

自主學習之成果展現

學習科目

科技－資訊科技

計畫名稱

來攻略 python(＜)＞

計畫內容

python 算是一個新的程式語言，對於初學者來將比較簡單易懂(雖然我不這麼認為)，而且可以應用在許多地方，網路爬蟲，遊戲都可以用 python 寫。

這次我們要嘗試的是從 0 開始學習 python，寫出一個簡單的遊戲。

參考的書是 Python 好好玩：趣學電玩遊戲程式設計

裡面有教如何使用 python 還有做簡單的遊戲。

但是我們不會把書所有的部分做完，預計跑 1/2 本書。做 36 小時。

預計進度

週次	內容
1	熟悉 python IDLE
2	練習使用 python IDLE File Editor
3	簡單了解 python 的基本語法
4	認識 import 這個東西
5	print 使用和神奇的\
6	def 的使用
7	random mod and time mod
8	Debugger 的用法
9	流程圖的畫法

10	list
11	in 運算子
12	字串處理
13	字典資料型別，鍵-值對
14	keys() 和 values() 字典用法
15	人工智慧的使用
16	人工智慧的使用
17	串列參照，短路運算求值，None 值
18	擴增指定運算子

計畫執行情形描述

原本計畫要看書學習，但後來由於高一同時有參加台大的資訊之芽，學習 C++，所以高一當時的自主學習課程都用來精進 C++ 了。

但我並未因為這次計畫未成功執行而放棄學習 Python，高三學測完後的我，開始有了許多程式的構思，為了成功寫出腦中的想法，我便開始學習 Python。

這次學習是觀看網路上的程式教學，我參考了許多教學及網站，習得我需要的技巧及能力，整個學習過程的筆記我都記錄在 [hackmd](#) 上，並附上觀看的教學以及參考連結。([hackmd 連結](#))

 發表於 2 個月前

 讚賞
  收藏
  已訂閱

 0 篇留言

Python學習紀錄

字串函式
 數字函式
 List
 Tuple
 Dictionary
 檔案讀入、寫入
 Tkinter學習筆記
 Reference

全部展開
 回到頂部
 移至底部

Python學習紀錄

字串函式

- lower 將字串改為小寫
- upper 將字串改為大寫
- islower 偵測字串是否全為小寫
- isupper 偵測字串是否全為大寫
- index 輸出某字元在字串中的位置
- replace 替換某段字串

數字函式

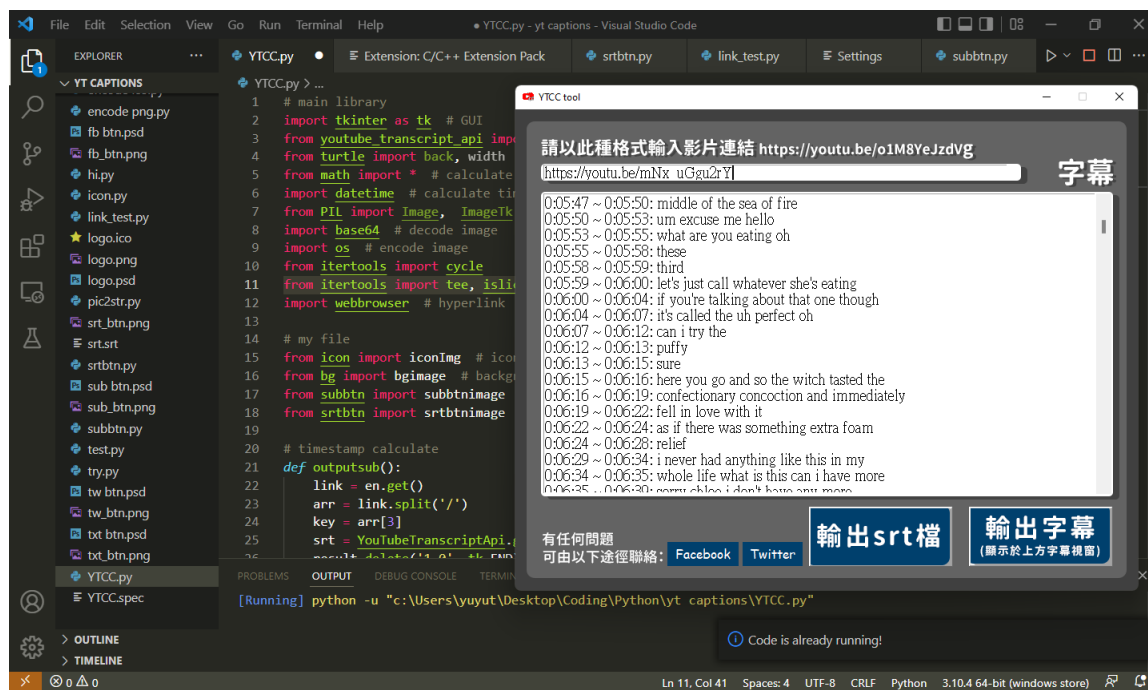
- // 整數除法
- abs() 絕對值
- pow() 指數次方
- max() 回傳最大值
- min() 回傳最小值
- round() 四捨五入

