

# 深入選課：選課志願序之Python爬蟲實作

王均琦	林沂萱	曾子恬
資工三B	資工三A	資工三A
107502545	107502507	107502506

## 摘要

本文旨在介紹小組開發之選課志願序爬蟲軟體，以及其實作細節。首先，我們將介紹其他大專院校學生開發的選課輔助程式，並分析其優缺點，並藉此凸顯小組開發此專門軟體的必要性。接著，將介紹小組開發選課志願序爬蟲軟體的實作相關技術。最後，將說明本軟體的貢獻及未來展望。

## Abstract

This article aims to give readers a brief profile of our “course-winner” app and its implementation details. Firstly, we’ll give a small introduction about other course selection apps developed by students from other universities. Additionally, we’ll point out each of their advantages and weaknesses in the matter of implementation, by doing so, readers could also see why we need a course selection app specially designed for NCU students. Later, we’ll touch on the implementation techniques about our app. Finally, we’ll wrap up this article by pointing out the app’s contributions and its future prospects.

**關鍵字：**網路爬蟲、Python、HTML

## 1 Introduction

## 2 Related Work

### 2.1 政治大學選課志願序查詢程式

政治大學選課志願序查詢程式是由政治大學資料科學系的同學自行完成的一個選課志願序查詢程式，使用者可以下載壓縮檔安裝程式，查詢政治大學108學年上

下學期一二階選課的志願序回報。執行程式後，可以在初始畫面輸入關鍵字，如圖1所示。如果關鍵字超過一個中間需要以空格分隔，如圖2所示。按下送出鍵後，得到搜尋結果，取得課程相關資料。由於政治大學的選課機制與中央大學不同，所以這套系統無法直接應用在中央大學。但是其圖形介面和功能都是我們可以參考的實作方向。

