



# 파이썬 데이터 분석 개발 환경



## ◆ Anaconda 설치하기

---

## ✓ Python 개발환경

---

◆ Anaconda 설치하기 → 접속 URL

<https://www.anaconda.com>

## ✓ Python 개발환경

◆ Anaconda 다운로드



## Anaconda Installers



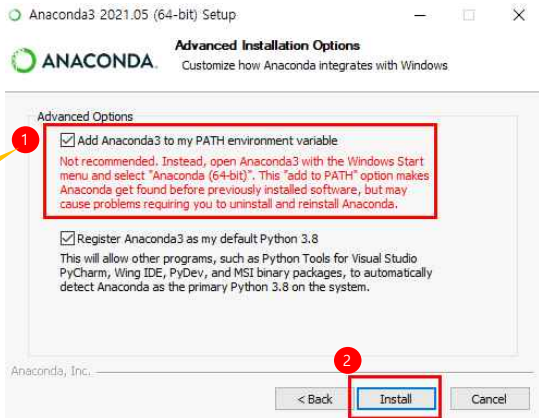
Windows

Python 3.11

↓ 64-Bit Graphical Installer (904.4M)

## ✓ Python 개발환경

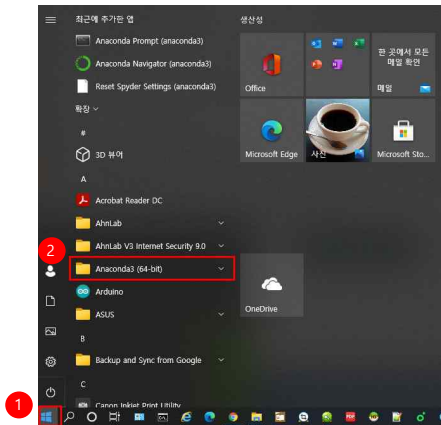
◆ Anaconda 설치하기 → [Add Path] 체크 → [Install] 버튼 선택 → 설치 시작됨



PC에 파이썬(Python)이  
이미 설치되어 있는  
경우에는  
체크하지 마세요!!

## ✓ Python 개발환경

◆ Anaconda 설치하기 → Anaconda 실행 파일들 위치 → [윈도우 아이콘] 선택 → [Anaconda3] 선택



## ✓ Python 개발환경

◆ Jupyter Notebook 시작하는 방법 → [윈도우] 에서 시작 [Jupyter Notebook(anaconda3)] 선택



## ✓ Python 개발환경

- ◆ Jupyter Notebook 시작하는 방법 → [윈도우] 에서 시작 → [Jupyter Notebook] 실행 화면
- ◆ Command 창(검정 화면)이 실행 됨 → 프로그램 중에는 종료하면 안됨

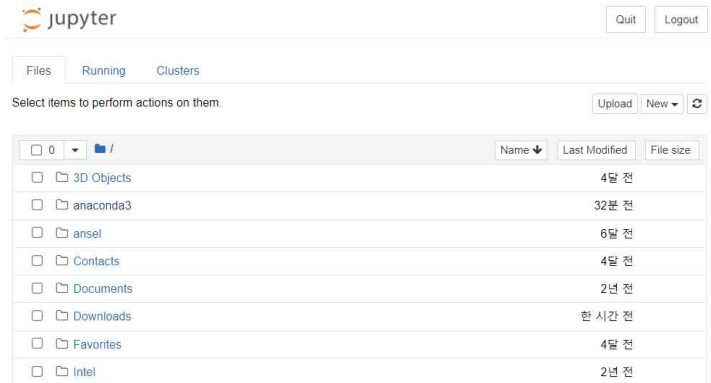
```
Jupyter Notebook (anaconda3)
[W 2021-07-07 13:46:01.016 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to
ServerApp. Be sure to update your config before our next release.
[W 2021-07-07 13:46:01.016 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to
ServerApp. Be sure to update your config before our next release.
[I 2021-07-07 13:46:01.026 LabApp] JupyterLab extension loaded from C:\Users\mento\Anaconda3\lib\site-packages\jupyterlab
[I 2021-07-07 13:46:01.027 LabApp] JupyterLab application directory is C:\Users\mento\Anaconda3\share\jupyter\lab
[I 13:46:01.031 NotebookApp] Serving notebooks from local directory: C:\Users\mento
[I 13:46:01.031 NotebookApp] Jupyter Notebook 6.3.0 is running at:
[I 13:46:01.031 NotebookApp] http://localhost:8888/?token=9049aca1346dfba04c8d97f5a84bb2b6f54b5ec45361f703
[I 13:46:01.032 NotebookApp] or http://127.0.0.1:8888/?token=9049aca1346dfba04c8d97f5a84bb2b6f54b5ec45361f703
[I 13:46:01.032 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 13:46:01.282 NotebookApp]

To access the notebook, open this file in a browser:
    file:///C:/Users/mento/AppData/Roaming/Jupyter/runtime/nbserver-21368-open.html
Or copy and paste one of these URLs:
    http://localhost:8888/?token=9049aca1346dfba04c8d97f5a84bb2b6f54b5ec45361f703
    or http://127.0.0.1:8888/?token=9049aca1346dfba04c8d97f5a84bb2b6f54b5ec45361f703
```



## ✓ Python 개발환경

- ◆ Jupyter Notebook 시작하는 방법 → [윈도우] 에서 시작 → [Jupyter Notebook] 실행 화면
- ◆ Command 창(검정 화면)이 실행 후에 [Jupyter Notebook] 실행화면이 보임



The screenshot shows the Jupyter Notebook web interface. At the top, there is a 'jupyter' logo and two buttons: 'Quit' and 'Logout'. Below the logo, there are three tabs: 'Files', 'Running', and 'Clusters'. The 'Files' tab is selected. Below the tabs, there is a text prompt 'Select items to perform actions on them.' and three buttons: 'Upload', 'New', and a refresh icon. Below this, there is a file browser table with columns for 'Name', 'Last Modified', and 'File size'. The table lists several folders: '3D Objects', 'anaconda3', 'ansel', 'Contacts', 'Documents', 'Downloads', 'Favorites', and 'Intel'. Each folder has a checkbox to its left.

