

How to run a program with Colab

1. Why Colab?

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

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

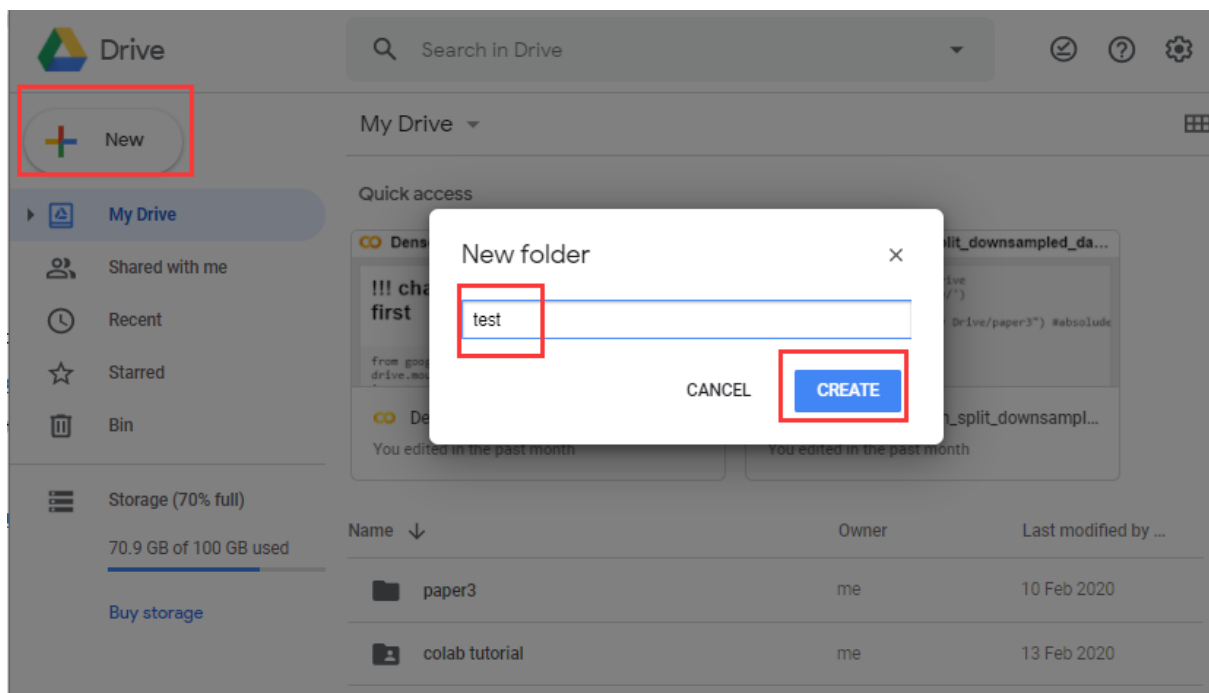
Whether you're a student, a data scientist or an AI researcher, Colab can make your work easier. Below is an official tutorial link.

