

Deep Learning Programming

Lecture 1.2: Pytorch Introduction

Sangryul Jeon

School of Computer Science and Engineering

srjeonn@pusan.ac.kr

PART 1: Colab

Slides borrowed from Naver Connect Foundation

파이썬의 기본 실행환경

Python Shell

코드 편집도구

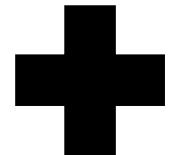
```
C:\Users\sc82choi\Downloads\cmdr_mini
(base) λ python
Python 3.7.4 (default, Aug  9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc
. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> 10 + 10
20
>>>
```



1.2_Python

한번 합쳐보자!

Python Shell



코드 편집도구



데이터 분석을 위한 파이썬 IDE

- IPython(<http://ipython.org>) 커널을 기반으로 한 대화형 파이썬 셀
- 일반적인 터미널 셀 + 웹 기반 데이터 분석 Notebook 제공
- 미디어, 텍스트, 코드, 수식 등을 하나의 문서로 표현 가능
- 사실상의 데이터 분석 Interactive Shell의 표준
- Julia + Python + R

- jupyter 설치

```
conda install jupyter
```

- jupyter 실행

```
jupyter notebook
```

- jupyter 설치

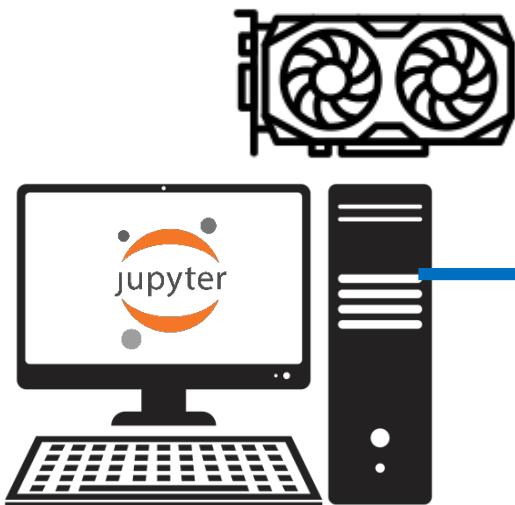
conda install jupyter

- jupyter 실행

jupyter notebook

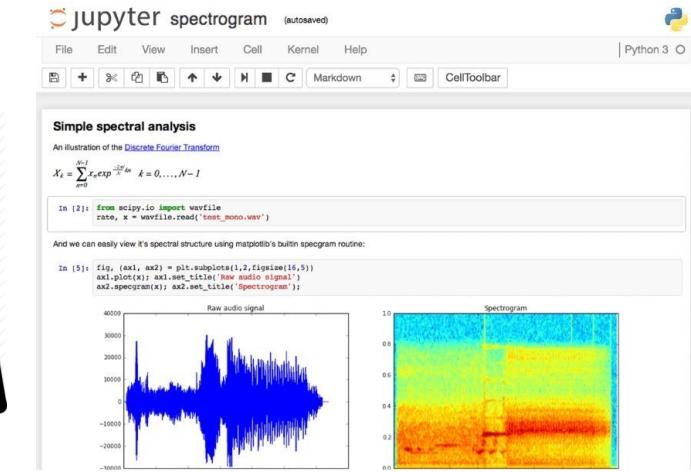
- <http://localhost:8888/tree> 주소로 jupyter 가 실행됨

Running Jupyter on local desktop



123.45.67.89

123.45.67.89:8888



- 터미널 창의 token 정보를 사용

```
[I 21:35:00.179 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice
to skip confirmation).
[C 21:35:00.180 NotebookApp]
Copy/paste this URL into your browser when you connect for the first time,
to login with a token:
http://localhost:8888/?token=7c8e9d6a158470b699bc77c989efa7d3838d6ecd7323552a
[I 21:35:00.342 NotebookApp] Accepting one-time-token-authenticated connection from ::1
[W 21:35:01.471 NotebookApp] 404 GET /static/components/moment/locale/ko.js?v=20171216213459 (::1) 6.93ms referer=http://localhost:8888/tree
pygmentize -f rtf FILE | pbcopy
```

[Logout](#)[Files](#)[Running](#)[Clusters](#)

Select items to perform actions on them.

[Upload](#)[New ▾](#)

0

[Name ▾](#)[Last Modified](#)

The notebook list is empty.

[Upload](#)[New ▾](#)

Notebook:

Python 3

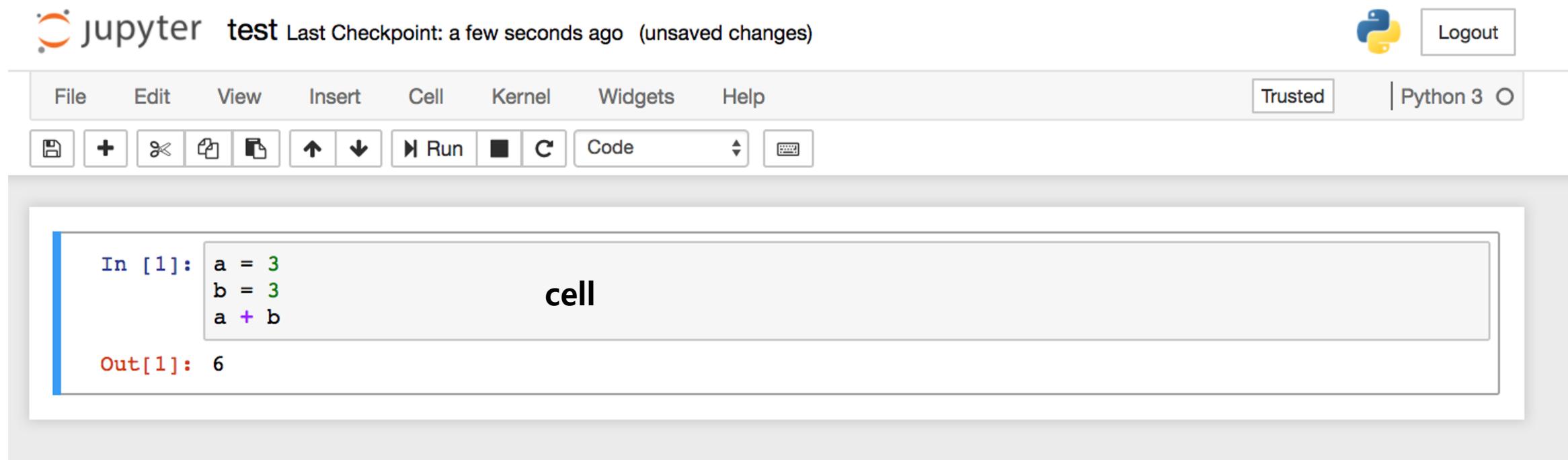
Other:

Text File

Folder

Terminal

- cell 단위로 실행 & 실행 시점에 해당 코드가 memory에 올라감
- 실행 명령어 ctrl + enter , shift + enter, alt + enter





단축키 존재



jupyter-notebook command group

change cell to code	(command mode)	[y]
change cell to heading 1	(command mode)	[1]
change cell to heading 2	(command mode)	[2]
change cell to heading 3	(command mode)	[3]
change cell to heading 4	(command mode)	[4]
change cell to heading 5	(command mode)	[5]
change cell to heading 6	(command mode)	[6]
change cell to markdown	(command mode)	[M]
change cell to raw	(command mode)	[R]
clear all cells output		

- 툴팁 표시하기 : Shift + Tab
- 들여쓰기 사용하기 : ctrl +] or ctrl + [
- 셀 나누기 : ctrl + shift + -

- 아래 셀이랑 합치기 : shift + M
- 셀 오려두기 : x 셀 copy : c 셀 붙여넣기 : v or shift + v
- 셀 지우기 : d, d' 셀 지우기 취소 : z
- Markdown 변환 : m,m' Code로 변환 : y, y'

- 구글이 개발 클라우드 기반의 jupyter notebook
- 구글 드라이브 + GCP + jupyter 등이 합쳐져서 사용자가 손쉽게 접근
- 초반 여러가지 모듈 설치의 장점을 가짐
- 구글 드라이브의 파일을 업로드하여 사용가능한 장점 가짐
- VSCode 등과 연결해서 사용가능
- V100 이상의 GPU를 무료로 쓸 수 있다는 장점을 가짐

- Colab pro을 사용할 경우 안정적인 colab의 활용이 가능해짐
- colab과 jupyter는 비슷한 듯 다른 단축키를 가짐

Actions	Colab	Jupyter
show keyboard shortcuts	Ctrl/Cmd M H	H
Insert code cell above	Ctrl/Cmd M A	A
Insert code cell below	Ctrl/Cmd M B	B
Delete cell/selection	Ctrl/Cmd M D	DD
Interrupt execution	Ctrl/Cmd M I	
Convert to code cell	Ctrl/Cmd M Y	Y
Convert to text cell	Ctrl/Cmd M M	M
Split at cursor	Ctrl/Cmd M -	Ctrl Shift -

 Colab Pro

Colab 최대한 활용하기

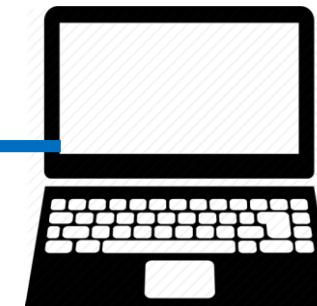
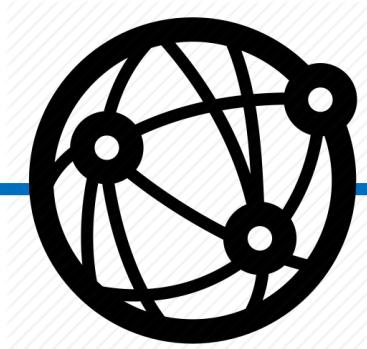
지금 업그레이드

\$9.99/월

반복 결제 · 언제든지 취소 가능

Google Colab

- Google Colab = **Google drive + Jupyter notebook**
- A virtual machine having GPUs is provided by Google.
 - The Google server must be connected in order to interpret your source code.
- Program codes can be accessed with a link like other documents in Google drive.
 - Easy to share codes

