

Covid-19 台灣疫情地圖

組員：林君襄 洪語涵
陳思雅 林采葳

目錄

01

...

選題動機

02

...

研究方法

03

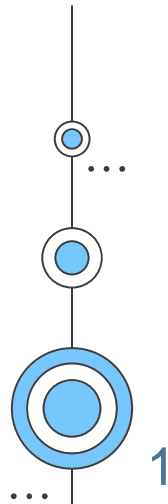
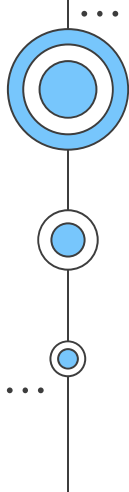
...

問題討論



01

選題動機



動機



啟發

隨著疫情逐漸攀升，確診數來到破萬的數字，以往整理成文字、表格的方法較不易觀看及比較。



思考

是否能夠利用課堂上所學，將資料進行視覺化，用圖表或地圖的方式方便大家觀看。



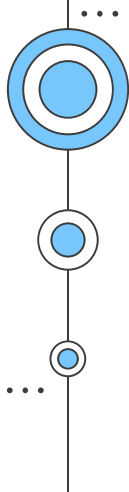
呈現

利用API的key及爬蟲抓取所需資料，再以tkinter設計GUI介面，呈現一個能夠標示最近半年，全台或是各縣市的確診分布地圖介面。

02

研究方法

1. 實驗步驟
2. 程式解說
3. 展示影片



實驗步驟

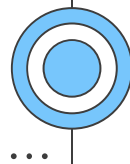
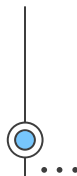
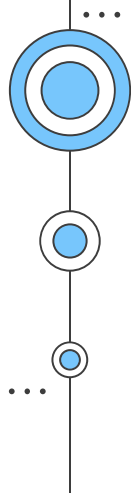
進行
網路爬蟲



資料視覺化



設計GUI介面



實驗步驟

進行網路爬蟲的網頁：[HTTPS://COVID-19.NCHC.ORG.TW/INDEX.PHP](https://COVID-19.NCHC.ORG.TW/INDEX.PHP)





程式解說



匯入所需套件

```
from tkinter import *///匯入Tkinter所有函數及屬性
import pandas as pd
import requests
import re
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg///預設tkagg///畫布及渲染器
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
import numpy as np
///以上為匯入所需套件
```


程式解說

定義函數

```
def printSelection():///定義函數printSelection
    year = []///將year設定為空list
    value = []///將value設定為空list

    url = urllist[var.get()]///將多個網址放在一個列表頁面上顯示
    response = requests.get(url)///使用GET方式下載網頁
    m = re.search("(\\[.*\\])",response.text)
        ///利用search函數搜尋字符串，找到匹配字符串並傳回

    str1 = m.group(1)///存放對比到的第一筆字符串
```



程式解說



提取所需資料

```
#取出圖表中數值
///讀去圖表資料後，將list分割、重組成第二個list，再取出所需資料
list_first = str1.split(",")///切割
value_connect = ":"///用以連接的符號
value_str = value_connect.join(list_first)///連接
value_list = value_str.split(":")///切割
value_len = len(value_list)///長度為value_list

for k in range(value_len):///所需資料所在list的位置
    if k % 6 == 1:
        year.append(value_list[k])
    if k % 6 == 3:
        value.append(value_list[k])

count = int(len(year)/6)
```

