



주교재 "파이썬과 함께 배우는 신호와 시스템" 의 Appendix E

보조자료

파이썬

- 객체지향적 고급 범용 프로그래밍 언어 (**무료**)
- 대화형 인터프리트 언어 (즉시 결과 확인)
- 간결하고 읽기 쉬운 문법
- 플랫폼에 독립적
- 다양한 패키지 (라이브러리 활용, 결과를 그래프로 쉽게 표시)
- 최근 인공지능 개발에 활용
- 패키지 관리 및 환경관리 시스템: **Anaconda**

[참조] **MATLAB** (<https://kr.mathworks.com/>)

- 공학용 "Application Software Package"로 프로그래밍 언어 아님 (**유료, 고가**)
- 변수와 데이터를 입력할 수 있는, 제품화된 ToolBox를 구입하여 사용
- 수식계산에 탁월한 기능과 간편성을 제공 (결과를 그래프로 쉽게 표시)
- 2024년 현재 대학에서 지원

시작하기

- Anaconda 설치(파이썬 포함)

- <https://www.anaconda.com/download>에서 PC OS에 맞게 다운 후 설치

ANACONDA. Products Solutions Resources Partners Company Free Download Sign Up Sign In

Distribution

Free Download*

Register to get everything you need to get started on your workstation including Cloud Notebooks, Navigator, AI Assistant, Learning and more.

- ✓ 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
- ✓ Distribution installation on Windows, MacOS, or Linux

*Use of Anaconda's Offerings at an organization of more than 200 employees requires a Business or Enterprise license. [See Pricing](#)

Provide email to download Distribution

Email Address: ← 메일주소 입력

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](#)

✓

Success!

Please check your email for the download link.

| | | | |
|-------|-------------------|---|-------|
| 보낸 사람 | 주소록추가 | Anaconda, Inc. <account@anaconda.cloud> | 수신 거부 |
| 보낸 날짜 | 8. 31 오전 11:59:53 | | |
| 받는 사람 | mcw@inje.ac.kr | | |

메일 확인하여 다운로드

ANACONDA.

Anaconda Distribution Download

Thank you for choosing Anaconda and Conda packages.
Continue with your download.

Download Now

시작하기

- 다운로드 후 실행




Download Now

For installation assistance, refer to [Troubleshooting](#).

Download Distribution by choosing the proper installer for your machine.

[Download](#)

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) |
|--|---|---|

Anaconda3 2024.02-1 (64-bit) Setup

Welcome to Anaconda3 2024.02-1 (64-bit) Setup

Setup will guide you through the installation of Anaconda3 2024.02-1 (64-bit).