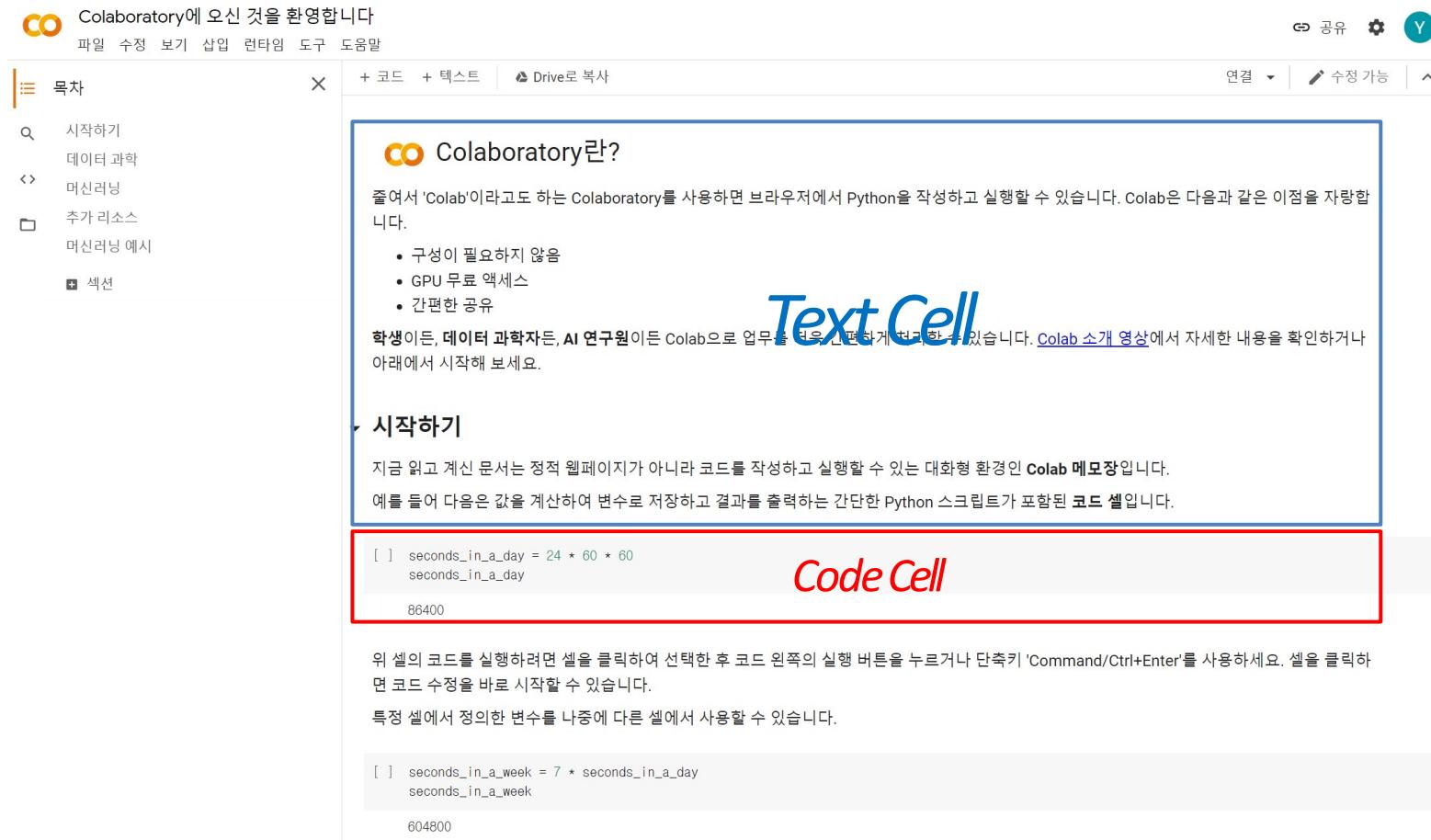


A screenshot of the Google Colab interface. On the right, a Jupyter notebook is open with several code cells. One cell contains Python code for mounting Google Drive. On the left, a file browser shows a directory structure under 'My Drive' with files like '07_structured_data.ipynb', 'BigQuery recipes', and 'Copy of nima colab.ipynb'. The interface includes standard Jupyter notebook navigation buttons for CODE, TEXT, CELL, and FILE.

Google server

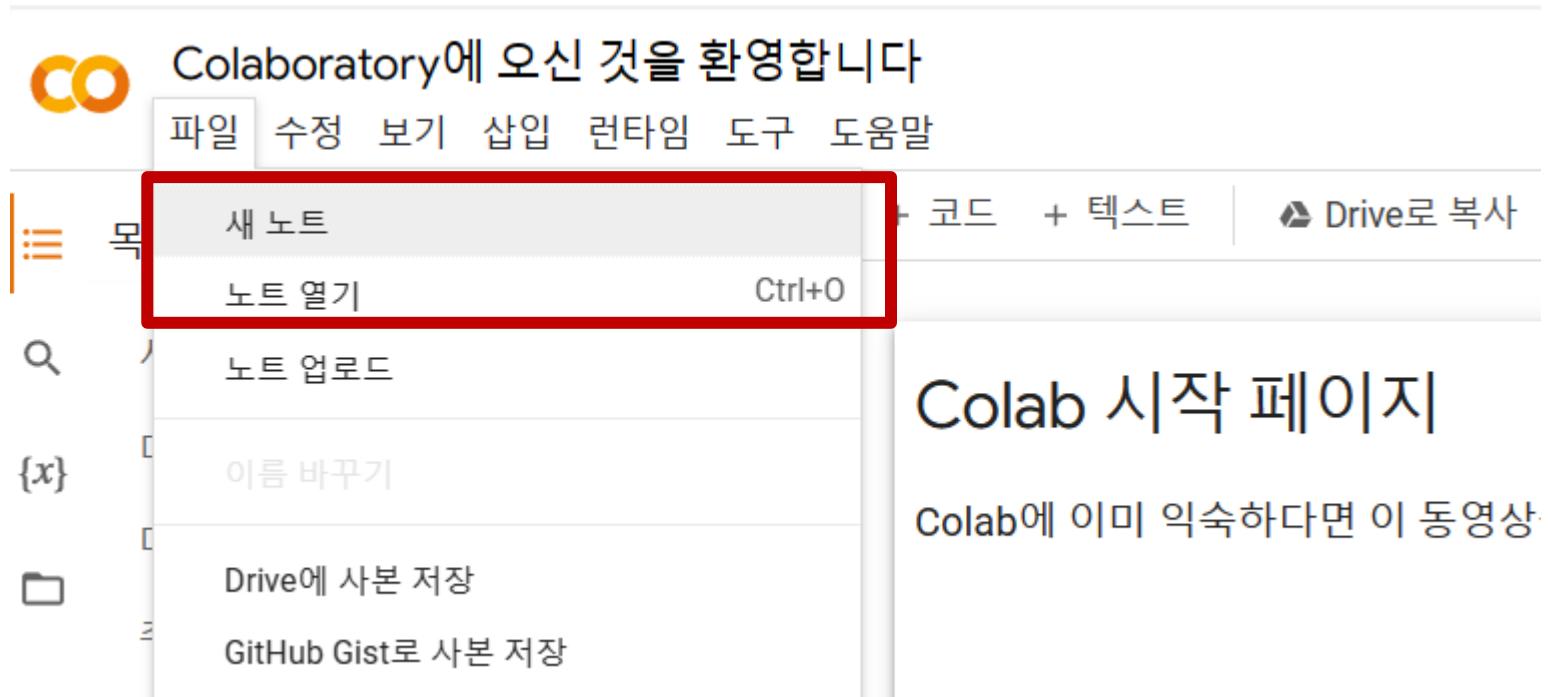
How to use Colab (1/7)