程式解說

繪製圖表

```
#圖表
fig = plt.figure(figsize=(9,5),facecolor="gray")///建立畫布
cv = Canvas(win)///圖形繪製
cv.place(x=50, y=250)///圖形繪製位置
plt.subplot(111)///介面分割
plt.clf()///清除圖形
plt.title(citytitlelist[var.get()]+ " Daily confirmed cases", fontproperties="SimHei", fontsize=16)
///圖表標題
|
plt.plot(year,value,'g')///將兩個list放於x及y軸
ax = plt.gca()///獲得坐標軸

ax.xaxis.set_major_locator(ticker.MultipleLocator(count))///設置主刻度
plt.yticks(np.linspace(0, len(value), 7),['0','5000','10000','15000','20000','25000','30000'])///設置y軸

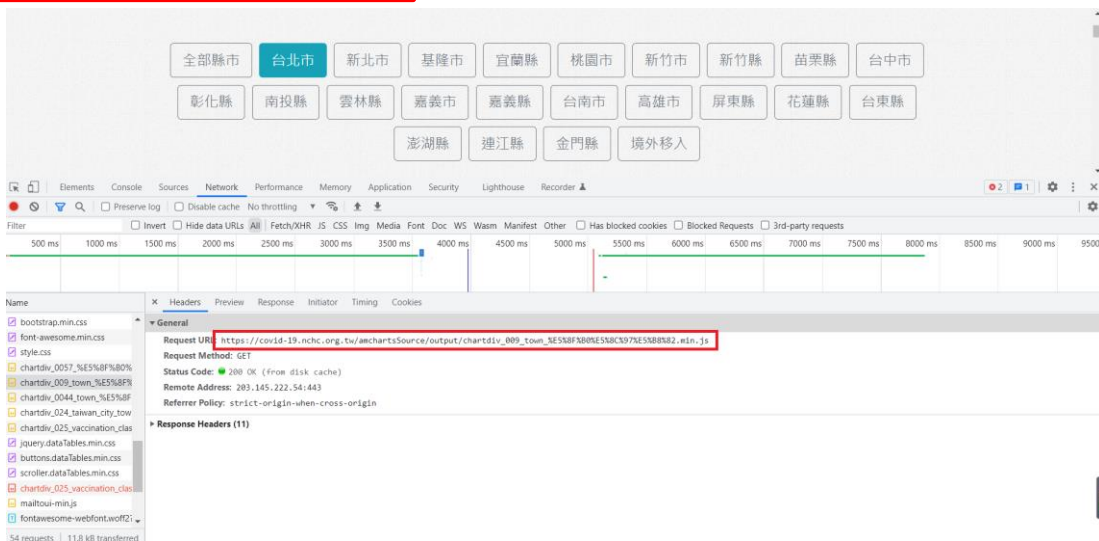
plt.grid()///繪製網格

canvas = FigureCanvasTkAgg(fig, master=cv)///在Tkinter內嵌入matplotlib
canvas.draw()///更新繪圖
canvas.get_tk_widget().pack(side=TOP, fill=BOTH, expand=1)
///放置位置 ///向x、y軸填滿 ///自動擴展
plt.ioff()///關閉互動模式
```

程式解說

取得網址

```
#取得網址
///建立資料讀取URL
url_head = "https://covid-19.nchc.org.tw/amchartsSource/output/chartdiv_009_town."
url_tail = ".min.js"
url_taipei = url_head + "台北市" + url_tail
url_newtaipei = url_head + "新北市" + url_tail
url_keeelung = url_head + "基隆市" + url_tail
url_yilan = url_head + "宜蘭縣" + url_tail
url_taoyuan = url_head + "桃園市" + url_tail
url_hsinchucity = url_head + "新竹市" + url_tail
url_hsinchucounty = url_head + "新竹縣" + url_tail
url_miaoli = url_head + "苗栗縣" + url_tail
url_changchung = url_head + "台中市" + url_tail
url_changhua = url_head + "彰化縣" + url_tail
url_nantou = url_head + "南投縣" + url_tail
url_yunlin = url_head + "雲林縣" + url_tail
url_chiayicity = url_head + "嘉義市" + url_tail
url_chiayicounty = url_head + "嘉義縣" + url_tail
url_tainan = url_head + "台南市" + url_tail
url_kaohsiung = url_head + "高雄市" + url_tail
url_pingtung = url_head + "屏東縣" + url_tail
url_hualien = url_head + "花蓮縣" + url_tail
url_tiatung = url_head + "台東縣" + url_tail
url_penghu = url_head + "澎湖縣" + url_tail
url_lianjiang = url_head + "連江縣" + url_tail
url_kinmen = url_head + "金門縣" + url_tail
url_imported = url_head + "境外移入" + url_tail
```





程式解說



建立視窗

```
#建立視窗
win=Tk()///建立視窗
win.config(bg="lightskyblue")///背景顏色
win.geometry("1000x800")///視窗大小
win.title("期末報告")///視窗標題
win.update()///更新

labell = Label(win, text="各城市Covid-19即時資訊", pady=3, fg="sandybrown",
                ,bg="forestgreen", font=("標楷體",25))///標題(各城市Covid-19即時資訊)
labell.place(x=300, y=10)///標題位置

var = IntVar()///顯示城市
var.set(0)
```

