

CODIGO:

```
import tkinter as tk

from tkinter import ttk, messagebox

import requests

import matplotlib.pyplot as plt

from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg


def fetch_data():
    """
    Conecta con la API de Open-Meteo y obtiene velocidad del viento horaria
    de Ciudad de México (últimas 24 horas).
    Devuelve dos listas: horas y velocidades.
    """
    try:
        url = (
            "https://api.open-meteo.com/v1/forecast"
            "?latitude=19.43&longitude=-99.13"
            "&hourly=windspeed_10m&past_days=1"
            "&timezone=auto"
        )

        response = requests.get(url, timeout=15)
        response.raise_for_status()
        data = response.json()

        horas = data["hourly"]["time"]
        velocidades = data["hourly"]["windspeed_10m"]
```

```

        return horas, velocidades

except Exception as e:

    messagebox.showerror("Error", f"No se pudieron obtener los datos:\n{e}")

    return [], []


def create_line_chart(horas, vel):

    """Gráfica de línea."""

    fig, ax = plt.subplots(figsize=(6, 3))

    ax.plot(horas, vel, linestyle="-", marker="o", markersize=3, color="blue")

    ax.set_title("Velocidad del viento en CDMX (línea)")

    ax.set_xlabel("Hora")

    ax.set_ylabel("Velocidad (km/h)")

    ax.tick_params(axis="x", rotation=45)

    fig.tight_layout()

    return fig


def create_bar_chart(horas, vel):

    """Gráfica de barras."""

    fig, ax = plt.subplots(figsize=(6, 3))

    ax.bar(horas, vel, color="green")

    ax.set_title("Velocidad del viento en CDMX (barras)")

    ax.set_xlabel("Hora")

    ax.set_ylabel("Velocidad (km/h)")

    ax.tick_params(axis="x", rotation=45)

    fig.tight_layout()

```

```
return fig
```

```
def mostrar_graficas(frm, horas, vel):
```

```
    """Inserta las gráficas en el frame de la ventana tkinter."""
```

```
    # Línea
```

```
    fig1 = create_line_chart(horas, vel)
```

```
    canvas1 = FigureCanvasTkAgg(fig1, master=frm)
```

```
    canvas1.draw()
```

```
    canvas1.get_tk_widget().pack(pady=10, fill="x")
```

```
    # Barras
```

```
    fig2 = create_bar_chart(horas, vel)
```

```
    canvas2 = FigureCanvasTkAgg(fig2, master=frm)
```

```
    canvas2.draw()
```

```
    canvas2.get_tk_widget().pack(pady=10, fill="x")
```

```
def open_win_canvas(parent: tk.Tk):
```

```
    """
```

```
    Crea la ventana secundaria con gráficas de la API.
```

```
    """
```

```
    win = tk.Toplevel(parent)
```

```
    win.title("Canvas con API (Open-Meteo) y gráficas")
```

```
    win.geometry("960x1000")
```

```
    frm = ttk.Frame(win, padding=12)
```