- Access and log in the site <https://colab.research.google.com> with the Google ID

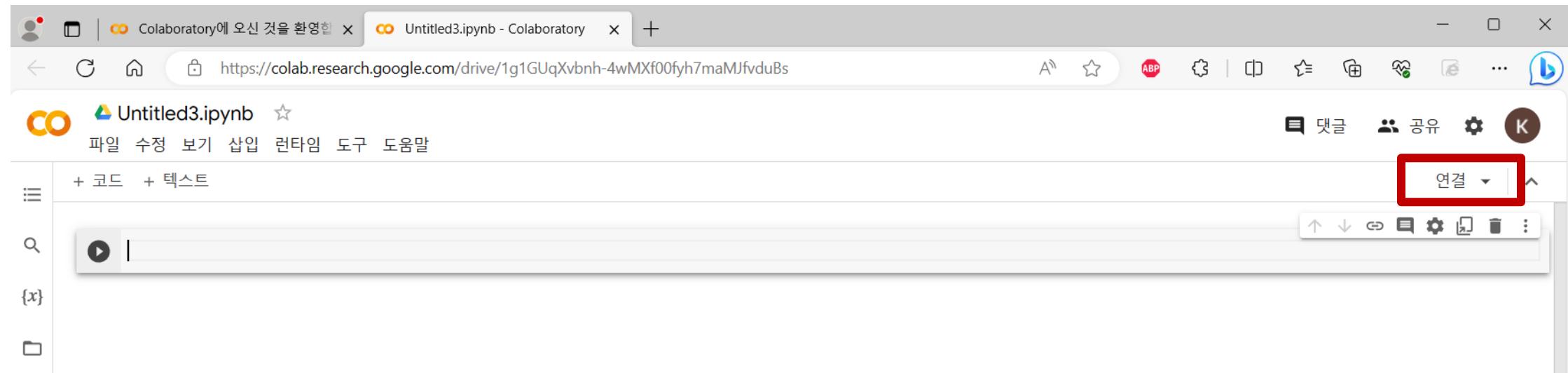


How to use Colab (2/7)

- File → New Note or Note Open



How to use Colab (3/7)



How to use Colab (4/7)

- Colab interface



- File extension: .ipynb

- You can download your codes on your PC as a .py file using “File → .py Download”.

How to use Colab (5/7)

- Code cell
 - Usual Python code
 - Each cell is the unit to be run at a time
 - Execution result of each cell is stored in memory and affects the following cells.
- The memory is reset if the run-time restarts.

Run Order

```
[1] # Code Cell!
a = 1
b = 2
print(a+b)

# Ctrl+Enter로 해당 코드 셀 실행
```

Run (Ctrl + Enter)

```
3
▶ # 각 셀은 한번에 실행할 단위를 뜻함
# 실행 이후에도 메모리는 그대로 유지되어 다른 셀의 실행에 영향을 줌
a += 3
b -= 1
print(a+b)
```

Result

```
5
```



Stop of Run
Restart Runtime

How to use Colab (6/7)

- Text cell
 - Markdown syntax format
 - Automatic creation of table-of-contents

The screenshot shows the Google Colab interface. On the left, there is a toolbar with various icons for text styling (bold, italic, etc.). Below the toolbar, a code cell contains the following text:

```
# Text Cell!
* 마크다운 형식의 텍스트 셀입니다
* docstring 등 여러 줄 주석을 효과적으로 시각화 할 수 있습니다
* 마크다운 문법을 숙지하시고 사용하시면 좋습니다
* https://heropy.blog/2017/09/30/markdown/
```

To the right of the code cell, a text cell displays the same content with a dashed border separating it from the code cell. The text in the text cell is bolded.

The screenshot shows the Google Colab interface with a sidebar on the left. The sidebar has tabs for "목차" (Table of Contents), "코드 스니펫" (Code Snippets), and "파일". The "목차" tab is highlighted with a red border. A yellow box highlights the "목차 생성" (Table of Contents Generation) button. The main content area shows a text cell with the following content:

▼ Text Cell!