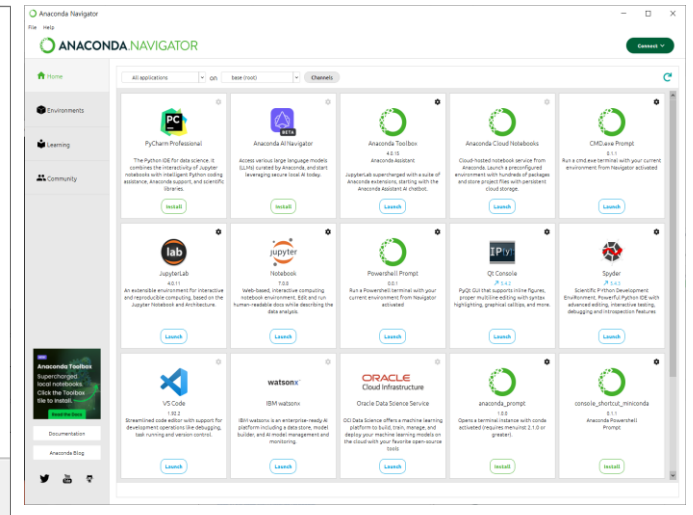
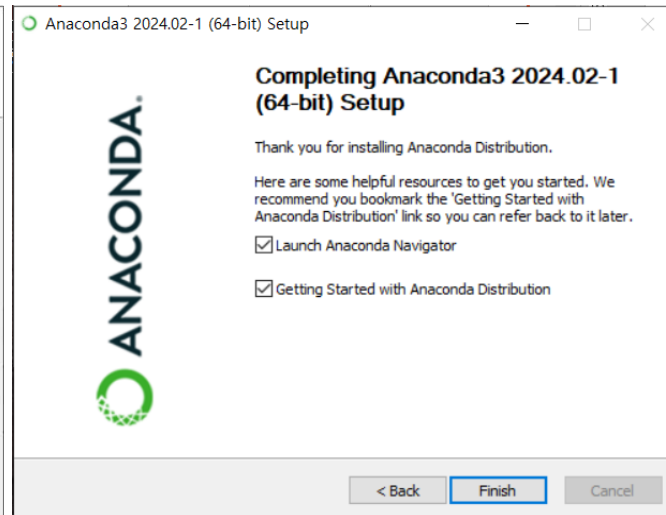
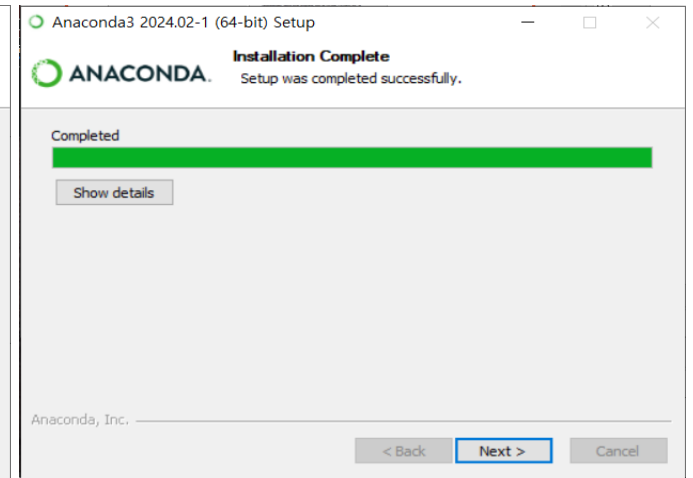
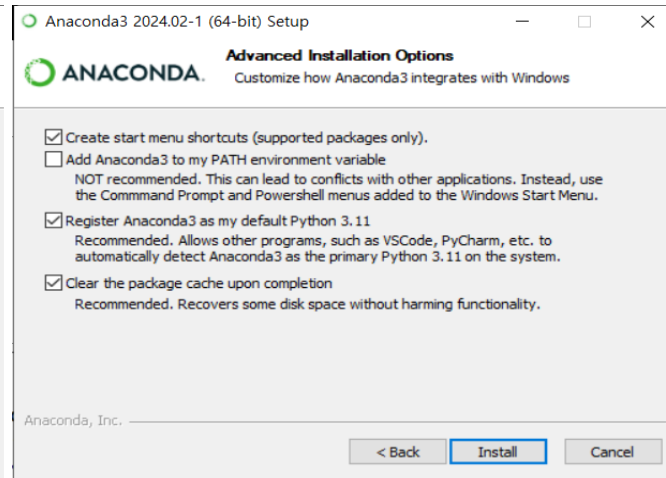
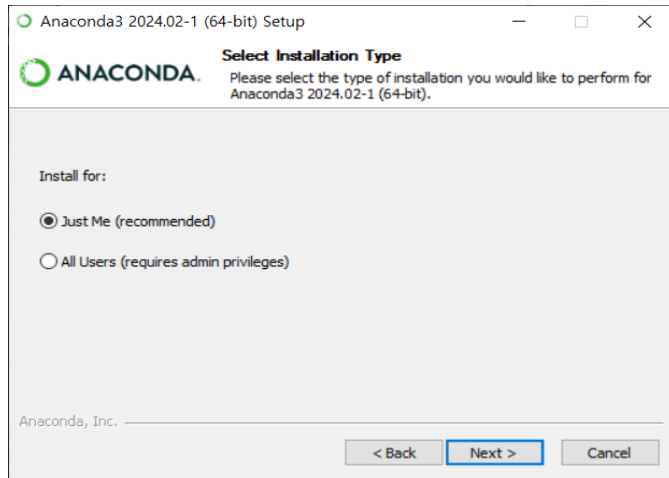
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

[Next >](#) [Cancel](#)

