

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

High Performance Multiprocessor Systems



Outline

- Introduction
- What Colab Offers You?
- Your First Colab Notebook
- Code & Text
- Getting started with Google Drive
- Mounting Drive
- Listing Drive Contents
- Running Python Code
- Enabling GPU
- Install pycuda
- Import pycuda

Introduction

- Colab is a free notebook environment that runs entirely in the cloud. It lets you and your team members edit documents, the way you work with Google Docs. Colab supports many popular machine learning libraries which can be easily loaded in your notebook.
- Google is quite aggressive in AI research. Over many years, Google developed AI framework called **TensorFlow** and a development tool called **Colaboratory**. Today TensorFlow is open-sourced and since 2017, Google made Colaboratory free for public use. Colaboratory is now known as Google Colab or simply **Colab**.
- Another attractive feature that Google offers to the developers is the use of GPU. Colab supports GPU and it is totally free. The reasons for making it free for public could be to make its software a standard in the academics for teaching machine learning and data science. It may also have a long term perspective of building a customer base for Google Cloud APIs which are sold per-use basis.
- Irrespective of the reasons, the introduction of Colab has eased the learning and development of machine learning applications.

What Colab Offers You?

➤ As a programmer, you can perform the following using Google Colab.

- Write and execute code in Python
- Document your code that supports mathematical equations
- Create/Upload/Share notebooks
- Import/Save notebooks from/to Google Drive
- Import/Publish notebooks from GitHub
- Import external datasets e.g. from Kaggle
- Integrate PyTorch, TensorFlow, Keras, OpenCV
- Free Cloud service with free GPU

Your First Colab Notebook



Note – As Colab implicitly uses Google Drive for storing your notebooks, ensure that you are logged in to your Google Drive account before proceeding further.

Step 1 – Open the following URL in your browser

– <https://colab.research.google.com> Your browser would display the following screen (assuming that you are logged into your Google Drive)

https://colab.research.google.com/?utm_source=scs-index

The screenshot displays the Google Colaboratory interface. The top navigation bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. A 'Cannot save changes' warning is visible. The 'File' menu is open, showing options: 'New notebook', 'Open notebook' (Ctrl+O), 'Upload notebook', 'Rename', 'Save a copy in Drive', 'Save a copy as a GitHub Gist', and 'Save a copy in GitHub'. The main content area shows a 'Welcome To Colaboratory' message and a list of notebooks. The notebooks list includes:

Title	Last opened	First opened	Actions
Welcome To Colaboratory	3:15 PM	August 10	[Icon]
Welcome To Colaboratory	3:12 PM	3:12 PM	[Icon]
tutorial_deep_learning_basics.ipynb	2:56 PM	2:56 PM	[Icon] [Icon]
Untitled2.ipynb	November 7	November 7	[Icon] [Icon]
01.05-IPython-And-Shell-Commands.ipynb	November 7	November 7	[Icon] [Icon]

At the bottom right, there are buttons for 'New notebook' and 'Cancel'.

Code & Text

1

Copy of hello1.ipynb

File Edit View Insert Runtime Tools Help Last saved

+ Code + Text

Insert code cell below
Ctrl+M B

```
import numpy
a =numpy.array([1,2,3])
print(a)
```

[1 2 3]

hello

2

Copy of hello1.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

```
[ ] import numpy
a =numpy.array([1,2,3])
print(a)
```

[1 2 3]

import numpy as np

3

Copy of hello1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Add text cell

```
[ ] import numpy
a =numpy.array([1,2,3])
print(a)
```

[1 2 3]

import numpy as np

hello



Getting started with Google Drive

- **Google Drive** is a free service from Google that allows you to store files **online** and access them anywhere using the **cloud**. Google Drive also gives you access to **free web-based applications** for creating **documents, spreadsheets, presentations**, and more.
- Why use Google Drive?
 - Google Drive is one of the most popular cloud storage services available today. If you've never used a cloud-based storage service like Google Drive before, take a moment to consider the **advantages** of keeping your files online. Because files can be accessed from any computer with an Internet connection, Drive eliminates the need to email or save a file to a USB drive. And because Drive allows you to **share** files, working with others becomes much easier.