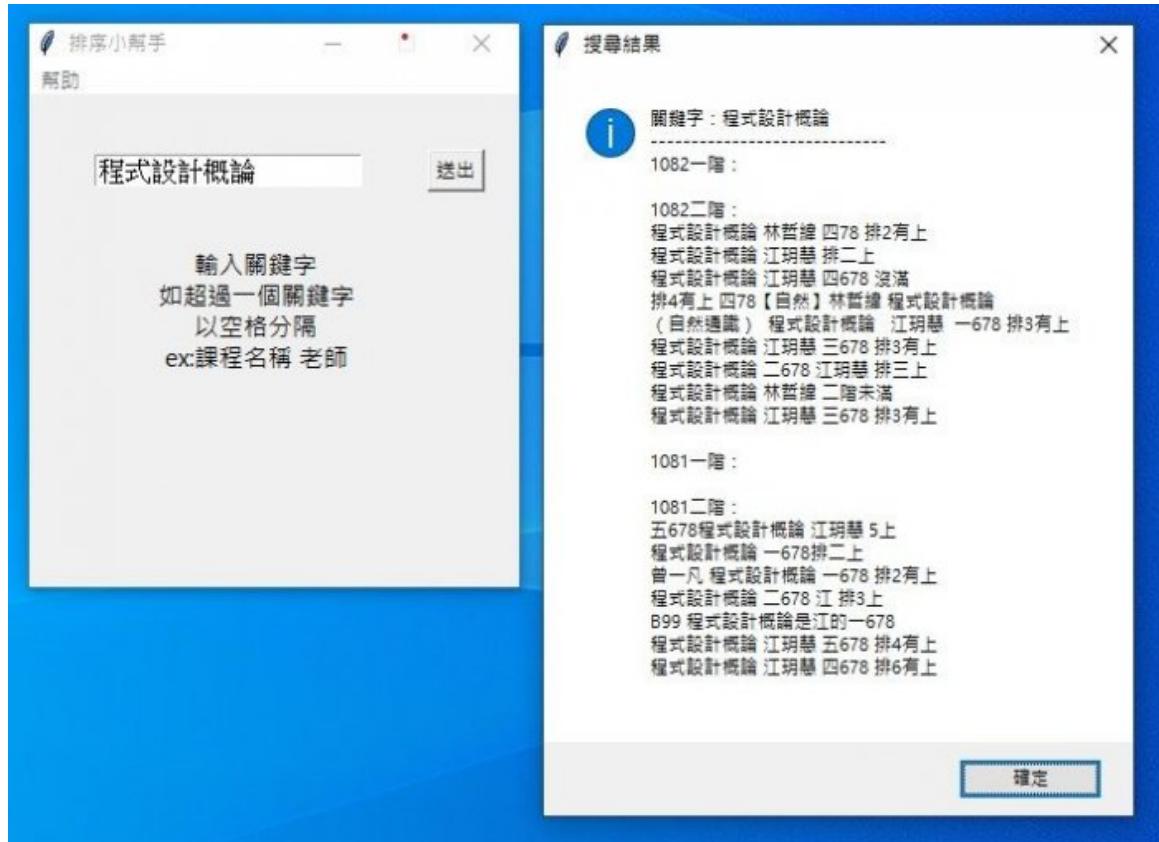


圖 1: 政治大學選課志願序查詢程式之輸入一個關鍵字

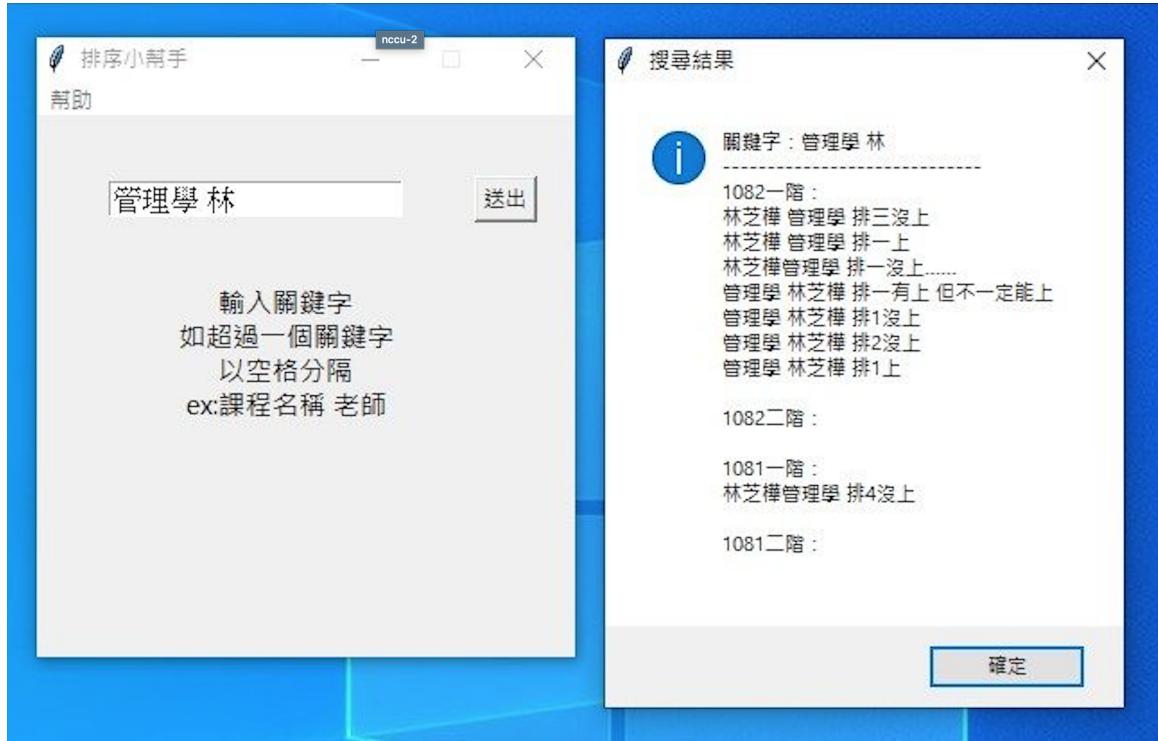


圖 2: 政治大學選課志願序查詢程式之輸入多個關鍵字

## 2.2 成大選課小幫手

成大選課小幫手是一個提供成功大學的同學使用的選課輔助網站。進入網站後，首先，選擇自己所屬的系級與班級（例如：資訊系三年級甲班），如圖3所示。點選確認後，畫面上即顯示課表和必修課程、以及必修課程的相關資訊，包含課名、學分、教師姓名、上課時間、上課地點及備註等，如圖4。而將網頁下拉後，可以勾選擬想要了解的課程，包含系選修和通識課，也可以選擇外系的課程進行找課的動作，如圖5。然而，這個系統主要是方便使用者查詢想要了解的課程以及其基本資訊，協助排課，並沒有實作出我們希望的計算中選率的功能，但仍能作為參考。