시작하기

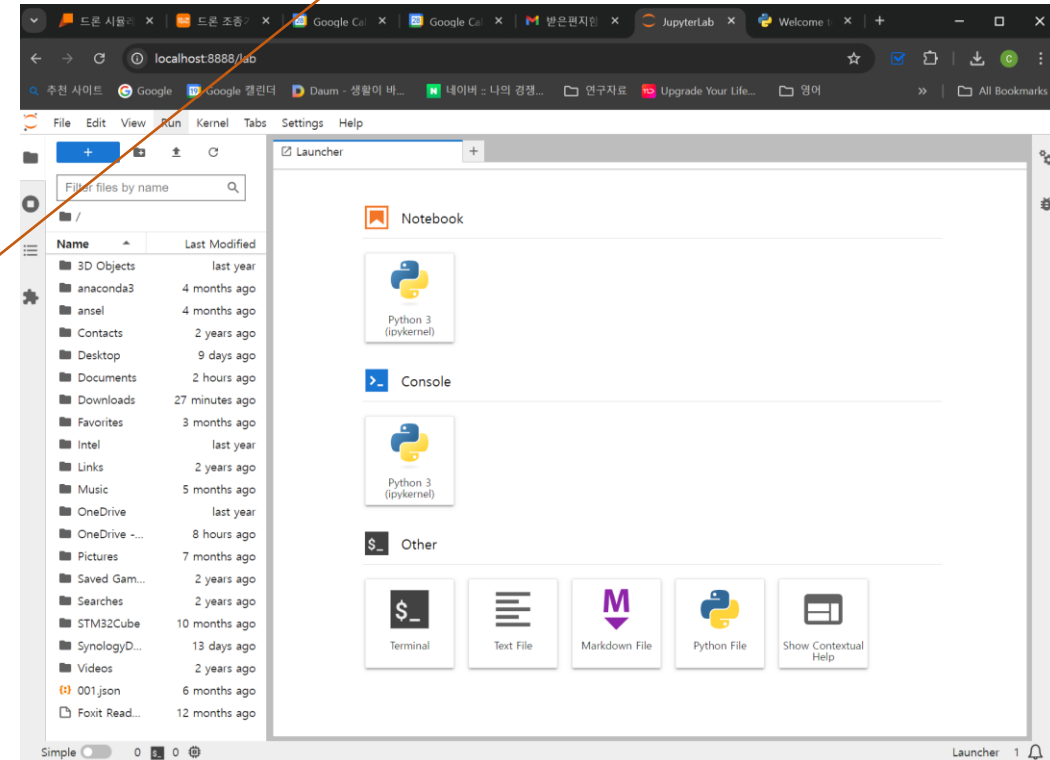
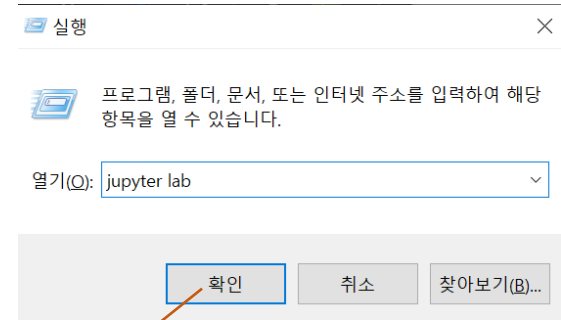
- 계속 설치



시작하기

- 통합개발환경(IDE): **Jupyter Lab**
 - 많은 IDE: Visual Studio, VS code,
 - ipython shell 환경 제공
- 설치