	Name	Last Modified	File size
<input type="checkbox"/>	0	/	
<input type="checkbox"/>	3D Objects	4달 전	
<input type="checkbox"/>	anaconda3	32분 전	
<input type="checkbox"/>	ansel	6달 전	
<input type="checkbox"/>	Contacts	4달 전	
<input type="checkbox"/>	Documents	2년 전	
<input type="checkbox"/>	Downloads	한 시간 전	
<input type="checkbox"/>	Favorites	4달 전	
<input type="checkbox"/>	Intel	2년 전	

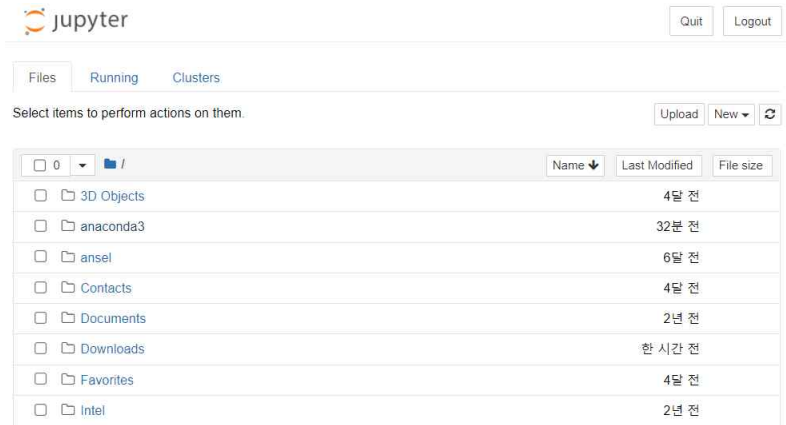


## ◆ Jupyter Notebook Tool 이해하기

---

## ✓ Python 개발환경

### ◆ Jupyter Notebook Tool 이해하기 → [Jupyter Notebook] 실행하기

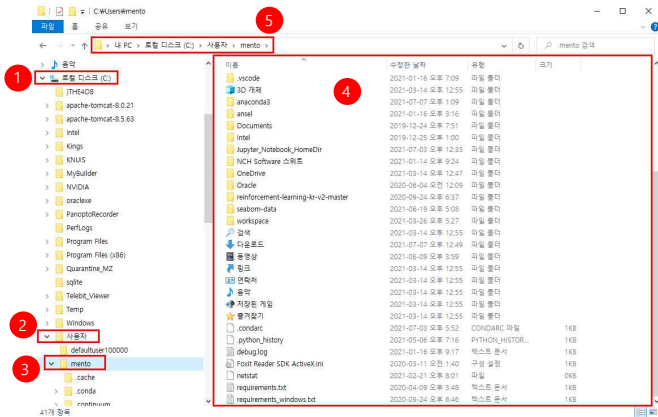


The screenshot displays the Jupyter Notebook web interface. At the top, the Jupyter logo is on the left, and 'Quit' and 'Logout' buttons are on the right. Below the logo, there are three tabs: 'Files' (selected), 'Running', and 'Clusters'. A message 'Select items to perform actions on them.' is shown, followed by 'Upload', 'New' (with a dropdown arrow), and a refresh icon. The main area is a file browser showing a list of files and folders. The table has columns for 'Name', 'Last Modified', and 'File size'. The files listed are: 3D Objects, anaconda3, anseI, Contacts, Documents, Downloads, Favorites, and Intel.

	Name	Last Modified	File size
<input type="checkbox"/>	0		
<input type="checkbox"/>	3D Objects	4달 전	
<input type="checkbox"/>	anaconda3	32분 전	
<input type="checkbox"/>	anseI	6달 전	
<input type="checkbox"/>	Contacts	4달 전	
<input type="checkbox"/>	Documents	2년 전	
<input type="checkbox"/>	Downloads	한 시간 전	
<input type="checkbox"/>	Favorites	4달 전	
<input type="checkbox"/>	Intel	2년 전	

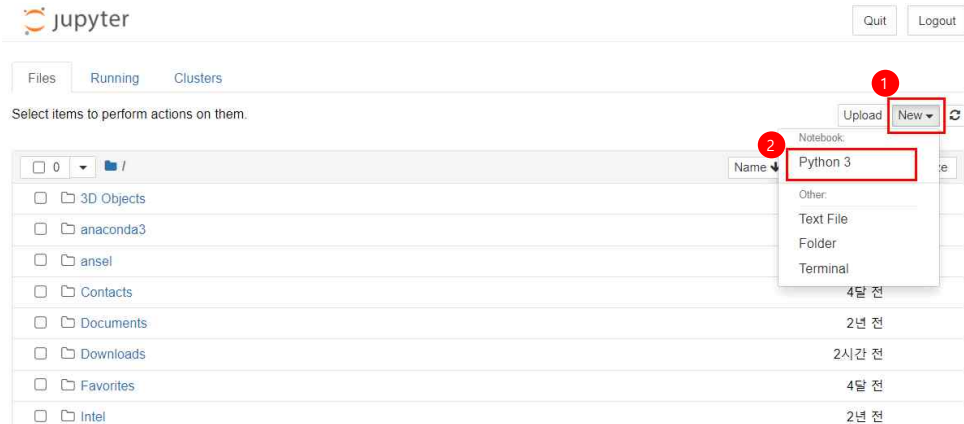
## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → [Jupyter Notebook] 기본 작업 파일 위치 → 폴더 확인



## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → [Python] 프로그램 시작하기 → [New > Python3] 선택



The screenshot shows the Jupyter Notebook web interface. At the top, there's a header with the Jupyter logo and 'Quit' and 'Logout' buttons. Below the header, there are tabs for 'Files', 'Running', and 'Clusters'. A message says 'Select items to perform actions on them.' Below this, there's a file browser showing a directory structure. A dropdown menu is open, showing options for creating new files or folders. The 'New' button is highlighted with a red circle and the number 1. The 'Python 3' option under the 'Notebook' section is highlighted with a red box and the number 2.

Quit Logout

Files Running Clusters

Select items to perform actions on them.

0 /

- 3D Objects
- anaconda3
- ansel
- Contacts
- Documents
- Downloads
- Favorites
- Intel

Upload New

Notebook:

- Python 3
- Other:
- Text File
- Folder
- Terminal

4달 전


2년 전

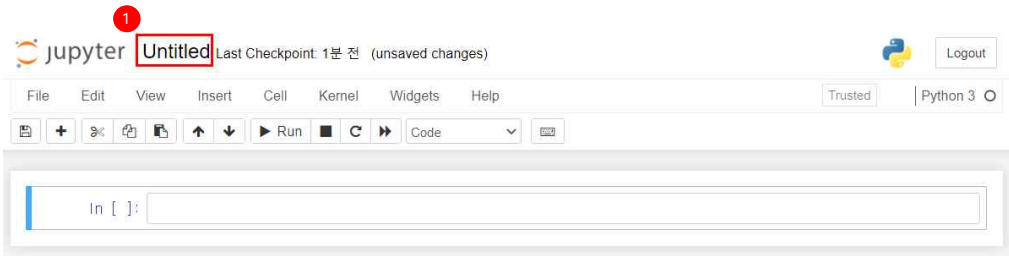
2시간 전

4달 전


2년 전

## ✓ Python 개발환경

- ◆  이해하기 → 작업 파일 위치 확인
- ◆ “Untitled”는 현재 작업의 제목으로 파일명과 동일하다.



## ✓ Python 개발환경

◆  이해하기 → 작업 파일 위치 확인

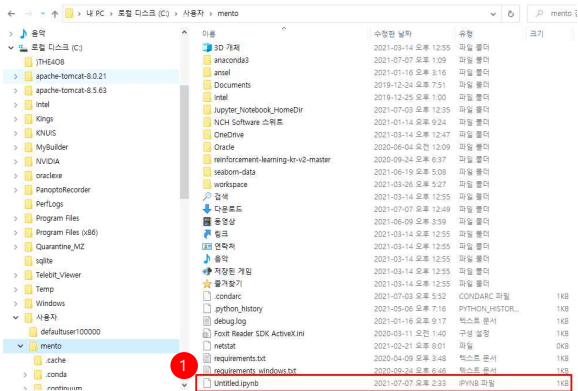
◆ [Jupyter Notebook 디렉토리]에서 "Untitled" 제목의 파일 위치 확인 → 확장자는 "ipynb"

<input type="checkbox"/>	OneDrive	4달 전
<input type="checkbox"/>	Oracle	일 년 전
<input type="checkbox"/>	reinforcement-learning-kr-v2-master	9달 전
<input type="checkbox"/>	Saved Games	4달 전
<input type="checkbox"/>	seaborn-data	18일 전
<input type="checkbox"/>	Searches	4달 전
<input type="checkbox"/>	Videos	한 달 전
<input type="checkbox"/>	workspace	3달 전
<input type="checkbox"/>	 Untitled.ipynb	Running 1분 전 576 B
<input type="checkbox"/>	debug.log	6달 전 105 B
<input type="checkbox"/>	Foxit Reader SDK ActiveX.ini	일 년 전 29 B

## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기      **작업 파일 위치 확인**

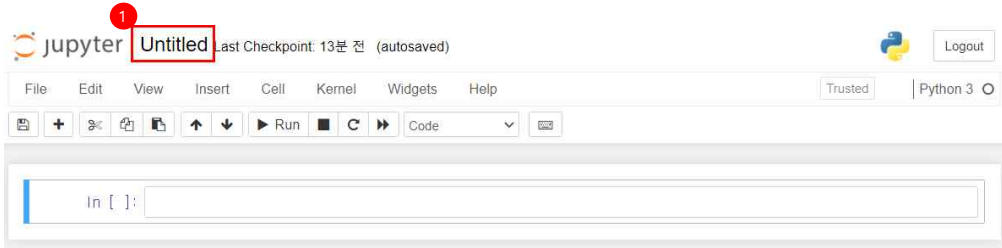
◆ [실제 디렉토리]에서 "Untitled" 제목의 파일 위치 확인 → 확장자는 "ipynb"





## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook > 작업 파일]에서 직접 수정 → “Untitled” 제목 클릭



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook > 작업 파일]에서 직접 수정 → “Untitled” 제목 클릭 후 화면