https://Colab.research.Google.com/notebooks/intro.ipynb#scrollTo=5fCEDCU_qrC0

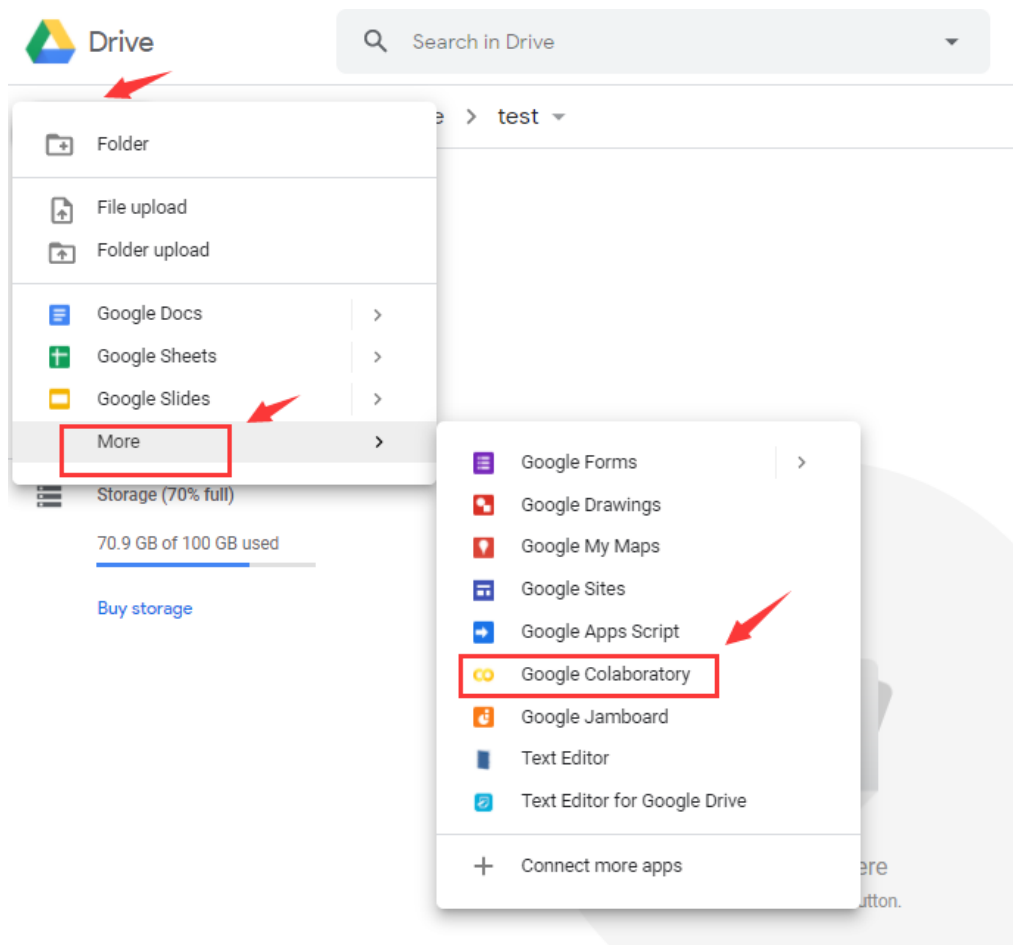
2. How to use Colab to run your program?

step 1: Log in your Google Drive and create a folder to hold your files.

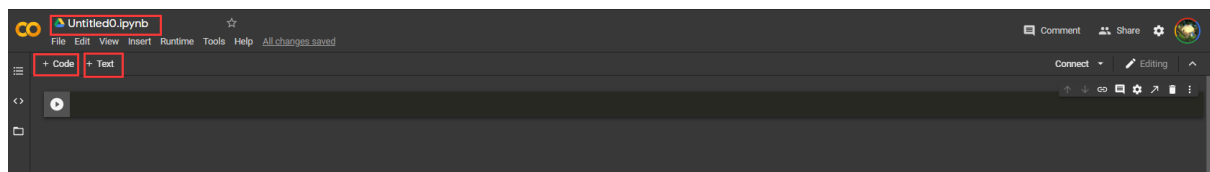
<https://drive.google.com/drive/u/0/my-drive>