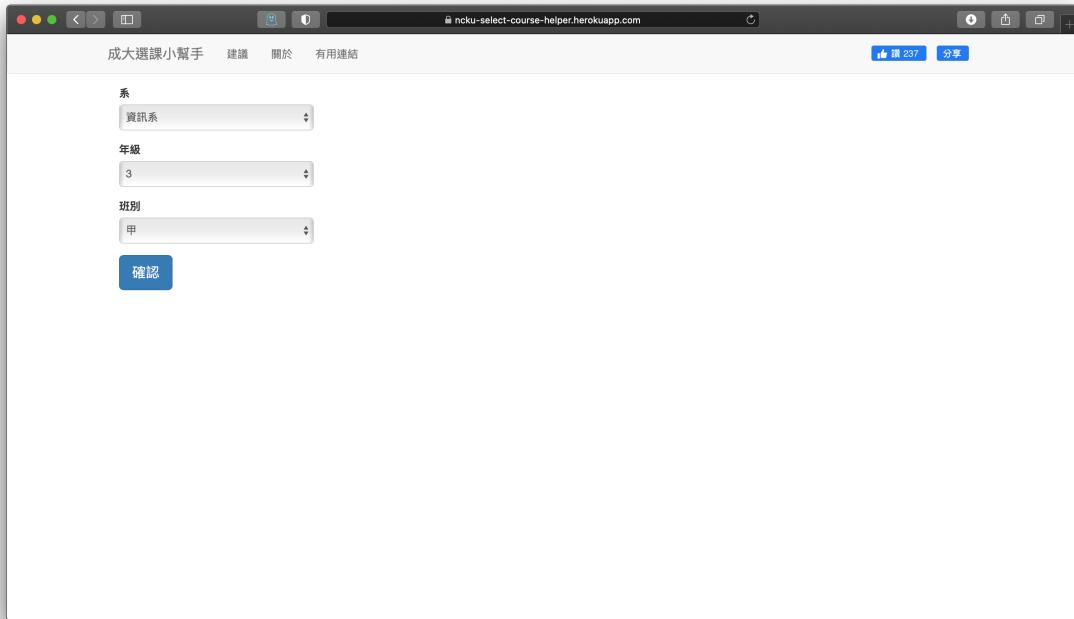


圖 3: 成大選課小幫手起始網頁

A screenshot of the NCKU Course Selection Helper showing course search results. The top navigation bar includes '清除系所設定' (Clear Department Setting). A yellow banner at the top right says '課程人數餘額的資料可能不是最即時的' (Course enrollment data may not be the most up-to-date). The main content area is divided into two sections: a weekly schedule grid and a detailed course list. The weekly grid shows days 1 through 9 with various course names listed. To the right, a table titled '目前修課' (Current Courses) lists three courses: 微算機原理與應用 (含實驗) by 周中平, 演算法 by 謝孫源, and 作業系統 by 張大綱. Each row in the table includes columns for '課名' (Course Name), '學分' (Credits), '教師姓名' (Teacher's Name), '上課時間' (Lecture Time), '上課地點' (Lecture Location), and '備註' (Remarks).

圖 4: 成大選課小幫手必修課查詢

The screenshot shows a web application interface for course selection. At the top left, there are filters for '系上專業選修課' (Major-specific courses) with options for 大一, 大二, 大三 (selected), and 大四; '核心通識' (Core General Education) with options for 基礎國文, 大一英文, 公民與歷史, and 哲學與藝術; and '跨領域通識' (Cross-Disciplinary General Education) with options for 科際整合, 自然與工程科學, 生命科學與健康, 人文學, and 社會科學. On the right, there is a search bar for '外系課程' (Courses from other departments) with fields for '系' (Chinese Dept.) and '年級' (Grade), and a dropdown menu showing '1'. Below the search bar are two buttons: '找課' (Search Courses) and '清空搜尋結果' (Clear Search Results). A large table titled '大三選修' displays course information for five courses:

序號	班別	年級	類別	英語授課	課程名稱	必選修	學分	教師姓名	已選課人數	餘額	時間	教室	備註
+ F7 112	甲乙	3	講義	否	製造資訊與系統概論	選修	3	陳裕民 蘭碧亮 鄭芳田 陳朝鈞 禹佩璇 李家岩*	64	36	[1]5~7	照坤精密儀器大樓 95405	
+ F7 115	甲乙	3	講義	否	作業研究	選修	3	李家岩	28	32	[5]2~4	資訊系館 4261	
+ F7 121	甲乙	3	講義	否	無線通訊網路	選修	3	鄭蕙宗	108	額滿	[4]2~4	資訊系館 4203	甲乙合班
+ F7 122	甲乙	3	講義	否	網頁應用及程式設計	選修	3	盧文祥	70	額滿	[5]2~4	資訊新館 65304	
+ F7 123	甲乙	3	講義	否	多處理機平行程式設計	選修	3	未定	48	2	[2]2~4	資訊新館 65105	

圖 5: 成大選課小幫手課程搜尋

### 2.3 中大選課 Ctrl+F很EAZY!!!

目前大多數中央大學的同學使用的協助選課的工具是Facebook的社團「中大選課 Ctrl+F很EAZY!!!」。社團中提供的功能包括自由發文提問、各個課程評價的歸納版面，而最容易的使用方法是善用Facebook中的Control+F搜尋功能，查詢是否有自己想要了解的課程的相關資訊。然而，在社團中只能搜尋到同學們對各個課程的主觀評價，沒有所有課程的資料，也沒有提供學校官方的人數限制、分發條件及課程綱要等資訊。因此，使用者無法在這個社團中獲得課程中選率等資訊。社團操作畫面如圖6所示。



圖 6: 中大選課 Ctrl+F很EAZY!!!

### 3 Method

本程式的實作目的在於降低計算志願序的困難度，並提供選課排志願序時的參考。學校的選課系統有提供選課名單及其志願序於網站上，為HTML語法，本次實作使用Python的Pandas 進行網路爬蟲，包括套件讀取網頁、處理資料並且計算，最後藉由Python的Tkinter套件建立使用者圖形介面(Graphical User Interface)。

#### 3.1 資料擷取：爬蟲

在緒論中已經介紹了各種網路爬蟲的分類，而本次實作使用的是焦點爬蟲(Focused Crawler)中的關鍵字搜尋方法(Keyword Based Approach)。透過URL位址抓取資料型態為HTML的網站，在Python上使用Pandas套件實行最為容易，匯入套件如圖7，其中把pandas 套件簡稱為pd，而程式碼的實行如圖8，顯示抓取的資料如圖8和圖9。

```
1 import pandas as pd
```

圖 7: 匯入套件Pandas

	學號	姓名	系所	年級班別	性別	選修別	志願序	籤號	選課狀態
0	課程相關資訊		課程相關資訊	NaN	NaN	NaN	NaN	NaN	NaN
1	流水號 / 課號		07016 / CC0318 - *	NaN	NaN	NaN	NaN	NaN	NaN
2	課程名稱/備註	電子科技與近代文明發展	Evolution of Electronic Technolog...	NaN	NaN	NaN	NaN	NaN	NaN
3	授課教師	陳竹一 / Chen, Jwu-E	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	時間/教室	星期四 7 (15:00 - 15:50)   工程二館(電機館大樓) 124	星期四 8 ...	NaN	NaN	NaN	NaN	NaN	NaN
5	選修別		選修	NaN	NaN	NaN	NaN	NaN	NaN
6	學分		2	NaN	NaN	NaN	NaN	NaN	NaN
7	全/半		半	NaN	NaN	NaN	NaN	NaN	NaN
8	授課語言		國語	NaN	NaN	NaN	NaN	NaN	NaN
9	採用密碼卡		部份使用	NaN	NaN	NaN	NaN	NaN	NaN
10	人數限制		120	NaN	NaN	NaN	NaN	NaN	NaN