- 마크다운 형식의 텍스트 셀입니다
- docstring 등 여러 줄 주석을 효과적으로 시각화 할 수 있습니다
- 마크다운 문법을 숙지하시고 사용하시면 좋습니다
- <https://heropy.blog/2017/09/30/markdown/>

At the bottom of the main content area, there is a code cell with the following content:

```
# Code Cell!
a = 1
b = 2
```

How to use Colab (7/7)

• Tools → Shortcut Keys

• Ctrl+Enter

- Run the current code cell

• Alt+Enter

- After running the current code cell and insert a new code cell

• Ctrl+F8

- Run all code cells above

• Ctrl+F9

- Run all code cells

• Ctrl+F10

- Run all code cells below

키보드 환경설정

편집기 키 바인딩
default

Enter 키를 눌러 제안 수락

단축키

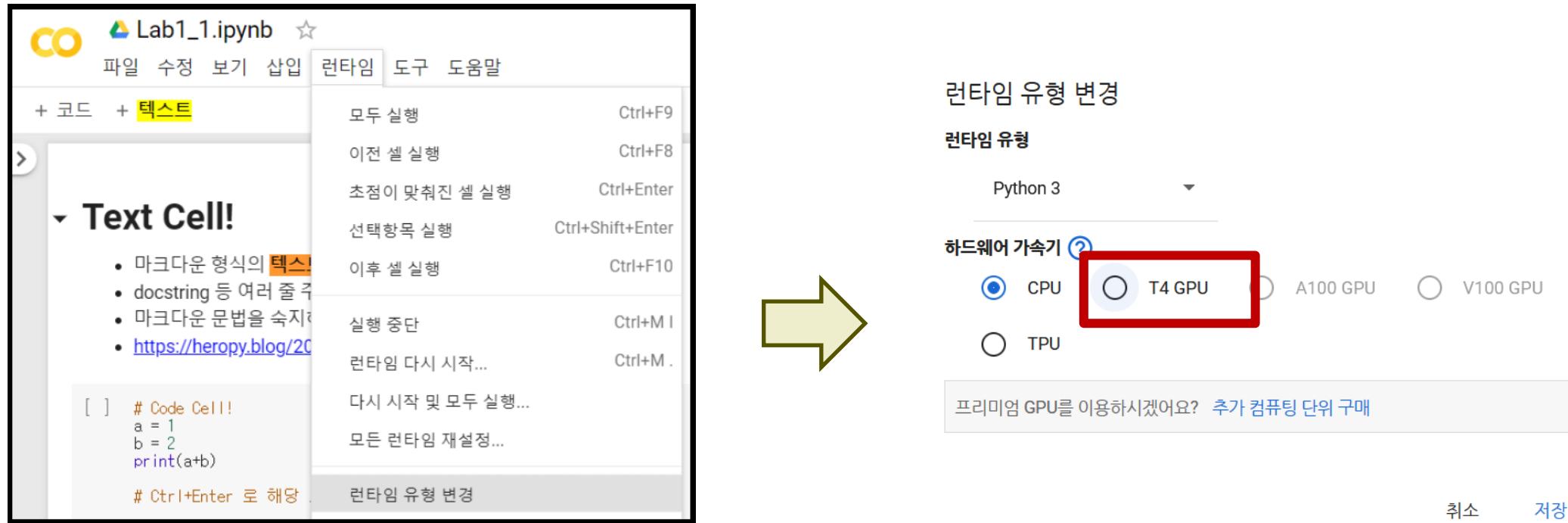
단축키를 추가하거나 변경하려면 키 조합을 클릭한 다음 새 키를 입력하세요. Ctrl+M을(를) 다중 키-이벤트 단축키의 접두어로 사용할 수 있습니다.

단축키 설정 .ipynb 다운로드	단축키 설정 설정 열기
단축키 설정 .py 다운로드	단축키 설정 세로형 탭 레이아웃 보기
단축키 설정 GitHub Gist로 사본 저장	단축키 설정 세션 관리
단축키 설정 GitHub에 사본 저장	단축키 설정 섹션 제목 셀 추가
단축키 설정 Google Drive의 노트 별표표시/별 표표시 해제	단축키 설정 셀 또는 선택 항목 복사
단축키 설정 Unmount ColabFS	단축키 설정 셀 또는 선택 항목 잘라내기
단축키 설정 가로형 탭 레이아웃 보기	Ctrl+Click 셀 선택 토큰
단축키 설정 노트 설정 열기	Ctrl+Shift+Y 셀 작업 다시 실행
단축키 설정 노트 업로드	Ctrl+M D 셀/선택항목 삭제
Ctrl+O 노트 열기	Shift+Enter 셀을 실행하고 다음 셀 선택
Ctrl+P 노트 인쇄	Alt+Enter 셀을 실행하고 새 셀 삽입
Ctrl+S 노트 저장	단축키 설정 스크래치 셀로 복사
단축키 설정 노트 출처 보기	Ctrl+Alt+N 스크래치 코드 셀 열기
단축키 설정 노트 파일 정보 표시	단축키 설정 실습 모드에서 열기
단축키 설정 노트를 휴지통으로 이동	Ctrl+M I 실행 중단
단축키 설정 노트북 공유	단축키 설정 실행된 코드 기록 표시
Ctrl+F9 노트의 모든 셀 실행	Ctrl+M B 아래에 코드 셀 삽입
Ctrl+M N 다음 셀	Ctrl+M F 양식 보기 순환
Shift+Down 다음 셀을 포함하도록 선택 영역을 확장합니다.	단축키 설정 양식 입력란 추가
	단축키 설정 양식 추가
	Ctrl+M A 위에 코드 셀 삽입

Use GPU (1/2)

● Enabling and testing the GPU

- Navigate to Edit→Notebook Settings
- select GPU from the Hardware Accelerator drop-down