- 윈도우에서 WINDOW+R ↓ → 실행 창
- 실행 창에 'jupyter lab'을 입력하고 확인



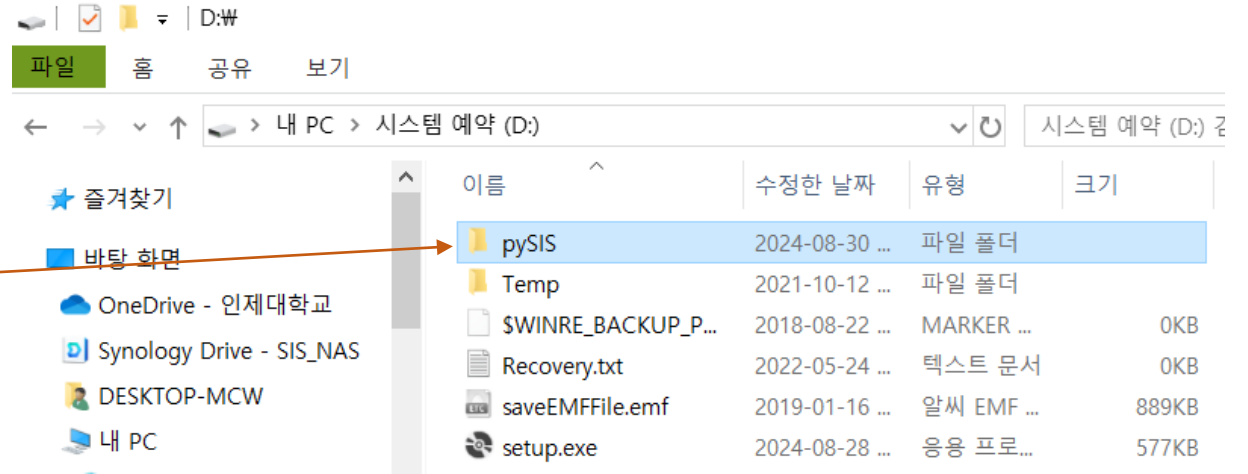
```
C:\Users\WDESKTOP-MCW\Anaconda3\Scripts\jupyter.exe
2024-08-28 16:59:38.341 ServerApp Package jupyterlab took 0.0000s to import.
2024-08-28 16:59:38.408 ServerApp Package jupyter_lsp took 0.0664s to import
2024-08-28 16:59:38.448 ServerApp A ``jupyter_server_extension_points`` function was not found in jupyter_lsp. Instead, a ``jupyter_server_extension_paths`` fu
nction was found and will be used for now. This function name will be deprecated in future releases of Jupyter Server.
2024-08-28 16:59:38.447 ServerApp Package jupyter_server_terminals took 0.0390s to import
2024-08-28 16:59:38.452 ServerApp Package notebook took 0.0000s to import
2024-08-28 16:59:38.452 ServerApp Package notebook_shim took 0.0000s to import
2024-08-28 16:59:38.452 ServerApp A ``jupyter_server_extension_points`` function was not found in notebook_shim. Instead, a ``jupyter_server_extension_paths``
function was found and will be used for now. This function name will be deprecated in future releases of Jupyter Server.
2024-08-28 16:59:39.925 ServerApp Package panel.io.jupyter_server_extension took 1.4699s to import
2024-08-28 16:59:39.925 ServerApp jupyter_lsp | extension was successfully linked.
2024-08-28 16:59:39.930 ServerApp jupyter_server_terminals | extension was successfully linked.
2024-08-28 16:59:39.936 ServerApp jupyterlab | extension was successfully linked.
2024-08-28 16:59:39.942 ServerApp notebook | extension was successfully linked.
2024-08-28 16:59:39.948 ServerApp Writing Jupyter server cookie secret to C:\Users\WDESKTOP-MCW\AppData\Roaming\jupyter\runtime\jupyter_cookie_secret
2024-08-28 16:59:40.769 ServerApp notebook_shim | extension was successfully linked.
2024-08-28 16:59:40.769 ServerApp panel.io.jupyter_server_extension | extension was successfully linked.
2024-08-28 16:59:40.827 ServerApp notebook_shim | extension was successfully loaded.
2024-08-28 16:59:40.829 ServerApp jupyter_lsp | extension was successfully loaded.
2024-08-28 16:59:40.850 ServerApp jupyter_server_terminals | extension was successfully loaded.
2024-08-28 16:59:40.856 ServerApp JupyterLab extension loaded from C:\Users\WDESKTOP-MCW\Anaconda3\lib\site-packages\jupyterlab
2024-08-28 16:59:40.837 LabApp JupyterLab application directory is C:\Users\WDESKTOP-MCW\Anaconda3\share\jupyter\lab
2024-08-28 16:59:40.838 LabApp Extension Manager is 'pypi'.
2024-08-28 16:59:40.840 ServerApp jupyterlab | extension was successfully loaded.
2024-08-28 16:59:40.847 ServerApp notebook | extension was successfully loaded.
2024-08-28 16:59:40.847 ServerApp panel.io.jupyter_server_extension | extension was successfully loaded.
2024-08-28 16:59:40.848 ServerApp Serving notebooks from local directory: C:\Users\WDESKTOP-MCW
2024-08-28 16:59:40.849 ServerApp Jupyter Server 2.10.0 is running at:
2024-08-28 16:59:40.849 ServerApp http://localhost:8888/lab?token=7e8dd88ef61dd0a6c94070533a9f45118fee7d03cdc43a3a
2024-08-28 16:59:40.851 ServerApp http://127.0.0.1:8888/lab?token=7e8dd88ef61dd0a6c94070533a9f45118fee7d03cdc43a3a
```

