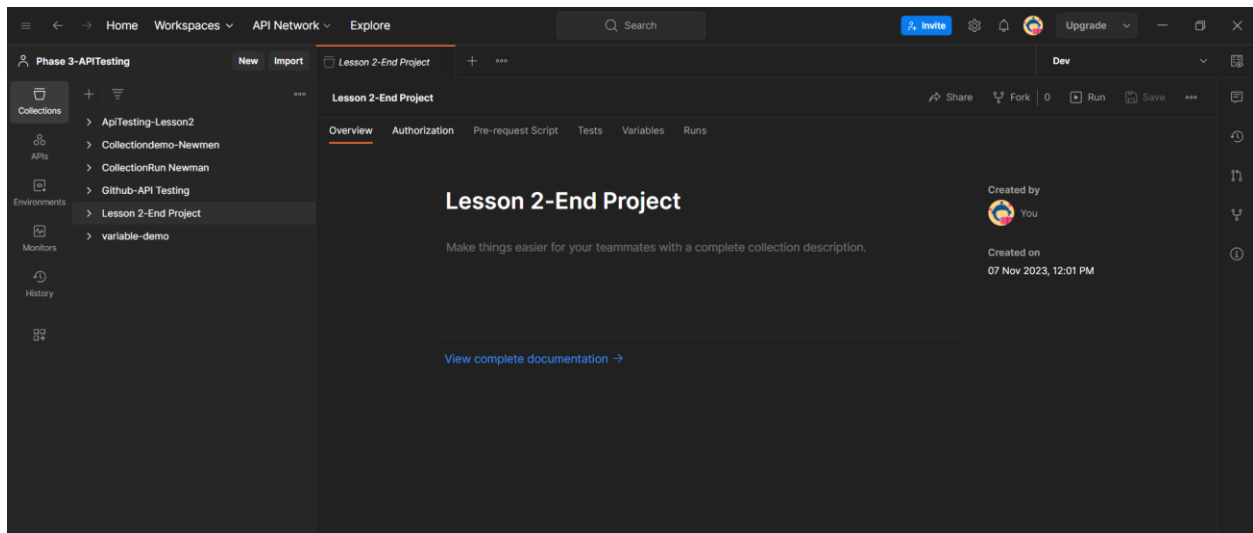


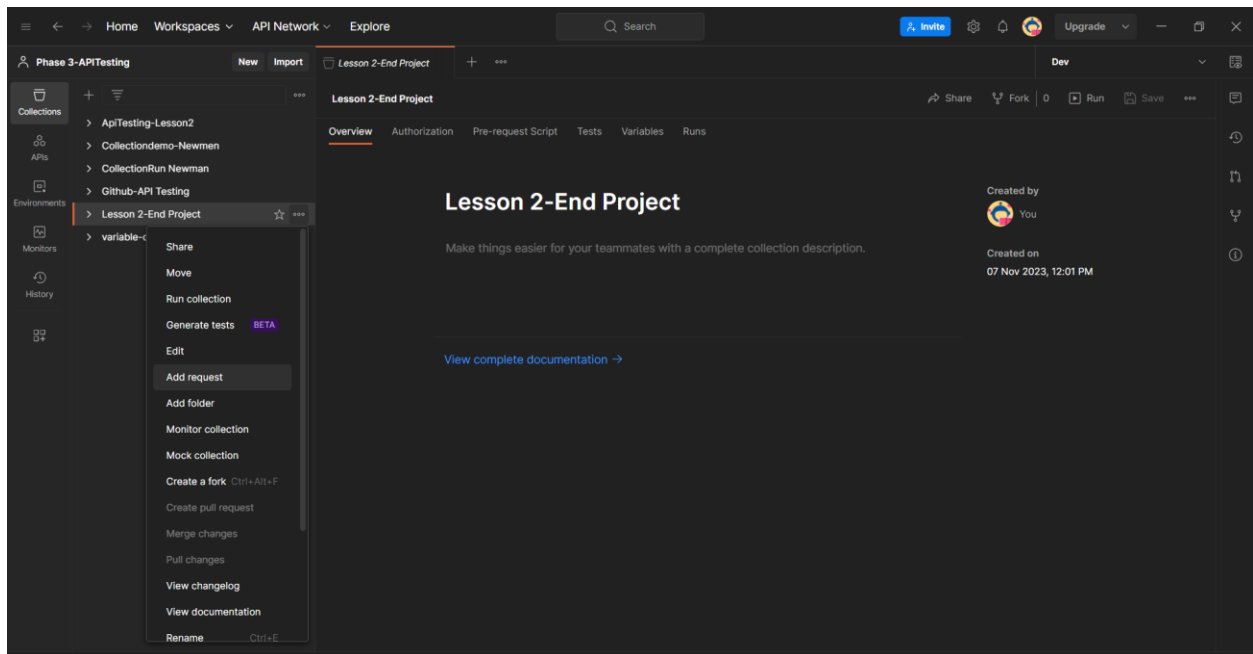
Lesson-end Project : Creating Well-Structured .Output for API Clients Using Postman to Get Weather Report.

