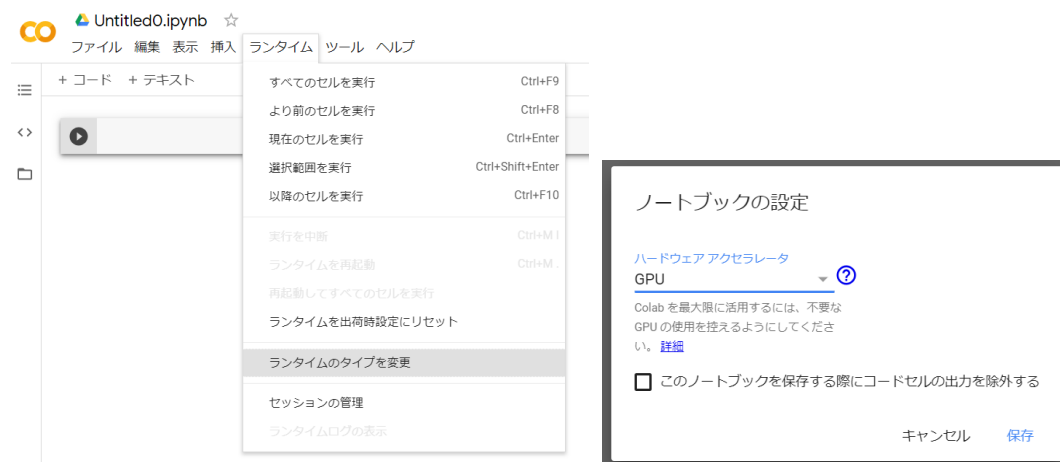


0.事前準備:[OBI-Models\\_Samples.zip](#)をダウンロードし解凍し、その内容を自分の Google ドライブにアップロードする (例えば下の図のように):

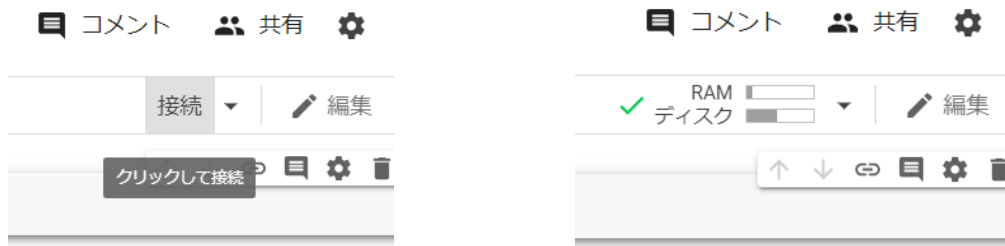
マイドライブ > OBI-Models ▾



1.[Colaboratory](#) に接続し、自分の Google アカウントでログインする。その後、ノートブックを新規作成し、ランタイムのタイプを GPU に変更する:



ページ右上の「接続」をクリックすると、GPU の設定が完了した。(右の図のように)



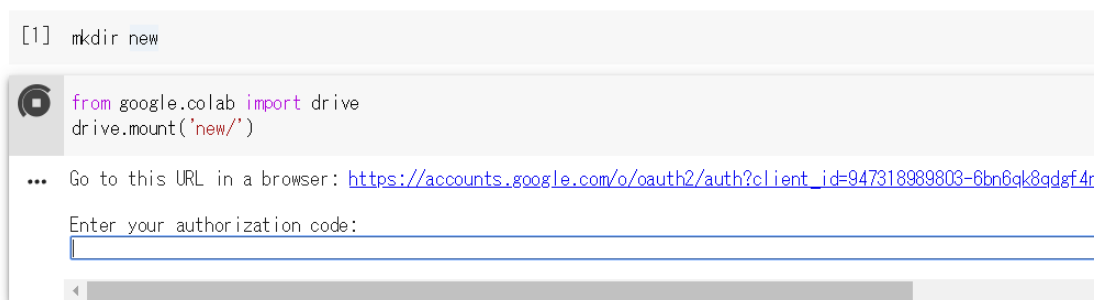
2.コードの挿入 (左の図) と実行 (右の図) :