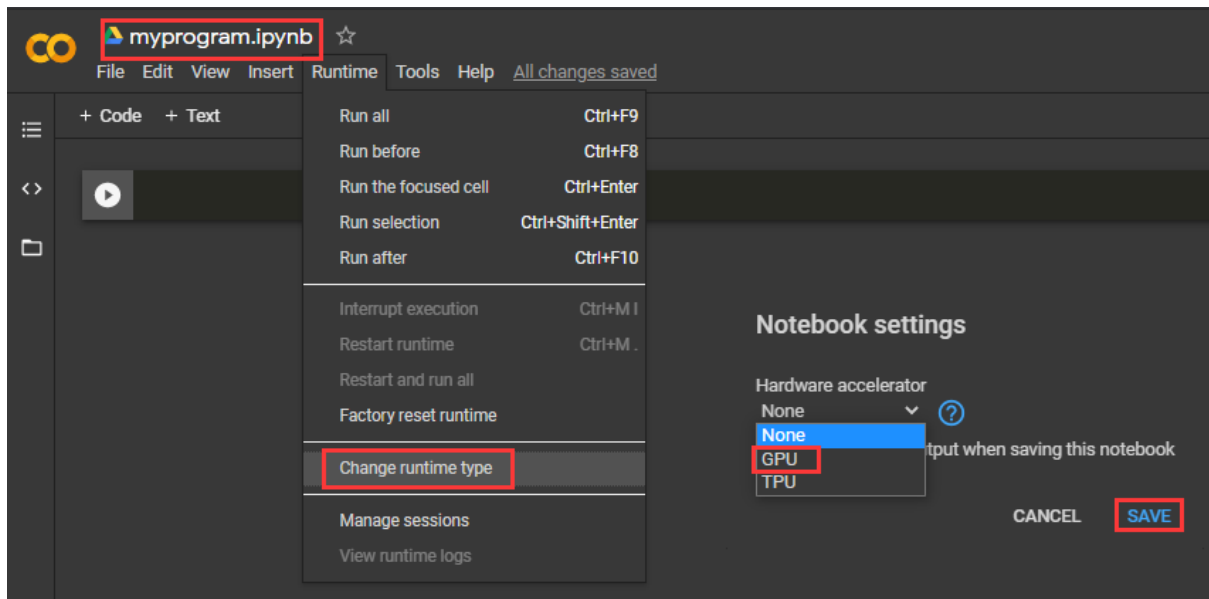
step 2: Enter the created folder and create a new Colab notebook.



You can change the name and add more 'code' blocks or 'test' blocks as shown below.

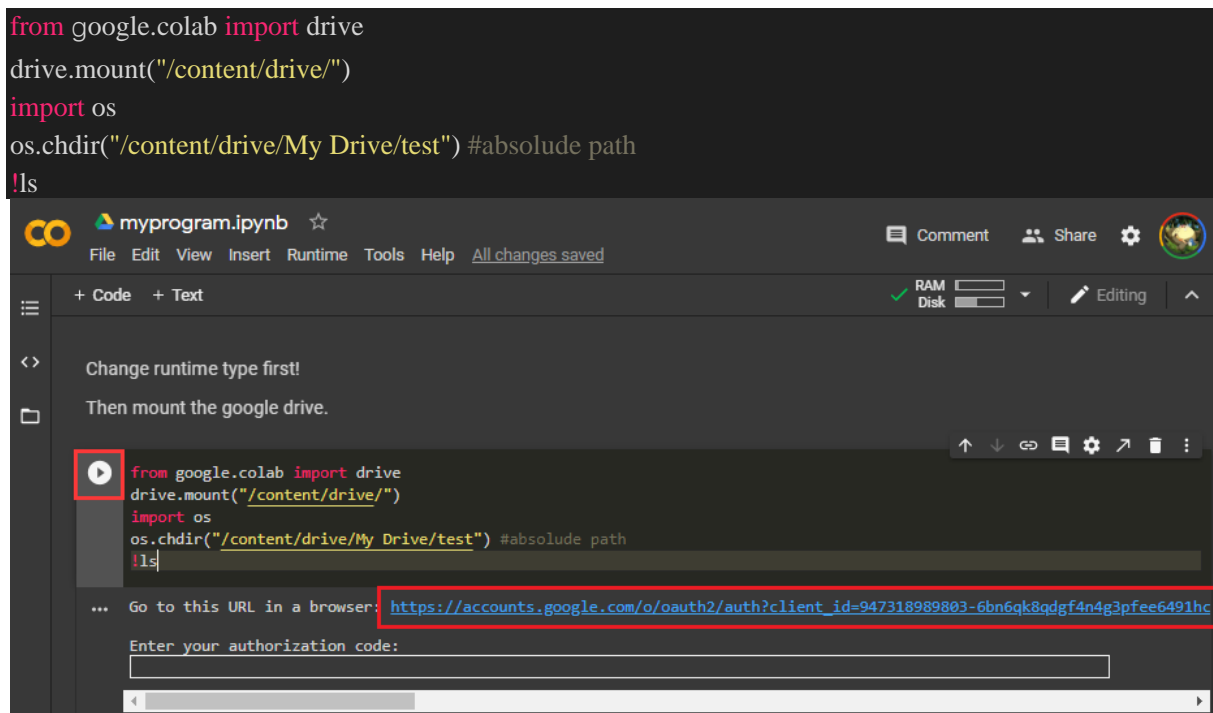


step 3: Change runtime type and choose GPU as the hardware accelerator



step 4: Mount Google Drive.