圖 8: Pandas 讀取HTML與輸出df[0]表格

	學號	姓名	系所	年級班別	性別	選修別	志願序	籤號	選課狀態
0	10*****	000	光電科學與工程學系	4	男	選修	1	0	待分發
1	10*****	000	資訊工程學系	4-2	男	選修	1	0	待分發
2	10*****	000	中國文學系	4	女	選修	3	0	待分發
3	10*****	000	英美語文學系	4	女	選修	3	0	待分發
4	10*****	000	英美語文學系	4	女	選修	1	0	待分發
...	...	...	...	...	...	...	...	...	...
631	10*****	000	生命科學系	1	女	選修	3	0	待分發
632	10*****	000	生命科學系	1	女	選修	1	0	待分發
633	10*****	000	生命科學系	1	男	選修	2	0	待分發
634	10*****	000	生命科學系	1	男	選修	1	0	待分發
635	10*****	000	生命科學系	1	男	選修	3	0	待分發

636 rows × 9 columns

圖 9: 輸出讀取的df[1]表格

藉由把讀取的資料輸出，得知df[0]為課程相關資訊，如圖8，而df[1]為選課名單，如圖9，抓下來的資料型態為Pandas的Dataframe，命名為df，總共有兩個表格，df[1]有9個欄位，636筆資料，可以提供後續的資料處理。

### 3.2 處理資料

為了計算填前三志願序的中選機率，我們需要擷取這門課的人數限制，而如果這門課沒有人數上限，也就是顯示「無」的時候，就預設上限為999人，如圖8的第一、二行。如果輸入多門課，則需要擷取課程名稱加以辨識，而其他課程相關資料，為了方便瀏覽，經過評估，就不特別顯示。人數上限儲存在變數max，課程名稱儲存在name字串，如圖8所示。

```

1 if df[0].loc[10]['姓名']=='無':
2     max=999
3 else:
4     max=int(df[0].loc[10]['姓名'])
5 name=df[0].loc[2]['姓名']

```

圖 10: 程式碼擷取人數限制與課程名稱

### 3.3 計算

從顯示df[1]的圖9得知，我們需要的資料在「志願序」欄位。使用value\_counts()函數可檢視資料中各個欄位類型的數量，套用在「志願序」欄位輸出如圖11下半部。

```

1 print(df[1]['志願序'].value_counts())
2
3 one=df[1]['志願序'].value_counts()[1]
4 if (len(df[1]['志願序'].value_counts())>2):
5     two=df[1]['志願序'].value_counts()[2]
6 else:
7     two=0
8 if len(df[1]['志願序'].value_counts())>3:
9     three=df[1]['志願序'].value_counts()[3]
10 else:
11     three=0

1    439
2    56
3    44
4    31
5    21
6    10
7    8
9    6
8    6
11   3
10   2
14   2
17   2
23   1
12   1
13   1
15   1
16   1
56   1
Name: 志願序, dtype: int64

```

圖 11: 志願序與其人數的程式碼和輸出

參數one、two和three各自儲存填前三志願序的人數，為了避免沒有人選填前三

志願序的狀況而導致讀取錯誤，我們加上了判斷式，如果有人選第二、三志願，才抓取資料，不然預設參數two、three為0。

接著計算你選填前三志願的機率，基本公式如公式1，百分比為人數限制除以選填此志願序的人數加一，再乘以100，加一的原因為要自己算進去，乘以100以取得百分比，最後四捨五入去除小數點，方便瀏覽。

$$\text{percentage} = \text{Round}(\max / (\text{one} + 1) * 100) \quad (1)$$

但這個公式只能套用到志願序一，而且第一，如果選填志願一的人數低於人數上限，算出來的值會超過100，第二，如果第一志願選的人數超出人數上限，那選填接下來志願的人都上不了，所以還要加上判斷式，詳細內容如圖12。

```
1 one_p = round(max/(one+1)*100)
2 if one_p<=100:
3     two_p = 0
4     three_p = 0
5 else:
6     one_p=100
7     two_p = round((max-one)/(two+1)*100)
8     if two_p <=100:
9         three_p = 0
10    else:
11        two_p=100
12        three_p = round((max-one-two)/(three+1)*100)
13        if three_p >100:
14            three_p=100
15 print(name)
16 print('人數限制: '+str(max))
17 print('第一志願: '+str(one_p)+'%')
18 print('第二志願: '+str(two_p)+'%')
19 print('第三志願: '+str(three_p)+'%\n')

電子科技與近代文明發展 Evolution of Electronic Technology and Modern Civilization
人數限制: 120
第一志願: 26%
第二志願: 0%
第三志願: 0%
```