<https://edu.gcfglobal.org/en/googledriveanddocs/all-about-google-drive/1/>

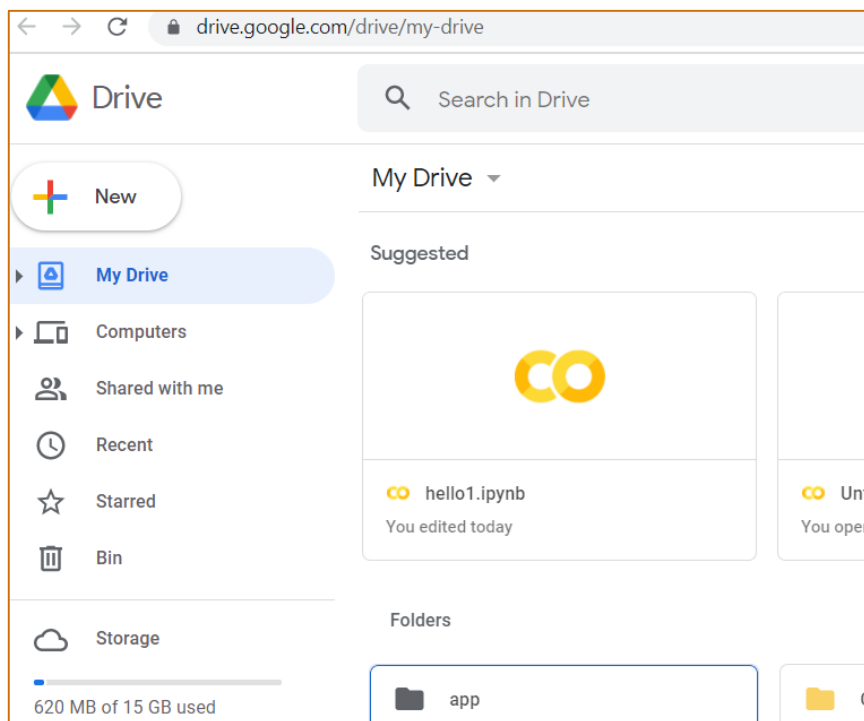
<https://edu.gcfglobal.org/en/googledriveanddocs/getting-started-with-google-drive/1/>



Getting started with Google Drive

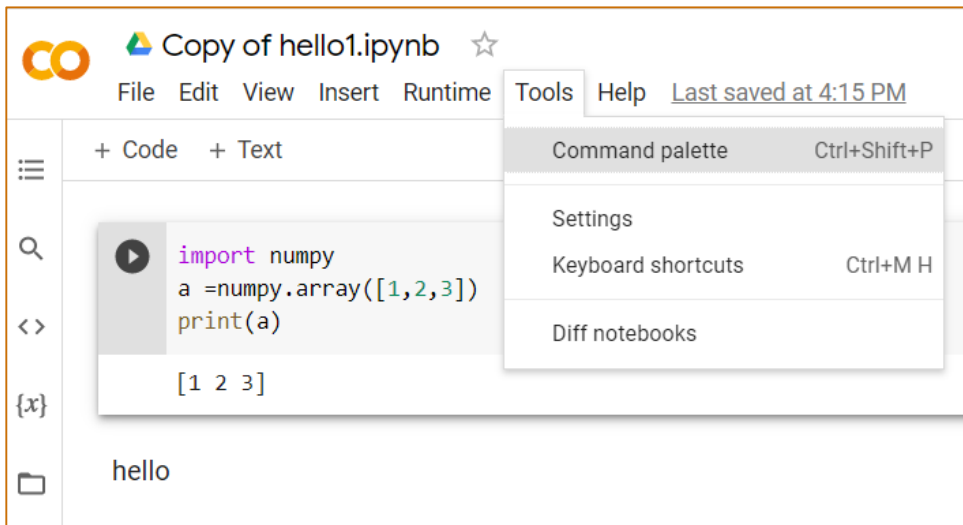
<https://www.google.com/drive/>

[Go to Drive](#)