시작하기

- **작업 폴더 만들기**

예) D:\mkdir pySIS

D:\pySIS



- **Jupyter Lab 경로 변경**

- 설정된 폴더(D:\pySIS)에서 jupyter lab 시작하게 함
[검색 및 참조]

<https://velog.io/@yellofi/%ED%8C%8C%EC%9D%B4%EC%8D%AC%EC%A3%BC%ED%94%BC%ED%84%B0-Jupyter-Lab-%EA%B2%BD%EB%A1%9C-%EB%B3%80%EA%B2%BD>

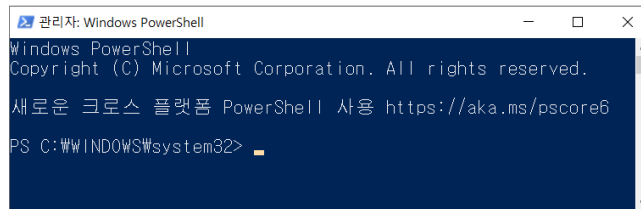
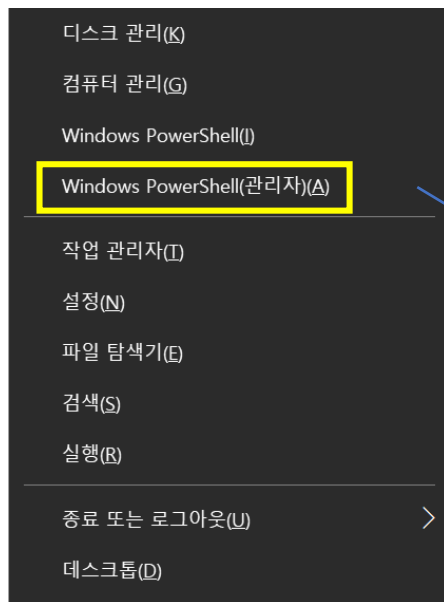
- 명령창에서 `jupyter lab --generate-config`

- C:\사용자(users)\(사용자 폴더 이름)\jupyter\jupyter_config.py 생성
- C.ServerApp.notebook_dir="" 찾아 좌 그림으로 수정

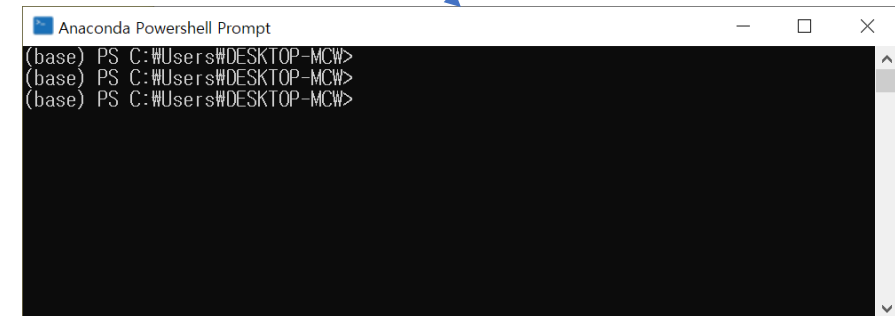
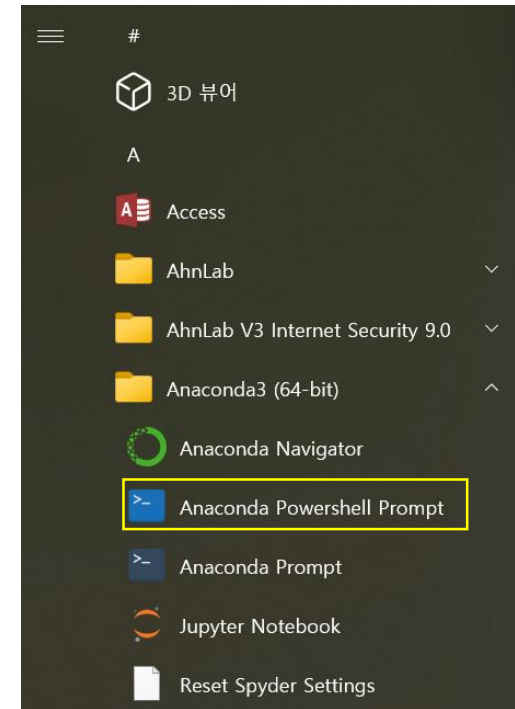
```
# Default: ''  
c.ServerApp.notebook_dir = 'D:\pySIS'
```