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

```
# Botón para cargar datos y graficar

def cargar():

    horas, vel = fetch_data()

    if horas and vel:

        mostrar_graficas(frm, horas, vel)


ttk.Button(frm, text="Cargar y mostrar gráficas", command=cargar).pack(pady=10)


# Para pruebas independientes (opcional)

if __name__ == "__main__":

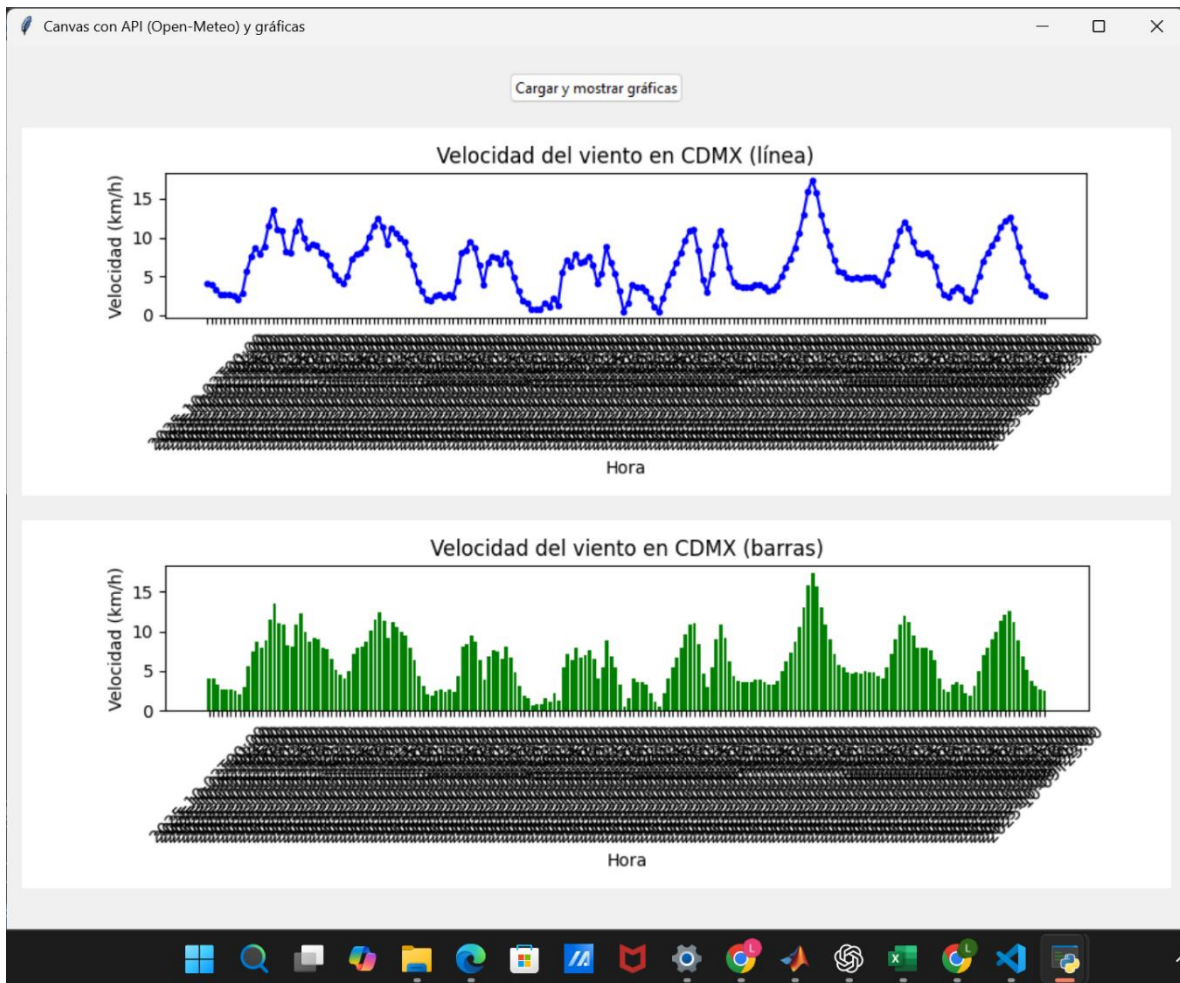
    root = tk.Tk()

    root.title("Prueba win_canvas")

    ttk.Button(root, text="Abrir ventana Canvas", command=lambda:
open_win_canvas(root)).pack(pady=20)

    root.mainloop()
```

## captura de pantalla



## cambios hechos:

Cambié el código para que en lugar de mostrar la temperatura de León ahora muestre la velocidad del viento en la Ciudad de México. Para eso cambié la latitud y longitud en la API y sustituí el parámetro `temperature_2m` por `windspeed_10m`. También actualicé los títulos, las etiquetas y las unidades en las gráficas para que indiquen "Velocidad del viento (km/h)". El programa ahora genera dos gráficas (línea y barras) con los datos de las últimas 24 horas en CDMX.