程式解說

城市串列

```
///標題的縣市
citytitlelist = {0:"Taipei", 1:"Newtaipei", 2:"Keelung", 3:"Yilan", 4:"Taoyuan", 5:"Hsinchu City",
6:"Hsinchu County", 7:"Miaoli", 8:"Taichung", 9:"Changhua", 10:"Nantou", 11:"Yunlin",
12:"Chiayi City", 13:"Chiayi County", 14:"Tainan", 15:"Kaohsiung", 16:"Pingtung", 17:"Hualien",
18:"Tiatung", 19:"Penghu", 20:"Lianjiang", 21:"Kinmen", 22:"Imported"}

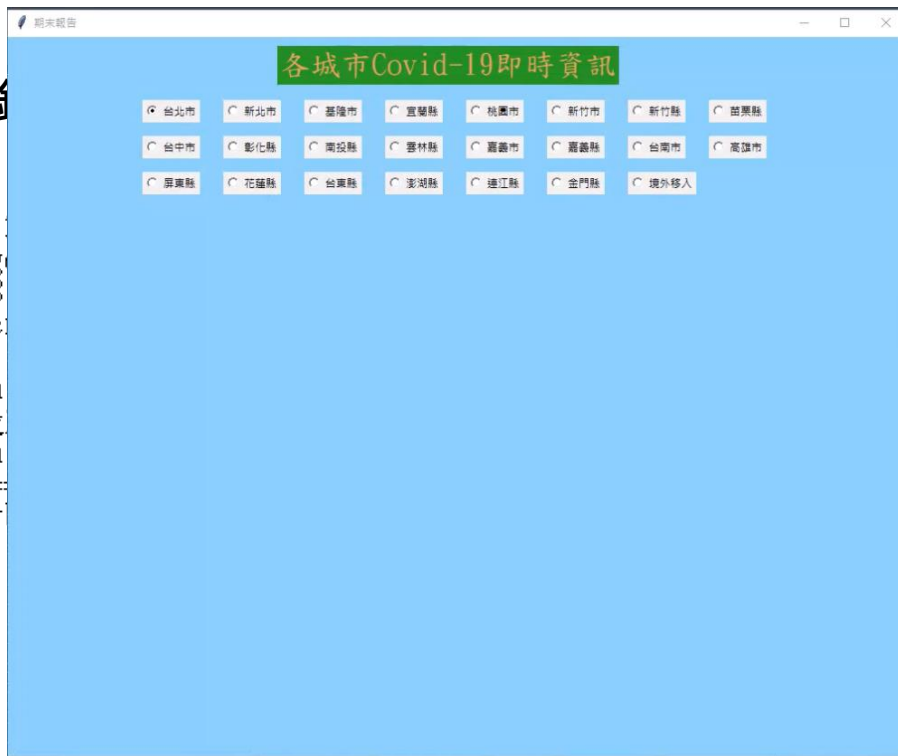
///城市串列
citylist = {0:"台北市", 1:"新北市", 2:"基隆市", 3:"宜蘭縣", 4:"桃園市", 5:"新竹市",
6:"新竹縣", 7:"苗栗縣", 8:"台中市", 9:"彰化縣", 10:"南投縣", 11:"雲林縣",
12:"嘉義市", 13:"嘉義縣", 14:"台南市", 15:"高雄市", 16:"屏東縣", 17:"花蓮縣",
18:"台東縣", 19:"澎湖縣", 20:"連江縣", 21:"金門縣", 22:"境外移入"}

///城市串列
urllist = {0:url_taipei, 1:url_newtaipei, 2:url_keelung, 3:url_yilan,
4:url_taoyuan, 5:url_hsinchucity, 6:url_hsinchucounty, 7:url_miaoli,
8:url_taichung, 9:url_changhua, 10:url_nantou, 11:url_yunlin,
12:url_chiayicity, 13:url_chiayicounty, 14:url_tainan, 15:url_kaohsiung,
16:url_pingtung, 17:url_hualien, 18:url_tiatung, 19:url_penghu,
20:url_lianjiang, 21:url_kinmen, 22:url_imported}
```

