

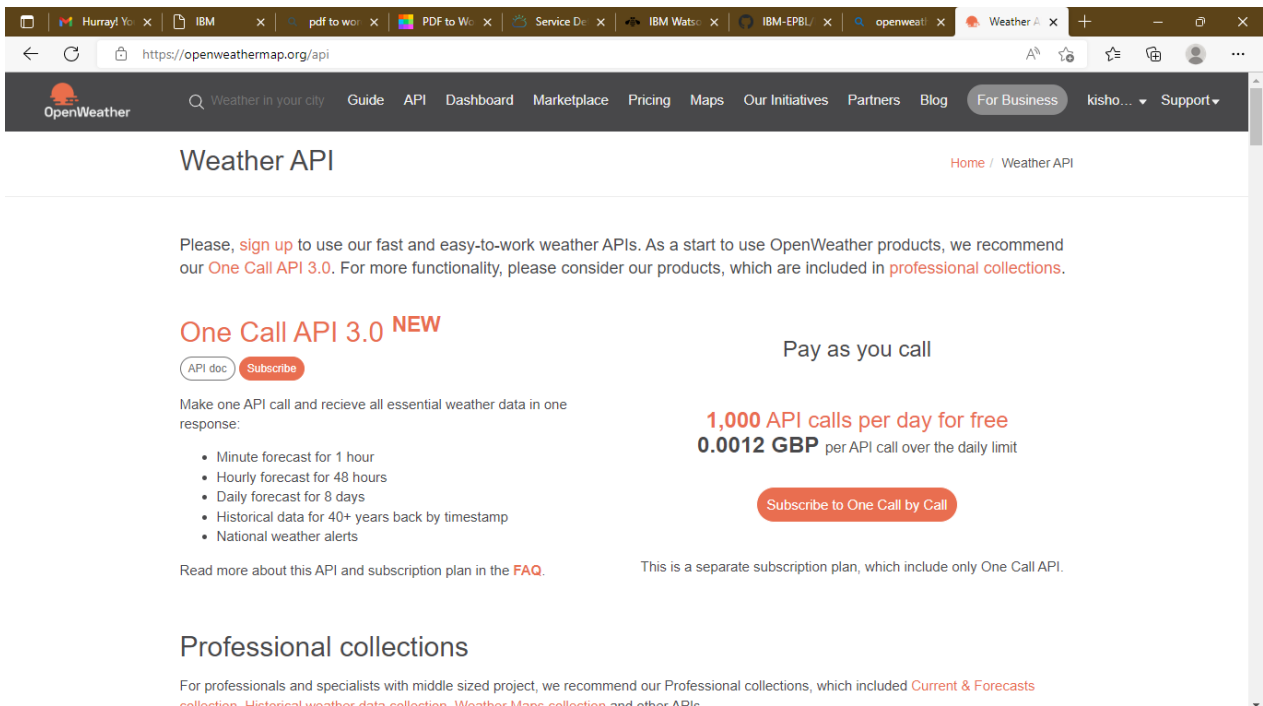
Develop a Python script

Date	12 September 2022
Team ID	PNT2022TMID13499
Project Name	Project - IoT Based Safety Gadget for Child Safety Monitoring & Notification
Maximum Marks	4 Marks

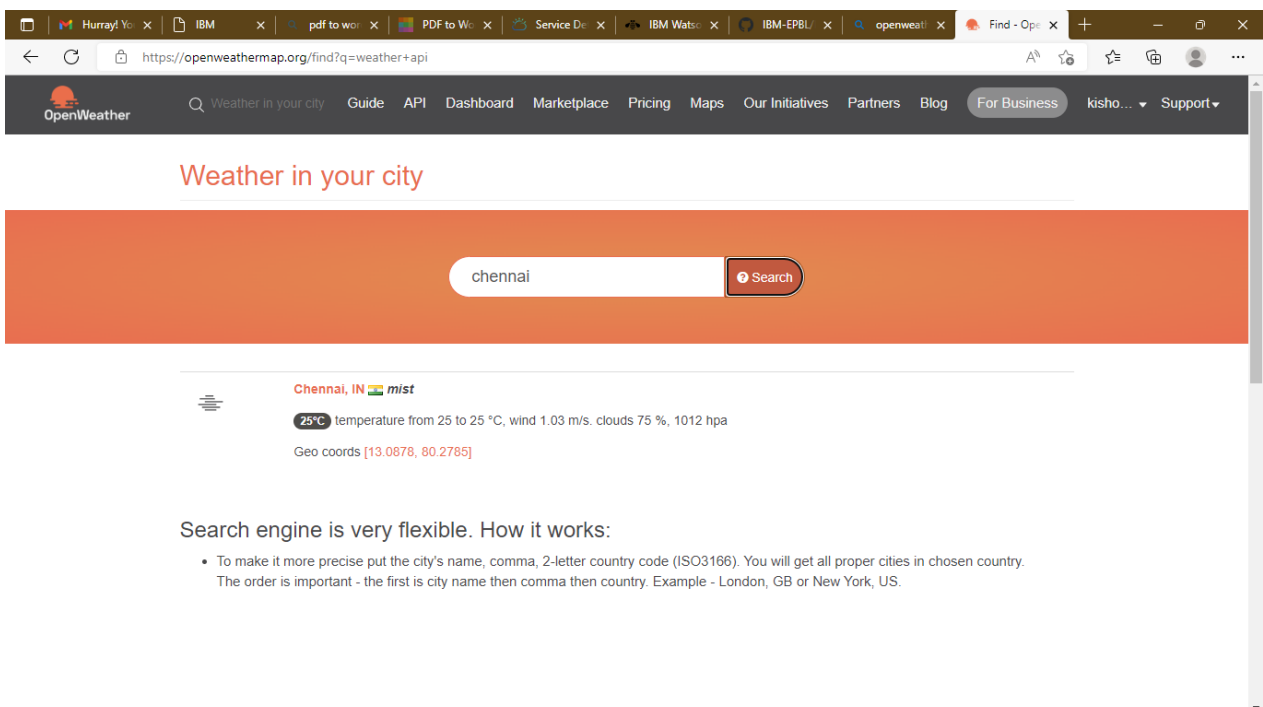
Safety Gadget for Child Safety Monitoring & Notification

