Colab Tutorial

TA: 孟妍 (Yen Meng)
2022.02.18
ntu-ml-2022spring-ta@googlegroups.com

Link of this Tutorial

https://reurl.cc/Epg3M0

(Things mentioned in this tutorial will be covered in this notebook)

Outline

- Introduction
- Getting Started
- Changing Runtime
- Executing Code Block
- Check GPU type
- File Manipulation
- Mounting Google Drive
- Saving Notebook
- Useful Linux Commands
- Problems You May Encounter... (very important)
- References

Introduction

What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

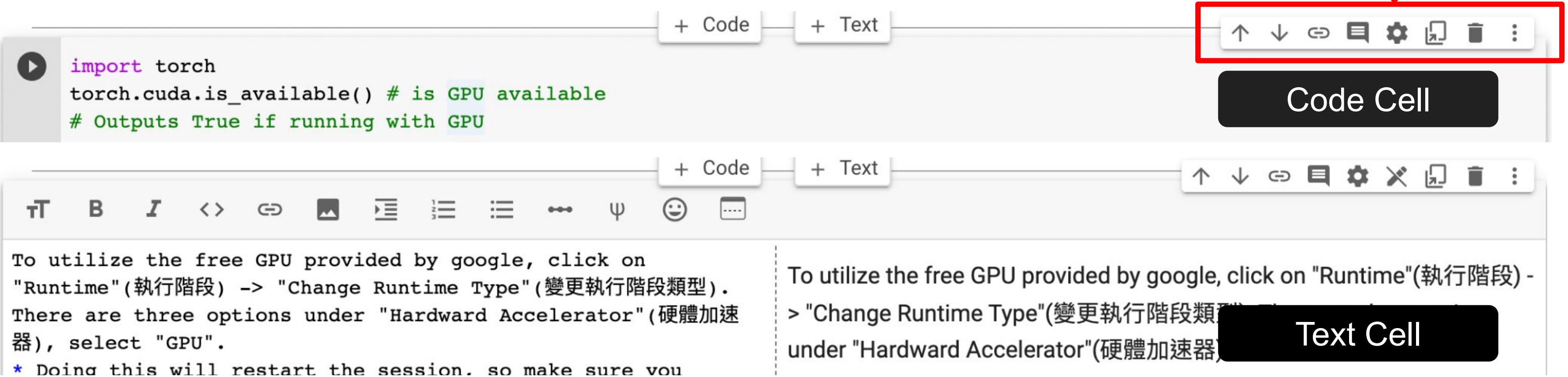
- Zero configuration required
- Free access to GPUs
- Easy sharing

Getting Started

Creating a new cell

You can create a new code cell by clicking on + Code , clicking on + Text generates a text cell

There are options for moving your cell up/down or copy or delete it



```
import torch
torch.cuda.is_available() # is GPU available
# Outputs True if running with GPU
```

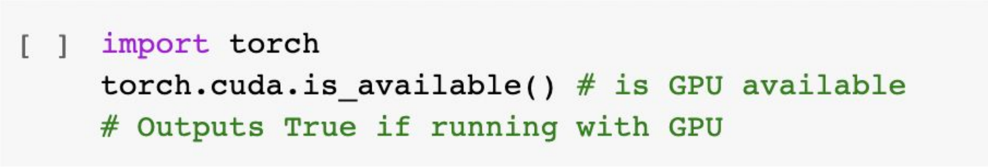
```
To utilize the free GPU provided by google, click on
"Runtime"(執行階段) -> "Change Runtime Type"(變更執行階段類型).
There are three options under "Hardward Accelerator"(硬體加速
器), select "GPU".
* Doing this will restart the session, so make sure you
```

To utilize the free GPU provided by google, click on "Runtime"(執行階段) -> "Change Runtime Type"(變更執行階段類

under "Hardward Accelerator"(硬體加速器

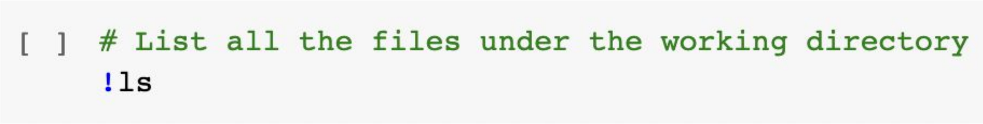
Getting Started

You can type python code in the code cell, or use a leading exclamation mark ! to change the code cell to treating the input as a shell script



```
[ ]  import torch
     torch.cuda.is_available() # is GPU available
     # Outputs True if running with GPU
```

→ **python**



```
[ ]  # List all the files under the working directory
     !ls
```

→ **shell script**

Getting Started

Using an exclamation mark (!) starts a new shell, does the operations, and then kills that shell, while percentage (%) affects the process associated with the notebook, and it is called a magic command.

