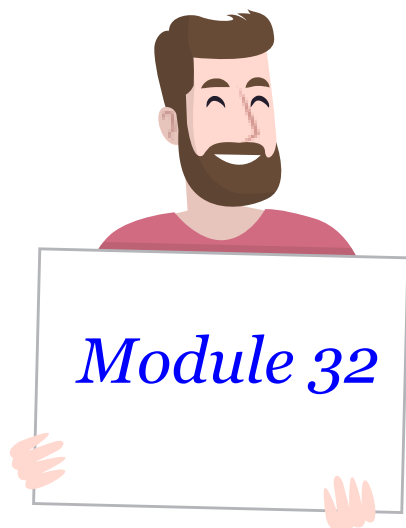




# Colab 介紹



designed by  freepik

Estimated time:  
**45** min.

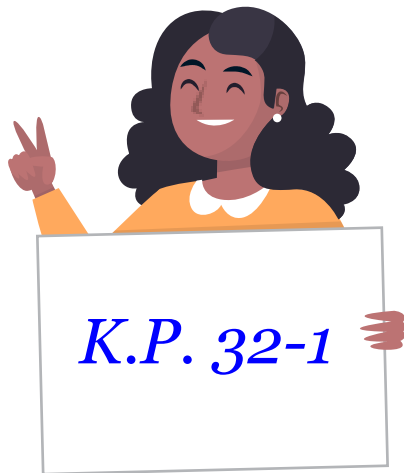
# 學習目標

- 32-1:Colab介紹
- 32-2:Colab使用
- 32-3:在Colab上訓練神經網路



# 32-1: Colab介紹

- **Google Colab**
- **使用Colab好處**
- **進入Colab頁面**



designed by freepik

# Google Colab
















- Colab是Google提供的雲端Jupyter Notebook開發環境
  - 可以在上面開發深度學習的
  - 提供免費GPU使用
  - 每次開啟有12小時的連續使用時間限制，超過需重新開啟才能使用

# 使用Colab好處

- 不用安裝任何東西
  - 只需要打開瀏覽器並搜尋Colab即可使用
- 但須注意Colab要有gmail帳號才能使用
  - 可以在以下連結申請gmail
  - [https://www.google.com/intl/zh-TW\\_tw/gmail/about/](https://www.google.com/intl/zh-TW_tw/gmail/about/)

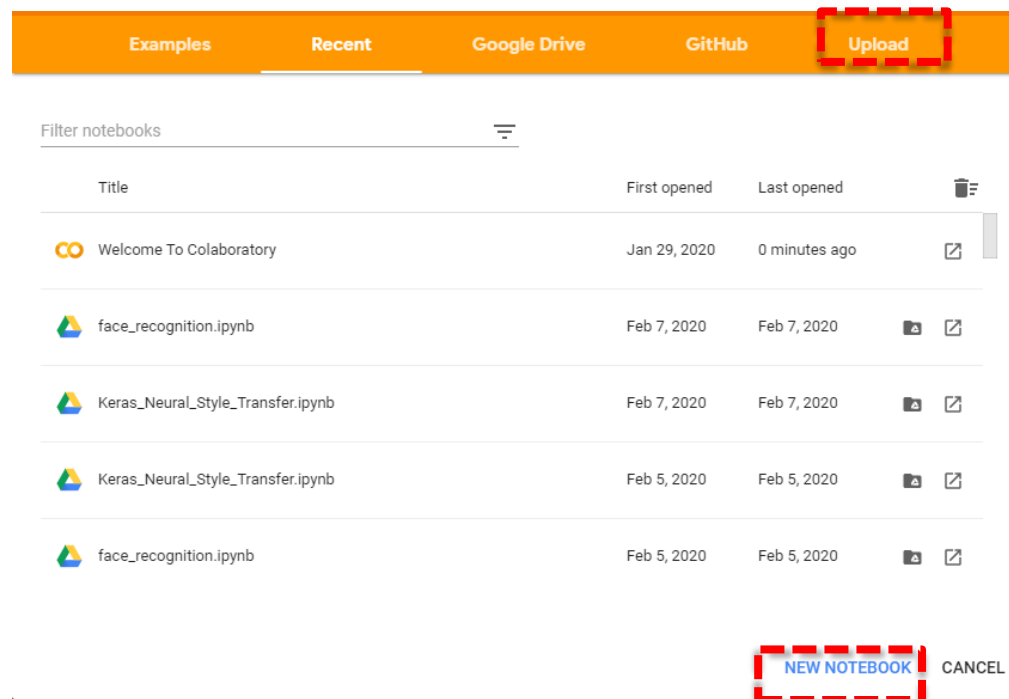
# 進入Colab頁面

- 登入Google帳號並進入以下連結
  - <https://colab.research.google.com/>
- 之後在Colab所建立的jupyter notebook檔案都會儲存在此Google帳號裡的Google drive空間當中

