

*** 請大家用以下的格式分享共筆:

=====

日期 ~自己的名字:

內容內容內容內容內容內容內容

~其他人的名字:

討論討論討論

=====

09/20 ~葉佩怡

剛開始學寫 python 總會製造一堆bug, 透過 pdb (The python debugger) 可以幫助自己找到bug! 有介紹pdb怎麼使用的網址 ↓

<https://docs.python.org/3/library/pdb.html>

歸納有兩種使用pdb的方法:

1. python3 -m pdb 我寫的python程式.py, 接著用命令代碼去debug
 2. 在程式內先 import pdb, 再利用 pdb module裡的函式來debug
-

09/20 ~王俐迦

初學剛開始練習python看這網址

<https://kopu.chat/2017/01/18/%E4%B8%80%E5%B0%8F%E6%99%82python%E5%85%A5%E9%96%80-part-1/>

覺得Jupyter Notebook 是很不錯用的編譯器

<http://jupyter.org/> 有網路版的編譯器可以拿來練習寫python

9/20~廖婕茹

[Python 入門指南 V2.00]

<https://pydoing.blogspot.com/2014/07/python-guide.html>

↑ 真心推薦這個網址, 裡面有題目也有例題跟答案



變數命名規則

#2個英文字母應避免使用

- 小寫的 l
- 大寫的 O

英文小寫 l 與數字1, 英文大寫O與數字0會搞混

#不能用數字當變數開頭

 合法變數 a x words num 	 合法類別名稱(大寫開頭) MyClass Complex FirstClass FoodExpert
 合法函數 fib() fib2() raw_input() split() 	 合法常數名稱 MAX_OVERFLOW TOTAL

縮排

#Python 程式裡不能隨意縮排, 因縮排是 Python 劃分程式區塊的方式

```
a = 1
print(a)
a = 3
print(a)
```

9/20~吳崇嘉

[Python 資料視覺化呈現, 實作機器學習方法]

http://www.cc.ntu.edu.tw/chinese/epaper/0041/20170620_4105.html



- 1.假如對於Python環境沒有那麼熟悉, 且是初學的話建議使用網址內的開發環境 **Anaconda**。
- 2.本文也會教大家用**Anaconda**, 從零一步一步將開發環境建構。
- 3.**Anaconda**除了寫python code 外, 還能方便建構數據使其視覺化。

9/20~李宜浩

如果過去有寫程式的經驗, 但沒有寫過Python, 我滿推薦這一系列的短片:

https://www.youtube.com/watch?v=Z1Yd7upQsXY&list=PLBZBJbE_rGRWeh5mlBhD-hhDwSEDxogDg

每部影片都不會太長, 以function為例, 花大約十分鐘就可以簡單地了解Python怎麼去定義函式。我覺得對於有寫過程式的人來說可以很快入門, 不過如果沒寫過程式的話可能就沒辦法從影片中學到觀念, 因為影片的內容著重在那些寫程式會用到的小工具(variable, if else, for loop...)在Python上的實作

9/20~李少琪

之前已經有使用過Python寫過程式, 現在在學習如何資料視覺化

我推薦這個網站給大家:<https://plot.ly/python/>

這網站裡有許多可視化的例子, 可以選擇適合資料或自己喜歡的來實作看看><

9/20~盧柏翔

找到一個python可用的開發環境, Visual Studio Code (教學:

<https://oranwind.org/python-vscode/>)

他是一套跨平台 (Windows、OS X、Linux) 的 IDE

9/20~葉文智

成為Python數據分析達人的第一課的第一課

<http://moocs.nccu.edu.tw/course/123/intro>

1. Anaconda的安裝教學

2. 如何使用jupyter notebook

3. python 一些基本的用法(迴圈、條件判斷、套件的使用.....等)

4. 簡單的預測分析、機器學習教學

適合剛入門之新手進行學習，每篇影片都滿短的非常容易吸收。

9/20~林彥如

Coursera的Programming for Everybody (Getting Started with Python)是設計給完全沒有coding經驗的人學習，內容對於有coding經驗的人會太簡單，不過可以學到一些常用的python語法，開課的密西根大學教授也寫了一本書，可在不清楚語法的時候當工具書查詢。

課程網址: <https://zh-tw.coursera.org/learn/python>

電子書Python for Everybody – Exploring Data Using Python 3 by Charles R.