圖 12: 計算填前三志願上的機率的程式碼與輸出

考量志願序一中選的機率，如果選填第一志願選的人數超出人數上限，那第二、三志願上的機率都是0，如果志願序一中選的機率大於等於100%，那就輸出100%，並且以公式1計算志願序二中選的機率，但這次要以人數限制減掉選填第一志願的人數除以第二志願選填的人數加一，再做計算。以此類推。最後以圖8輸入「電子科技與近代文明發展課程」URL作為範例輸出，如圖12。以上程式碼可於此連結檢視、執行：<https://colab.research.google.com/drive/1hyca61nUeyUNU098KPoolAz5gl96BWpM?usp=sharing>。

## 3.4 套入GUI

以上部分為零散的程式碼，因此，我們實作對使用者友善的圖形介面，而Google Colab筆記本不支援使用者圖形介面，所以我們將開發環境轉移到Pycharm IDE，使用Anaconda 解譯器(Interpreter)，而部分程式碼也需要簡化改善，以下分為程式碼精簡和套入GUI兩部分說明。

### 3.4.1 程式碼精簡

將以上程式碼建成一個函式surf()如圖13。其中，把多餘的print()輸出去除，url網址字串為輸入，並以line字串收集輸出並回傳，以供之後顯示於使用者圖形介面。

```
def surf(url):
    df = pd.read_html(url)
    max = int(df[0].loc[10]['姓名'])
    name = df[0].loc[2]['姓名']
    one = df[1]['志願序'].value_counts()[1]
    two = df[1]['志願序'].value_counts()[2]
    three = df[1]['志願序'].value_counts()[3]
    one_p = round(max / (one + 1) * 100)
    if one_p <= 100:
        two_p = 0
        three_p = 0
    else:
        one_p = 100
        two_p = round(max / (one + two + 1) * 100)
        if two_p <= 100:
            three_p = 0
        else:
            two_p = 100
            three_p = round(max / (one + two + three + 1) * 100)
            if three_p > 100:
                three_p = 100
    global line
    line = ""
    line = line + name + '\n'
    line = line + '人數限制：' + str(max) + '\n'
    line = line + '第一志願：' + str(one_p) + '%\n'
    line = line + '第二志願：' + str(two_p) + '%\n'
    line = line + '第三志願：' + str(three_p) + '%\n'
    return line
```

圖 13: surf()函式

而考慮到使用者可能會有一次輸入多個URL網址的需求，因此，我們另外建一個函式surfs()，如圖14所示，以arg\*的形式輸入，即可用一個函式處理多型態的輸入，其實作結果如圖15。

```

def surfs(*aug):
    line = ""
    for many in aug:
        line = line + surf(many) + '\n'
    return line

```

圖 14: surfs()函式

```

1 surfs('https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=07006',
2           'https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=07016',
3           'https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=09009',
4           'https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=09061',
5           'https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=07010')

腦與學習 The Learning Brain
人數限制: 120
第一志願: 100%
第二志願: 100%
第三志願: 86%

電子科技與近代文明發展 Evolution of Electronic Technology and Modern Civilization
人數限制: 120
第一志願: 27%
第二志願: 0%
第三志願: 0%

說一個良心的故事責任消費與生產 A Story about Conscience- Responsible Consumption and Production
人數限制: 50
第一志願: 45%
第二志願: 0%
第三志願: 0%

醫療與健康 Medicine and Health
人數限制: 120
第一志願: 36%
第二志願: 0%
第三志願: 0%

神祕的宇宙 Mysteries of The Universe
人數限制: 120
第一志願: 72%
第二志願: 0%
第三志願: 0%

```

圖 15: surfs()函式使用範例

### 3.4.2 GUI 實作

建置使用者圖形介面，首先我們先匯入Tkinter套件，呼叫時簡化為tk，而messagebox功能需要另外匯入，如圖16。

```

import tkinter as tk # for making GUI
import tkinter.messagebox