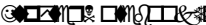


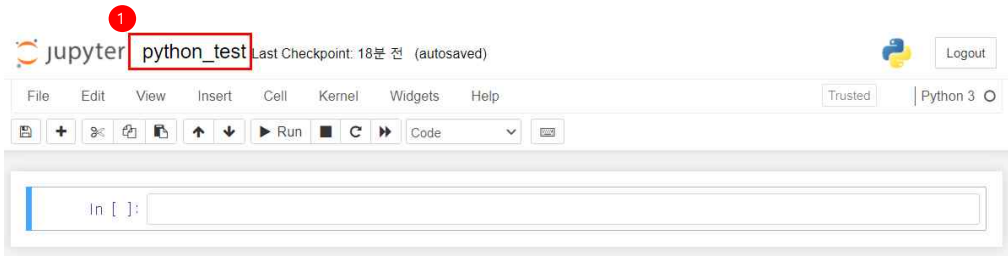
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook > 작업 파일]에서 직접 수정 → 제목을 “python\_test”로 수정 → [Rename] 선택




## ✓ Python 개발환경

- ◆  이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook > 작업 파일]에서 직접 수정 → 수정된 화면 확인



## ✓ Python 개발환경

◆  이해하기 → 제목(파일명) 수정하는 방법

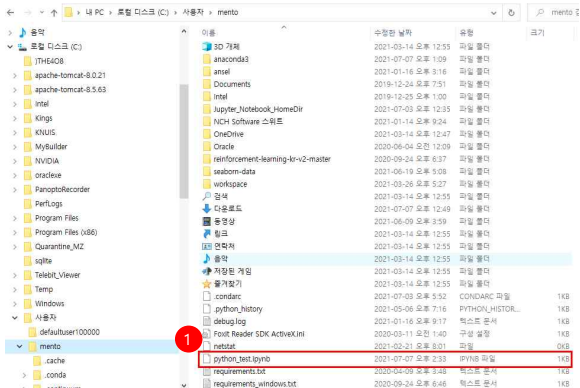
◆ [Jupyter Notebook > 작업 파일]에서 직접 수정 → 수정된 파일 확인 [Jupyter Notebook] 디렉토리

<input type="checkbox"/>	📁 Saved Games		4달 전
<input type="checkbox"/>	📁 seaborn-data		18일 전
<input type="checkbox"/>	📁 Searches		4달 전
<input type="checkbox"/>	📁 Videos		한 달 전
<input type="checkbox"/>	📁 workspace		3달 전
1 <input type="checkbox"/>	 python_test.ipynb	Running 17분 전	576 B
<input type="checkbox"/>	📄 debug.log		6달 전 105 B
<input type="checkbox"/>	📄 Foxit Reader SDK ActiveX.ini		일 년 전 29 B
<input type="checkbox"/>	📄 netstat		5달 전 0 B
<input type="checkbox"/>	📄 requirements.txt		일 년 전 97 B

## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법

◆ [Jupyter Notebook > 작업 파일]에서 직접 수정 → 수정된 파일 확인 [실제 디렉토리] 확인



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook 홈 디렉토리]에서 수정 → 작업 파일 닫기 → 수정 파일 체크 후 [Shutdown] 선택

The screenshot shows the Jupyter Notebook web interface. At the top, there's a 'jupyter' logo and 'Quit' and 'Logout' buttons. Below the logo, there are tabs for 'Files' (highlighted with a red circle and the number '2'), 'Running', and 'Clusters'. Under the 'Files' tab, there are buttons for 'Duplicate', 'Shutdown' (highlighted with a red box), 'View', 'Edit', and a trash icon. To the right of these buttons are 'Upload', 'New', and a refresh icon. The main area displays a file list with columns for 'Name', 'Last Modified', and 'File size'. The file list includes folders like '3D Objects', 'anaconda3', 'ansel', 'Contacts', 'Documents', 'Videos', and 'workspace', and files like 'python\_test.ipynb' (which is checked with a checkbox and highlighted with a red box and the number '1'), 'debug.log', and 'Foxit Reader SDK ActiveX.ini'.

	Name	Last Modified	File size
<input type="checkbox"/>	3D Objects	4달 전	
<input type="checkbox"/>	anaconda3	2시간 전	
<input type="checkbox"/>	ansel	6달 전	
<input type="checkbox"/>	Contacts	4달 전	
<input type="checkbox"/>	Documents	2년 전	
<input type="checkbox"/>	Videos	한 달 전	
<input type="checkbox"/>	workspace	3달 전	
<input checked="" type="checkbox"/>	python_test.ipynb	Running 23분 전	576 B
<input type="checkbox"/>	debug.log	6달 전	105 B
<input type="checkbox"/>	Foxit Reader SDK ActiveX.ini	일 년 전	29 B

## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법

◆ [Jupyter Notebook 홈 디렉토리]에서 수정 → 작업 파일 닫기 → 수정 파일 체크 후 [Rename] 선택

The screenshot shows the Jupyter Notebook web interface. At the top, there's a 'jupyter' logo and 'Quit' and 'Logout' buttons. Below the logo, there are tabs for 'Files', 'Running', and 'Clusters'. The 'Files' tab is active, showing a file list. A red circle with the number '2' highlights the 'Rename' button in the toolbar. Another red circle with the number '1' highlights the checkbox next to the file 'python\_test.ipynb' in the file list. The file list has columns for 'Name', 'Last Modified', and 'File size'.