Examples   Recent   Google Drive   GitHub   Upload				
Filter notebooks				
Title	First opened	Last opened		
 Welcome To Colaboratory	Jan 29, 2020	0 minutes ago		
 face_recognition.ipynb	Feb 7, 2020	Feb 7, 2020		
 Keras_Neural_Style_Transfer.ipynb	Feb 7, 2020	Feb 7, 2020		
 Keras_Neural_Style_Transfer.ipynb	Feb 5, 2020	Feb 5, 2020		
 face_recognition.ipynb	Feb 5, 2020	Feb 5, 2020		
<a href="#">NEW NOTEBOOK</a> <a href="#">CANCEL</a>				

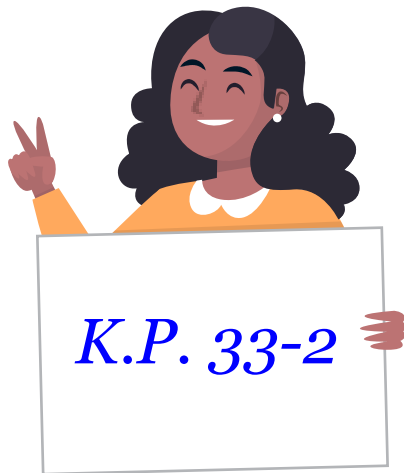
# 進入Colab頁面

- 進入Colab頁面後
  - 可以選擇”upload”去上傳一個ipynb檔案
  - 也可以按下”New Notebook”來開啟一個新的ipynb檔案



## 32-2: Colab使用

- 連接Google Cloud
- 新增程式碼區塊及文字區塊
- 上傳檔案
- 程式碼行數顯示

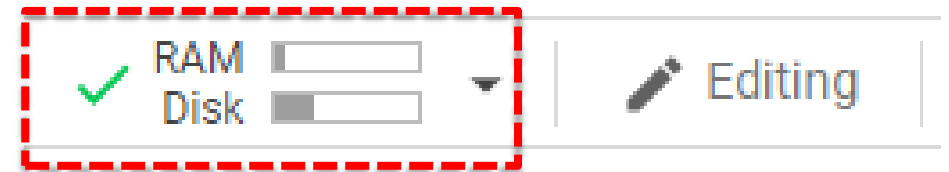
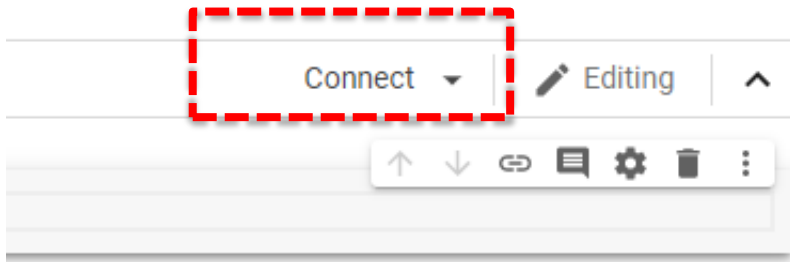