(Writeup/Source Code):

With the first we will create a Blank collection named as Lesson 2-End Project as below



For step 2: we will right click on the collection Lesson 2-End Project and Add request



After that we will go to the below link

<https://openweathermap.org/api>

Signup and login to this application

Now create API keys -> go to your account -> click on my API keys

The screenshot shows the OpenWeather API dashboard. At the top, there's a navigation bar with links like Guide, API, Dashboard, Marketplace, Pricing, Maps, Our Initiatives, Partners, Blog, For Business, Diks..., and Support. Below this, a sub-navigation bar includes New Products, Services, API keys, Billing plans, Payments, Block logs, My orders, My profile, and Ask a question. A message states: "You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them." Below this is a table of API keys:

Key	Name	Status	Actions
9ac152fe94812dc749193708b0bbe2f0	Default	Active	Toggle Edit Delete
d18b84521ee5451b16029e5356464351	APIkeypostman	Active	Toggle Edit Delete

To the right of the table is a "Create key" form with a text input for "API key name" and a "Generate" button. Below the table, there are three sections: "Product Collections" (Current and Forecast APIs, Historical Weather Data, Weather Maps, Weather Dashboard), "Subscription" (How to start, Pricing, Subscribe for free), and "Company" (OpenWeather is a team of IT experts and data scientists that has been practising deep weather data science since 2014. For each point on the globe, OpenWeather provides historical, current and forecasted weather data via REST and API. Headquarters in London, UK).

The api request we will use is :

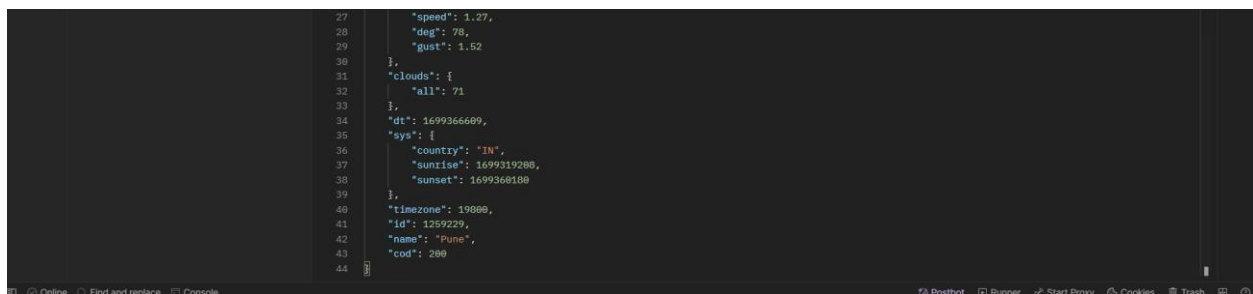
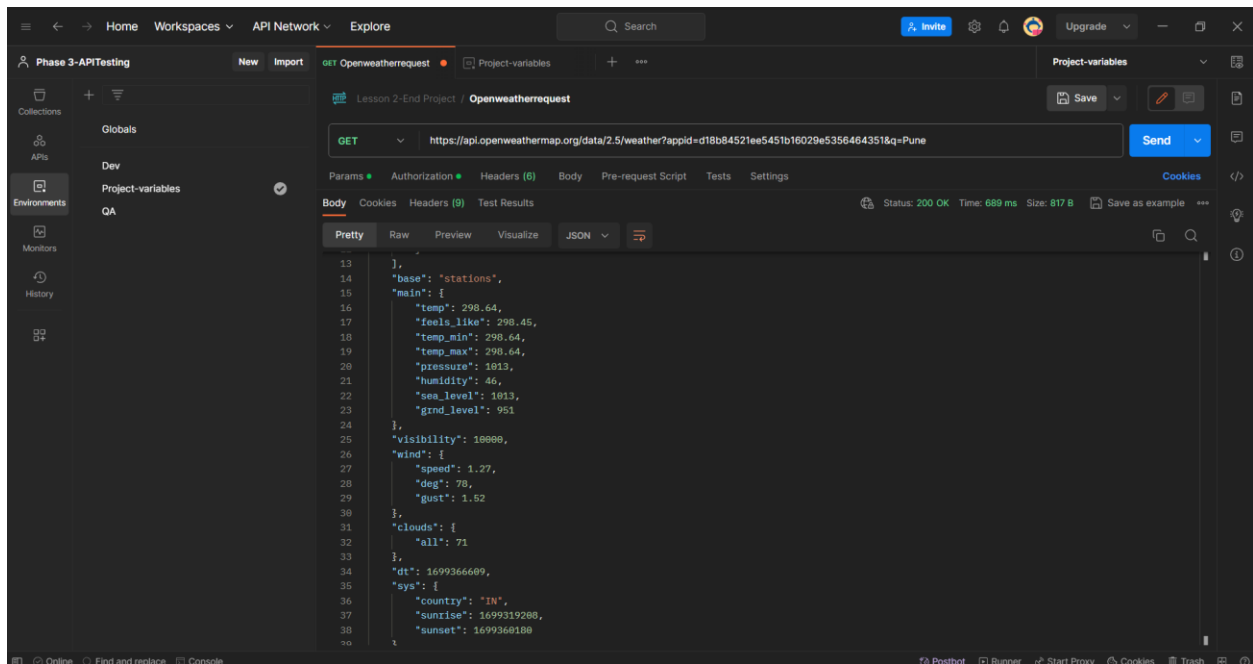
<https://api.openweathermap.org/data/2.5/weather?q=London&appid={give your api key}>