This step is optional but very recommended. Because it enables you to import files from your Google Drive folder or save results to your Google Drive folder conveniently.



The first two lines of codes will ask for getting the authorization code by logging into your Google account.



Sign in

Please copy this code, switch to your application and paste it there:

4/3wFTHzcu30E7jKW-
n_ikwFQTsqUW3qsCiwqE3jzhi3U3g03107eQTey



Then, copy and paste the authorization code and press Enter.

If everything goes well, you should see the response “Mounted at /content/Drive”

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client\_id=947318989803-6bn6qk8qdgf4n4g3pfee6491hc
Enter your authorization code:
4/3wEe6v8nSkrOKmsaAG7yM5rA0yDnPP5uXjtuvLTK_8hEHR_TupD5GK8
Mounted at /content/drive/
myprogram.ipynb
```

The next two lines change the current folder to the new created folder ‘test’.

```
import os
os.chdir("/content/drive/My Drive/test") #absolute path
```

The last line ‘!ls’ is used to double-check whether the Drive folder is properly mounted to ‘test’. We can see the notebook ‘myprogram.ipynb’ is in the folder ‘test’.

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client\_id=947318989803-6bn6qk8qdgf4n4g3pfee6491hc
Enter your authorization code:
4/3wEe6v8nSkrOKmsaAG7yM5rA0yDnPP5uXjtuvLTK_8hEHR_TupD5GK8
Mounted at /content/drive/
myprogram.ipynb
```

step 5: Use the code block as a Linux terminal.

Add a ‘!’ before a command, then the code block will work as a Linux terminal. For example, run ‘!nvidia-smi’ you will see the usage status of the current Active sessions.

```
!!nvidia-smi
```

```
[Wed Sep 9 12:02:16 2020]
+-----+
| NVIDIA-SMI 450.66                Driver Version: 418.67                CUDA Version: 10.1                |
+-----+-----+
| GPU   Name                     Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf    Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|=====-=+=====+=====+=====+=====+=====+=====+=====+
|  0   Tesla T4                  Off          | 00000000:00:04.0 Off |                    0 |
|N/A   31C    P8              9W / 70W | 0MiB / 15079MiB |      0%   Default  |
+-----+-----+
| Processes: |
| GPU   GI   CI        PID   Type   Process name                      GPU Memory |
| ID    ID   ID          |          |           | Usage      |
+-----+-----+
| No running processes found |
+-----+-----+

```

Run ‘!pip list’, you will see many packages like tensorflow and torch are pre-installed by Colab.

```
!pip list
statsmodels 0.10.2
sympy 1.1.1
tables 3.4.4
tabulate 0.8.7
tblib 1.7.0
tensorboard 2.3.0
tensorboard-plugin-wit 1.7.0
tensorboardcolab 0.0.22
tensorflow 2.3.0
tensorflow-addons 0.8.3
tensorflow-datasets 2.1.0
tensorflow-estimator 2.3.0
tensorflow-gcs-config 2.3.0
tensorflow-hub 0.9.0
tensorflow-metadata 0.23.0
tensorflow-privacy 0.2.2
tensorflow-probability 0.11.0
termcolor 1.1.0
terminado 0.8.3
testpath 0.4.4
text-unidecode 1.3
textblob 0.15.3
textgenrnn 1.4.1
Theano 1.0.5
thinc 7.4.0
tifffile 2020.8.25
toml 0.10.1
toolz 0.10.0
torch 1.6.0+cu101
torchsummary 1.5.1
torchtext 0.3.1
torchvision 0.7.0+cu101
```

step 6: Install and uninstall packages.