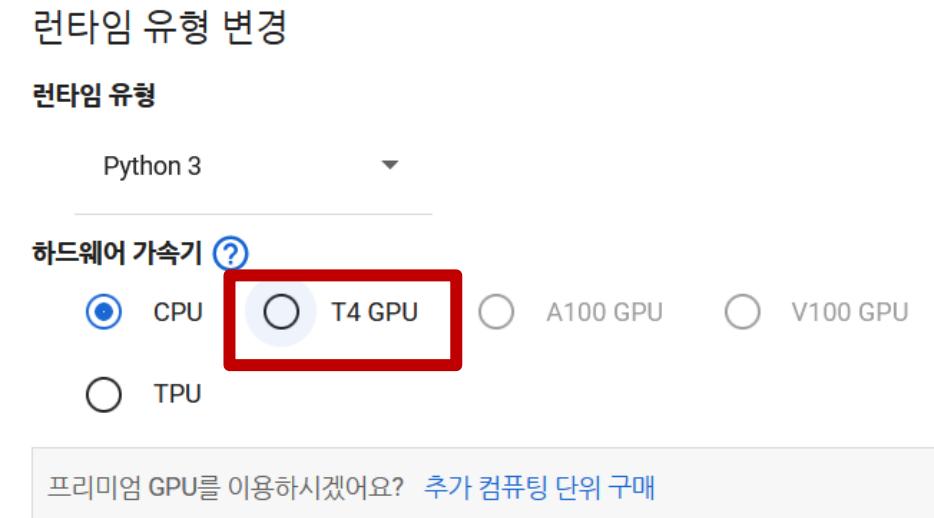
Use GPU (2/2)

- Click and Execute

- <https://colab.research.google.com/notebooks/gpu.ipynb>

```
[ ] import tensorflow as tf  
device_name = tf.test.gpu_device_name()  
if device_name != '/device:GPU:0':  
    raise SystemError('GPU device not found')  
print('Found GPU at: {}'.format(device_name))
```

TensorFlow 2.x selected.
Found GPU at: /device:GPU:0



취소 저장

Connecting Google Drive to Colab (1/3)

- Step 1. Mounting our Google Drive using the following code

```
from google.colab import drive  
drive.mount('/content/gdrive/', force_remount=True)
```

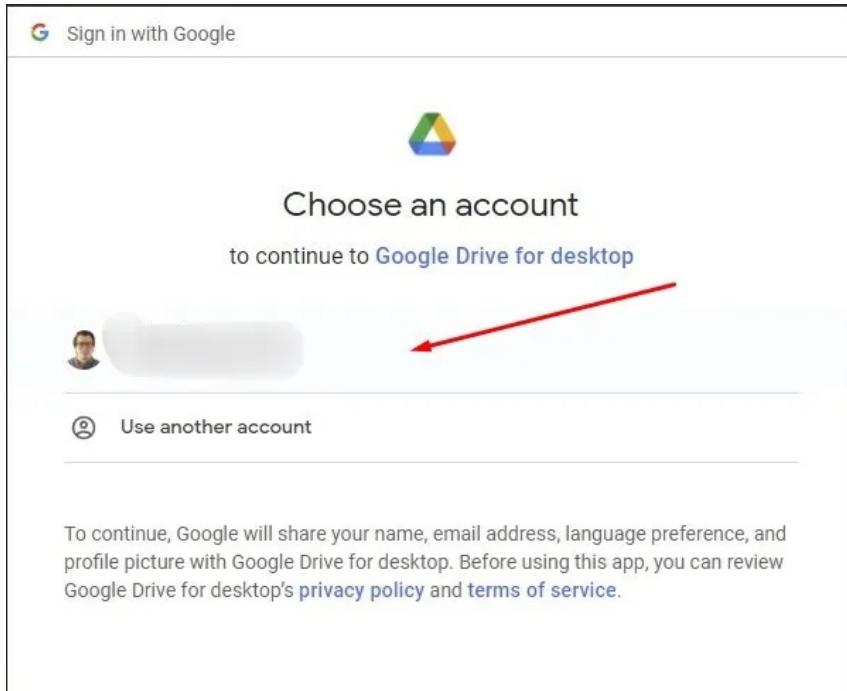
- Step2. Pop up

- Click “Connect Google Drive”