Severance: http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf

9/20~張庭華

<https://docs.python.org/3/index.html>

官方的document就提供很多基本的操作，其中還有簡單的例子，若遇到無法解決的問題也可以上stack overflow上找尋答案，通常80%以上的問題都有人提出過了

9/20~顏均惟

Python讀檔

利用open()函式

`f = open(file_name, access_mode, buffering)`

例如：

`f = open('abc.txt', 'r')`

其中的access_mode表示文檔打開的模式

r - 只用於讀取

rb - 以二進制格式打開且只用於讀取

W - 用於寫入

a - 用於新添加至已有內容之後，如果該文檔不存在則創建新文檔進行寫入

而讀取的方式例如

`line = f.read()` 其中

`read()` - 讀取整個文檔(存成string)

`readline()` - 讀取一行(存成string),所以可以搭配while達到讀取所有行

`readlines()` - 一次性讀取文檔所有行(存成list),所以可以搭配迴圈再操作list

其他還有一些像是`getline`、`split`等操作

可以再參考資料來源：

<https://hk.saowen.com/a/1bed9d29bccc1de483d0bbb36a91def7348a27628ed7d2d33fa464a446a114c3>

<https://www.jianshu.com/p/a672f39287c4>

9/21~鄭富澤

python套件

python 有支援許多套件，像是Numpy支援非常多的科學計算，包含矩陣運算、線性代數、傅立葉轉換等，Matplotlib可以畫出各種圖型，scikit-learn是可用於機器學習的套件，包含內建的分群分類計算、回歸、統計等功能，可用pip install 或conda install來安裝套件。

9/21~李宛臻

python 網路爬蟲程式的教學

<https://pala.tw/python-web-crawler/>

什麼是爬蟲??

<https://pala.tw/what-is-web-crawler/>

爬蟲程式可以幫助我們從網頁上抓需要的資料下來，因為我們這組應該需要相關的技巧，因此我把它放上來當個紀錄也讓有興趣的人可以用看看

另外記錄一下我比較常用的python編譯器

1. Jupyter notebook

網頁版的編譯器，非常輕巧好用。而且各個block各自run，對於測試自己的寫法對不對很有幫助。

2. Pycharm

安裝

https://pygame.hackersir.org/Lessons/01/PyCharm_install.html

安裝函式庫

<https://ithelp.ithome.com.tw/articles/10194620?sc=iThelpR>

Pycharm的介面比較適合專案開發，而且還可以設定break point~~~

缺點是比較吃資源

3. Visual Studio Code

<https://ithelp.ithome.com.tw/articles/10195139?sc=iThelpR>

跟visual studio是同源(微軟的孩子)，不過我也剛開始用而已，通常是用這個編輯我的code，然後用cmd跑。另一個好處是，也可以拿來寫其他語言。

最後說一下用anaconda的好處：虛擬環境

<http://yenlung-blog.logdown.com/posts/257347-anaconda-in-the-virtual-environment-and-package-management>

我們可以使用conda指令，創建不同的虛擬環境，這樣就不用了一個專案裡面，包進所有函式庫(也就是客製化的環境)。

基本的建立虛擬環境指令：

```
conda create python=3 -n myEnv
```

```
activate myEnv
```

```
deactivate myEnv
```

9/21~蘇于安

如果要從頭開始學習Python推薦莫煩的影片

教學影片每部都不長而且都是中文發音

整體來說滿好吸收的

<https://morvanzhou.github.io/tutorials/python-basic/>

另外也推薦一本入門書

"精通 Python: 運用簡單的套件進行現代運算"

從最基本的語法開始教學

相當淺顯易懂

9/21 ~ 吳慈娟

Pycharm編輯器: 有中斷點及多個程式檔案管理的功能, 也可快速在多個檔案或library中切換, 可有效率地開發較大程式, 詳細的教學可參考影片如下

- 基礎功能: <https://youtu.be/mlQkSifzVb8>
 - 使用中斷點: <https://youtu.be/oSUck7QZNNE>
-

9/21 ~ 謝立凡

https://nthu-datalab.github.io/ml/labs/01_Scientific-Python-101/01_Scientific-Python-101.html