If the packages you used are not pre-installed. You can install it by yourself. For example, when I try to import Transformers (a State-of-the-art Natural Language Processing Package for PyTorch and TensorFlow 2.0. <https://huggingface.co/transformers/index.html>), an error occurred --- ModuleNotFoundError: No module named 'transformers'. Therefore, I need to install the package by myself.

```
!pip install transformers
```

```
!pip install transformers
Collecting transformers
  Using cached https://files.pythonhosted.org/packages/ae/95/c8c55b600308dc04e95100dc8ad8a244dd800fe75dfafcf1d6348c6f6209/transformers-3.1.0-py3-none-any.whl
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.6/dist-packages (from transformers) (4.41.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.6/dist-packages (from transformers) (3.0.12)
Requirement already satisfied: requests in /usr/local/lib/python3.6/dist-packages (from transformers) (2.23.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.6/dist-packages (from transformers) (1.18.5)
Requirement already satisfied: sentencepiece==0.1.92 in /usr/local/lib/python3.6/dist-packages (from transformers) (0.1.91)
Requirement already satisfied: dataclasses; python_version < "3.7" in /usr/local/lib/python3.6/dist-packages (from transformers) (0.7)
Requirement already satisfied: packaging in /usr/local/lib/python3.6/dist-packages (from transformers) (20.4)
Requirement already satisfied: regex==2019.12.17 in /usr/local/lib/python3.6/dist-packages (from transformers) (2019.12.20)
Requirement already satisfied: tokenizers==0.8.1.rc2 in /usr/local/lib/python3.6/dist-packages (from transformers) (0.8.1rc2)
Requirement already satisfied: sacremoses in /usr/local/lib/python3.6/dist-packages (from transformers) (0.0.43)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.6/dist-packages (from requests->transformers) (3.0.4)
Requirement already satisfied: certifi==2017.4.17 in /usr/local/lib/python3.6/dist-packages (from requests->transformers) (2020.6.20)
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python3.6/dist-packages (from requests->transformers) (1.24.3)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.6/dist-packages (from requests->transformers) (2.10)
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from packaging->transformers) (1.15.0)
Requirement already satisfied: pyparsing=2.0.2 in /usr/local/lib/python3.6/dist-packages (from packaging->transformers) (2.4.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.6/dist-packages (from sacremoses->transformers) (0.16.0)
Requirement already satisfied: click in /usr/local/lib/python3.6/dist-packages (from sacremoses->transformers) (7.1.2)
Installing collected packages: transformers
Successfully installed transformers-3.1.0
```

On the other hand, you can uninstall packages when you don't need them anymore or you want another version. Below is an example of uninstalling packages.

```
!pip uninstall transformers
```

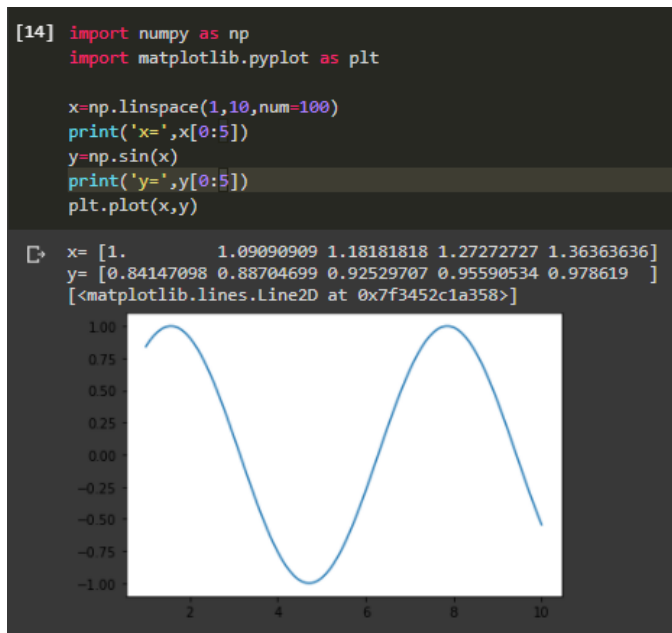
```
!pip uninstall transformers

Uninstalling transformers-3.1.0:
  Would remove:
    /usr/local/bin/transformers-cli
    /usr/local/lib/python3.6/dist-packages/transformers-3.1.0.dist-info/*
    /usr/local/lib/python3.6/dist-packages/transformers/*
  Proceed (y/n)? y
  Successfully uninstalled transformers-3.1.0
```

step 7: Code and run.

```
import numpy as np
import matplotlib.pyplot as plt

x=np.linspace(1,10,num=100)
print('x=',x[0:5])
y=np.sin(x)
print('y=',y[0:5])
plt.plot(x,y)
```