Use % instead of ! for cd (change directory) command

other magic commands are listed here

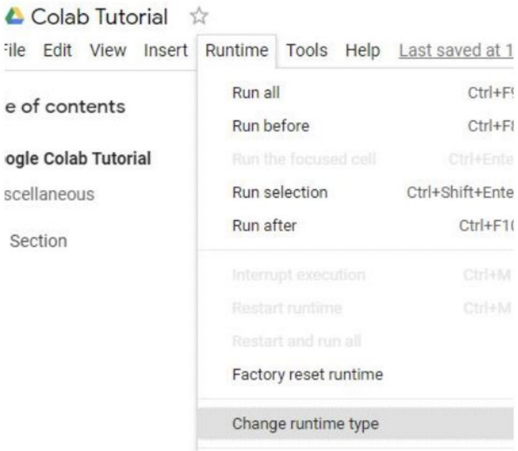
Changing Runtime

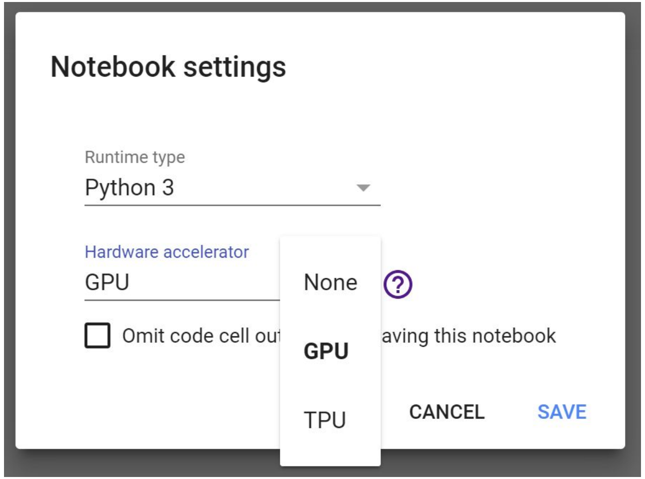
To utilize the free GPU provided by google,

click on "Runtime"(執行階段) → "Change Runtime Type"(變更執行階段類型).

select "**GPU**" for "Hardware Accelerator"(硬體加速器)

Doing this will restart the session, so make sure you change to the desired runtime before executing any code.




