import os

import tkinter as tk

from tkinter import ttk, messagebox

try:

from openai import OpenAI

except ImportError:

OpenAI = None # Permitimos abrir la ventana aunque no esté la librería

# Paleta (coherente con tu app)

PRIMARY\_BG = "#E97612"

SURFACE\_BG = "#2A250F"

ACCENT = "#22C55E"

TEXT = "#FFFFFF"

MUTED = "#9CA3AF"

def open\_win\_consejos(parent: tk.Tk):

"""

Abre una ventana (Toplevel) con un botón para pedir

una frase motivacional a OpenAI y mostrarla en pantalla.

Sigue el mismo patrón de 'open\_win\_form'.

"""

win = tk.Toplevel(parent)

win.title("Consejos motivacionales")

win.geometry("420x420")

win.configure(bg=PRIMARY\_BG)

# --------- Estilos ttk ---------

style = ttk.Style(win)

style.theme\_use("clam")

style.configure("TFrame", background=PRIMARY\_BG)

style.configure("TLabel", background=PRIMARY\_BG, foreground=TEXT)

style.configure("Muted.TLabel", background=PRIMARY\_BG, foreground=MUTED)

style.configure("Surface.TFrame", background=SURFACE\_BG)

style.configure("Accent.TButton", background=ACCENT, foreground=TEXT, borderwidth=0, padding=8)

style.map("Accent.TButton", background=[("active", "#16A34A")])

# --------- Contenedor principal ---------

frm = ttk.Frame(win, padding=16, style="TFrame")

frm.pack(fill="both", expand=True)

# Título

ttk.Label(frm, text="🌿 Consejos motivacionales", font=("Segoe UI", 16, "bold")).grid(

row=0, column=0, columnspan=3, sticky="w", pady=(0, 8)

)

ttk.Label(frm, text="Presiona el botón para recibir inspiración del día.", style="Muted.TLabel").grid(

row=1, column=0, columnspan=3, sticky="w", pady=(0, 14)

)

# Tarjeta de frase

card = ttk.Frame(frm, padding=14, style="Surface.TFrame")

card.grid(row=2, column=0, columnspan=3, sticky="nsew")

frm.rowconfigure(2, weight=1)

frm.columnconfigure(0, weight=1)

lbl\_frase = ttk.Label(card, text="(Aquí aparecerá tu frase)", wraplength=360,

font=("Segoe UI", 11, "italic"))

lbl\_frase.pack(fill="x")

# --------- Lógica para obtener frase ---------

def obtener\_frase():

if OpenAI is None:

messagebox.showerror("Falta dependencia",

"No se encontró la librería 'openai'. Instálala con:\n\npip install openai")

return

api\_key = os.getenv("OPENAI\_API\_KEY", "sk-proj-EWYVTHPoGzCTXc\_ekIoZEVWan0aDXyuUm7e7KLszC31PIHTayPkp7\_vBXdzmilyy6b1h0Y46EdT3BlbkFJW7994fheEXCL1xUECSVZ6WCBdTtlQng\_bj6Izb6f5dxjsIOti-cE1CptNquNQxwFf5oa6kIuEA").strip()

if not api\_key:

messagebox.showerror("API Key requerida",

"No se encontró la variable de entorno OPENAI\_API\_KEY.\n"

"Configúrala antes de continuar.")

return

try:

client = OpenAI(api\_key=api\_key)

# Si quieres, cambia el modelo por el que tengas disponible en tu cuenta

modelo = "gpt-3.5-turbo"

rsp = client.chat.completions.create(

model=modelo,

messages=[

{"role": "system",

"content": "Eres un generador de frases motivacionales breves, positivas y en español."},

{"role": "user",

"content": "Dame una frase motivacional para hoy (máx. 20 palabras)."}

],

max\_tokens=40,

temperature=0.9,

)

frase = rsp.choices[0].message.content.strip().strip("“”\"'")

if not frase:

raise ValueError("La respuesta vino vacía.")

lbl\_frase.config(text=f"“{frase}”")

except Exception as e:

messagebox.showerror("Error", f"Ocurrió un problema al generar la frase:\n\n{e}")

def copiar\_frase():

texto = lbl\_frase.cget("text").strip()

if not texto or texto.startswith("("):

messagebox.showinfo("Nada que copiar", "Aún no has generado una frase.")

return

win.clipboard\_clear()

win.clipboard\_append(texto)

messagebox.showinfo("Copiado", "Frase copiada al portapapeles.")

# --------- Botones ---------

ttk.Button(frm, text="💬 Obtener frase", style="Accent.TButton",

command=obtener\_frase).grid(row=3, column=0, sticky="w", pady=14)

ttk.Button(frm, text="Copiar", command=copiar\_frase).grid(row=3, column=1, sticky="w", pady=14, padx=(8, 0))

ttk.Button(frm, text="Cerrar", command=win.destroy).grid(row=3, column=2, sticky="e", pady=14)

# Accesibilidad / foco inicial

win.focus\_set()

win.grab\_set()