程式解說

設定按鈕

```
for i in range(0,10):
    for j in range(0,10):
        n = i * 8
        if(n < len(cityl):
            cityl[n] = cityl[n] + str(i) + str(j)
            rbtem[n] = rbtem[n] + str(i) + str(j)
        else:
            if(n==len(cityl)):
                cityl.append('')
                rbtem.append('')
```



```
command=printSelection)
//呼叫指令def
置
```

The diagram illustrates a network structure. On the left, a node (represented by a blue circle with a white center) is connected to another node (represented by a blue circle with a white center) via a line. A third node (represented by a blue circle with a white center) is connected to the second node via a line. An ellipsis (...) is placed between the second and third nodes. In the center, the text '所有程式碼' (All Code) is displayed. On the right, a node (represented by a blue circle with a white center) is connected to another node (represented by a blue circle with a white center) via a line.

```
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\lan\Downloads\COVID-19\COVID-19.py
final.py x COVID-19.py x
1 from tkinter import *
2 import pandas as pd
3 import requests
4 import re
5 from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
6 import matplotlib.pyplot as plt
7 import matplotlib.ticker as ticker
8 import numpy as np
9
10 def printSelection():
11     year = []
12     value = []
13
14     url = urllist[var.get()]
15     response = requests.get(url)
16     m = re.search("(\\[.\\.\\])", response.text)
17
18     str1 = m.group(1)
19
20     #取出圖表中數值
21     list_first = str1.split(",")
22     value_connect = ":"
23     value_str = value_connect.join(list_first)
24     value_list = value_str.split(":")
25     value_len = len(value_list)
26
27     for k in range(value_len):
28         if k % 6 == 1:
29             year.append(value_list[k])
30             if k % 6 == 3:
31                 value.append(value_list[k])
32
33     count = int(len(year)/6)
34
35     #圖表
36     fig = plt.figure(figsize=(9,5),facecolor="gray")
37     cv = Canvas(win)
```

The image shows a Spyder Python IDE window with a file named `COVID-19.py` open. The script is designed to generate a chart showing daily confirmed COVID-19 cases for various cities in Taiwan. The chart is a multi-locator bar chart with the following characteristics:

- Figure Size:** (9, 5) with a gray facecolor.
- Canvas:** A TkAgg canvas with a width of 50 and a height of 250.
- Subplot:** A single subplot with a width of 111.
- Title:** The title is dynamically generated based on the city selected from a list (e.g., "Daily confirmed cases").
- Y-axis:** The y-axis is labeled 'g' and has major ticks at 0, 5000, 10000, 15000, 20000, 25000, and 30000.
- X-axis:** The x-axis is labeled 'g' and has major ticks at 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 7

