import tkinter as tk

from tkinter import ttk, messagebox, simpledialog

from datetime import datetime

def register():

reg\_window = tk.Toplevel(root)

reg\_window.title("Registrasi")

tk.Label(reg\_window, text="Username:").grid(row=0, column=0, padx=5, pady=5)

reg\_username = tk.Entry(reg\_window)

reg\_username.grid(row=0, column=1, padx=5, pady=5)

tk.Label(reg\_window, text="Password:").grid(row=1, column=0, padx=5, pady=5)

reg\_password = tk.Entry(reg\_window, show='\*')

reg\_password.grid(row=1, column=1, padx=5, pady=5)

def save\_registration():

username = reg\_username.get()

password = reg\_password.get()

if username and password:

with open('users.txt', 'a') as file:

file.write(f'{username},{password}\n')

messagebox.showinfo("Registrasi Berhasil", "Registrasi berhasil. Silakan login.")

reg\_window.destroy()

else:

messagebox.showwarning("Registrasi Gagal", "Username dan password harus diisi.")

tk.Button(reg\_window, text="Register", command=save\_registration).grid(row=2, columnspan=2, pady=5)

def login():

login\_window = tk.Toplevel(root)

login\_window.title("Login")

tk.Label(login\_window, text="Username:").grid(row=0, column=0, padx=5, pady=5)

login\_username = tk.Entry(login\_window)

login\_username.grid(row=0, column=1, padx=5, pady=5)

tk.Label(login\_window, text="Password:").grid(row=1, column=0, padx=5, pady=5)

login\_password = tk.Entry(login\_window, show='\*')

login\_password.grid(row=1, column=1, padx=5, pady=5)

def check\_login():

username = login\_username.get()

password = login\_password.get()

if username and password:

try:

with open('users.txt', 'r') as file:

users = file.readlines()

for user in users:

stored\_user, stored\_pass = user.strip().split(',')

if stored\_user == username and stored\_pass == password:

messagebox.showinfo("Login Berhasil", f"Selamat datang, {username}")

login\_window.destroy()

show\_main\_window() # Menampilkan program absensi setelah login berhasil

return

messagebox.showerror("Login Gagal", "Username atau password salah.")

except FileNotFoundError:

messagebox.showerror("Error", "File users.txt tidak ditemukan.")

else:

messagebox.showwarning("Login Gagal", "Username dan password harus diisi.")

tk.Button(login\_window, text="Login", command=check\_login).grid(row=2, columnspan=2, pady=5)

def absensi():

nama = entry\_nama.get()

status = combo\_status.get()

mata\_kuliah = combo\_matkul.get()

semester = combo\_semester.get()

tanggal = entry\_tanggal.get()

if nama and mata\_kuliah and tanggal:

today\_date = datetime.now().strftime("%Y-%m-%d")

if tanggal == today\_date:

if status == "Izin":

tree.insert('', 'end', values=(nama, mata\_kuliah, status, "", semester, tanggal))

elif status == "Sakit":

alasan = entry\_alasan.get()

tree.insert('', 'end', values=(nama, mata\_kuliah, status, alasan, semester, tanggal))

else:

tree.insert('', 'end', values=(nama, mata\_kuliah, status, "", semester, tanggal))

entry\_nama.delete(0, 'end')

entry\_alasan.delete(0, 'end')

combo\_status.set("Hadir")

combo\_matkul.set("Pilih Mata Kuliah")

combo\_semester.set("Semester 1")

entry\_tanggal.delete(0, 'end')

else:

messagebox.showerror("Error", "Absensi hanya dapat dilakukan untuk hari ini.")

else:

messagebox.showerror("Error", "Harap lengkapi semua data.")