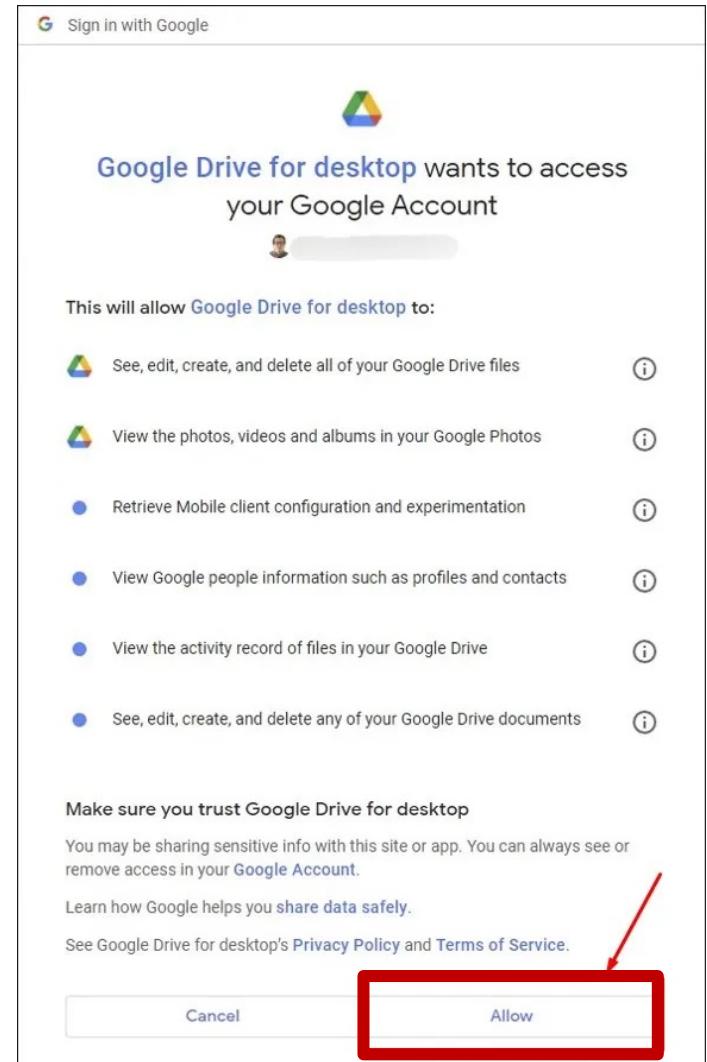


Connecting Google Drive to Colab (2/3)

- Step 3: Choose your account



- Step 4: Click Allow



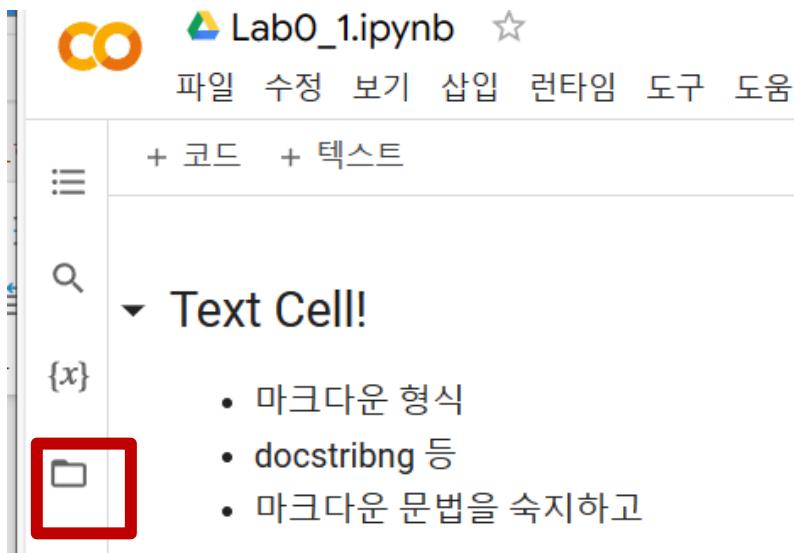
Connecting Google Drive to Colab (3/3)

● Step5. Mount Completed

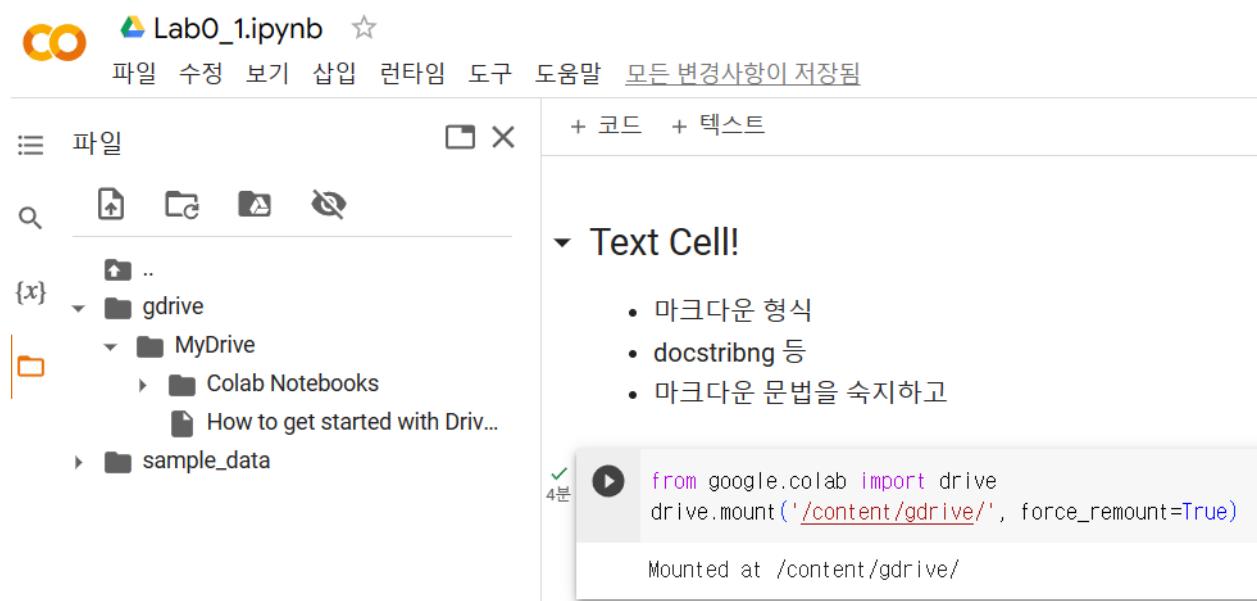
```
✓ 4분
▶ from google.colab import drive
drive.mount('/content/gdrive/', force_remount=True)

Mounted at /content/gdrive/
```

● Step 6. Click the folder icon



● Step7. Check the folder



From Google Drive (1/2)

