**The Global Axis project is a Python-based application that provides real-time weather updates and local time information for cities worldwide. Using the OpenWeatherMap API, it fetches temperature, weather conditions, and time zone data, making it a useful tool for travelers, businesses, and researchers.**

**With globalization and remote work on the rise, quick access to accurate weather and time data is essential for trip planning, international meetings, and weather monitoring. The project utilizes Python, Requests library, JSON parsing, and the Datetime module to ensure efficient data retrieval and processing.**

**Future enhancements could include a graphical user interface (GUI), weather forecasting, mobile/web applications, and automated alerts for extreme weather conditions. By integrating API-driven data collection, Global Axis serves as a practical tool for real-time global weather and time tracking.**

**CODE:**

import requests

from datetime import datetime, timedelta

# Constants

API\_KEY = "your\_openweathermap\_api\_key" # Replace with your actual API key

BASE\_URL = "https://api.openweathermap.org/data/2.5/weather"

# Function to get city weather and time info

def get\_city\_info(city\_name):

try:

params = {

"q": city\_name,

"appid": API\_KEY,

"units": "metric"

}

response = requests.get(BASE\_URL, params=params)

if response.status\_code != 200:

print("Unable to fetch data. Error:", response.json().get("message", "Unknown error"))

return

data = response.json()

# Extract weather details

weather\_desc = data['weather'][0]['description'].capitalize()

temperature = data['main']['temp']

timezone\_offset = data['timezone']

# Get local time

utc\_time = datetime.utcnow()

city\_time = utc\_time + timedelta(seconds=timezone\_offset)

formatted\_time = city\_time.strftime("%H:%M:%S")

formatted\_date = city\_time.strftime("%d %B %Y")

formatted\_day = city\_time.strftime("%A")

# Display info

print(f"\nCity: {city\_name}")

print(f"Local Time: {formatted\_time}")

print(f"Date: {formatted\_date}")

print(f"Day: {formatted\_day}")

print(f"Weather: {weather\_desc}")

print(f"Temperature: {temperature}°C")

except Exception as e:

print("An error occurred:", e)

# Main execution block

if \_\_name\_\_ == "\_\_main\_\_":

city = input("Enter city name: ")

get\_city\_info(city)