Notebook settings

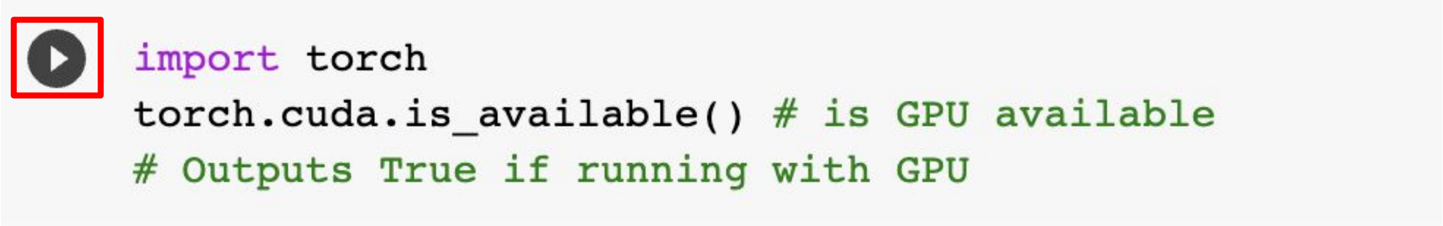
Runtime type
Python 3

Hardware accelerator
GPU

□ Omit code cell out…aving this notebook

Executing Code Block

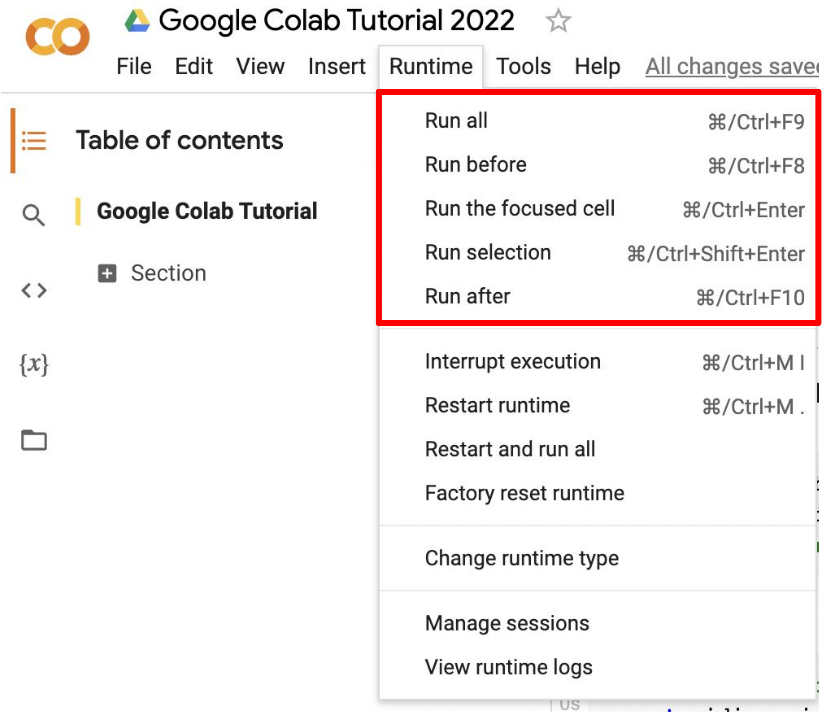
Click on the play button to execute a specific code cell



```
import torch
torch.cuda.is_available() # is GPU available
# Outputs True if running with GPU
```

Executing Code Block

Other options to run your code



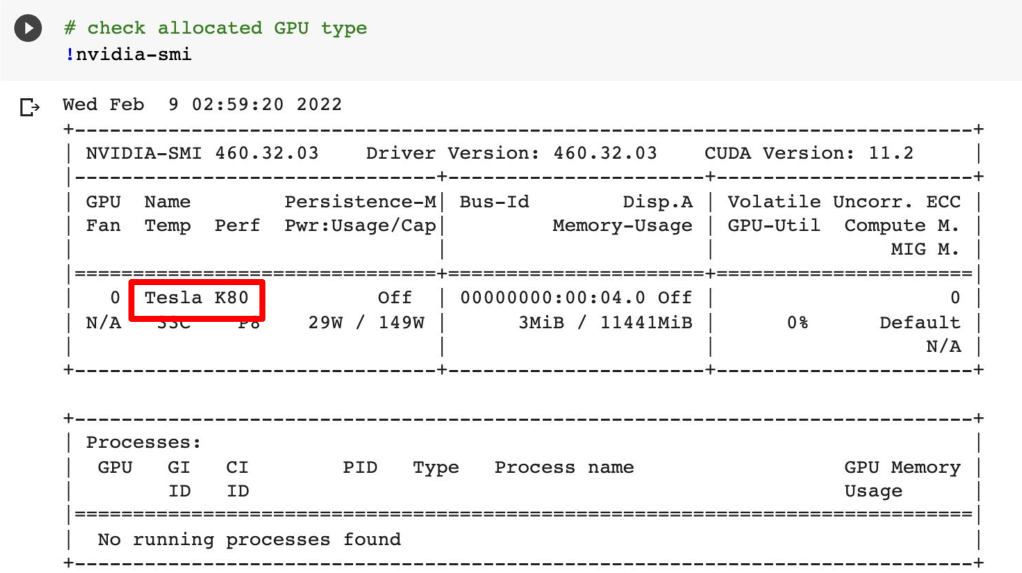
Check GPU Type

Use the command **nvidia-smi** to check the allocated GPU type

Available GPUs:

P100 > T4 > K80

(but most of the time you get K80 using the free Colab)



```
# check allocated GPU type
!nvidia-smi
```

```
Wed Feb  9 02:59:20 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.32.03    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla K80           Off  | 00000000:00:04.0 Off |                    0 |
| N/A   33C    P8    29W / 149W |      3MiB / 11441MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

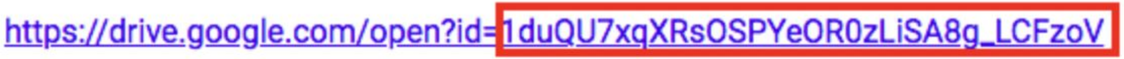
File Manipulation

Download files via Google Drive

A file stored in Google Drive has the following sharing link：

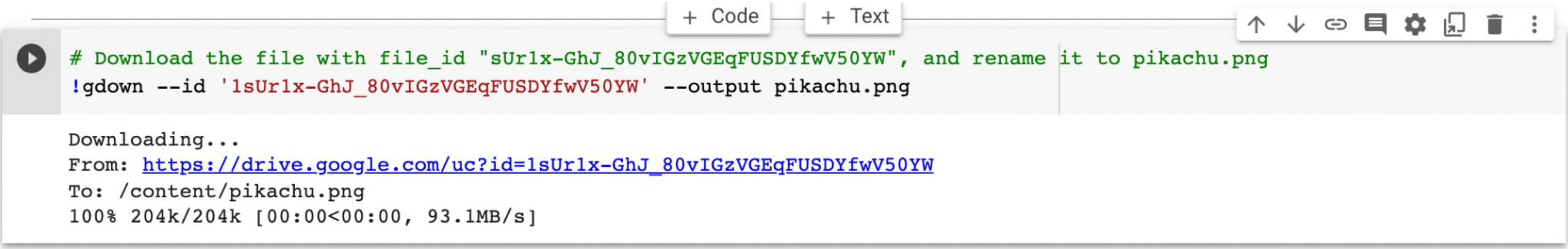
https://drive.google.com/open?id=1duQU7xqXRsOSPYeOR0zLiSA8g_LCFzoV

The random string after "open?id=" is the **file_id**



https://drive.google.com/open?id=1duQU7xqXRsOSPYeOR0zLiSA8g_LCFzoV

It is possible to download the file via Colab knowing the **file_id**, using the following command.



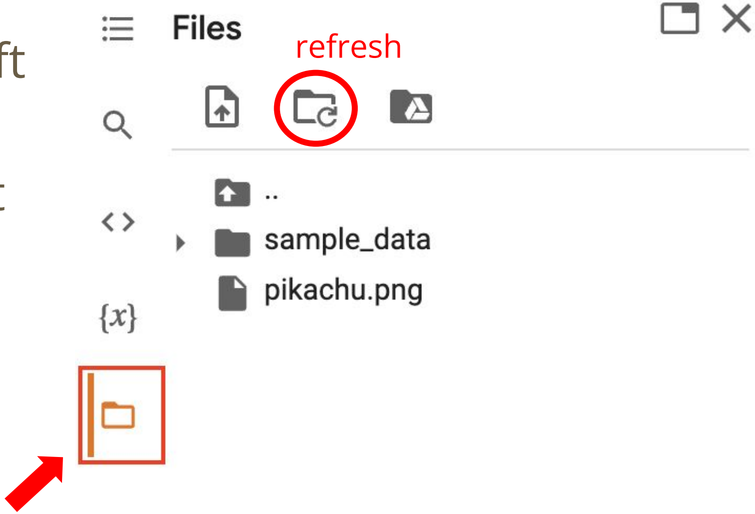
```
# Download the file with file_id "sUr1x-GhJ_80vIGzVGEqFUSDYfwV50YW", and rename it to pikachu.png
!gdown --id '1sUr1x-GhJ_80vIGzVGEqFUSDYfwV50YW' --output pikachu.png
```