	Name	Last Modified	File size
<input type="checkbox"/>	3D Objects	4달 전	
<input type="checkbox"/>	anaconda3	2시간 전	
<input type="checkbox"/>	ansel	6달 전	
<input type="checkbox"/>	Contacts	4달 전	
<input type="checkbox"/>	Documents	2년 전	
<input type="checkbox"/>	Downloads	2시간 전	
<input type="checkbox"/>	Favorites	4달 전	
<input checked="" type="checkbox"/>	python_test.ipynb	27분 전	576 B



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook 홈 디렉토리]에서 수정 → [Rename] 선택 후 화면



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [Jupyter Notebook 홈 디렉토리]에서 수정 → 파일명을 "python\_test2.ipynb"로 수정 → [Rename] 선택



## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법

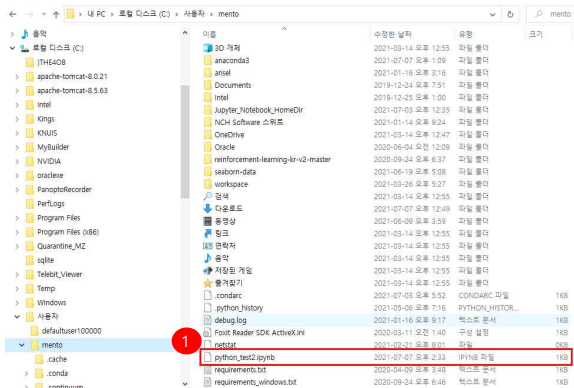
◆ [Jupyter Notebook 홈 디렉토리]에서 수정 → 수정된 파일 확인 → [Jupyter Notebook] 디렉토리

<input type="checkbox"/>	Oracle	일 년 전	
<input type="checkbox"/>	reinforcement-learning-kr-v2-master	9달 전	
<input type="checkbox"/>	Saved Games	4달 전	
<input type="checkbox"/>	seaborn-data	18일 전	
<input type="checkbox"/>	Searches	4달 전	
<input type="checkbox"/>	Videos	한 달 전	
<input type="checkbox"/>	workspace	3달 전	
1 <input type="checkbox"/>	 python_test2.ipynb	31분 전	576 B
<input type="checkbox"/>	debug.log	6달 전	105 B
<input type="checkbox"/>	Foxit Reader SDK ActiveX.ini	일 년 전	29 B
<input type="checkbox"/>	netstat	5달 전	0 B

## ✓ Python 개발환경

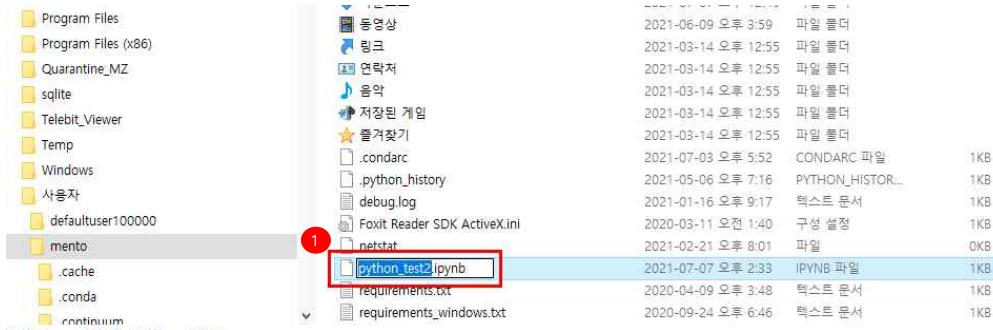
◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법

◆ [Jupyter Notebook 홈 디렉토리]에서 수정 → 수정된 파일 확인 [실제 디렉토리] 확인



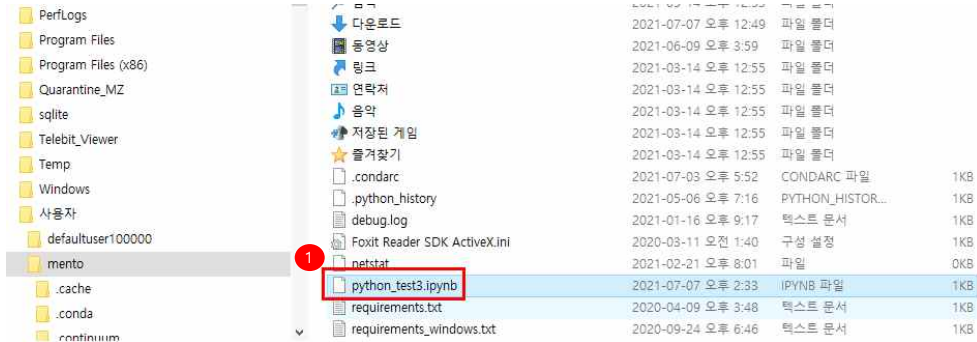
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [실제 폴더]에서 수정 → 수정할 파일 선택 → [F2 키 선택] 화면



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법
- ◆ [실제 폴더]에서 수정 → 수정할 파일 선택 → 파일명을 “python\_test3” 입력 후 [Enter]




## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → 제목(파일명) 수정하는 방법

◆ [실제 폴더]에서 수정 → 파일 선택    수정된 파일 확인 → [Jupyter Notebook] 디렉토리(새로고침)

<input type="checkbox"/>	📁 OneDrive		4달 전
<input type="checkbox"/>	📁 Oracle		일 년 전
<input type="checkbox"/>	📁 reinforcement-learning-kr-v2-master		9달 전
<input type="checkbox"/>	📁 Saved Games		4달 전
<input type="checkbox"/>	📁 seaborn-data		18일 전
<input type="checkbox"/>	📁 Searches		4달 전
<input type="checkbox"/>	📁 Videos		한 달 전
<input type="checkbox"/>	📁 workspace		3달 전
1 <input type="checkbox"/>	📄 python_test3.ipynb	39KB	전 576 B
<input type="checkbox"/>	📄 debug.log	6달 전	105 B
<input type="checkbox"/>	📄 Foxit Reader SDK ActiveX.ini	일 년 전	29 B

