import tkinter as tk

from tkinter import messagebox

from tkinter import font

import requests

API\_KEY = "your\_api\_key"

# Function to fetch weather details

def fetch\_weather(city):

url = f"http://api.openweathermap.org/data/2.5/weather?q={city}&appid={API\_KEY}&units=metric"

try:

response = requests.get(url)

response.raise\_for\_status()

data = response.json()

weather = {

"city": data["name"],

"temperature": data["main"]["temp"],

"humidity": data["main"]["humidity"],

"description": data["weather"][0]["description"],

"wind\_speed": data["wind"]["speed"]

}

return weather

except requests.exceptions.RequestException as e:

messagebox.showerror("Error", f"Unable to retrieve data: {e}")

return None

# Function to display weather in the GUI

def display\_weather():

city = city\_entry.get()

if city:

weather = fetch\_weather(city)

if weather:

result = (f"City: {weather['city']}\n"

f"Temperature: {weather['temperature']}°C\n"

f"Humidity: {weather['humidity']}%\n"

f"Description: {weather['description'].capitalize()}\n"

f"Wind Speed: {weather['wind\_speed']} m/s")

result\_label.config(text=result, fg="white")

else:

result\_label.config(text="No data found.", fg="red")

else:

messagebox.showwarning("Input Error", "Please enter a city name.")

# Setting up the Tkinter window

root = tk.Tk()

root.title("Weather Monitor")

root.geometry("500x400")

root.configure(bg="#1e3d59") # Set a background color

# Custom font styles

header\_font = font.Font(family="Helvetica", size=16, weight="bold")

text\_font = font.Font(family="Helvetica", size=12)

# City input label

city\_label = tk.Label(root, text="Enter City:", font=header\_font, bg="#1e3d59", fg="#ff6b6b")

city\_label.pack(pady=10)

# City entry field

city\_entry = tk.Entry(root, width=30, font=text\_font, bd=3)

city\_entry.pack(pady=10)

# Search button with customized style

search\_button = tk.Button(root, text="Get Weather", command=display\_weather, bg="#ff6b6b", fg="white",

font=text\_font, activebackground="#ff3b3b", activeforeground="white", bd=0)

search\_button.pack(pady=15)

# Result display label with enhanced style

result\_label = tk.Label(root, text="", font=text\_font, bg="#1e3d59", fg="white", justify="left", wraplength=400)

result\_label.pack(pady=20)

root.mainloop()