最速入門python + numpy + matplotlib

就一個頁面, 究極懶人包

9/21 ~ 林宜謙

<http://tech-marsw.logdown.com/blog/2014/09/03/getting-started-with-python-in-ten-minute>

python 10分鐘入門

有程式基礎參考這篇文章, 可以很快搞懂python與其他語言不同之處

- python定義函式用`def`開頭, 同樣以冒號`:`結尾, 還有縮排。(不同於c++, 好常
 - 忘記要冒號, 也不像c++, python用縮排表示同一個block)
 - Class的初始化函式是由兩條底線包含著`__init__`做宣告。
 - `from time import time` `"""從time檔案插入time() 函式, 使用time() 前不需要加檔名。`
 - 其中python字串內建的分割函式`string.split()`很好用, 可以將字串依指定的字元(字串)切割
 - 字串可用雙引號`"`或用單引號`'`來進行標示
 - dictionary像是hash-table一樣有一個key對應一個變數
-

9/21 ~ 陳友信

我利用anaconda來訓練python能力, 這是我的環境安裝筆記可以參考

[Anaconda_5.2_64-bit \(Python_3.6\) 一步一步安裝圖文操作](#)

其中"Jupyter"與"Spyder"是我目前使用過的, 以下附上比較詳細的介紹

[Jupyter Notebook介紹](#)

[Spyder. 維基百科](#)

另外, 這是我利用python爬蟲的範例, 也可以參考

[Python_利用Jupyter Notebook抓取'股票個股日成交資訊' 輸出成csv檔 一步一步圖文操作](#)

9/24 ~ 張乃琪

<http://www.runoob.com/python3/python3-string.html>

這個教學網站有很多整理好的python code寫法, 適合給已經會寫code但是對python語法不熟的大大們使用

9/25 ~劉宗晁

<https://ithelp.ithome.com.tw/articles/10187702>

這個教學網站主要是在介紹tensorflow套件做深度學習的運算,

安裝tensorflow:

(1)可運用pip3 install tensorflow

(2)運用Anaconda 環境進行開發

程式:

主要是先列出一個簡單的 $ax+b$ 函數, 再讓電腦用監督式學習的方式進行演練

最後再由tensorflow套件進行運算處理

當loss達最小化值時, 運算才會終止。

而常用的套件如下:

名詞	定義
Graphs	建立運算元
Sessions	執行運算
Tensors	資料
Variables	變數
Feeds	資料輸入
Fetches	資料輸出

9/26 ~宋泉儒

<http://moocs.nccu.edu.tw/course/123/intro>

第一周從基本的變數或字串開始說明, 淺顯易懂。第二周學習迴圈以及LIST的切割。

<https://morvanzhou.github.io/>

這也是有許多基礎的語法，爬蟲教學，也有套件的應用，如:tensorflow。

9/26~顏均惟

若要處理文字資料，則必須將文字數值化

其中一個方式為word2vec

將文字轉成向量，進而再做之後的處理

詳細word2vec可以參考此篇

<http://cpmarkchang.logdown.com/posts/773062-neural-network-word2vec-part-1-overview>

而如何實作word2vec

python有一個套件叫做gensim

實際範例可參考這篇

他利用gensim訓練出英文維基百科的word2vec模型

<https://textminingonline.com/training-word2vec-model-on-english-wikipedia-by-gensim>

9/27 ~羅大郡

<https://morvanzhou.github.io/tutorials/data-manipulation/np-pd/>

Numpy 和 Pandas的介紹網頁，對於會用到矩陣運算的同學應該會是個不錯用的工具。

9/27~王佑誠

蠻推薦使用anaconda去建立python環境

個人OS系統和windows系統都很方便地完成安裝

可以很方便的安裝python其它套件

像是python-opencv、還有tensorflow等上課之後可能會使用到的東西

只要開啟anaconda的管理器就可以直接安裝

然後再使用spyder去測試即可

9/27~謝璨安

<http://www.learnpython.org/> 學習python的網站，除了文字解釋也有範例練習，能初步了解python基本工具

9/27~吉烜瑾

https://www.edx.org/?gclid=EAlalQobChMI0lqUjfla3QIVS6SWCh3HqwUEEAAYASAAEgKh_d_D_BwE 一個很不錯的學習網站，對Python學習有不同的課程，Python for data science適合較快速的初學者，會介紹很多data science中python的實用技巧，還用machine learning, visualization 等

Python類比Java 真的很方便，everything in Python is object, 不用專門去定義。

9/27~葉鈺萱

https://www.codecademy.com/courses/learn-python/lessons/python-lists-and-dictionaries/exercises/access-by-index-1?action=resume_content_item&link_content_target=interstitial_undefined