## ✓ Python 개발환경

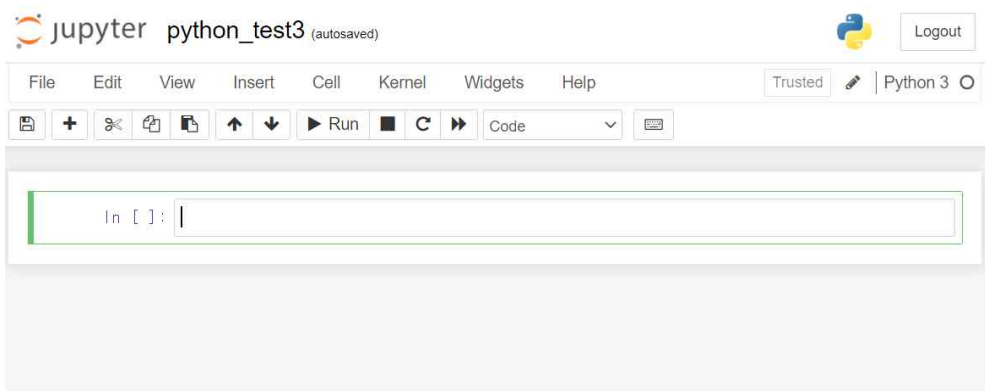
◆  이해하기 → 제목(파일명) 수정하는 방법

파일명 수정, 폴더 생성 등 작업은  
[실제 폴더]에서 수행!!



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ "[python\\_test3.ipynb](#)" 파일 클릭하여 Open 후 화면



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ "Hello Python" 출력해 보기
- ◆ "Hello Python" 입력 후 [Ctrl + Enter]

```
In [1]: "Hello Python"
```

```
Out[1]: 'Hello Python'
```

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 코드(Code) 실행을 위한 단축키 알아보기
- ◆ **[Ctrl + Enter]** : 입력한 박스(Box) 영역만 실행

```
In [6]: print("Hello Python")  
        print("Ctrl + Enter")
```

```
Hello Python  
Ctrl + Enter
```

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 코드(Code) 실행을 위한 단축키 알아보기
- ◆ **[Shipt + Enter]** : 입력한 박스(Box) 영역 실행 후 다음 박스(Box)로 이동(다음 박스가 있는 경우)
- ◆ 다음 박스(Box)가 없으면 새로운 박스(Box) 생성

```
In [5]: print("Hello Python")  
        print("Shipt + Enter")
```

```
Hello Python  
Shipt + Enter
```

```
In [ ]:
```

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 코드(Code) 실행을 위한 단축키 알아보기
- ◆ **[Alt + Enter]** : 입력한 박스(Box) 영역 실행 후 다음 영역에 새로운 박스(Box) 생성

```
In [9]: print("Hello Python")  
        print("Ctrl + Enter")
```

Hello Python  
Ctrl + Enter

```
In [10]: print("Hello Python")  
         print("Alt + Enter")
```

Hello Python  
Alt + Enter

```
In [ ]:
```

```
In [8]: print("Hello Python")  
        print("Shipt + Enter")
```

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 코드(Code) 실행을 위한 단축키 알아보기
- ◆ **[dd]** : 선택한 박스(Box) 영역 삭제하기

```
In [9]: print("Hello Python")  
        print("Ctrl + Enter")
```

Hello Python  
Ctrl + Enter

```
In [11]: print("Hello Python")  
         print("Alt + Enter")
```

Hello Python  
Alt + Enter

1

```
In [ ]: |
```

```
In [8]: print("Hello Python")  
        print("Shipt + Enter")
```

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 코드(Code) 실행을 위한 단축키 알아보기
- ◆ **[dd]** : 선택한 박스(Box) 영역 삭제 후 화면

```
In [9]: print("Hello Python")  
        print("Ctrl + Enter")
```

Hello Python  
Ctrl + Enter

```
In [10]: print("Hello Python")  
         print("Alt + Enter")
```

Hello Python  
Alt + Enter

```
In [8]: print("Hello Python")  
        print("Shipt + Enter")
```

Hello Python  
Shipt + Enter

## ✓ Python 개발환경

◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법

◆ 주석(#) 넣기 : 실행되지 않는 코드(Code) → 코드에 대한 설명을 작성할 때 "#을 붙인 후 작성"

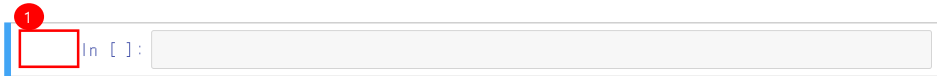
```
# 샵(#)은 주석 : 실행되지 않습니다.  
## : 샵은 여러개 넣어도 됨..  
print("주석 # 처리하기")
```

주석 # 처리하기



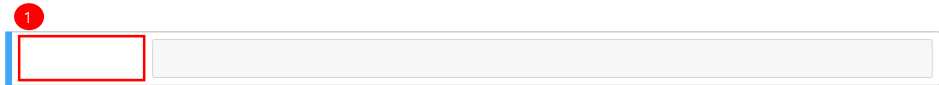
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> **in[]** 왼쪽의 빈 공간을 마우스로 선택



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> "m" 키워드 입력 → in[] 이 없어진다.

