

Jupyter Notebook

Start



- 從 Anaconda 開啟 Jupyter Notebook 會跑出命令提示字元的視窗
- 會在Google中開啟網頁

```
Microsoft Windows [版本 10.0.18362.900]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。

C:\Users\lee>jupyter notebook
[I 14:47:47.997 NotebookApp] JupyterLab extension loaded from c:\users\lee\appdata\local\programs\python\python37\lib\site-packages\jupyterlab
[I 14:47:47.997 NotebookApp] JupyterLab application directory is c:\users\lee\appdata\local\programs\python\python37\share\jupyter\lab
[I 14:47:47.997 NotebookApp] Serving notebooks from local directory: D:/
[I 14:47:47.997 NotebookApp] The Jupyter Notebook is running at:
[I 14:47:47.997 NotebookApp] http://localhost:8888/?token=e305e6d72517dd873fd89c3d52ca8512fa71a71aefa90a1c
[I 14:47:47.997 NotebookApp] or http://127.0.0.1:8888/?token=e305e6d72517dd873fd89c3d52ca8512fa71a71aefa90a1c
[I 14:47:47.997 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 14:47:48.091 NotebookApp]

To access the notebook, open this file in a browser:
    file:///C:/Users/lee/AppData/Local/programs/python/python37/share/jupyter/runtime/nbserver-5304-open.html
Or copy and paste one of these URLs:
    http://localhost:8888/?token=e305e6d72517dd873fd89c3d52ca8512fa71a71aefa90a1c
    or http://127.0.0.1:8888/?token=e305e6d72517dd873fd89c3d52ca8512fa71a71aefa90a1c
```

Debug:

1. 網頁若沒有開啟則可以複製網址開啟
2. 另一種開啟方法為開啟Anaconda的命令提示字元輸入jupyter notebook

DIR

- 開啟後右上角會有一個New可以新增python ipynb檔
- Upload可以從別處上傳檔案到這個地方
- 開啟的預設路徑皆為使用者底下目錄



更改起始目錄:

- 1.開啟 Anaconda 命令提示字元
- 2.輸入jupyter notebook --generate-config
- 3.輸入時會跟你講位址要記住
- 4.找到jupyter_notebook_config.py以txt開啟
- 5.用ctrl + F找到

```
## The directory to use for notebooks and kernels.
#c.NotebookApp.notebook_dir = ""
```
- 6.把#拿掉並輸入自己想預設的位置例如下

```
## The directory to use for notebooks and kernels.
c.NotebookApp.notebook_dir = ' D:\Python '
```
- 7.重啟即可

Interface

點一下可以重新命名

jupyter Untitled Last Checkpoint: 2019年11月27日 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Run Code



Logout

Trusted

Python 3

Connecting to kernel

Trusted

Python 3

稱為Cell

In [4]:

```
1 import sys
2 import cv2
3 import dlib
```

In [5]:

```
1 Face_LandMarkModel_Path = 'D:/Model/Dlib_Face_LandMark/shape_predictor_68_face_landmarks.dat'
2 detector = dlib.get_frontal_face_detector()
3 predictor = dlib.shape_predictor(Face_LandMarkModel_Path)
```

In [12]:

```
1 Camera = cv2.VideoCapture(0)
2
3 while True:
4     ret, frame = Camera.read()
5     frame = cv2.flip(frame, 1)
6     if not ret:
7         break
8     face_rects, scores, idx = detector.run(frame, 0)
9     for i, d in enumerate(face_rects):
10         x1 = d.left()
```

一整個叫一個Kernel
有連接到系統會顯示圓圈
沒有連接到系統會顯示斷裂error
通常跳出斷裂為不小心把
命令提示字元視窗關閉重啟即可

Hot Key

四個常用快捷鍵

1.Shift + Enter 執行

2.Esc + a 上方加入cell

3.Esc + b 下方加入cell

4.Esc + d + d (ESC案住D雙擊)

查詢其他快捷指令用esc + h

Cell未執行過

```
In [ ]: 1 for i in range(100000):  
        2 print(i)
```

Cell執行中顯示*

```
In [*]: 1 for i in range(100000):  
        2 print(i)
```

Cell執行完畢顯示這個
kernel執行完畢幾次

```
In [1]: 1 for i in range(100000):  
        2 print(i)
```

Interface

複製cell內容

執行等同shift + enter

Kernel重啟

剪下cell內容

貼上cell內容

程式中斷(停止執行)

下方加入一個cell
等同esc + b

存檔等同esc + s

The image shows the Jupyter Notebook interface with a toolbar and a code cell. The toolbar contains the following buttons from left to right: a save icon (floppy disk), a plus icon (add cell), a scissors icon (cut), a copy icon, a paste icon, up and down arrow icons, a 'Run' button (play icon), a square button (interrupt), a circular arrow icon (restart kernel), and a right arrow icon. Below the toolbar is a code cell with the prompt 'In [*]:' and the following code:

```
1 i = 0
2 while i==0:
3     print(i)
```

The output of the code cell shows seven '0' characters on separate lines.

