

Lab 4: Python GUI Programming Report

學號：110511254 姓名：徐煜絨

1. 請簡述 Python Tkinter GUI 程式主體架構（如何引入模組.宣告主視窗及物件…等等）

Tkinter 這個名字來自於 Tk interface，他是 Python 裡的標準 GUI 工具包，也是其內建的函式庫，提供使用者實現以程式語言做出希望出現的圖形介面，如這次實驗中的案以及視窗等。

首先要先引用 Tkinter 模組，用 import 直接導入：

```
import tkinter as tk
```

在使用 GUI 介面時，我們通常需要建立一個主視窗：

```
window = tk.Tk()
```

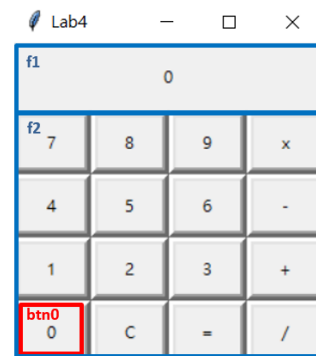
以下是 tkinter 各種元件的使用方式：

	名稱	用途
容器		
1.	window	建立視窗
2.	Frame	建立框架
3.	Labelframe	建立標籤框架
4.	Panedwindow	建立拼布可滑動視窗
執行指令		
5.	Button	按鈕
輸入或輸出		
6.	Label	建立標籤
7.	Entry	建立單行文字輸入
8.	Text	建立多行文字輸入
9.	Scale	建立滑桿
10.	Spinbox	建立數字方塊
11.	Listbox	建立列式方塊
選擇控制		
12.	Radiobutton	選項按鈕(單選)
13.	Checkbutton	核取按鈕(多選)
繪圖與顯示影像		
14.	Canvas	繪圖
15.	PhotoImage	圖像顯示
訊息傳達		
16.	Messagebox	訊息方塊
檔案管理		
17.	Askopenfilename	開檔
18.	Asksaveasfilename	存檔
功能表		
19.	Menu	下拉式選單

這邊用 button 和 label 做解說：

```
btn0 = tk.Button(f2, text='0', borderwidth=5, width=6, height=2,
command=lambda: click('0')).grid(column=0, row=3))
```

這邊宣告了 btn0 這個物件是 button，括號中的設定依序為要顯示在哪個 frame（lab3 因為沒有分 frame，所以直接寫 window）、按鈕的邊框寬度、按鈕寬度、按下去後要執行的動作、用 grid 指定物件的位子（左下角）。



接著是 label 的設定：

```
tk.Label(f1, textvariable=var, height=3).grid(column=0, row=0)
```

label 是用來設定顯示的文字，其中的 f1, height, grid 如前面 button 所述，而 textvariable=var 表示顯示的文字會隨著 var 的變動一起改變，如果不會改變文字，則可以寫 text=var。

再來是物件的排版：

- pack()**：如果用 pack()，則物件預設會由上往下依序排列，如果需要其他排列方式，可以使用 side，e.g. pack(side='left')。Lab4 是用 pack 來設定 frame1 和 frame2 的位子（f1.pack(), f2.pack()），因為 f1.pack() 比 f2.pack() 早出現，根據預設，f1 就會在 f2 上面。
- grid(column=?, row=?)**：這是用在如九宮格的排版上，這次 lab 的 button 就是以 grid 安排位子，像上面 btn0 的範例，藉由設定 column=0, row=3，能讓 btn0 被放到左下角。
- place(x=?, y=?)**：給定絕對座標，並指定 widget 在座標上的特定位子。

最後是 mainloop，他的功能能夠使程式不斷執行，並根據使用者每次按下不同按鍵，做不同動作、給予不同回應。他的程式碼是 `window.mainloop()`。

2. 請解釋 Python 計算機的程式碼(將程式貼上並加上註解)。
(請詳細標注能實現的功能，如：`/0` 能顯示錯誤訊息，`0/`任何數字為 `0`，第一次計算完的結果能繼續做運算，負數計算…等等)

```
import tkinter as tk

def SetValue():
    # label 的字串會在不同 func 中改變，所以用 var
    global var
    # 也是因為字串內容會改變，所以選擇 textvariable = var
    tk.Label(f1, textvariable=var, height = 3).grid(column=0, row=0)

out = []

def click(num):
    global var
    global out
    # 當使用者按 C，會清除所有東西
    if(num=='C'):
        var.set('0')
        clear()
    elif(num.isdigit()):
        num = int(num)
        # 還沒有輸入或是已輸入一組數字加上運算符號
        if(len(out)==0 or len(out)==2):
            out.append(num)
        # 要在第一組數字或第二組數字後面加上其他數字
        # e.g. 讓 21 變成 215
        elif(len(out)==1 or len(out)==3):
            # 特例，可能是負數
            if((len(out)==1)and(out[0]=='-')):
                out[0] = num*(-1)
            elif((len(out)==3)and(out[2]=='-')):
                out[2] = num*(-1)
            print(num)
            # 可能會是多個位數的數字
        else:
            temp4 = int(out[(len(out)-1)])
```

```

        # 在負數最後面加一位
        if(temp4<0):
            temp = temp4*10 - num
        # 在正數最後面加一位
        else:
            temp = temp4*10 + num
        # 轉為 str 存入 out
        out[len(out)-1] = str(temp)

    var.set(out)

# 輸入的是運算符號（不含=，包含負數的負號）
elif(num!='='):
    out.append(num)
    var.set(out)
# 輸入=，開始計算
else:
    calculate()
    var.set(out)

def clear():
    global var
    global out
    out.clear() # 清空 out[]
    var.set('0') # 清空，顯示 0

#加減乘除
def calculate():
    global var
    global out
    result1 = int(out[0])
    result2 = int(out[2])
    if(out[1] == '+'):
        ans = result1 + result2
    elif(out[1] == '-'):
        ans = result1 - result2
    elif(out[1] == 'x'):
        ans = result1 * result2