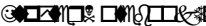



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 1개 입력 후 제목 작성하기 → 가장 큰 Font

# "#" 1개 -> 가장 큰 Font

## ✓ Python 개발환경

- ◆  이해하기 → Editor 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 1개 입력 후 제목 작성하기 →  실행

"#" 1개 -> 가장 큰 Font

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)  
제목 넣는 방법 -> 박스(Box)에 "#을 2개 입력 후 제목 작성하기

## "#" 2개 -> 두번째로 큰 Font

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.  
제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 2개 입력 후 제목 작성하기 [Shift+Enter] 실행

"#" 2개 -> 두번째로 큰 Font

In [ ]: |

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 3개 입력 후 제목 작성하기

### "#" 3개 -> 세번째로 큰 Font

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 3개 입력 후 제목 작성하기 → [Shipt+Enter] 실행

"#" 3개 -> 세번째로 큰 Font

In [ ]:



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 4개 입력 후 제목 작성하기

#### "#" 4개 -> 네번째로 큰 Font

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 4개 입력 후 제목 작성하기 → [Shipt+Enter] 실행

"#" 4개 -> 네번째로 큰 Font

In [ ]:

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 5개 입력 후 제목 작성하기

##### "#" 5개 -> 다섯번째로 큰 Font

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 5개 입력 후 제목 작성하기 [Shipt+Enter] 실행

"#" 5개 -> 다섯번째로 큰 Font

In [ ]:

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)
- ◆ 제목 넣는 방법 -> 박스(Box)에 "#을 6개 이상 입력 후 제목 작성하기

##### "#" 6개 -> 여섯개 이상 입력하면 변화 없음

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)  
제목 넣는 방법 -> 박스(Box)에 "#을 6개 입력 후 제목 작성하기 → [Shipt+Enter] 실행 - 변화없음

"#" 5개 -> 다섯번째로 큰 Font

"#" 6개 -> 여섯개 이상 입력하면 변화 없음

In [ ]: |

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor 작성 방법**
- ◆ 제목 넣기 : 작성하는 코드에 대한 "제목"을 넣을 수 있다.
- ◆ 제목은 "#"의 개수에 따라 Font의 Size가 달라짐 (#의 개수가 많을 수록 Font는 작아짐, 최대 5개)

**전체 Font Size 비교**

**"#" 1개 -> 가장 큰 Font**

**"#" 2개 -> 두번째로 큰 Font**

**"#" 3개 -> 세번째로 큰 Font**

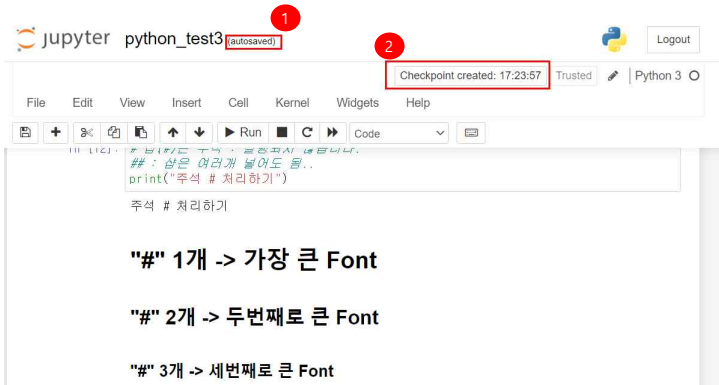
**"#" 4개 -> 네번째로 큰 Font**

**"#" 5개 -> 다섯번째로 큰 Font**

**"#" 6개 -> 여섯개 이상 입력하면 변화 없음**

## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 저장하기 : [Ctrl + S] → 저장은 수시로 하기

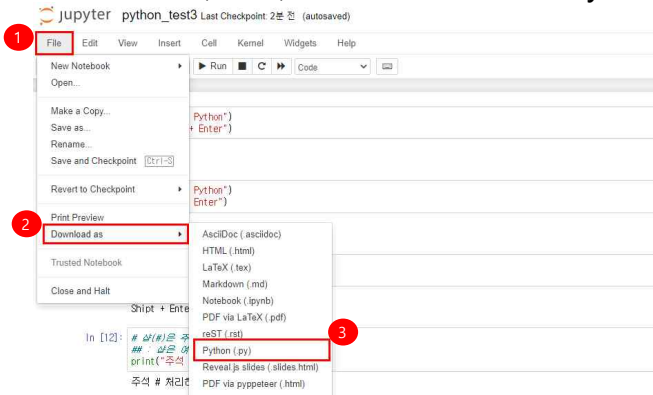




## ✓ Python 개발환경

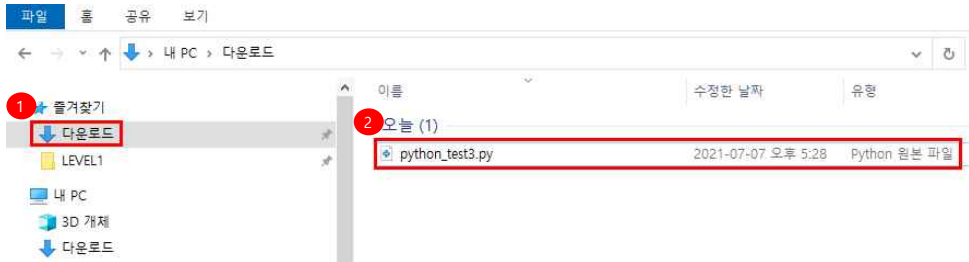
### ◆ Jupyter Notebook Tool 이해하기 Editor 작성 방법

### ◆ 다운로드 하기 : 원하는 파일 포맷(Format) 형식으로 다운로드 하기 → Python 파일 선택



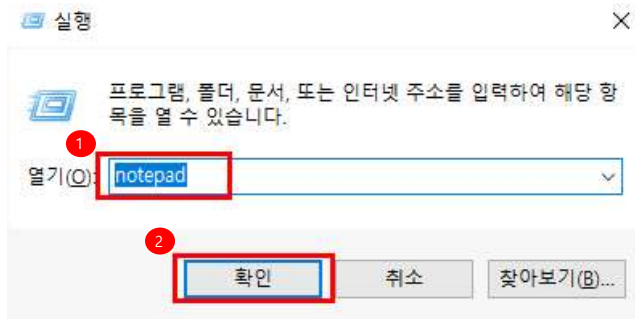
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 **Python** 파일 확인하기