```
Downloading...
From: https://drive.google.com/uc?id=1sUr1x-GhJ_80vIGzVGEqFUSDYfwV50YW
To: /content/pikachu.png
100% 204k/204k [00:00<00:00, 93.1MB/s]
```

File Manipulation

File Structure

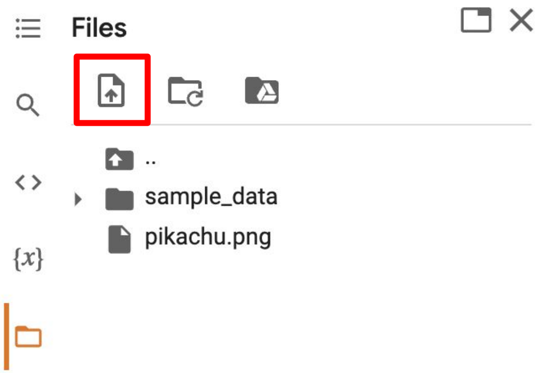
- You may click on the folder icon on the left to view your current files
- After downloading files, if the files are not immediately shown, click the refresh button
- Files are temporarily stored, and will be removed once you end your session.



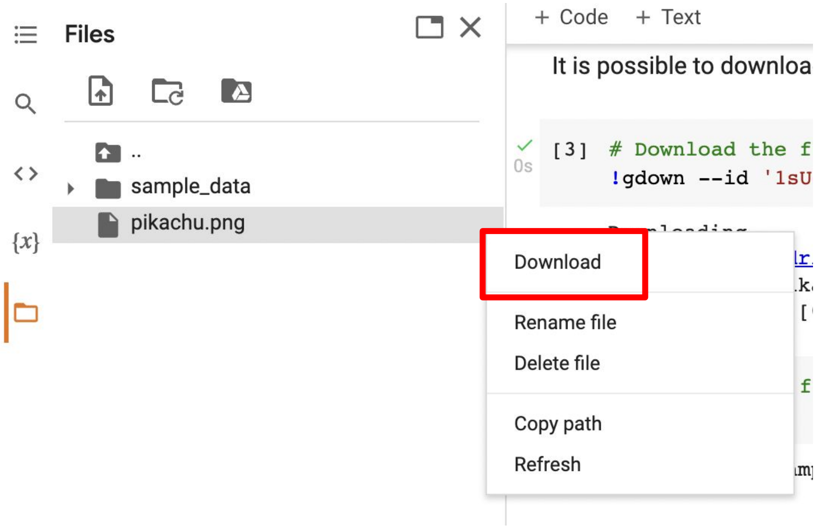
File Manipulation

Upload and Download Files

Click the upload icon to upload local files to your session



click ⋮ to download files to your local



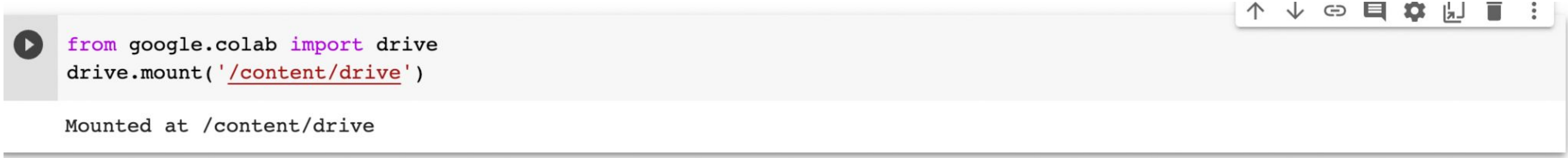
It is possible to downloa

