있습니다.

셀을 세 가지 형태^{type}로 변경할 수 있습니다.

- Code는 주석과 코드를 입력할 수 있습니다.
- Markdown은 마크다운 문서로 작성할 수 있습니다.
- Raw는 코드를 원형으로 보관할 때 사용합니다.

[참고] 마크다운은 텍스트 기반의 마크업 언어이며 HTML 또는 PDF로 쉽게 변환할 수 있습니다.

JupyterLab에서 Notebook 셀 모드

- 편집모드는 셀이 활성화된 상태이며, 셀에 코드와 주석을 입력하고 실행합니다.
- 명령모드는 셀이 비활성화된 상태이며, 단축키로 셀 관련 작업을 실행합니다.

편집모드



- ① 키보드에서 **esc** 키를 누릅니다.
- ② 마우스로 셀 바깥쪽을 클릭합니다.

- ① 키보드에서 **enter** 키를 누릅니다.
- ② 마우스로 셀 안쪽을 클릭합니다.

명령모드



여러 셀 복사/붙여넣기, 셀 위/아래로 이동, 셀 위/아래 추가 및 셀 삭제 메뉴입니다.

JupyterLab 환경설정

The screenshot shows the JupyterLab interface with the Settings menu open. Red dashed arrows point from Korean text annotations to specific settings:

- JupyterLab Dark**: # Jupyter Notebook 테마를 설정합니다. Dark 테마는 눈의 피로를 줄여줍니다.
- Autosave Documents**: # Jupyter Notebook을 자동 저장하도록 설정되어 있는데 해제할 수 있습니다.
- Auto Close Brackets**: # 여는 괄호를 입력할 때 닫는 괄호를 자동으로 추가하려면 체크합니다.
- Advanced Settings Editor**: # 고급 설정 항목에서 폰트 및 크기를 설정할 수 있습니다. (다음 페이지)

The interface also shows a file browser on the left and a bottom toolbar with icons for Terminal, Text File, Markdown File, Python File, and Show Contextual Help.

JupyterLab 환경설정 (계속)

Settings

Search...

MODIFIED

- Extension Manager
- Keyboard Shortcuts
- Notebook**
- Theme

SETTINGS

- Cell Toolbar
- Code Console**
- CodeMirror
- Command Palette
- CSV Viewer
- Debugger
- Document Manager
- Document Search
- File Browser
- File Browser Widget
- HTML Viewer
- JupyterLab Shell

JSON Settings Editor

Notebook

Notebook settings.

Restore to Defaults

Code Cell Configuration

The configuration for all code cells.

☐ Auto Closing Brackets

Cursor Blinking Rate

530

Half-period in milliseconds used for cursor blinking. The default blink rate is 530ms. By setting this to zero, blinking can be disabled. A negative value hides the cursor entirely.

Font Family

NanumGothicCoding

코드 기본 폰트를 설정합니다.

Default:

Font Size

14

폰트 기본 크기를 설정합니다.

Default:

Line Height

☐ Show Line Numbers

Line Wrap

off

☒ Match Brackets

☐ Read Only

☒ Insert Spaces

End of Document