Reference

- [【python】4hours beginner tutorial](#)
- [\[Python教學\]Python List必學實作](#)
- [Python有了串列\(list\)，為何還要有元組\(tuple\)？](#)
- [Python 初學第九講 — 字典](#)
- [Python GUI - Tkinter](#)

依靠過去學其他語言的經驗，我大概花了 2 個小時就熟悉了 python，過去寫 python 程式都是照樣造句，看完教學後我才了解了每行程式碼背後的意義，並自行寫出自己的程式。

實際產出與成效



此為我在自學 python 完後寫的程式，也是這個程式讓我想把 python 學好，在這個程式中我學會了如何使用 tkinter 模組、將 png 圖檔轉成 base64 編碼、利用 api 從 YouTube 取得字幕資料。

會有寫這個程式式因為，我自己本身有在經營翻譯影片的頻道，大多翻譯英文影片，多次翻譯下來，我發現 YouTube 上有許多原文為英文的影片會自動產生英文字幕，準確率頗高，如果能善加利用，便會提高翻譯的效率。

所以我利用 api 取得 YouTube 字幕的資料，再解析、整理資料，並做出可視化界面，讓人便於操作。

```
40 # just sec 2 time
41 def sec2time(sec, n_msec=3):
42     ''' Convert seconds to 'D days, HH:MM:SS.FFF' '''
43     if hasattr(sec, '__len__'):
44         return [sec2time(s) for s in sec]
45     m, s = divmod(sec, 60)
46     h, m = divmod(m, 60)
47     d, h = divmod(h, 24)
48     if n_msec > 0:
49         pattern = '%02d:%02d:%02d.%df' % (n_msec+3, n_msec)
50     else:
51         pattern = r'%02d:%02d:%02d'
52     if d == 0:
53         return pattern % (h, m, s)
54     return ('%d days, ' + pattern) % (d, h, m, s)
55
56 # so that I can get the next element of the list
57 def previous_and_next(some_iterable):
58     items, nexts = tee(some_iterable, 2)
59     nexts = chain(islice(nexts, 1, None), [None])
60     return zip(items, nexts)
61
62 # srt file output
63 def outputsrt():
64     link = en.get()
65     arr = link.split('/')
66     key = arr[3]
67     srt = YouTubeTranscriptApi.get_transcript(key)
68     num = 1
69     srtcy = cycle(srt)
70     with open("srt.srt", "w+") as f:
71         f.write(str(1) + "\n" + "00:00:00.000 --> 00:00:00.001" +
72               "\n" + "start" + "\n\n")
73         for item, nxt in previous_and_next(srt):
74             if item['text'] == "[Music]" or item['text'] == "[Laughter]":
75                 continue
76             else:
77                 num = num+1
78                 begin = str(sec2time(item['start']))
79                 end = str(sec2time(item['start']+item['duration']))
80                 if nxt != None:
```

也學會了較為複雜的函式計算，爲了達成某項功能，我花了三個自訂函式才達成該功能。

以下為程式測試影片，整體功能皆正常運作。

測試影片連結: <https://youtu.be/tC0tWLqr-LQ>

GitHub 原始碼: <https://reurl.cc/qOM9rR>

學習心得

經過這次自行學習 python 的經驗，我更加了解該如何取得自己需要的資料，在遇到問題時自己上網找解決方法，Stack overflow 和 Geeksforgeeks 都在學習時讓我取得了大量的資料。

這次的經驗也讓我知道了註解的重要性，因為整個程式大約花了兩個星期才做完，每次重新開啟檔案，先前的思緒都會消失，需要再重新閱讀去理解自己之前寫的東西，這時註解就顯得十分重要，能讓自己更快回到狀態。

在未來，我想要擁有自己獨立開發一款程式的能力，而這些都是要達成此目標不可缺少的能力，希望大學四年後，各個能力都可以更上一層樓。