```
[3] # Download the f
    !gdown --id '1sU
```

Mounting Google Drive

If you don’t want to download the data every time you start a new session, or you want some files to be saved permantly,

you can mount your own google drive to colab and directly download/save the data to your google drive.

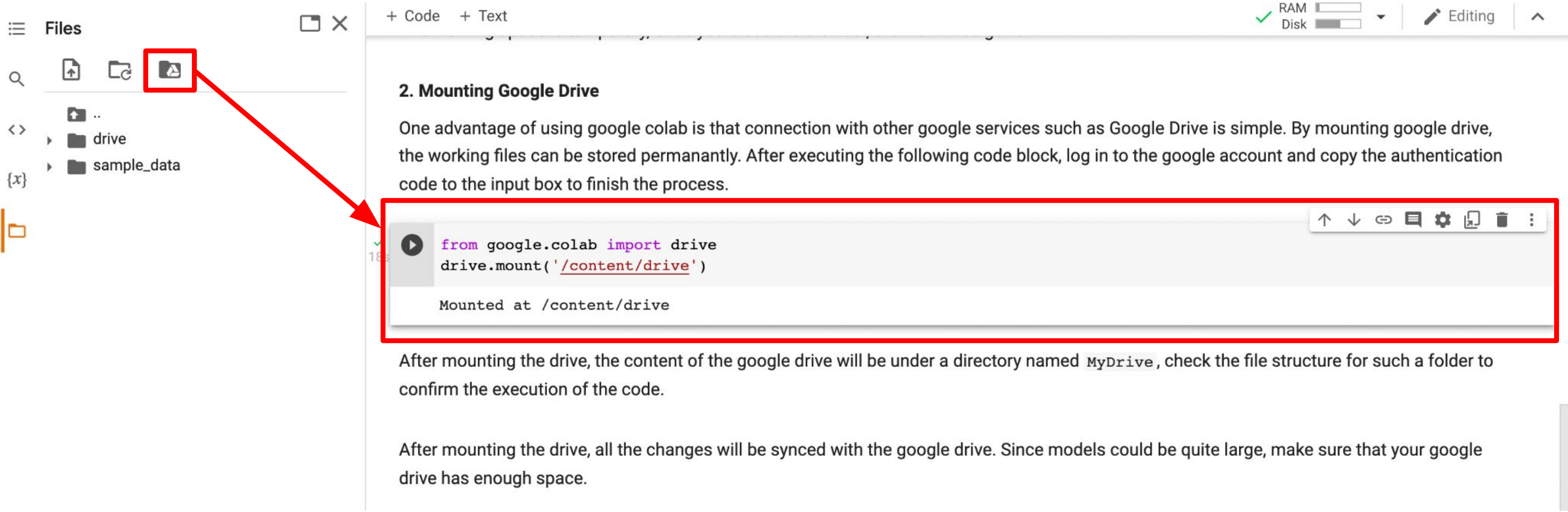


```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

Mounting Google Drive

Click on the Google Drive icon, the **Mount Drive** code block will be generated



2. Mounting Google Drive

One advantage of using google colab is that connection with other google services such as Google Drive is simple. By mounting google drive, the working files can be stored permanantly. After executing the following code block, log in to the google account and copy the authentication code to the input box to finish the process.

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

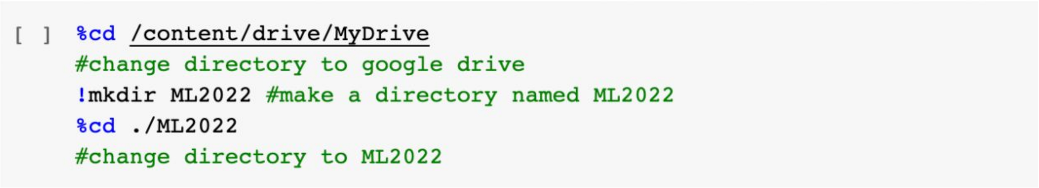
After mounting the drive, the content of the google drive will be under a directory named `MyDrive`, check the file structure for such a folder to confirm the execution of the code.

After mounting the drive, all the changes will be synced with the google drive. Since models could be quite large, make sure that your google drive has enough space.

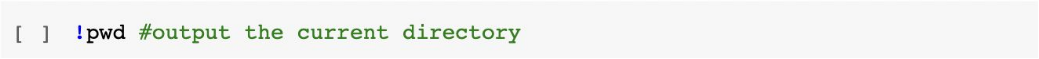
Mounting Google Drive

Execute the following three code blocks in order

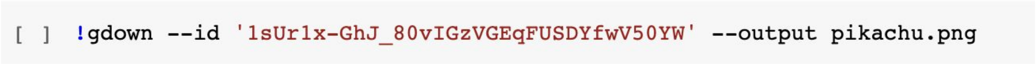
This will download the image to your google drive, and you can access it later