所有程式碼

```
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Lin\Downloads\COVID-19\COVID-19.py

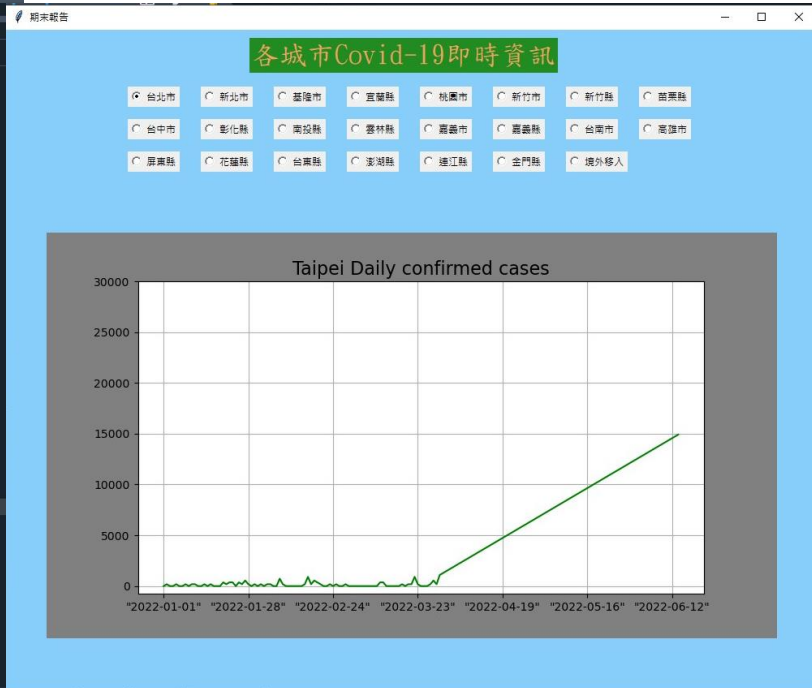
COVID-19.py x
58 #取得網址
59 url_head = "https://covid-19.nchc.org.tw/amchartsSource/output/chartdiv_009_town_"
60 url_tail = ".min.js"
61 url_taipei = url_head + "台北市" + url_tail
62 url_newtaipei = url_head + "新北市" + url_tail
63 url_keelung = url_head + "基隆市" + url_tail
64 url_yilan = url_head + "宜蘭縣" + url_tail
65 url_taoyuan = url_head + "桃園市" + url_tail
66 url_hsinchucity = url_head + "新竹市" + url_tail
67 url_hsinchucounty = url_head + "新竹縣" + url_tail
68 url_miaoli = url_head + "苗栗縣" + url_tail
69 url_taichung = url_head + "台中市" + url_tail
70 url_changhua = url_head + "彰化縣" + url_tail
71 url_nantou = url_head + "南投縣" + url_tail
72 url_yunlin = url_head + "雲林縣" + url_tail
73 url_chiayicity = url_head + "嘉義市" + url_tail
74 url_chiayicounty = url_head + "嘉義縣" + url_tail
75 url_tainan = url_head + "台南市" + url_tail
76 url_kaohsiung = url_head + "高雄市" + url_tail
77 url_pingtung = url_head + "屏東縣" + url_tail
78 url_hualien = url_head + "花蓮縣" + url_tail
79 url_tiatung = url_head + "台東縣" + url_tail
80 url_penghu = url_head + "澎湖縣" + url_tail
81 url_lianjiang = url_head + "連江縣" + url_tail
82 url_kinmen = url_head + "金門縣" + url_tail
83 url_imported = url_head + "境外移入" + url_tail
84
85 #建立視窗
86 win=Tk()
87 win.config(bg="LightSkyBlue")
88 win.geometry("1000x800")
89 win.title("期末報告")
90 win.update()
```

```
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Lin\Downloads\COVID-19\COVID-19.py

COVID-19.py x
85 #建立視窗
86 win=Tk()
87 win.config(bg="LightSkyBlue")
88 win.geometry("1000x800")
89 win.title("期末報告")
90 win.update()
91
92 label1 = Label(win, text="各城市Covid-19即時資訊", pady=3, fg="sandybrown",
93               ,bg="forestgreen", font=("標楷體",25))
94 label1.place(x=300, y=10)
95
96 var = IntVar()
97 var.set(0)
98 citytitlelist = {0:"Taipei", 1:"Newtaipei", 2:"Keelung", 3:"Yilan", 4:"Taoyuan", 5:"Hsinchu City",
99                6:"Hsinchu County", 7:"Miaoli", 8:"Taichung", 9:"Changhua", 10:"Nantou", 11:"Yunlin",
100               12:"Chiayi City", 13:"Chiayi County", 14:"Tainan", 15:"Kaohsiung", 16:"Pingtung", 17:"Hualien",
101               18:"Tiatung", 19:"Penghu", 20:"Lianjiang", 21:"Kinmen", 22:"Imported"}
102
103 citylist = {0:"台北市", 1:"新北市", 2:"基隆市", 3:"宜蘭縣", 4:"桃園市", 5:"新竹市",
104            6:"新竹縣", 7:"苗栗縣", 8:"台中市", 9:"彰化縣", 10:"南投縣", 11:"雲林縣",
105            12:"嘉義市", 13:"嘉義縣", 14:"台南市", 15:"高雄市", 16:"屏東縣", 17:"花蓮縣",
106            18:"台東縣", 19:"澎湖縣", 20:"連江縣", 21:"金門縣", 22:"境外移入"}
107
108 urllist = {0:url_taipei, 1:url_newtaipei, 2:url_keelung, 3:url_yilan,
109           4:url_taoyuan, 5:url_hsinchucity, 6:url_hsinchucounty, 7:url_miaoli,
110           8:url_taichung, 9:url_changhua, 10:url_nantou, 11:url_yunlin,
111           12:url_chiayicity, 13:url_chiayicounty, 14:url_tainan, 15:url_kaohsiung,
112           16:url_pingtung, 17:url_hualien, 18:url_tiatung, 19:url_penghu,
113           20:url_lianjiang, 21:url_kinmen, 22:url_imported}
114
115
116 for i in range(0,3):
117     for j in range(0,8):
118         n = i * 8 + j
119         if(n < len(citylist)):
120             city1 = citylist[n]
```