```

圖 16: 匯入GUI套件

我們使用tk套件手刻出一個初始介面和一個對話框(Message Box)，並彙整成一個GUI()函式，其完整程式碼如圖17。初始頁面大小為610x250，名稱為「選課超快樂」，其中有一個文字區域給使用者輸入，並且搭配上可互動的滾動式畫面，

方便使用者輸入大量URL網址時的編輯與查看，最後加上一個按鈕，觸發便會執行surfs()函式，將結果顯示在視窗中。

```
44     def gui(): # a function to use GUI
45         window = tk.Tk() # create a window
46         window.title('選課超快樂') # set title of window
47         window.geometry('610x250') # set how big the window is
48         window.configure(background='orange') # set background to orange!
49
50     def execute(): # function to close window and get inputs
51         global url # set global variables
52         url = url_text.get(1.0, "end-1c") # get entry of the user
53         url = url.replace(' ', '')
54         url = url.split("\n")
55         line = surfs(*url)
56         tkinter.messagebox.showinfo("搜尋結果", message=line, default='ok')
57
58     o_frame = tk.Frame(window, width=20) # create a frame
59     o_frame.pack(side=tk.LEFT) # pack to window
60     t_frame = tk.Frame(window, width=20) # create a frame
61     t_frame.pack(side=tk.RIGHT) # pack to window
62     url_label = tk.Label(text='請輸入網址:\n(可換行分段)') # set what to display
63     url_label.pack(side=tk.TOP) # start from left
64     url_frame = tk.Frame(window) # create a frame
65     url_frame.pack(side=tk.TOP) # pack to window
66
67     text_scr = tk.Scrollbar(url_frame)
68     url_text = tk.Text(url_frame, height=5, yscrollcommand=text_scr.set) # where user input
69     text_scr.pack(side=tk.RIGHT, fill=tk.Y)
70     url_text.insert(0.0,
71                     'https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=07006\n'
72                     'https://cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=07007')
73     # set default value
74     url_text.pack(side=tk.LEFT) # start from left
75     text_scr.config(command=url_text.yview)
76     exe_btn = tk.Button(window, text='Go!', command=execute) # a button to execute
77     exe_btn.pack() # pack the button
78     window.mainloop() # loop to continue showing the window
```

圖 17: GUI()函式

最後只要在main呼叫gui函式，就可以順利執行，如圖18。

```
if __name__ == '__main__':
    gui()
```

圖 18: main主程式

### 3.5 結果：GUI 成品 實際操作

我們最後使用Python 打包程式Pyinstaller，把整個程式打包成一個.exe檔，可於此連結下載：<https://drive.google.com/file/d/1hsSgjmX4-bkyGKQXK1KtcoKUFjIUSoX1/view?usp=sharing>。首先到「選課系統」的「課程查詢」欄位尋找課程，在想要計算選填前三志願上的機率的課程右手邊點選「修課名單」，就會跳出如圖19的頁面，複製其URL網址輸入至「選課超快樂」程式頁面的文字區，如圖20。

修課名單 - Google Chrome  
cis.ncu.edu.tw/Course/main/query/byKeywords?courselist=07016

課程相關資訊								
流水號 / 課號	07016 / CC0318 - *							
課程名稱/備註	電子科技與近代文明發展 <i>Evolution of Electronic Technology and Modern Civilization</i>							
授課教師	陳竹一 / Chen, Jwu-E							
時間/教室	星期四 7 (15:00 - 15:50)   工程二館(電機館大樓) 124 星期四 8 (16:00 - 16:50)   工程二館(電機館大樓) 124							
選修別	選修							
學分	2							
全/半	半							
授課語言	國語							
採用密碼卡	部份使用							
人數限制	120							
學生修課明細								
學號	姓名	系所	年級班別	性別	選修別	志願序	簽號	選課狀態
10*****	○○○	光電科學與工程學系	4	男	選修	1	0	待分發
10*****	○○○	資訊工程學系	4-2	男	選修	1	0	待分發
10*****	○○○	中國文學系	4	女	選修	3	0	待分發
10*****	○○○	英美語文學系	4	女	選修	3	0	待分發
10*****	○○○	英美語文學系	4	女	選修	1	0	待分發