designed by freepik



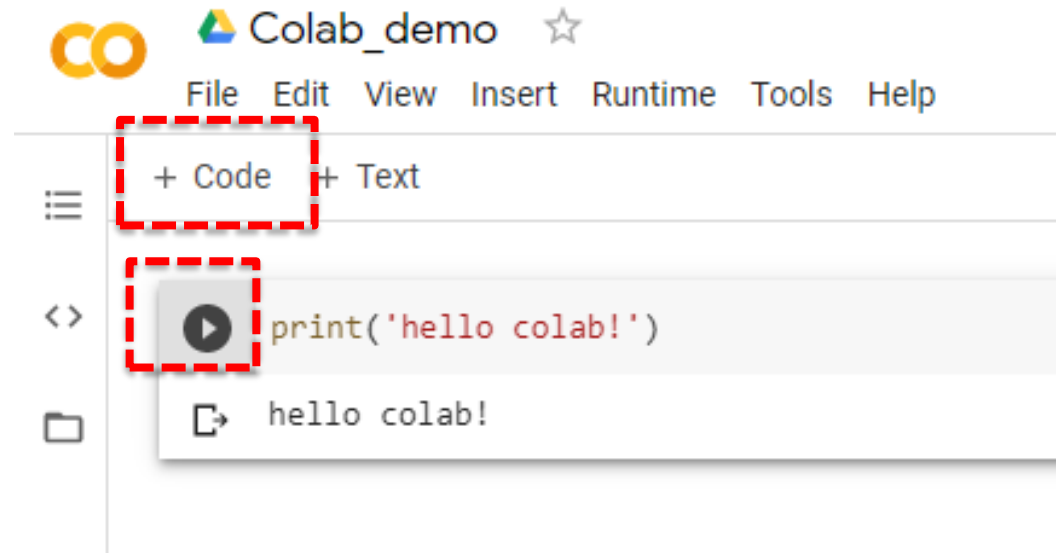
# 連接Google Cloud

- 按下Colab環境右上角的”Connect”
  - 它會在Google Cloud產生新的主機來讓使用者跑接下來的程式碼
  - 等待一陣子後會出現右下圖片，顯示目前RAM以及Disk用量



# 新增程式碼區塊及文字區塊

- 按下”+Code”即可新增程式碼區塊
  - 當輸入完新的程式碼後，可以按下箭頭符號執行(或是按下快捷鍵 **shift+Enter**也行)
  - 結果將會顯示在程式碼正下方，這跟Jupyter Notebook操作上是一樣的



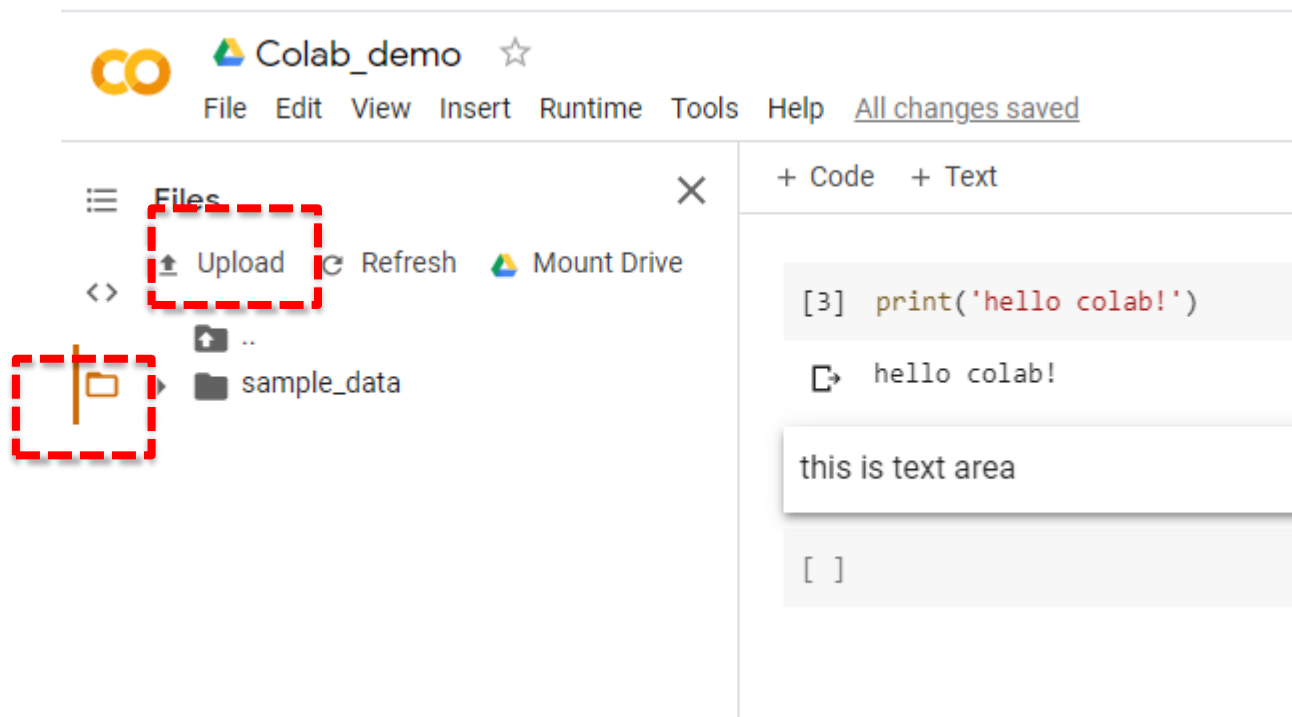
# 新增程式碼區塊及文字區塊

- 按下”+Text”即可新增文字區塊
  - 當輸入完新的文字後，可以按下箭頭符號執行(或是按下快捷鍵**shift+Enter**也行)