파이썬 위젯 사용방법

- ipywidget 모듈 설치
 - 윈도우 키+X → 팝업 창
 - Windows PowerShell(관리자)(A)
 - PowerShell 창 (권장)



또는
시작메뉴에서 Anaconda3 (64-bit)에서
Anaconda Powershell Prompt 클릭



설치방법

> conda install -c conda-forge ipyml

> conda install nodejs

```
PS C:\Users\DESKTOP-MCW> conda install -c conda-forge ipyml
Channels:
- conda-forge
- defaults
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: C:\Users\DESKTOP-MCW\anaconda3

added / updated specs:
- ipyml

The following packages will be downloaded:
```

| package | build | | |
|--------------------------|--------------|--------|-------------|
| ca-certificates-2024.7.4 | h56e8100_0 | 151 KB | conda-forge |
| certifi-2024.7.4 | pyhd8ed1ab_0 | 156 KB | conda-forge |
| ipyml-0.9.4 | pyhd8ed1ab_0 | 209 KB | conda-forge |
| openssl-3.3.1 | h2466b09_3 | 8.0 MB | conda-forge |
| ucrt-10.0.22621.0 | h57928b3_0 | 1.2 MB | conda-forge |

```
done
```

```
PS C:\Users\DESKTOP-MCW> conda install nodejs
Channels:
- defaults
- conda-forge
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: C:\Users\DESKTOP-MCW\anaconda3

added / updated specs:
- nodejs

The following packages will be downloaded:
```

| package | build | |
|------------------|-----------------|---------|
| conda-24.7.1 | py311haa95532_0 | 1.2 MB |
| frozendict-2.4.2 | py311haa95532_0 | 38 KB |
| nodejs-18.18.2 | haa95532_0 | 24.2 MB |
| Total: | | 25.6 MB |

The following NEW packages will be INSTALLED:

```
frozendict      pkgs/main/win-64::frozendict-2.4.2-py311haa95532_0
nodejs          pkgs/main/win-64::nodejs-18.18.2-haa95532_0
```

The following packages will be UPDATED:

The following packages will be SUPERSEDED by a higher-priority channel:

```
certifi         conda-forge/noarch::certifi-2024.7.4~ --> pkgs/main/win-64::certifi-2024.7.4-py311haa95532_0
```

Proceed ([y]/n)? y

Downloading and Extracting Packages:

```
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

설치방법 (jupyterlab-manager)

> Jupyter labextension install @jupyter-widgets/jupyterlab-manager

```
PS C:\Users\DESKTOP-MCW> jupyter labextension install @jupyter-widgets/jupyterlab-manager
(Deprecated) Installing extensions with the jupyter labextension install command is now deprecated and will be removed
in a future major version of JupyterLab.

Users should manage prebuilt extensions with package managers like pip and conda, and extension authors are encouraged
to distribute their extensions as prebuilt packages
Building jupyterlab assets (production, minimized)
```

설치방법 (jupyter-matplotlib)

> jupyter labextension install jupyter-matplotlib

```
PS C:\Users\DESKTOP-MCW> conda install jupyter-matplotlib
Channels:
- defaults
- conda-forge
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: failed

PackagesNotFoundError: The following packages are not available from current channels:

- jupyter-matplotlib

Current channels:

- defaults
- https://conda.anaconda.org/conda-forge/win-64
- https://conda.anaconda.org/conda-forge/noarch

To search for alternate channels that may provide the conda package you're
looking for, navigate to