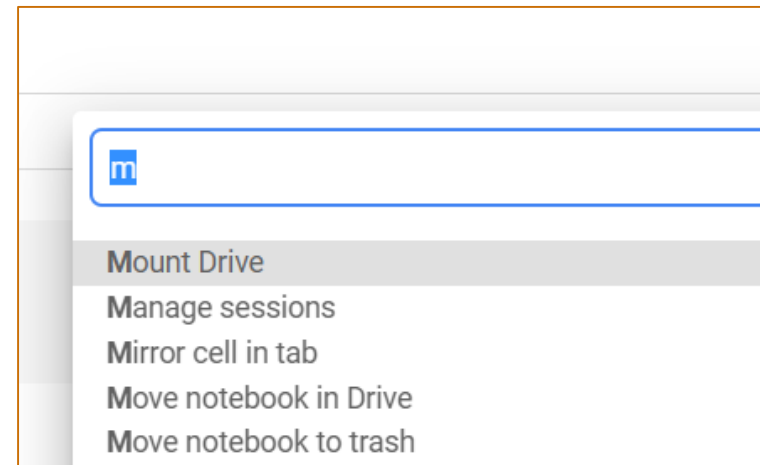


Mounting Drive

1



2



3

Permit this notebook to access your Google Drive files?

Connecting to Google Drive will permit code executed in this notebook to modify files in your Google Drive until access is otherwise revoked.

No thanks

Connect to Google Drive

Listing Drive Contents Running Python Code

1

Listing Drive Contents

```
✓ [6] !ls "/content/drive/My Drive/Colab Notebooks"
0s
'Copy of hello1.ipynb'    hello1.ipynb    Untitled0.ipynb    Untitled2.ipynb
'Copy of Untitled3.ipynb'  Untitled        Untitled1.ipynb    Untitled3.ipynb
```

Running Python Code

2

```
✓ 3s ▶ !python3 "/content/drive/My Drive/multiprocessing_example2.py"

Starting foo_process

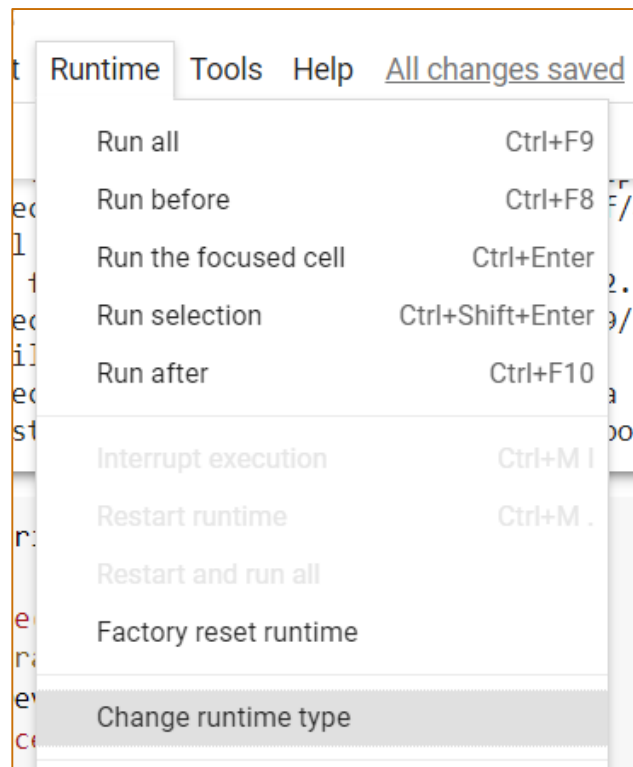
Starting Process-2

Exiting foo_process

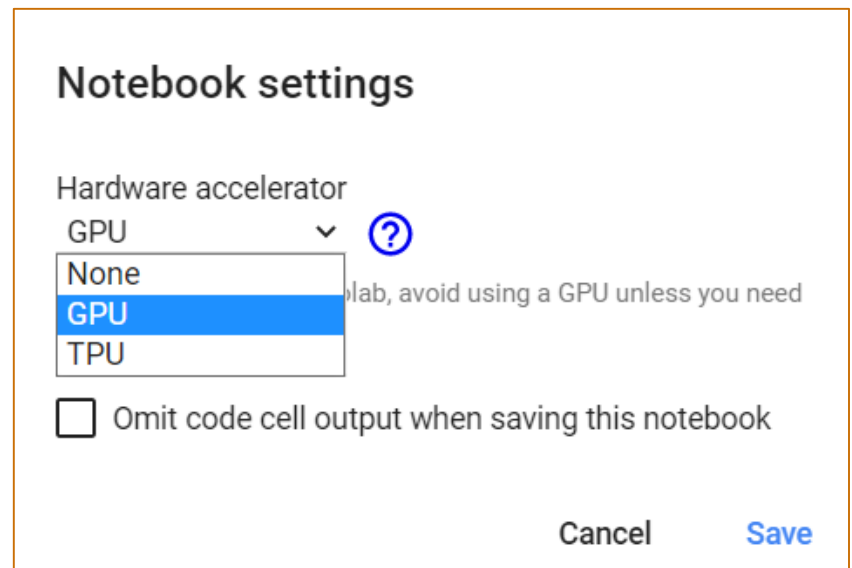
Exiting Process-2
```

