**README FILE**

**BY**

**ANDREW IGNATIUS BROWN**

**JustIT SKILLS BOOTCAMP**

**January 2025.**

**TOPIC: PYTHON WEATHER APP PROJECT CODE.**

# IMPORT API/components

import requests

import json

# WELCOME

user\_name = input("Hi, please enter your name: ")

print(f'\nWelcome {user\_name} to the Weather APP!')

# Function to input city

def city\_forecast():

return input("\nPlease, type a city in the UK for a weather forecast or type 'stop' to exit: ")

# Function to fetch weather data

def city\_weather(city):

try:

# Attach the inputted city to the API link to obtain data as latitude and longitude

attach\_city = f'http://api.openweathermap.org/geo/1.0/direct?q={city},GB&appid=23521458b54d1731b680e7098bc4954a'

response = requests.get(attach\_city) # Apply the API request function to obtain city details

# Check if the response is successful and convert the result to a JSON format

if response.status\_code == 200: # 200 represents a successful response

data = response.json()

if data: # If the result from API is successful and not empty

lat, lon = data[0]["lat"], data[0]["lon"] # Obtain latitude and longitude values of the city

# Fetch weather details using the second API link

weather\_api = f'https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&appid=23521458b54d1731b680e7098bc4954a'

weather\_data = requests.get(weather\_api)

if weather\_data.status\_code == 200:

data2 = weather\_data.json() # Keep it as a JSON object

# Weather info

print(f"\nTHE WEATHER FOR {city.upper()}:") # Print weather info title

celsius = round(data2['main']['temp'] - 273.15, 2) # Convert temperature from Kelvin to Celsius

print(f"Current temperature: {celsius}°C.")

print(f"Weather condition: {data2['weather'][0]['description']}.")

print(f"Wind speed: {data2['wind']['speed']} m/s.")

print(f"Humidity: {data2['main']['humidity']}%.")

else:

print(f"Unable to fetch detailed weather data for {city}. Error {weather\_data.status\_code}. Please try again later.")

else:

print(f"No location data found for the city: {city}. Please check the city name and try again.")

return False # Signal invalid input to re-prompt in the main loop

else:

print(f"Error: Unable to fetch data. HTTP Status Code: {response.status\_code}")

return False # Signal invalid input to re-prompt in the main loop

except requests.exceptions.RequestException as e:

print(f"An error occurred while fetching data: {e}")

return False # Signal invalid input to re-prompt in the main loop

return True # Signal successful execution

# Main Workflow

count = 1

while count <= 1:

city = city\_forecast()

if city.lower() == 'stop': # Allow the user to exit by typing 'stop'

break

else:

success = city\_weather(city)

if success: # Only increment count if the weather fetch was successful

count += 1

# Thank the user

print(f'\nThank you for using the Weather APP, {user\_name}.')