def show\_main\_window():

main\_window = tk.Toplevel(root)

main\_window.title("Aplikasi Absensi Mahasiswa")

frame\_input = ttk.Frame(main\_window)

frame\_input.pack(padx=10, pady=10)

label\_nama = ttk.Label(frame\_input, text="Nama Mahasiswa:")

label\_nama.grid(row=0, column=0, padx=5, pady=5, sticky="w")

global entry\_nama

entry\_nama = ttk.Entry(frame\_input)

entry\_nama.grid(row=0, column=1, padx=5, pady=5)

label\_matkul = ttk.Label(frame\_input, text="Mata Kuliah:")

label\_matkul.grid(row=1, column=0, padx=5, pady=5, sticky="w")

matkul\_options = {

"Semester 1": ["Pilih Mata Kuliah", "Kalkulus", "Fisika", "Kimia"],

"Semester 2": ["Pilih Mata Kuliah", "Biologi", "Bahasa Inggris", "Sejarah"],

"Semester 3": ["Pilih Mata Kuliah", "Ekonomi", "PMS", "Pancasila"],

"Semester 4": ["Pilih Mata Kuliah", "Algoritma", "Pemrograman", "Statistika"],

"Semester 5": ["Pilih Mata Kuliah", "Jaringan Komputer", "Sistem Operasi", "Basis Data"],

"Semester 6": ["Pilih Mata Kuliah", "PBO", "Struktur Data", "Kecerdasan Buatan"],

"Semester 7": ["Pilih Mata Kuliah", "Kriptografi", "Keamanan Jaringan", "Big Data"],

"Semester 8": ["Pilih Mata Kuliah", "IoT", "Cloud Computing", "Cybersecurity"]

}

global combo\_matkul

combo\_matkul = ttk.Combobox(frame\_input, state="readonly")

combo\_matkul.set("Pilih Mata Kuliah")

combo\_matkul.grid(row=1, column=1, padx=5, pady=5)

def update\_matkul(event):

selected\_semester = combo\_semester.get()

matkul\_list = matkul\_options[selected\_semester]

combo\_matkul['values'] = matkul\_list

combo\_matkul.set("Pilih Mata Kuliah")

global combo\_semester

combo\_semester = ttk.Combobox(frame\_input, values=["Semester 1", "Semester 2", "Semester 3", "Semester 4", "Semester 5", "Semester 6", "Semester 7", "Semester 8"], state="readonly")

combo\_semester.set("Semester 1")

combo\_semester.grid(row=2, column=1, padx=5, pady=5)

combo\_semester.bind("<<ComboboxSelected>>", update\_matkul)

label\_tanggal = ttk.Label(frame\_input, text="Tanggal (YYYY-MM-DD):")

label\_tanggal.grid(row=3, column=0, padx=5, pady=5, sticky="w")

global entry\_tanggal

entry\_tanggal = ttk.Entry(frame\_input)

entry\_tanggal.grid(row=3, column=1, padx=5, pady=5)

label\_status = ttk.Label(frame\_input, text="Status Kehadiran:")

label\_status.grid(row=4, column=0, padx=5, pady=5, sticky="w")

status\_options = ["Hadir", "Izin", "Sakit"]

global combo\_status

combo\_status = ttk.Combobox(frame\_input, values=status\_options, state="readonly")

combo\_status.set("Hadir")

combo\_status.grid(row=4, column=1, padx=5, pady=5)

label\_alasan = ttk.Label(frame\_input, text="Alasan Sakit:")

label\_alasan.grid(row=5, column=0, padx=5, pady=5, sticky="w")

global entry\_alasan

entry\_alasan = ttk.Entry(frame\_input)

entry\_alasan.grid(row=5, column=1, padx=5, pady=5)

button\_absen = ttk.Button(frame\_input, text="Absen", command=absensi)

button\_absen.grid(row=6, columnspan=2, padx=5, pady=5)

frame\_tabel = ttk.Frame(main\_window)

frame\_tabel.pack(padx=10, pady=10)

columns = ("Nama", "Mata Kuliah", "Status", "Alasan Sakit", "Semester", "Tanggal")

global tree

tree = ttk.Treeview(frame\_tabel, columns=columns, show="headings")

for col in columns:

tree.heading(col, text=col)

tree.pack()

root = tk.Tk()

root.title("Login dan Registrasi")

tk.Button(root, text="Login", command=login).pack(padx=10, pady=5)

tk.Button(root, text="Register", command=register).pack(padx=10, pady=5)

root.mainloop()