https://anaconda.org

and use the search bar at the top of the page.
```

- ipywidget 참고 사이트

<https://github.com/matplotlib/jupyter-matplotlib>

Voila

- Voila란?
 - 원래 Dashboard 형성이 목적 → 노트북에서 Dashboard로 바로 변환
 - Notebook 기반 코드의 프로토타입을 쉽게 만들어주는 라이브러리
 - 단 시간 내에 **간단한** 프로토타입 작성 가능해서 자주 활용
- Voila 장점
 - Notebook에서 별도 코드 추가없이 실행
 - Jupyter Notebook 결과를 쉽게 웹 형태로 띄울 수 있음

Voila 설치 및 설정

- ipywidget 사용에 편리
- Voila를 사용하기 위한 설치
 - > `conda install -c conda-forge voila`
 - > ~~`jupyter labextension install @jupyter-voila/jupyterlab-preview`~~
 - > `conda install @jupyter-voila/jupyterlab-preview`

[참조] voila에 대한 상세한 사항

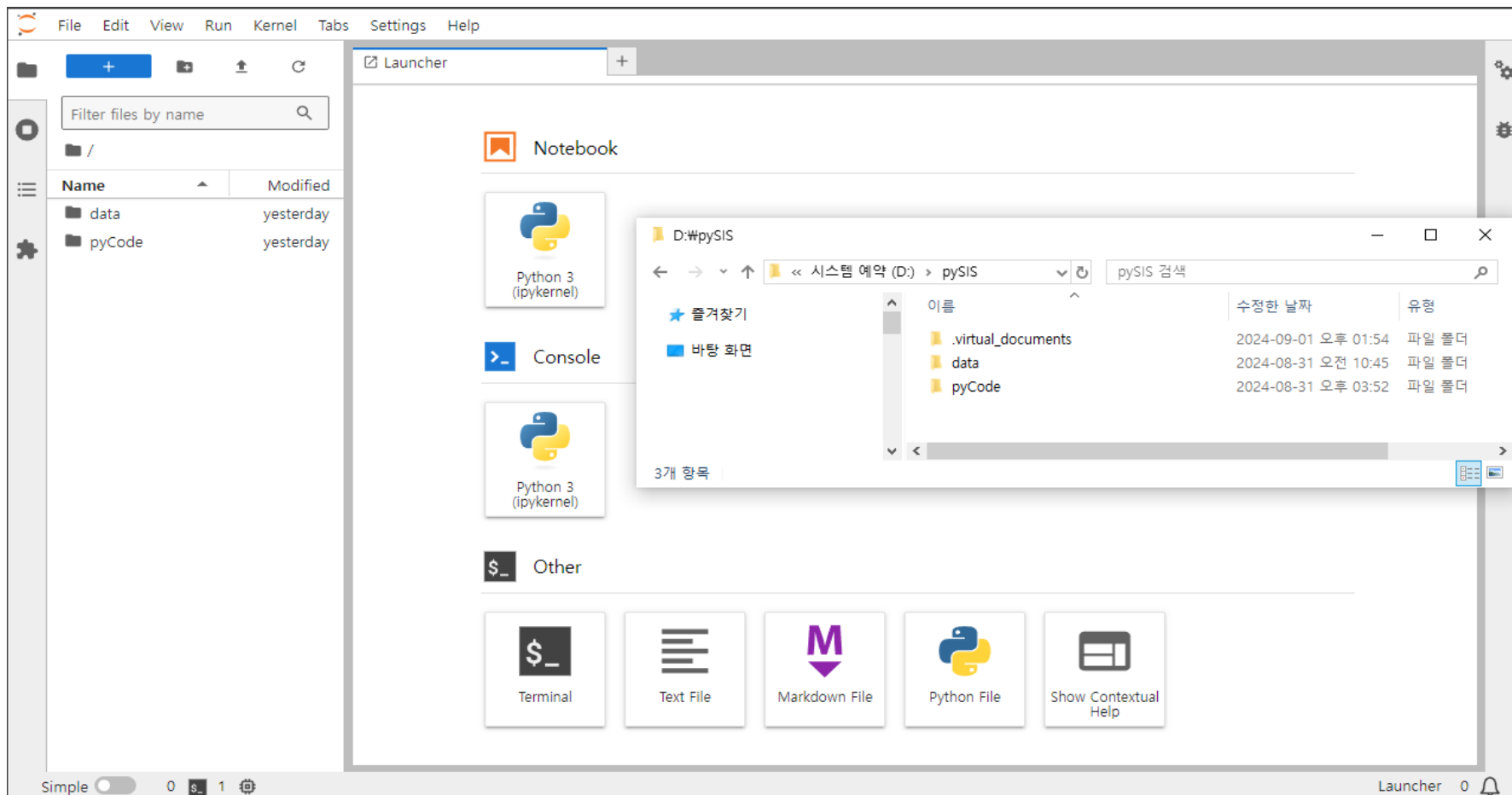
<https://github.com/voila-dashboards/voila>

Jupyter Lab 시작

- 명령 창 또는 PowerShell