新手練習python的好地方

9/27~楊順全

<https://www.learnpython.org/>

此網站不僅適合新手從基本的開始學起，也適合有程式經驗的各位學習進階的知識

<https://www.facebook.com/groups/180708015327157/>

這是他們網站在FB上的社團，可以在社團裡看到世界各國的人提供的資料或最新的資訊

9/27~張家齊

<https://www.youtube.com/watch?v=lqrFMiJfHBU&t=313s>

介紹基本的爬蟲概念，透過json將抓取資料做整理，並寫入至csv檔

<http://moocs.nccu.edu.tw/media/17888>

pandas可用來處理時間序列資料以及結構化資料，此教學簡單利用datereader讀取股票資料，並用matplotlib以圖表方式呈現

<https://www.youtube.com/watch?v=jV6eHoLzD2E>

爬蟲遇到限制時，如何加以突破，將程式的思路完整呈現出來

9/27~李宛臻

python 物件導向語法

python 雖然偏向script語言，不過物件導向功能也是支援的，而且很好用

<https://ithelp.ithome.com.tw/articles/10161285>

不熟悉物件導向的同學，可以參考下面這個網址：

<https://blog.miniasp.com/post/2009/08/27/OOP-Basis-What-is-class-and-object.aspx>

簡單介紹一下python class 建立的方法

```
class Human:
```

```
    def __init__(self, h=0, w=0):
```

```
        self.height=h
```

```
        self.weight=w
```

```
    def BMI(self):
```

```
        return self.weight / ((self.height/100)**2)
```

__init__ 讓我們定義類別成員，當我們創立這個類別的物件時，每個物件都會包含

__init__ 裡面的成員。

然後我們會用def去創造屬於這個class的方法，創建物件之後就可以呼叫該方法了。

這邊要注意的是self，是一個指向物件本身的reference。詳細作用可以看這裡：

<https://freedomknight.me/guan-yu-python-zhong-de-self/>

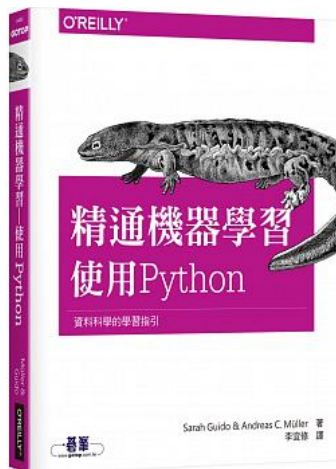
另外在第一個連結裡面也可以找到其他如何python使用物件導向的功能，例如繼承。歡迎大家去看看。

9/28~張庭華

<https://ithelp.ithome.com.tw/articles/10161708>

介紹Python的讀檔功能，不同的模式分別有不同的涵義，取決於使用者的需求。

9/28~葉文智



簡單介紹 Python 使用與機器學習(監督式、非監督式、演算法.....等)
新手容易讀懂。

9/28~李宜浩

Python之所以好用是因為有很多工具都已經幫你寫好了，不需要自己去implement那些function。其中我覺得對於新手來說很值得學習的就是NumPy，NumPy是Python中很重要的一個模組，特別是在處理大型資料的時候，他強大的矩陣運算工具，可以大大地提升處理數據的效率。

這邊提供大家NumPy官方的document：

<https://docs.scipy.org/doc/numpy/user/quickstart.html>

雖然官方的文件可能閱讀起來會比較生硬，不過我覺得它的內容寫得很清楚，有很多實際的例子去說明NumPy那些強大的功能該如何使用。有時間的話可以把文件列的功能都試一次看看，功力會瞬間大增哦

9/27~Phu:

I have been studying python through videos from YouTube, especially, sequences of videos from Siraj Raval which supplied knowledge not only about Python but also Artificial Intelligence: <https://www.youtube.com/channel/UCWN3xxRkmTPmbKwht9FuE5A/playlists>
Also, according to online websites such as Github, Udacity and Kaggle, I learned and practiced advanced algorithms to solve real projects.

There are several platforms running python projects such as Jupyter notebook, Charm and Spider which can be easily downloaded and installed on the Internet.