Enabling GPU

1



2



Install pycuda

```
!pip install pycuda # install cuda|

Collecting pycuda
  Downloading pycuda-2021.1.tar.gz (1.7 MB)
    |████████████████████████████████████████| 1.7 MB 3.6 MB/s
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing wheel metadata ... done
Requirement already satisfied: appdirs>=1.4.0 in /usr/local/lib/python3.7/dist-packages (from pycuda)
Collecting pytools>=2011.2
  Downloading pytools-2021.2.9.tar.gz (66 kB)
    |████████████████████████████████████████| 66 kB 4.0 MB/s
Collecting mako
  Downloading Mako-1.1.6-py2.py3-none-any.whl (75 kB)
    |████████████████████████████████████████| 75 kB 4.4 MB/s
Requirement already satisfied: numpy>=1.6.0 in /usr/local/lib/python3.7/dist-packages (from mako)
Requirement already satisfied: MarkupSafe>=0.9.2 in /usr/local/lib/python3.7/dist-packages (from mako)
Building wheels for collected packages: pycuda, pytools
  Building wheel for pycuda (PEP 517) ... done
  Created wheel for pycuda: filename=pycuda-2021.1-cp37-cp37m-linux_x86_64.whl size=140000 sha256=...
  Stored in directory: /root/.cache/pip/wheels/c4/ef/49/dc6a5feb8d980b37c83d465e...
  Building wheel for pytools (setup.py) ... done
  Created wheel for pytools: filename=pytools-2021.2.9-py2.py3-none-any.whl size=140000 sha256=...
  Stored in directory: /root/.cache/pip/wheels/41/b9/6e/94bb014f6484b15ec77e7877...
Successfully built pycuda pytools
Installing collected packages: pytools, mako, pycuda
Successfully installed mako-1.1.6 pycuda-2021.1 pytools-2021.2.9
```

Import pycuda

```
▶ import pycuda.driver as drv
   drv.init()
   print("%d device(s) found." % drv.Device.count())
   for ordinal in range(drv.Device.count()):
       dev = drv.Device(ordinal)
       print("Device #%d: %s" % (ordinal, dev.name()))
       print(" Compute Capability: %d.%d" % dev.compute_capability())
       print(" Total Memory: %s KB" % (dev.total_memory()//(1024)))
```

```
↳ 1 device(s) found.
   Device #0: Tesla K80
       Compute Capability: 3.7
       Total Memory: 11715776 KB
```