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → Editor 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 Python 파일 내용 확인하기 → 편집기를 이용해서 Open해야 함
- ◆ "Notepad" 텍스트 편집기 Open하기 → 윈도우 키 [F]키] → "notepad" 입력 → [확인] 선택



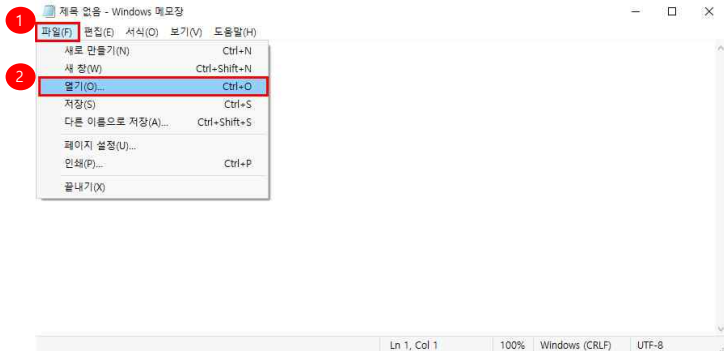
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 **Python** 파일 내용 확인하기 → 편집기를 이용해서 **Open**해야 함
- ◆ “Notepad” 텍스트 편집기 **Open**하기 → “Notepad” 최초 화면



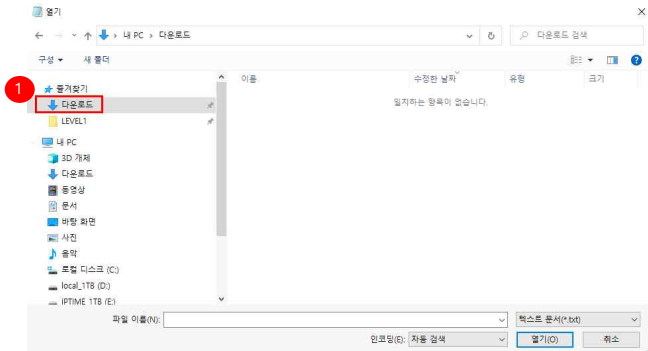
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → Editor 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 Python 파일 내용 확인하기 → 편집기를 이용해서 Open해야 함
- ◆ “Notepad” 텍스트 편집기에서 Python 파일 Open 하기 → [파일 > 열기] 선택



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → Editor 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 Python 파일 내용 확인하기 → 편집기를 이용해서 Open해야 함
- ◆ 다운받은 Python 파일이 위치한 폴더로 이동



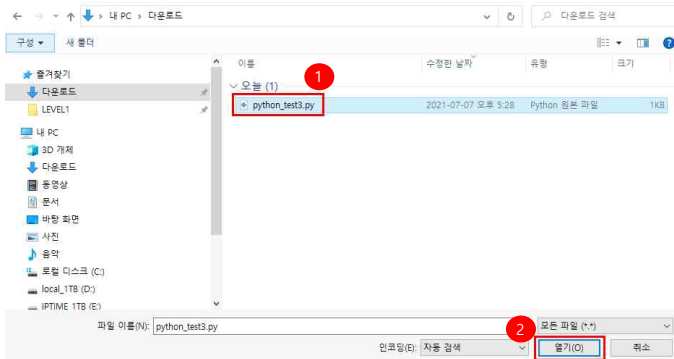
## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → Editor 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 Python 파일 내용 확인하기 → 편집기를 이용해서 Open해야 함
- ◆ 다운받은 Python 파일이 위치한 폴더로 이동 → 파일형식을 "모든 파일(\*)" 선택



## ✓ Python 개발환경

- ◆ Jupyter Notebook Tool 이해하기 → **Editor** 작성 방법
- ◆ 다운로드 하기 : 다운로드 받은 **Python** 파일 내용 확인하기 → 편집기를 이용해서 **Open**해야 함
- ◆ 다운받은 Python 파일 열기 → **".py"** 파일 선택 → [열기] 선택





## ✓ Python 개발환경

### ◆ Jupyter Notebook Tool 이해하기 → Editor 작성 방법

### ◆ 다운로드 하기 : 다운로드 받은 Python 파일 내용 확인하기      편집기를 이용해서 Open해야 함

### ◆ 다운받은 Python 파일 열기      파이썬(Python) 코드 보기

python\_test3.py - Windows 메모장  
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)  
#!/usr/bin/env python  
# coding: utf-8

# In[9]:

```
print("Hello Python")  
print("Ctrl + Enter")
```

# In[11]:

```
print("Hello Python")  
print("Alt + Enter")
```

# In[8]:

```
print("Hello Python")  
print("Shipt + Enter")
```

# In[12]:

```
# 샵(#)은 주석 : 실행되지 않습니다.  
## : 샵은 여러개 넣어도 됨..  
print("주석 # 처리하기")
```

# # "#" 1개 -> 가장 큰 Font

# ## "#" 2개 -> 두번째로 큰 Font

# ### "#" 3개 -> 세번째로 큰 Font

# #### "#" 4개 -> 네번째로 큰 Font

# ##### "#" 5개 -> 다섯번째로 큰 Font

# ##### "#" 6개 -> 여섯개 이상 입력하면 변화 없음

# In[ ]: