

Develop the Python Script

(Develop a Python script)

Team ID : PNT2022TMID18894

IBM ID : IBM-Project-31900-1660206003

Industry-specific intelligent fire management system

Create a code snippet using python to

1. Extract weather data from OpenWeatherMap using APIs
2. Send the extracted data to the cloud
3. Receive data from the cloud and view it in the python compiler

1*Team Member Weather account ID

We have sent the confirmation link to nsp789052@gmail.com. Please check your email.

New Products Services API keys Billing plans Payments Block logs My orders My profile Ask a question

Historical weather for any location

Our new technology, Time Machine, has allowed us to enhance the data in the [Historical Weather Collection](#).

- Historical weather data available for ANY coordinate
- The depth of historical data have been extended to **40 YEARS**

You can download data from [Personal account](#) or [contact us](#) to order it.

[Learn more](#) [Go to purchase](#)

We have sent the confirmation link to venkatsanthosh47@gmail.com. Please check your email.

New Products Services API keys Billing plans Payments Block logs My orders My profile Ask a question

Historical weather for any location

Our new technology, Time Machine, has allowed us to enhance the data in the [Historical Weather Collection](#).

- Historical weather data available for ANY coordinate
- The depth of historical data have been extended to **40 YEARS**

You can download data from [Personal account](#) or [contact us](#) to order it.

[Learn more](#) [Go to purchase](#)

Activities Google Chrome Nov 12 7:13 PM prabuprithvi20... 100 %

Wh... Men... Wea... IBM... You... IBM... IBM... IBM... IBM... IBM... IBM... IBM... +

home.openweathermap.org

Gmail Codeit Spotify - Web... IBM Watson Io... DHIRAJLAL G... SethupathiN... DIBM-Project... Other bookmarks

OpenWeather Weather in your city Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business prab... Support

We have sent the confirmation link to prabuprithvi2000@gmail.com. Please check your email.

New Products Services API keys Billing plans Payments Block logs My orders My profile Ask a question



Historical weather for any location

Our new technology, Time Machine, has allowed us to enhance the data in the [Historical Weather Collection](#).

- Historical weather data available for **ANY** coordinate
- The depth of historical data have been extended to **40 YEARS**

You can download data from [Personal account](#) or [contact us](#) to order it.

[Learn more](#)

[Go to purchase](#)



Weather Dashboard

Activities Google Chrome Nov 12 7:14 PM vinothkalam212... 100 %

Wh... Men... Wea... IBM... You... IBM... IBM... IBM... IBM... IBM... IBM... IBM... IBM... IBM... +

home.openweathermap.org

Gmail Codeit Spotify - Web... IBM Watson Io... DHIRAJLAL G... SethupathiN... DIBM-Project... Other bookmarks

OpenWeather Weather in your city Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business vinot... Support

We have sent the confirmation link to vinothkalam2127@gmail.com. Please check your email.

New Products Services API keys Billing plans Payments Block logs My orders My profile Ask a question



Historical weather for any location

Our new technology, Time Machine, has allowed us to enhance the data in the [Historical Weather Collection](#).

- Historical weather data available for **ANY** coordinate
- The depth of historical data have been extended to **40 YEARS**

You can download data from [Personal account](#) or [contact us](#) to order it.

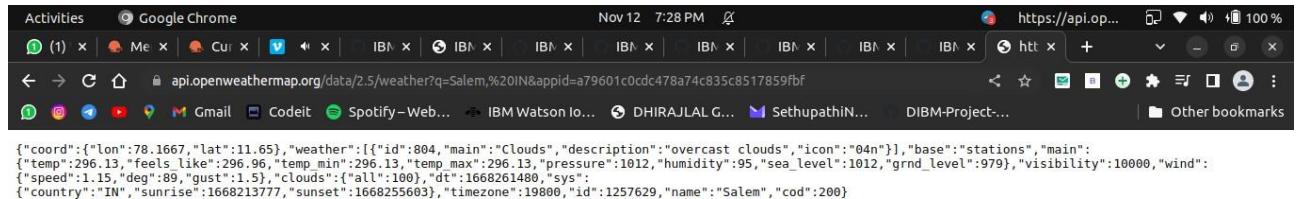
[Learn more](#)

[Go to purchase](#)



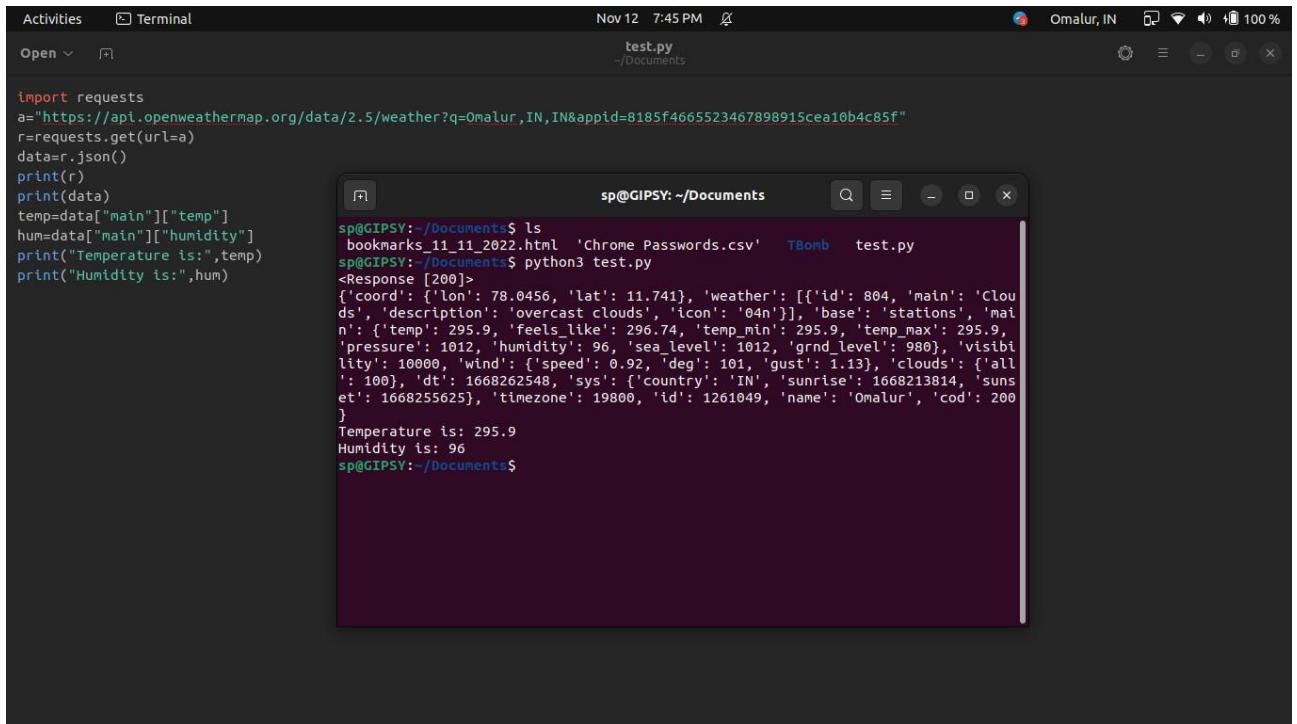
Weather Dashboard

3* python compiler



The screenshot shows a Google Chrome browser window with the URL <https://api.openweathermap.org/data/2.5/weather?q=Salem,%20IN&appid=a79601c0cdc478a74c835c8517859bf>. The page displays a JSON object representing the weather in Salem, India. Key fields include coordinates, weather conditions, temperature, humidity, pressure, visibility, wind speed and direction, and sunrise/sunset times.

```
{"coord":{"lon":78.1667,"lat":11.65},"weather":[{"id":804,"main":"Clouds","description":"overcast clouds","icon":"04n"}],"base":"stations","main":{"temp":296.13,"feels_like":296.96,"temp_min":296.13,"temp_max":296.13,"pressure":1012,"humidity":95,"sea_level":1012,"grnd_level":979}, "visibility":10000,"wind":{"speed":1.15,"deg":89,"gust":1.5}, "clouds":{"all":100}, "dt":1668261480, "sys": {"country": "IN", "sunrise": 1668213777, "sunset": 1668255603}, "timezone":19800, "id":1257629, "name": "Salem", "cod":200}
```



The screenshot shows a terminal window with Python code to fetch weather data from the OpenWeatherMap API and print it to the console. The code uses the requests library to make a GET request to the API endpoint for Omalur, India, and then prints the JSON response to the terminal. The terminal also shows the command to run the script and its execution.

```
import requests
a="https://api.openweathermap.org/data/2.5/weather?q=Omalur,IN,IN&appid=8185f4665523467898915cea10b4c85f"
r=requests.get(url=a)
data=r.json()
print(r)
print(data)
temp=data["main"]["temp"]
hum=data["main"]["humidity"]
print("Temperature is:",temp)
print("Humidity is:",hum)
```

```
sp@GIPSY:~/Documents$ ls
bookmarks_11_2022.html  'Chrome Passwords.csv'  TBomb  test.py
sp@GIPSY:~/Documents$ python3 test.py
<Response [200]>
{'coord': {'lon': 78.0456, 'lat': 11.741}, 'weather': [{"id": 804, 'main': 'Clouds', 'description': 'overcast clouds', 'icon': '04n'}], 'base': 'stations', 'main': {'temp': 295.9, 'feels_like': 296.74, 'temp_min': 295.9, 'temp_max': 295.9, 'pressure': 1012, 'humidity': 96, 'sea_level': 1012, 'grnd_level': 980}, 'visibility': 10000, 'wind': {'speed': 0.92, 'deg': 101, 'gust': 1.13}, 'clouds': {'all': 100}, 'dt': 1668262548, 'sys': {'country': 'IN', 'sunrise': 1668213814, 'sunset': 1668255625}, 'timezone': 19800, 'id': 1261049, 'name': 'Omalur', 'cod': 200}
Temperature is: 295.9
Humidity is: 96
sp@GIPSY:~/Documents$
```