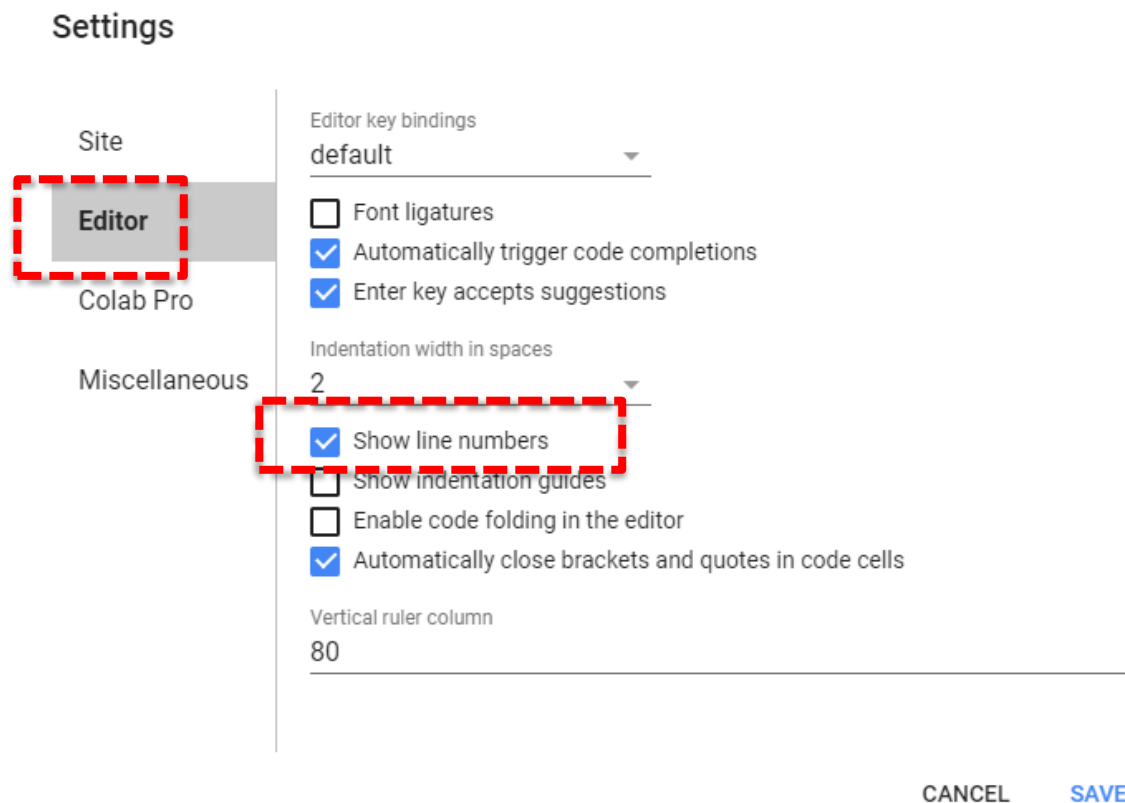
# 上傳檔案

- 如果需要上傳資料集到Colab上面，可以按下左側資料夾的圖示
  - 點選”upload”並上傳檔案



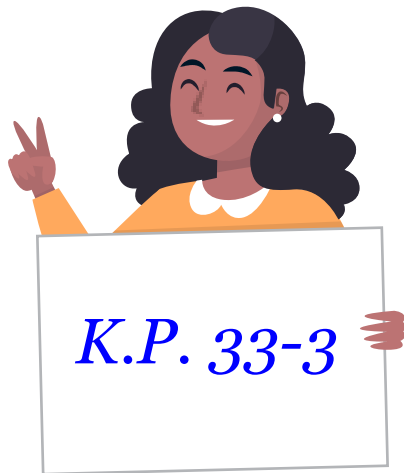
# 程式碼行數顯示

- 點選上方”Setting”
  - 在”Editor”底下有”Show line numbers”把它勾起來即可顯示程式碼行數



# 32-3: 在Colab上訓練神經網路

- TPU介紹
- 選取GPU/TPU做運算單元
- 在Colab上安裝套件
- 執行神經網路



designed by freepik

# TPU介紹

- 張量處理器(TPU)是Google為深度學習框架TensorFlow所設計的加速晶片
  - Google在2016年的Google I/O年會上首次公布了TPU
  - 在使用Colab時，也可以使用TPU