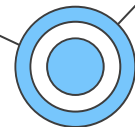
所有程式碼

```
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Li\Desktop\COVID-19\COVID-19.py
COVID-19.py x
92 label1 = Label(win, text="各城市Covid-19即時資訊", pady=3, fg="sandybrown",
93 ,bg="forestgreen", font=("標楷體",25))
94 label1.place(x=300, y=10)
95
96 var = IntVar()
97 var.set(0)
98 citytitlelist = {0:"Taipei", 1:"Newtaipei", 2:"Keelung", 3:"Yilan", 4:"Taoyuan", 5:"Hsinchu City",
99 6:"Hsinchu County", 7:"Miaoli", 8:"Taichung", 9:"Changhua", 10:"Nantou", 11:"Yunlin",
100 12:"Chiayi City", 13:"Chiayi County", 14:"Tainan", 15:"Kaohsiung", 16:"Pingtung", 17:"Hualien",
101 18:"Tiatung", 19:"Penghu", 20:"Lianjiang", 21:"Kinmen", 22:"Imported"}
102
103 citylist = {0:"台北市", 1:"新北市", 2:"基隆市", 3:"宜蘭縣", 4:"桃園市", 5:"新竹市",
104 6:"新竹縣", 7:"苗栗縣", 8:"台中市", 9:"彰化縣", 10:"南投縣", 11:"雲林縣",
105 12:"嘉義市", 13:"嘉義縣", 14:"台南市", 15:"高雄市", 16:"屏東縣", 17:"花蓮縣",
106 18:"台東縣", 19:"澎湖縣", 20:"連江縣", 21:"金門縣", 22:"境外移入"}
107
108
109 urlist = {0:url_taipei, 1:url_newtaipei, 2:url_keelung, 3:url_yilan,
110 4:url_taoyuan, 5:url_hsinchucity, 6:url_hsinchucounty, 7:url_miaoli,
111 8:url_taichung, 9:url_changhua, 10:url_nantou, 11:url_yunlin,
112 12:url_chiayicity, 13:url_chiayicounty, 14:url_tainan, 15:url_kaohsiung,
113 16:url_pingtung, 17:url_hualien, 18:url_tiatung, 19:url_penghu,
114 20:url_lianjiang, 21:url_kinmen, 22:url_imported}
115
116 for i in range(0,3):
117     for j in range(0,8):
118         n = i * 8 + j
119         if(n < len(citylist)):
120             city1 = citylist[n]
121             rbtem = Radiobutton(win, text=city1, variable=var, value=n, command=printSelection)
122             rbtem.place(x = j * 90 + 150, y = i * 40 + 70)
123             if(n==0):
124                 rbtem.select()
125
126 win.mainloop()
```





展示影片



Spyder (Python 3.9)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\Lin\Downloads\COVID-19\COVID-19.py

final.py x COVID-19.py x

```
1 from tkinter import *
2 import pandas as pd
3 import requests
4 import re
5 from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
6 import matplotlib.pyplot as plt
7 import matplotlib.ticker as ticker
8 import numpy as np
9
10 def printSelection():
11     year = []
12     value = []
13
14     url = urllist[var.get()]
15     response = requests.get(url)
16     m = re.search("(\\[.*\\])", response.text)
17
18     str1 = m.group(1)
19
20     #取出圖表中數值
21     list_first = str1.split(",")
22     value_connect = ":"
23     value_str = value_connect.join(list_first)
24     value_list = value_str.split(":")
25     value_len = len(value_list)
26
27     for k in range(value_len):
28         if k % 6 == 1:
29             year.append(value_list[k])
30         if k % 6 == 3:
31             value.append(value_list[k])
32
33     count = int(len(year)/6)
34
35     #圖表
36     fig = plt.figure(figsize=(9,5),facecolor="gray")
37     cv = Canvas(win)
38     cv.place(x=50, y=250)
39     plt.subplot(111)
```