圖 19: 選課系統之修課名單網頁



圖 20: 選課超快樂初始頁面

如果有輸入多個URL網址的需求，可以換行分段。最後按下「Go!」按鈕即顯示搜尋結果視窗，如圖21。

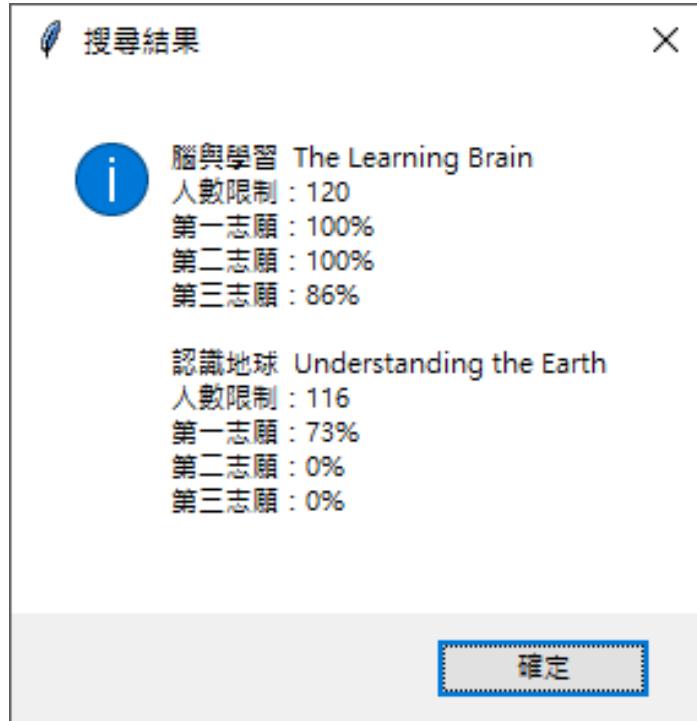


圖 21: 搜尋結果視窗

### 3.6 實作中遇到的問題

基本程式架構的建設並無特別的問題點，困難的是設法讓使用者操作上更方便，因此，我們優化使用者輸入與輸出的介面，使其符合現實需求，遇到的問題以及解決方法如下：

#### 1. 網路爬蟲：HTML資料抓取

一開始不知道如何抓取網頁資料，還使用F12查看HTML原始碼，尋找需要的資料部分，也嘗試過把整個網頁下載下來，但這些資料前處理太繁雜，不便使用者操作。

解決方法：最後找到了Pandas套件read\_html()函數，直接輸入URL網址即可完成HTML抓取。

#### 2. Tkinter：GUI 建構

在Google Colab 網頁上，是以筆記本的方式呈現、執行程式碼，還要自己執行編譯才能使用，輸入資料的地方也不明顯，對於沒使用過的使用者並不方便。

解決方法：使用Tkinter 套件建立GUI並且打包成.exe檔，即可直接執行。

#### 3. Python: \*args

考慮到使用者可能會輸入單一或數個網址，無法預測函式有多少輸入參數，也就無法使用一般函數輸入方式。

解決方法：以`*args` 發送一個非鍵值對的可變數量的參數列表給一個函數，使函式在接受引數時更有彈性。

## 4 Conclusion

深感於選課期間計算各課程中選率的不便性，我們實作了「選課超快樂」作為解決方案。首先，說明開發此應用程式的動機，以及希望可以實現的功能，讓同學們往後在選課期間都能事半功倍。接著，簡介以往被開發出來的選課輔助應用程式與網站，包含政治大學選課志願序查詢程式、成大選課小幫手以及中央大學學生普遍使用的中大選課Ctrl+F很EAZY!!!。藉由比較其優缺點，為實作的方向提供幫助。最後，透由程式碼介紹整個應用程式的實作過程、使用方式以及實作中遭遇的難題與其克服方式。

使用此應用程式，只要輸入選課系統的課程頁面網址，即可計算出所選課程的中選率，讓志願序排序不再是項大工程。未來，我們希望可以加入選課優先順序的功能，並實作手機應用程式提升其便攜性，讓更多人可以使用此選課輔助軟體，以期為中央大學的同學提供一個更為實用的選課工具。