```

```

elif(out[1] == '/'):
    # 分母不可為零
    if(out[2]==0):
        ans = 'ERROR'
    else:
        ans = int(result1/result2)
# 為了能直接用答案做下一次運算
# 先清空 out，再把 ans 作為 out 的第 0 項
out.clear()
out.append(str(ans))
var.set(out)

if __name__ == "__main__":
    # 創造 tkinter 的物件主視窗
    window = tk.Tk()
    window.title('Lab4')

    f1 = tk.Frame(window)
    f2 = tk.Frame(window)

    # frame.pack() --> 用來指定 widget 在視窗上的位置
    f1.pack()
    f2.pack()

    var = tk.StringVar()
    # 用來設定 var 字串變數的值，一開始顯示 0
    var.set('0')
    SetValue() # 使 stringVar 能成功顯示

    #設定個按鍵位置，包含 0~9, +, -, *, /, =, c

    btn7 = tk.Button(f2, text='7', borderwidth=5, width=6, height=2,
        command=lambda: click('7')).grid(column=0, row=0)
    btn8 = tk.Button(f2, text='8', borderwidth=5, width=6, height=2,
        command=lambda: click('8')).grid(column=1, row=0)
    btn9 = tk.Button(f2, text='9', borderwidth=5, width=6, height=2,
        command=lambda: click('9')).grid(column=2, row=0)
    btn_mul = tk.Button(f2, text='x',borderwidth=5, width=6, height=2,

```

```

        command=lambda: click('x')).grid(column=3, row=0)

btn4 = tk.Button(f2, text='4',borderwidth=5, width=6, height=2,
        command=lambda: click('4')).grid(column=0, row=1)
btn5 = tk.Button(f2, text='5',borderwidth=5, width=6, height=2,
        command=lambda: click('5')).grid(column=1, row=1)
btn6 = tk.Button(f2, text='6',borderwidth=5, width=6, height=2,
        command=lambda: click('6')).grid(column=2, row=1)
btn_sub = tk.Button(f2, text='-',borderwidth=5, width=6, height=2,
        command=lambda: click('-')).grid(column=3, row=1)

btn1 = tk.Button(f2, text='1',borderwidth=5, width=6, height=2,
        command=lambda: click('1')).grid(column=0, row=2)
btn2 = tk.Button(f2, text='2',borderwidth=5, width=6, height=2,
        command=lambda: click('2')).grid(column=1, row=2)
btn3 = tk.Button(f2, text='3',borderwidth=5, width=6, height=2,
        command=lambda: click('3')).grid(column=2, row=2)
btn_add = tk.Button(f2, text='+',borderwidth=5, width=6, height=2,
        command=lambda: click('+')).grid(column=3, row=2)

btn0 = tk.Button(f2, text='0',borderwidth=5, width=6, height=2,
        command=lambda: click('0')).grid(column=0, row=3)
btn_clear = tk.Button(f2, text='C', borderwidth=5, width=6,
height=2, command=lambda: click('C')).grid(column=1, row=3)
btn_equ = tk.Button(f2, text='=', borderwidth=5, width=6,
height=2, command=lambda: click('=')).grid(column=2, row=3)
btn_div = tk.Button(f2, text='/',borderwidth=5, width=6, height=2,
        command=lambda: click('/')).grid(column=3, row=3)

window.mainloop()

```

3. 心得

這次實驗課前一天晚上熬夜趕隔天要交的 layout，所以實驗時做得有點痛苦，但幸好最後還是成功 demo。在我的程式中有一行是 `num=int(num)`，因為傳入 `click` 的 `num` 其變數型態是 `str`，一開始並沒有發現，到後來輸入 `2+3` 會變成 `23`，我才用 `print(type(num))` 檢查。

我覺得比較困難的部分是有負數的運算，這樣就需要處理特例，我也曾經不小心沒有加上 else（右圖裡的），這會導致當輸入是負數（-x）時，會顯示-xx，而且用-xx 做運算，例如原本想計算 $3--9(=12)$ ，會變成 $3--99(=102)$ ，最後處理好也成功執行。

```
if((len(out)==1)and(out[0]=='-')):
    out[0] = num*(-1)
elif((len(out)==3)and(out[2]=='-')):
    out[2] = num*(-1)
    print(num)
# 可能會是多個位數的數字
else:
    temp4 = int(out[(len(out)-1)])
    # 在負數最後面加一位
    if(temp4<0):
        temp = temp4*10 - num
    # 在正數最後面加一位
    else:
        temp = temp4*10 + num
    # 轉為str存入out
    out[len(out)-1] = str(temp)
```

Reference:

1. <https://yayar.medium.com/python-tkinter> 各種元件使用-上-
dd9d680ee602