03

問題討論

1. 遇到的問題
2. 修改
3. 實際應用

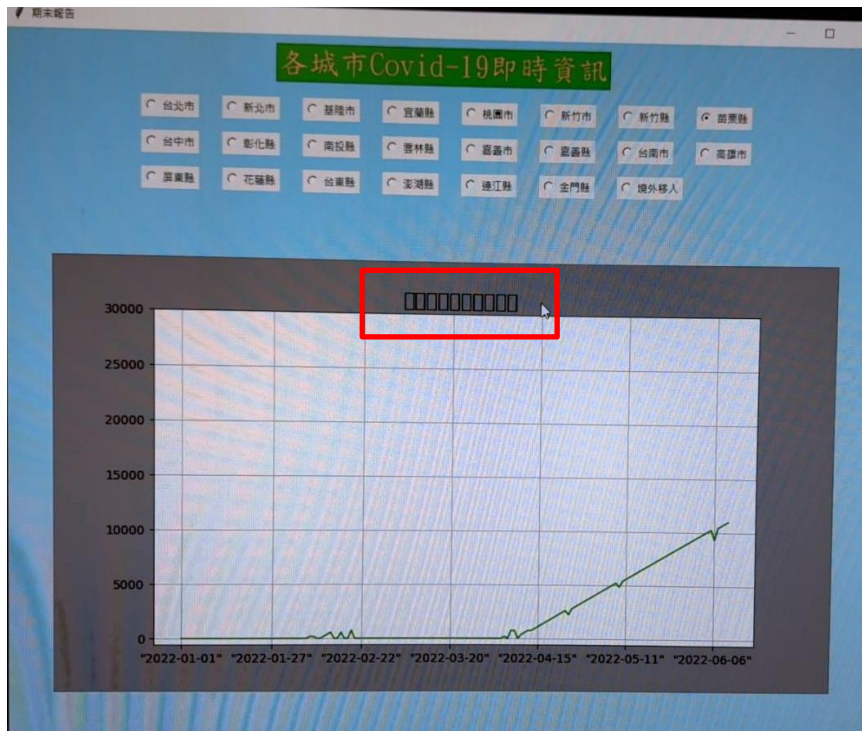
遇到的問題

1. 標題無法正常顯示中文

原程式

```
#圖表
fig = plt.figure(figsize=(9,5),facecolor="gray")
cv = Canvas(win)
cv.place(x=50, y=250)
plt.subplot(111)
plt.clf()
plt.title(citylist[var.get()]+"每日新增確診數", fontproperties="SimHei", fontsize=16)
plt.plot(year,value,'g')
ax = plt.gca()
```

2. 無法結合地圖功能



修改

問題 1 解決方法：將標題改成英文

更正此行

```
#圖表
fig = plt.figure(figsize=(9,5),facecolor="gray")///建立畫布
cv = Canvas(win)///圖形繪製
cv.place(x=50, y=250)///圖形繪製位置
plt.subplot(111)///介面分割
plt.clf()///清除圖形
plt.title(citytitlelist[var.get()]+ " Daily confirmed cases", fontproperties="SimHei", fontsize=16)
///圖表標題
```

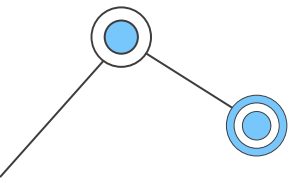
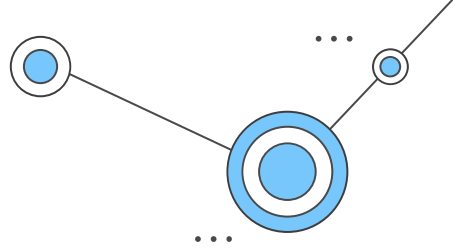
新增此行

```
///標題的縣市
citytitlelist = {0:"Taipei", 1:"Newtaipei", 2:"Keelung", 3:"Yilan", 4:"Taoyuan", 5:"Hsinchu City",
6:"Hsinchu County", 7:"Miaoli", 8:"Taichung", 9:"Changhua", 10:"Nantou", 11:"Yunlin",
12:"Chiayi City", 13:"Chiayi County", 14:"Tainan", 15:"Kaohsiung", 16:"Pingtung", 17:"Hualien",
18:"Tiatung", 19:"Penghu", 20:"Lianjiang", 21:"Kinmen", 22:"Imported"}
```

問題 2 解決方法：將地圖更改為以圖表呈現

實際應用

1. 疫情分布地圖
2. 熱門觀光區人流（預測）圖
3. 車流狀況（預測）圖





謝謝大家