```
관리자: Windows PowerShell
PS D:\#pySIS>
PS D:\#pySIS>
PS D:\#pySIS> jupyter lab
```

- 저장된 코드 편집
 - 해당 폴더로 이동
- 새 코드 편집
 - 우측 Launcher 아이콘 누름



Jupyter Lab 사용 및 종료

- 새 파일 이름.확장자 = **untitled.ipynb**

- Jupyter lab 셀 사용

- code 수정 및 실행

- Markdown 타입

- 입력:

- 출력:

The screenshot shows a Jupyter Lab code cell with a blue vertical bar on the left. The cell contains two parts: a code editor and an output area. The code editor has a light blue background and contains the following text: `## 파이썬 예제 1_1:`, `##### 연속시간 신호의 생성과 파형 그리기`, `[1]: import numpy as np`, and `import matplotlib.pyplot as plt`. The output area has a white background and contains the text: `파이썬 예제 1_1:`, `연속시간 신호의 생성과 파형 그리기`, and `import numpy as np`, `import matplotlib.pyplot as plt`. The output area also has a light blue background. The cell has a toolbar on the right with icons for expand, up, down, refresh, and delete.

- Jupyter lab 종료

- 명령 프롬프트 혹은 터미널에서 CTRL+C ↓

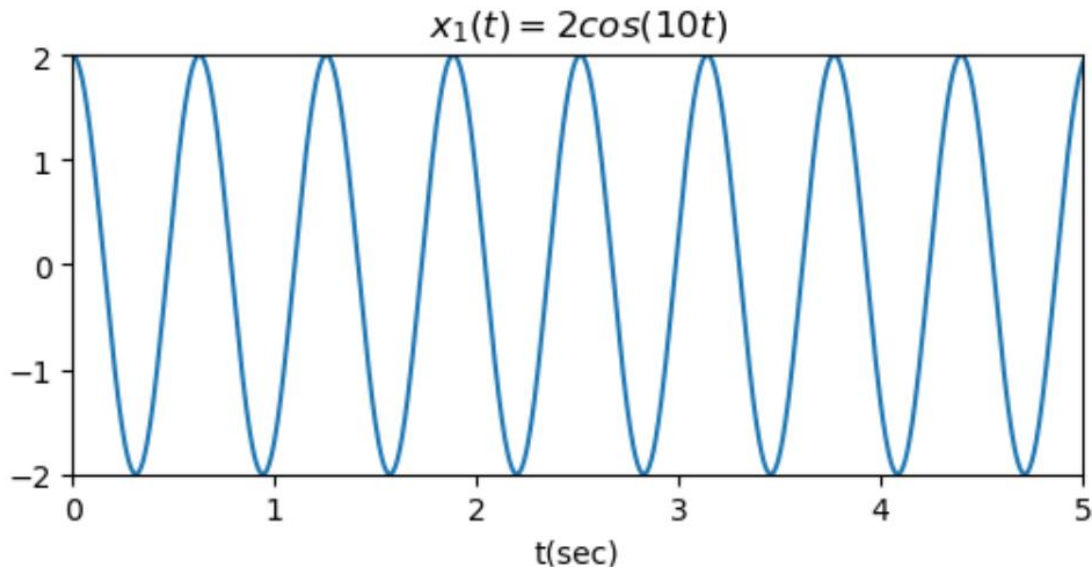
- 파일 메뉴에서 log out

사용 예

```
[1]: import numpy as np
import matplotlib.pyplot as plt
```

```
[2]: t = np.linspace(0, 5, 1001)
x1 = 2 * np.cos(10*t)
fig, ax = plt.subplots( figsize=(6, 2.5) )
ax.plot(t,x1)
ax.set_xlim(0,5)
ax.set_ylim(-2.0, 2.0)
ax.set_xlabel( 't(sec)')
ax.set_title('$x_1(t) = 2\cos(10t)$')
```

```
[2]: Text(0.5, 1.0, '$x_1(t) = 2\cos(10t)$')
```



```
[4]: t2 = np.arange(-1, 3, 0.001)
x2 = np.exp(-t2) * (t2>0)
fig, ax = plt.subplots( figsize=(6, 2.5) )
ax.plot(t2, x2)
ax.set_xlim(-1,3)
ax.set_xlabel('t(sec)')
ax.set_title('$x_2(t)=e^{-t}u(t)$')
```

```
[4]: Text(0.5, 1.0, '$x_2(t)=e^{-t}u(t)$')
```

