

Develop the Python Script

(Develop a Python script)

Date	28 October 2022
Team ID	PNT2022TMID19682
Project Name	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

The screenshot shows the OpenWeatherMap website. The header includes the OpenWeather logo, a search bar, and navigation links: Guide, API, Dashboard, Marketplace, Pricing, Maps, Our Initiatives, Partners, Blog, For Business, Mah..., and Support. Below the header is a secondary navigation bar with links: New Products, Services, API keys, Billing plans, Payments, Block logs, My orders, My profile, and Ask a question. The main content area features a large image of a sunset over a body of water. To the right of the image is the heading 'Historical weather for any location' and a paragraph: 'Our new technology, Time Machine, has allowed us to enhance the data in the Historical Weather Collection.' Below this are two bullet points: 'Historical weather data available for ANY coordinate' and 'The depth of historical data have been extended to 40 YEARS'. A line of text says 'You can download data from Personal account or contact us to order it.' At the bottom of this section are two buttons: 'Learn more' and 'Go to purchase'.



Weather Dashboard

The OpenWeather Dashboard is a lightweight and flexible visual tool for our customers who would like to be notified weather events to make informed decisions and plan actions based on the weather input.

- Track the main weather parameters: temperature, wind speed, precipitations
- Weather data are updated every hour
- Global coverage - Choose any location on the globe
- Email notifications

This is a duplicate of the screenshot above, showing the OpenWeatherMap website with the 'Historical weather for any location' section. It includes the same header, navigation bars, main content area with the sunset image, and the 'Weather Dashboard' section with its illustration and bullet points.

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

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.

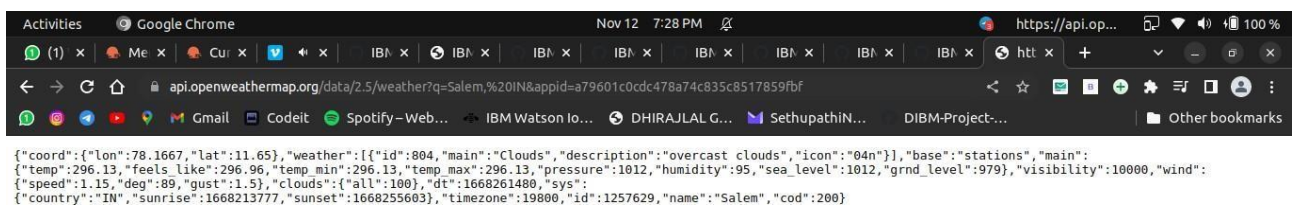
[Learn more](#)

[Go to purchase](#)

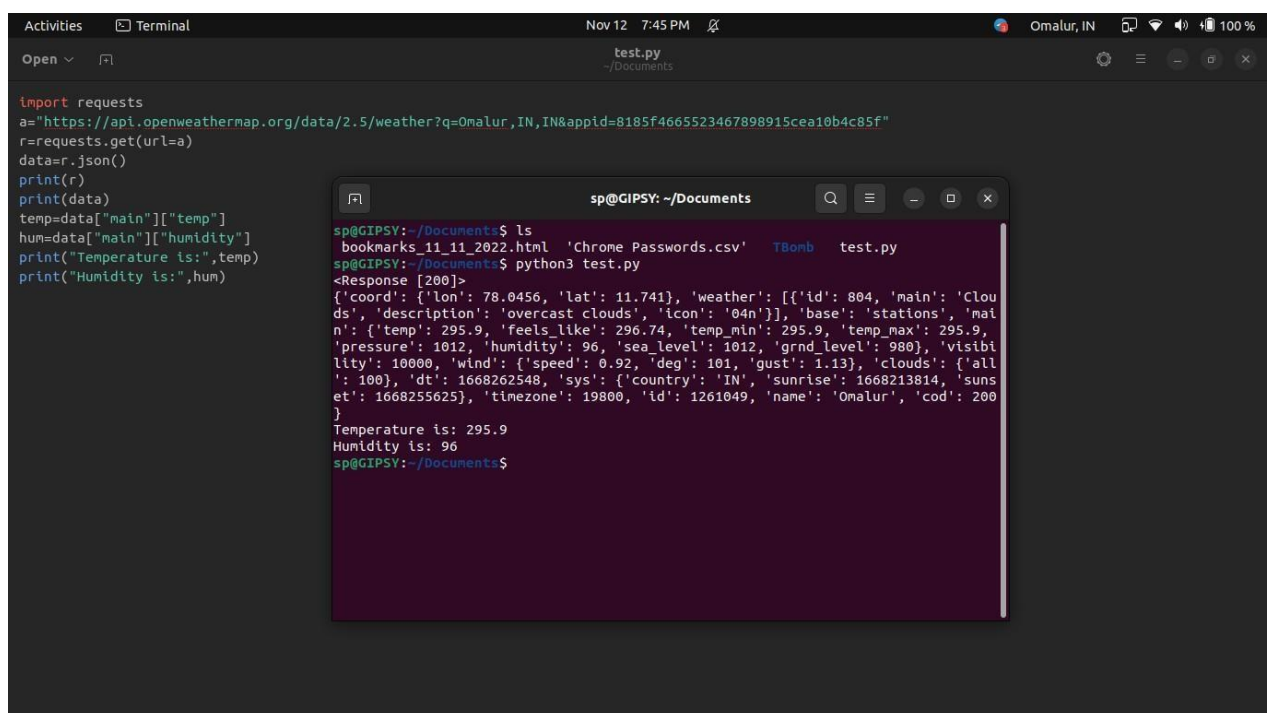


Weather Dashboard

3* python compiler



```
{
  "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
    }
  }
}
```



```
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
sp@GIPSY:~/Documents$ ls
bookmarks_11_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$
```