9/28~蔡晨鈴:

1.jupyter 使用說明:

<https://medium.com/@yehjames/%E8%B3%87%E6%96%99%E5%88%86%E6%9E%90-%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-%E7%AC%AC1-2%E8%AC%9B-jupyter-notebook%E4%BB%8B%E7%B4%B9-705f023e3720>

筆記: 1. Shift + tab ->可以幫助你看到函式的說明, 不需要再去google 相關的文件

2.Esc+h-> h是help的意思, 能列出所有的快捷鍵

3.Shift + enter ->執行這個cell

4.Esc + d,d 刪除目前的cell

2.python基礎學習:

<http://www.runoob.com/python/python-basic-syntax.html>

10/1~吉烜瑾:

$P(H|X)=P(X|H)/p(H)P(X)$

Naive Bayies Classifier在整理, 預測數據方面結合python有蠻多的應用
將數據train, 建模, 之後再去test;

但python結構也相對較為複雜, 多為深入machine learning的探討;

這種方法可以處理多類別問題, 但也麻煩在於對於input data的方式比較敏感

10/2 ~陳友信

這裡附上我實際執行Python的一些簡單範例與說明所撰寫的文章
讓初學者能快速上手Python~

[Python_常用基本指令與資料型態範例說明 一步一步圖文操作](#)

10/2 ~陳昭和

和大家分享一些做數據分析時, 非常好用的概念和語法。如果適當的利用coding的技巧, 可以提升coding效率, 和縮短程式碼。

單行 List Comprehension

```
1 numbers = [1, 2, 3, 4, 5]
2
3 doubled_odds = []
4 for n in numbers:
5     if n % 2 == 1:
6         doubled_odds.append(n * 2)
7
8
~
```

每次需要定義某種列表時都要寫 for 循環是很冗長的，好在 Python 有一種方法可以用一行程式碼解決這個問題。此語法可能有點難以理解，但是一旦熟悉了這種技巧，就會經常用到它。

Lambda 函數

coding的過程中經常為了實現最後的功能，創建一個又一個階段性的函數，這些函數往往就只用一兩次。程式碼變得很繁雜。這時候可以使用Lambda 函數解決此問題！

Lambda 函數用於在 Python 中創建小型的，一次性的和匿名的函數。

基本上，它們可以讓你「在不創建新函數的情況下」創建一個函數。

lambda 函數的基本語法如下：

lambda arguments: expression

請看下面的簡單示例

```
def max(m, n):
    return m if m > n else n
print(max(10, 3)) # 顯示 10
```

可以改寫為

```
max = lambda m, n: m if m > n else n
print(max(10, 3)) # 顯示 10
```

Pandas Apply

apply 類似於 map 函數，不過它是用於 Pandas DataFrames，或者更具體地說是用於 Series 的。

Apply 會根據你指定的內容向列或行中的每個元素發送一個函數。

[apply應用可以參考](#)

10/4~顏均惟

矩陣的定義及運算：

利用numpy的array定義矩陣

可進行一些矩陣運算像是加減乘、反矩陣等

如圖例

```

numpy_test.py
1  import numpy
2
3  # 定義矩陣
4  A = numpy.array([
5      ...[1, 3, -2],
6      ...[2, 5, -3],
7      ...[-3, 2, -4]
8  ])
9
10 B = numpy.array([
11     ...[6, 2, 5],
12     ...[3, 4, 3],
13     ...[8, 2, 1]
14 ])
15
16 print(A + B)  # 矩陣相加
17 print(4 * A)  # 乘純量
18 print(A.dot(B))  # 矩陣相乘
19 print(A.T)  # 轉置矩陣
20
21 A_inv = numpy.linalg.inv(A)  # 反矩陣
22 print(A.dot(A_inv))  # 驗證反矩陣
23
24 # 定義單位矩陣
25 Inv = numpy.eye(3, dtype='int')
26 print(Inv)

```

10/4~Phu

A good course for studying:

- + Basic of Python
- + Numpy: fundamental package for scientific computing with Python. It contains
 - a powerful N-dimensional array object.
 - sophisticated (broadcasting) functions
 - tools for integrating C/C++ and Fortran code
 - useful linear algebra, Fourier transform, and random number capabilities.
- + SciPy: python-based ecosystem of open-source software for mathematics, science, and engineering.
- + Matplotlib: a Python 2D plotting library which produces publication quality figures in a variety of hard-copy formats and interactive environments across platforms.

Link: <http://cs231n.github.io/python-numpy-tutorial/>

10/4~ 葉文智

python爬蟲系—— requests和BeautifulSoup庫的基本用法, 簡單了解爬蟲與使用爬蟲。

<https://tw.saowen.com/a/07458f50ff3500921c005039d21a8dce7b2d31063d21bae867c0d69c9379a8c6>

10/23~ 林宜謙

Python命名規則, 各種底線到底代表什麼意義?

<https://aji.tw/python%E4%BD%A0%E5%88%B0%E5%BA%95%E6%98%AF%E5%9C%A8%E5%BA%95%E7%B7%9A%E4%BB%80%E9%BA%BC%E5%95%A6/>