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



The screenshot shows the OpenWeatherMap API page. The header includes the OpenWeather logo and navigation links: Weather in your city, Guide, API, Dashboard, Marketplace, Pricing, Maps, Our Initiatives, Partners, Blog, For Business, kisho..., and Support. The main heading is "Weather API" with a link to "Home / Weather API". The text explains that users should sign up to use the fast and easy-to-work weather APIs. It recommends the "One Call API 3.0" for more functionality. A list of features for the One Call API 3.0 is provided: Minute forecast for 1 hour, Hourly forecast for 48 hours, Daily forecast for 8 days, Historical data for 40+ years back by timestamp, and National weather alerts. The pricing is "Pay as you call" with "1,000 API calls per day for free" and "0.0012 GBP per API call over the daily limit". A "Subscribe to One Call by Call" button is present. Below this, the "Professional collections" section is introduced, recommending professional collections for middle-sized projects, including Current & Forecasts collection, Historical weather data collection, Weather Maps collection, and other APIs.



The screenshot shows the "Weather in your city" page on OpenWeatherMap. The header is identical to the previous screenshot. The main heading is "Weather in your city". Below the heading is a search bar with the text "chennai" and a "Search" button. The search results show the weather for Chennai, IN, with a "mist" condition. The temperature is 25°C, with a range from 25 to 25 °C, wind 1.03 m/s, clouds 75 %, and 1012 hpa. The geo coordinates are [13.0878, 80.2785]. Below the search results, the text "Search engine is very flexible. How it works:" is followed by a list of instructions: To make it more precise put the city's name, comma, 2-letter country code (ISO3166). You will get all proper cities in chosen country. The order is important - the first is city name then comma then country. Example - London, GB or New York, US.

requests

```
a = "https://api.openweathermap.org/data/2.5/weather?q=Chennai,TN&appid=6d13d12f9cd34a07871a5795d01e2c47"
```

```
r = requests.get(us = a)
```

```
data = r.json()
```

```
pnnt(r)
```

```
pnnt(data)
```

```
temp = data["main"]["temp"]
```

```
hum = data["main"]["humidity"]
```

```
pnnt("Temperature is : ",temp)
```

```
pnnt("Humidity is : ",hum)
```

```
Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
<Response {200}>
temperature is : 298.14
»
===== RESTART: E:/IBM/pre/weatherMap.py =====
<Response {200}>
{'coord': {'lon': 80.2785, 'lat': 13.0878}, 'weather': [{'id': 701, 'main': 'Mist', 'description': 'mist', 'icons': '50n'}], {'id': 500, 'main': 'Rain', 'description': 'Tight rain', 'icons': '10n'}], 'base': 'stations', 'main': {'temp': 298.14, 'feels_like': 299.15, 'temp_min': 298.14, 'temp_max': 298.14, 'pressure': 1012, 'humidity': 94}, 'visibility': 2500, 'wind': {'speed': 1.54, 'deg': 350}, 'clouds': {'all': 75}, 'dt': 1667317416, 'sys': {'type': 1, 'id': 9218, 'country': 'TN', 'sunrise': 1667262751, 'sunset': 1667304738, 'timezone': 19800, 'ids': 1264527, 'name': 'Chennai', 'code': 200}
temperature is : 298.14
Humidity is : 94
```