# 選取GPU/TPU做運算單元

- 在使用Colab時，可以使用GPU/TPU的資源
  - 點選Colab上方按鈕”Runtime”的”change runtime type”

## Notebook settings

Runtime type  
Python 3

Hardware accelerator  
None

☐ Omit code cell output

GPU  
TPU

CANCEL SAVE



# 在Colab上安裝套件

- 如果需要在Colab上面安裝套件，只需要將pip指令輸入在程式碼區塊內並執行即可

+ Code + Text

```
[1] pip install pandas
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.6/dist-packages (0.25.3)  
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.6/dist-packages (from pandas) (2018.9)  
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.6/dist-packages (from pandas) (1.17.5)  
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.6/dist-packages (from pandas) (2.6.1)  
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.6/dist-packages (from python-dateutil>=2.6.1->pandas) (1.12.0)
```

```
[2] pip install numpy
```

```
Requirement already satisfied: numpy in /usr/local/lib/python3.6/dist-packages (1.17.5)
```

# 執行神經網路

- 可以將之前所學到的DNN神經網路TensorFlow程式碼拿來在Colab執行看看

```
cross_entropy = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits(labels=y, logits=y_predict))
train_step = tf.train.GradientDescentOptimizer(0.5).minimize(cross_entropy)

correct_prediction = tf.equal(tf.argmax(y_predict, 1), tf.argmax(y, 1))
accuracy = tf.reduce_mean(tf.cast(correct_prediction, tf.float32))

with tf.Session() as sess:
    sess.run(tf.global_variables_initializer())

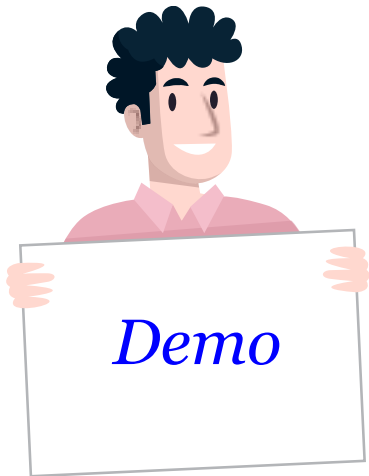
    for step in range(1000):
        batch_xs, batch_ys = mnist.train.next_batch(100)

        train_step_, cross_entropy_ = sess.run([train_step, cross_entropy], feed_dict={x: batch_xs, y: batch_ys})
        if step % 50 == 0:
            print("step {}: cross_entropy is {}".format(step, cross_entropy_))
        accuracy_ = sess.run(accuracy, feed_dict={x: mnist.test.images, y: mnist.test.labels})
        print('Testing..... accuracy is {}'.format(accuracy_))
```

```
Extracting MNIST_data/train-images-idx3-ubyte.gz
Extracting MNIST_data/train-labels-idx1-ubyte.gz
Extracting MNIST_data/t10k-images-idx3-ubyte.gz
Extracting MNIST_data/t10k-labels-idx1-ubyte.gz
step 0: cross_entropy is 2.3113460540771484
step 50: cross_entropy is 0.9799741506576538
step 100: cross_entropy is 0.7489251494407654
step 150: cross_entropy is 0.6375553011894226
step 200: cross_entropy is 0.4262787997722626
step 250: cross_entropy is 0.4993631839752197
step 300: cross_entropy is 0.46281278133392334
step 350: cross_entropy is 0.32882171869277954
step 400: cross_entropy is 0.5450003147125244
```

# Demo 32-3

- Colab開啟
- Colab上傳jupyter notebook
- 執行神經網路程式碼



designed by freepik

# 線上Corelab

- 題目1：使用Colab去執行DNN神經網路在MNIST上的分類
- 題目2：使用Colab去執行CNN神經網路在MNIST上的分類
- 題目3：使用Colab去執行RNN神經網路在MNIST上的分類

# 本章重點精華回顧

- Colab介紹
- Colab基礎使用
- Colab執行神經網路



# Lab: Colab環境使用

- Lab01: Colab開啟
- Lab02: Colab上傳jupyter notebook
- Lab03: 執行神經網路程式碼

Estimated time:

**20** minutes