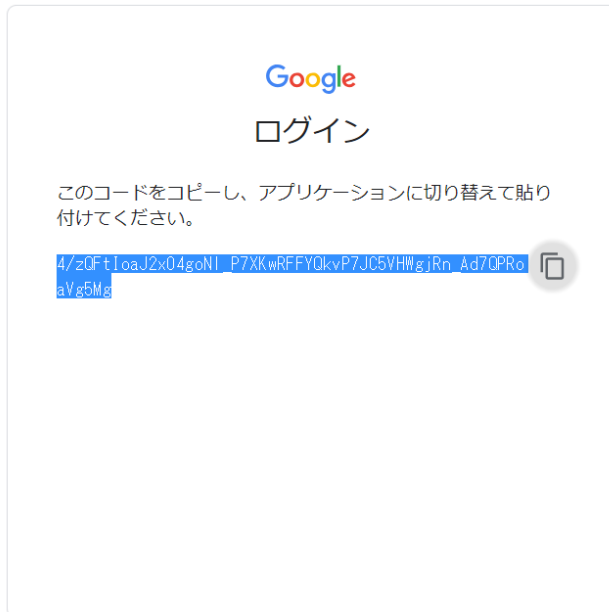
Colab と自分の Google ドライブを連携するために、次のコードを実行する :

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

+ コード + テキスト



青いリンクをクリックし、自分の Google アカウントでログインし、下の図のようなコードをコピーする。



コードを入力し、「Mounted at new/」を見ると、Colab のフォルダー「new」と自分の Google ドライブが連携した。

+ コード + テキスト

```
[1] mkdir new
```

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

Go to this URL in a browser: [https://accounts.google.com/o/oauth2/auth?client\\_id=947318989803-6b...](https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6b...)

Enter your authorization code:  
.....|

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

Go to this URL in a browser: <https://a...>

Enter your authorization code:  
.....  
Mounted at new/

3.前アップロードしたフォルダー（例えば OBI-Models）に入り、コード

**!python untitled0.py**

を入力しプログラムを実行する。（「!」は必要なもの）

```
+ コード + テキスト
ls
new/ sample_data/

[4] cd new
/content/new

[5] ls
'My Drive'/

[6] cd My Drive
/content/new/My Drive

[8] cd OBI-Models/
/content/new/My Drive/OBI-Models

[9] ls
Images/ untitled0.py
```

```
!python untitled0.py
... Using TensorFlow backend.
2020-05-01 09:58:04.518153: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcudart.so.10.1
2020-05-01 09:58:06.482873: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcuda.so.1
2020-05-01 09:58:06.535582: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:961] successful NUMA node read from SysFS had negative value (-
2020-05-01 09:58:06.536486: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1561] Found device 0 with properties:
pciBusID: 0000:00:04.0 name: Tesla K80 computeCapability: 3.7
coreClock: 0.8235GHz coreCount: 13 deviceMemorySize: 11.17GiB deviceMemoryBandwidth: 223.96GiB/s
2020-05-01 09:58:06.536543: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcudart.so.10.1
2020-05-01 09:58:06.777995: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcublas.so.10
2020-05-01 09:58:06.925908: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcufft.so.10
2020-05-01 09:58:06.947462: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcublas.so.10
2020-05-01 09:58:07.230344: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcusolver.so.10
2020-05-01 09:58:07.276465: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcusparse.so.10
```

```
Epoch 1/1000
2020-05-01 09:58:53.701613: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcublas.so.10
2020-05-01 09:58:55.130281: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcudnn.so.7
49/49 [=====] - 199s 4s/step - loss: 2.9536 - accuracy: 0.2026 - val_loss: 3.3678 - val_accuracy: 0.0625
Epoch 2/1000
49/49 [=====] - 86s 2s/step - loss: 1.5874 - accuracy: 0.5359 - val_loss: 3.4010 - val_accuracy: 0.0787
Epoch 3/1000
49/49 [=====] - 49s 1s/step - loss: 1.1783 - accuracy: 0.6513 - val_loss: 3.3300 - val_accuracy: 0.0337
Epoch 4/1000
49/49 [=====] - 49s 1s/step - loss: 0.8336 - accuracy: 0.7423 - val_loss: 3.1618 - val_accuracy: 0.0112
Epoch 5/1000
49/49 [=====] - 49s 1s/step - loss: 0.7698 - accuracy: 0.7641 - val_loss: 3.7438 - val_accuracy: 0.0417
Epoch 6/1000
49/49 [=====] - 50s 1s/step - loss: 0.6744 - accuracy: 0.8036 - val_loss: 4.3362 - val_accuracy: 0.0449
Epoch 7/1000
49/49 [=====] - 49s 1s/step - loss: 0.4118 - accuracy: 0.8923 - val_loss: 3.7340 - val_accuracy: 0.0674
Epoch 8/1000
49/49 [=====] - 49s 990ms/step - loss: 0.4347 - accuracy: 0.8634 - val_loss: 3.6703 - val_accuracy: 0.2022
Epoch 9/1000
49/49 [=====] - 50s 1s/step - loss: 0.3652 - accuracy: 0.8929 - val_loss: 2.0332 - val_accuracy: 0.5521
```

これで Colab で GPU を用いて深層学習プログラムを実行した。

4.深層学習モデルの利用について、次のページで参照できる：

<https://keras.io/ja/applications/>