Docstring

詳細註解

在括號內案
shift + tab 可
以顯示註解及
使用方式(看該
人員是否有註
解使用方式)

In [4]:

1 np.array()

Docstring:

array(object, dtype=None, copy=True, order='K', subok=False, ndmin=0)

Create an array.

In [5]: 1 import numpy as np

```
>>> np.array([1, 2, 3], ndmin=2)
array([[1, 2, 3]])
```

Type provided:

```
>>> np.array([1, 2, 3], dtype=complex)
array([ 1.+0.j,  2.+0.j,  3.+0.j])
```

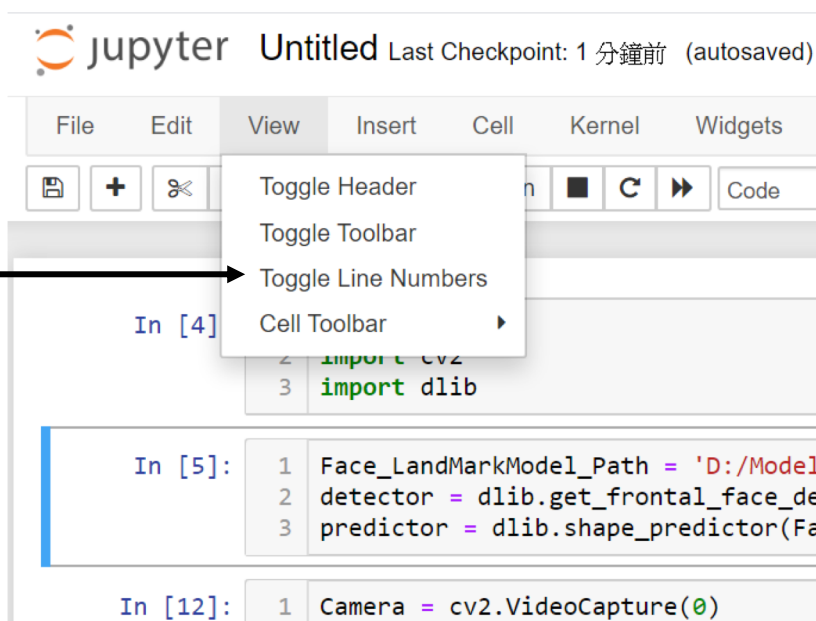
Data-type consisting of more than one element:

```
>>> x = np.array([(1,2),(3,4)], dtype=[('a', '<i4'), ('b', '<i4')])
>>> x['a']
array([1, 3])
```

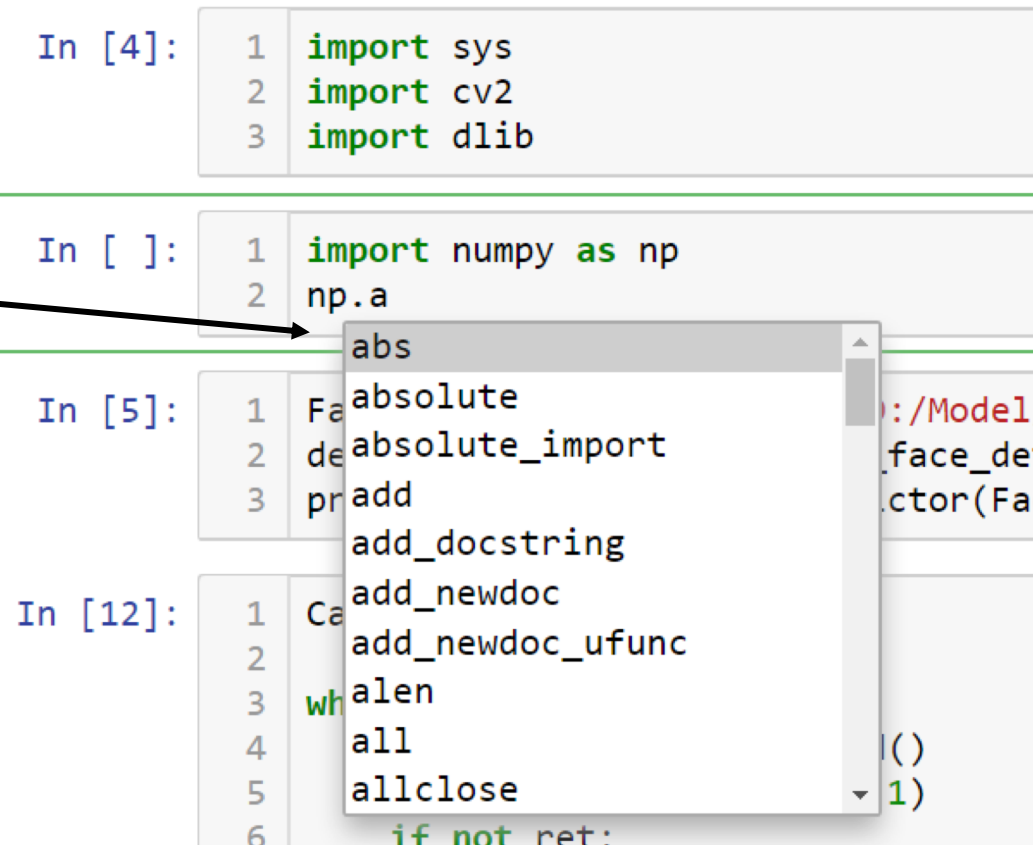
Creating an array from sub-classes:

Convenient

顯示行數Debug
好用



案tab可快速選取



Editor

同時縮排/反縮排:
在要縮排的區域反白
後按TAB會同時縮排
Shift + TAB 則可以同
時反縮排

```
2  
3 while True:  
4     → ret, frame = Camera.read()  
5     → frame = cv2.flip(frame, 1)  
6     → if not ret:  
7         → break  
8         face_rects, scores, idx = detector.run(frame, 0)  
9         for i, d in enumerate(face_rects):
```

取代:
Esc + F可以取代

Find and Replace

In []:

1	A
2	AA
3	A
4	AA
5	A

Find: Aa . * | AA

Replace: AAA|

2 matches

AA	AAA
AA	AAA

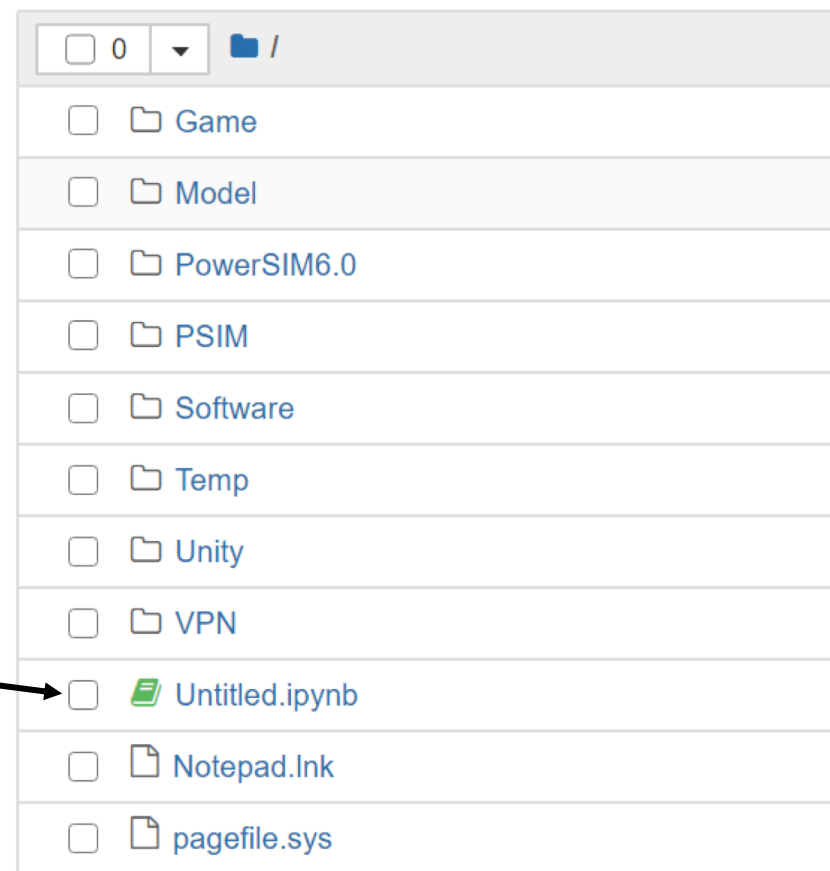
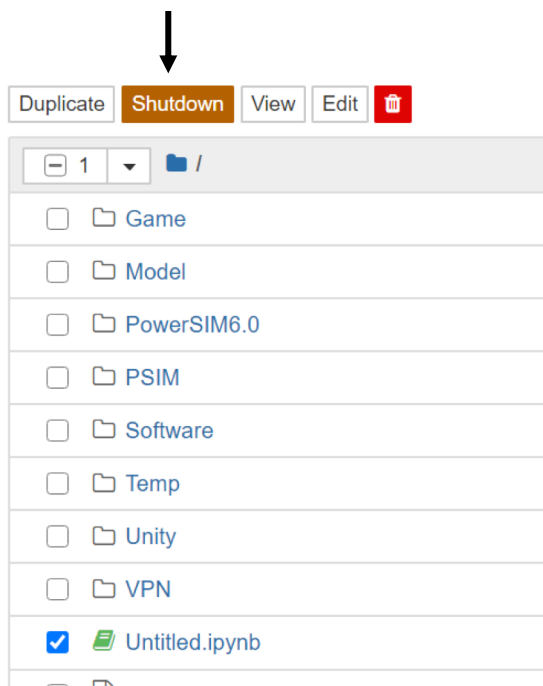
Replace All

In []:

1	A
2	AAA
3	A
4	AAA
5	A

Shutdown

儲存後關掉並不會結束kernel
勾選後選擇shutdown結束kernel



結束kernel後就不會亮著