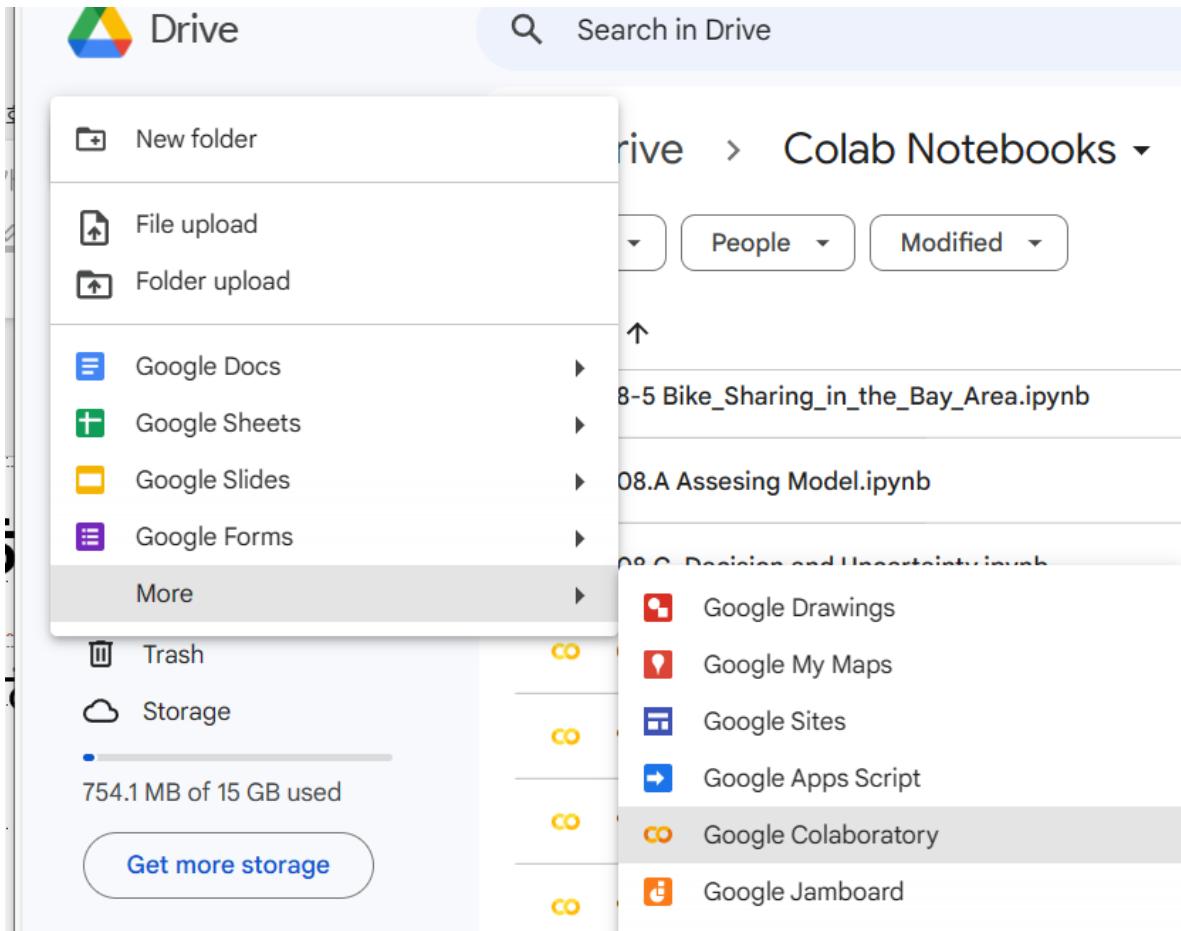
- Check Google Drive

The screenshot shows the Google Drive web interface. On the left, there's a sidebar with navigation links: '+ New', 'My Drive' (selected), 'Computers', 'Shared with me', 'Recent', 'Starred', 'Spam', 'Trash', 'Storage' (754.1 MB of 15 GB used), and a 'Get more storage' button. The main area is titled 'My Drive > Colab Notebooks'. It features a search bar and filters for 'Type', 'People', and 'Modified'. A table lists eight Jupyter Notebook files:

Name	Owner	Last modified	File size
03.B.Arrays	me	Mar 27, 2023	11 KB
04.A.IMDB.ipynb	me	Mar 27, 2023	47 KB
04.A.Pandas.ipynb	me	Mar 27, 2023	50 KB
04.A.Tables.ipynb	me	Mar 24, 2023	95 KB
04.B_pandas_example.ipynb	me	Mar 29, 2023	340 KB
05. C. FDS07Visualization_Pandas.ipynb	me	Apr 10, 2023	684 KB
05.B. Matplotlib_Seaborn.ipynb	me	Apr 5, 2023	1.2 MB
05.D. FDS08.4 Bike Sharing.ipynb	me	Sep 2, 2023	3.4 MB

From Google Drive (2/2)

- Click “+New” → Click more
 - We can select Google Colaboratory



Conclusion

- Issues or Tips
 - The 'maximum lifetime' of a running notebook is 12 hours (browser open)
 - An 'Idle' notebook instance cuts-off after 90 minutes
 - You can have a maximum of 2 notebooks running concurrently
 - If you close the notebook window and open it while the instance is still running, the cell outputs and variables will still persist. However if the notebook instance has been recycled, your cell outputs and variables will no longer be available.

Thank you

Prof. Jeon, Sangryul

Computer Vision Lab.

Pusan National University, Korea

Tel: +82-51-510-2423

Web: <http://sr-jeon.github.io/>

E-mail: srjeonn@pusan.ac.kr