

Lab 5: Python GUI Programming Report

學號：109511286

姓名：蔡佩蓉

1. 請貼上自己的程式碼並附上註解

```
# -*- coding: utf-8 -*-
"""
Created on Wed Dec 6 20:15:49 2023

@author: 109511286
"""

import tkinter as tk
from tkinter import messagebox
import pickle
from PIL import Image, ImageTk

# 登入按鈕功能
def login():
    # 獲取使用者輸入的帳號和密碼
    username = entry_username.get()
    password = entry_password.get()

    try:
        # 嘗試讀取帳號資料
        with open('accounts.pickle', 'rb') as file:
            accounts = pickle.load(file)
        # 檢查帳號是否存在，並驗證密碼是否正確
        if username in accounts:
            if accounts[username] == password:
                # 若帳號密碼正確，顯示登入成功訊息
                messagebox.showinfo("Login", "Login Successful")
            else:
                # 若密碼錯誤，顯示錯誤訊息
                messagebox.showerror("Login", "Incorrect Password")
        else:
            # 若帳號不存在，詢問是否要根據輸入建立新帳號
            response = messagebox.askyesno("User does not exist.", "Do you want to create an account by your input?")
            if response:
                # 直接將新帳號寫入pickle檔
                accounts[username] = password
                with open('accounts.pickle', 'wb') as file:
                    pickle.dump(accounts, file)
            except FileNotFoundError:
                # 若找不到檔案，詢問是否建立新帳號
                response = messagebox.askyesno("No accounts found.", "Do you want to create an account by your input?")
                if response:
                    # 如果找不到帳號檔案，則建立新的帳號pickle檔
                    accounts = {username: password}
                    with open('accounts.pickle', 'wb') as file:
                        pickle.dump(accounts, file)

def open_signup_window():
    # 開啟註冊視窗
    signup_window = tk.Toplevel(window)
    signup_window.title("Sign Up")
    signup_window.geometry("300x200")

# 註冊按鈕功能
def signup():
    # 獲取新帳號和密碼，並確認是否一致
    new_username = entry_new_username.get()
    new_password = entry_new_password.get()
    confirm_password = entry_confirm_password.get()

    try:
        try:
            with open('accounts.pickle', 'rb') as file:
                accounts = pickle.load(file)
        except EOFError:
            accounts = {}
        except FileNotFoundError:
            accounts = {}

        # 檢查帳號是否存在，以及兩次密碼輸入是否一致
        if new_username in accounts:
            # 若帳號已存在，顯示錯誤訊息
            messagebox.showerror("Sign Up", "Username already exists")
        elif new_password != confirm_password:
            # 若密碼不一致，顯示錯誤訊息
            messagebox.showerror("Sign Up", "Passwords do not match")
        else:
            # 將新帳號寫入pickle檔
            accounts[new_username] = new_password
            with open('accounts.pickle', 'wb') as file:
                pickle.dump(accounts, file)
            messagebox.showinfo("Sign Up", "Registration Successful")
            signup_window.destroy()

# 帳號、密碼輸入框及按鈕
label_new_username = tk.Label(signup_window, text="New Username:")
label_new_username.grid(column=0, row=0, padx=10, pady=10)
entry_new_username = tk.Entry(signup_window)
entry_new_username.grid(column=1, row=0, padx=10, pady=10)

label_new_password = tk.Label(signup_window, text="New Password:")
label_new_password.grid(column=0, row=1, padx=10, pady=10)
entry_new_password = tk.Entry(signup_window, show="*")
entry_new_password.grid(column=1, row=1, padx=10, pady=10)

label_confirm_password = tk.Label(signup_window, text="Confirm Password:")
label_confirm_password.grid(column=0, row=2, padx=10, pady=10)
entry_confirm_password = tk.Entry(signup_window, show="*")
entry_confirm_password.grid(column=1, row=2, padx=10, pady=10)

button_signup = tk.Button(signup_window, text="Sign Up", command=signup)
button_signup.grid(column=1, row=3, padx=0, pady=10)
```

```

if __name__ == "__main__":
    # 創建主視窗
    window = tk.Tk()
    window.title('Lab5')
    window.geometry('400x400')

    # 登入介面
    f1 = tk.Frame(window)
    f2 = tk.Frame(window)
    f1.pack()
    f2.pack()

    # 載入圖片並顯示在介面上
    image1 = Image.open('image.jpg').resize((400, 225))
    image1 = ImageTk.PhotoImage(image1)
    label_image = tk.Label(f1, image=image1)
    label_image.pack()

    # 帳號、密碼輸入框及按鈕
    label_username = tk.Label(f2, text="Username:")
    label_username.grid(column=0, row=0, padx=10, pady=10)
    entry_username = tk.Entry(f2)
    entry_username.grid(column=1, row=0, padx=10, pady=10)

    label_password = tk.Label(f2, text="Password:")
    label_password.grid(column=0, row=1, padx=10, pady=10)
    entry_password = tk.Entry(f2, show="*")
    entry_password.grid(column=1, row=1, padx=10, pady=10)

    button_login = tk.Button(f2, text="Log In", command=login)
    button_login.grid(column=0, row=2, padx=0, pady=10)

    button_signup = tk.Button(f2, text="Sign Up", command=open_signup_window)
    button_signup.grid(column=1, row=2, padx=0, pady=10)

    window.mainloop()

```

在 `login()` 裡有兩個可以直接建立新帳號的方式：

a. `accounts[username] = password`：

這個語法是在已存在的字典 `accounts` 中新增或更新一個鍵值對，鍵是 `username`，值是 `password`。如果 `username` 已經存在於 `accounts` 中，這個操作將會更新對應的密碼；如果不存在，則會新增一個鍵值對。

b. `accounts = {username: password}`：

這是創建了一個新的字典，其中只有一個鍵值對，鍵是 `username`，值是 `password`。這個操作會覆蓋掉原本 `accounts` 變數所指向的字典，並建立一個全新的字典。

當帳號不存在時，因要持續在原有的 `accounts` 字典中新增或更新資料，所以這裡會選擇使用 `accounts[username] = password`。而當找不到 `pickle` 檔案時，需要重新建立一個只有一個鍵值對的新字典，因此會選擇使用 `accounts = {username: password}`。

2. 心得或建議

在探索這段程式碼的過程中，我深入了解了如何使用 `tkinter` 建立簡單的登入系統。從登入驗證到帳號註冊，這個程式碼展示了使用者介面的基本互動和資料存儲的方法。透過 `pickle` 模組，程式能夠將帳號資訊以二進制形式保存，實現了基本的帳號管理功能。更重要的是，我理解了如何處理使用者的輸入，並根據其行為進行不同的操作，例如驗證密碼、檢查帳號是否存在以及創建新帳號等。這段程式碼不僅僅是技術的學習，也是開啟了對程式設計中使用使用者體驗和互動性考量的一扇門。