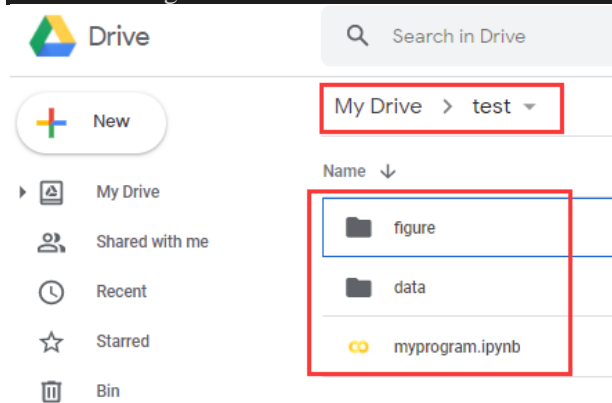


step 8: Save and load

In this step, I will show how to import data to Google Drive and save data to Google Drive.

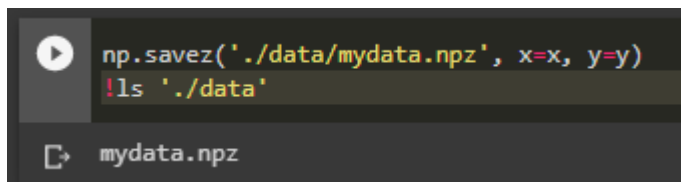
(1) Create a 'data' folder and a 'figure' folder under the 'test' folder.

```
!mkdir data figure
```



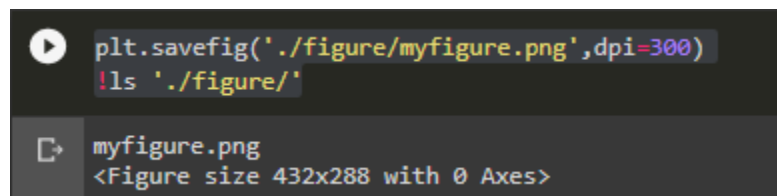
(2) Save data

```
np.savez('./data/mydata.npz', x=x, y=y)
!ls './data'
```



(3) Save figure

```
plt.savefig('./figure/myfigure.png',dpi=300)
!ls './figure/'
```



You may find that the figure is blank. Use the codes below to solve the problem.

```
import numpy as np
import matplotlib.pyplot as plt

x=np.linspace(1,10,num=100)
print('x=',x[0:5])
y=np.sin(x)
print('y=',y[0:5])

plt.figure() # creat a new figure
myfig = plt.gcf() # Get the current figure. If no current figure exists, a
new one is created using figure().
plt.plot(x,y) # plot on current figure. (and create another new figure,
which usually happens after plt.show().)
myfig.savefig('./figure/figure.png',dpi=300) #save myfig
```

```
data = np.load('./data/mydata.npz')
loaded_x=data['x']
loaded_y=data['y']
print('loaded_x',loaded_x[0:5])
print('loaded_y',loaded_y[0:5])
print(x==loaded_x)
print(y==loaded_y)
```

[illegible]

Colab is a very good platform for beginners to learn python programming. However, it also has the below limitations.

- “Google Colab notebooks have an idle timeout of 90 minutes and absolute timeout of 12 hours. This means, if user does not interact with his Google Colab notebook for more than 90 minutes, its instance is automatically terminated. Also, maximum lifetime of a Colab instance is 12 hours.”*

- (3) Very slow Google Drive disk access speed when a folder contains thousands of files.