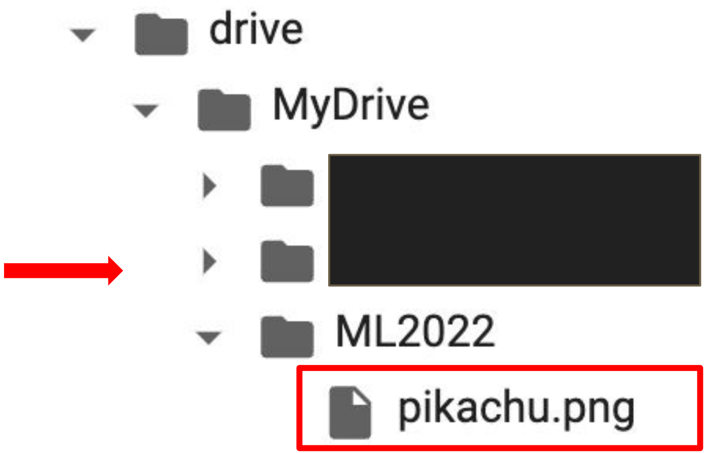
```
[ ]  %cd /content/drive/MyDrive
     #change directory to google drive
     !mkdir ML2022 #make a directory named ML2022
     %cd ./ML2022
     #change directory to ML2022
```



```
[ ]  !pwd #output the current directory
```

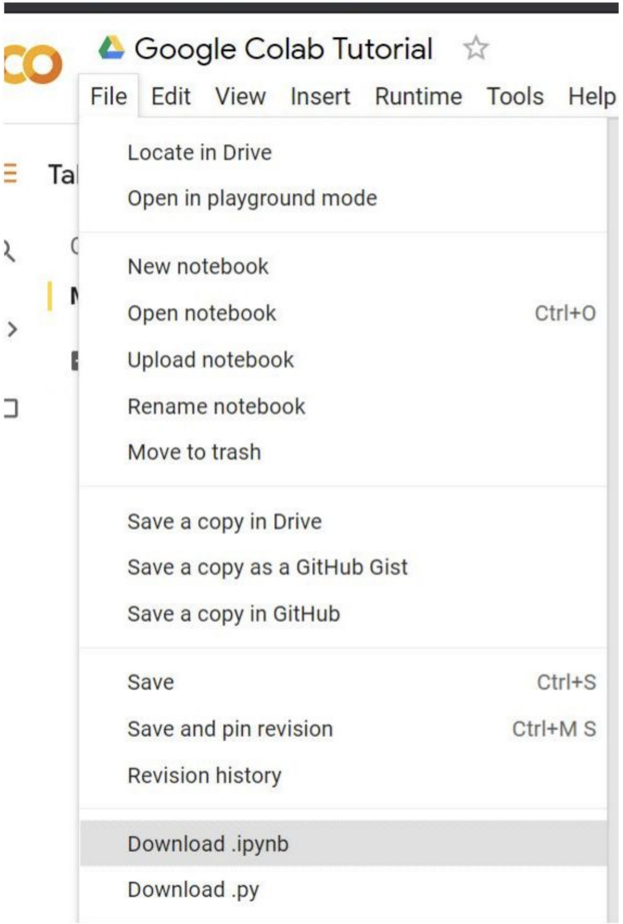


```
[ ]  !gdown --id '1sUr1x-GhJ_80vIGzVGEqFUSDYfwV50YW' --output pikachu.png
```



Saving Notebook

- Download the .ipynb file to your local device (File > Download .ipynb)
- Save the colab notebook to your google drive (File > Save a copy in Drive).
- Convert .ipynb to .py and download (File > Download .py)



Useful Linux Commands (in Colab)

ls : List all files in the current directory

ls -l : List all files in the current directory with more detail

pwd : Output the working directory

mkdir <dirname> : Create a directory <dirname>

cd <dirname> : Move to directory <dirname>

gdown : Download files from google drive

wget : Download files from the internet

python <python_file>: Executes a python file

Problems You May Encounter…

- Colab will **automatically disconnect** if idle timeout(90 min., sometimes varying) or when your screen goes black

  → solution: keep your screen on or try using javascript

- GPU usage is **not unlimited** ! (your account will be stopped for a period if you reached the max gpu usage 12 hrs)
  *** The cooldown period before you can connect to another GPU will extend from hours to days to weeks depending on your usage**

  → solution: open another account

Best solution:

1. buy colab pro :)
2. use your own resource (if able)

Reminder: TAs are not required to help you solve environment problems

Reference

- https://colab.research.google.com
- https://research.google.com/colaboratory/faq.html