As we click on Send we get the Response output with weather detail of city given

The screenshot shows a REST client interface with a "Phase 3-API Testing" workspace. A "GET Openweatherrequest" is selected. The URL is `https://api.openweathermap.org/data/2.5/weather?appid=d18b84521ee5451b16029e5356464351&q=Pune`. The "Query Params" section shows three parameters: "appid" (value: d18b84521ee5451b16029e5356464351), "appid" (value: d18b84521ee5451b16029e5356464351), and "q" (value: Pune). The "Body" tab is selected, showing the JSON response in "Pretty" format:

```
1 {
2   "coord": {
3     "lon": 73.8553,
4     "lat": 18.5196
5   },
6   "weather": [
7     {
8       "id": 803,
9       "main": "Clouds",
10      "description": "broken clouds",
11      "icon": "04n"
12    }
13  ],
14 }
```

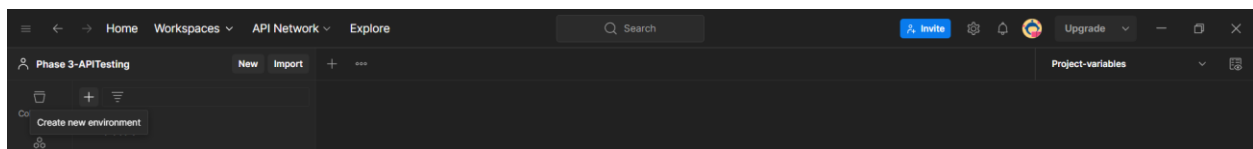
The status bar at the bottom indicates "Status: 200 OK", "Time: 689 ms", and "Size: 817 B".



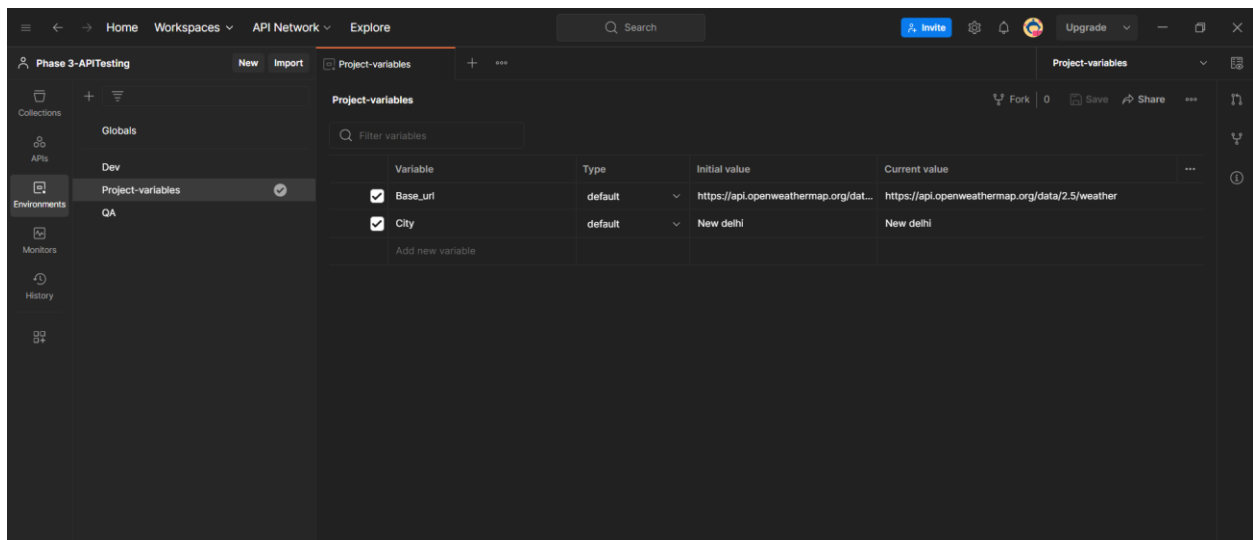
Variables & environments in Postman:

Variables are created using Environments in Postman, Environment can be considered as we created a Base file in java package.

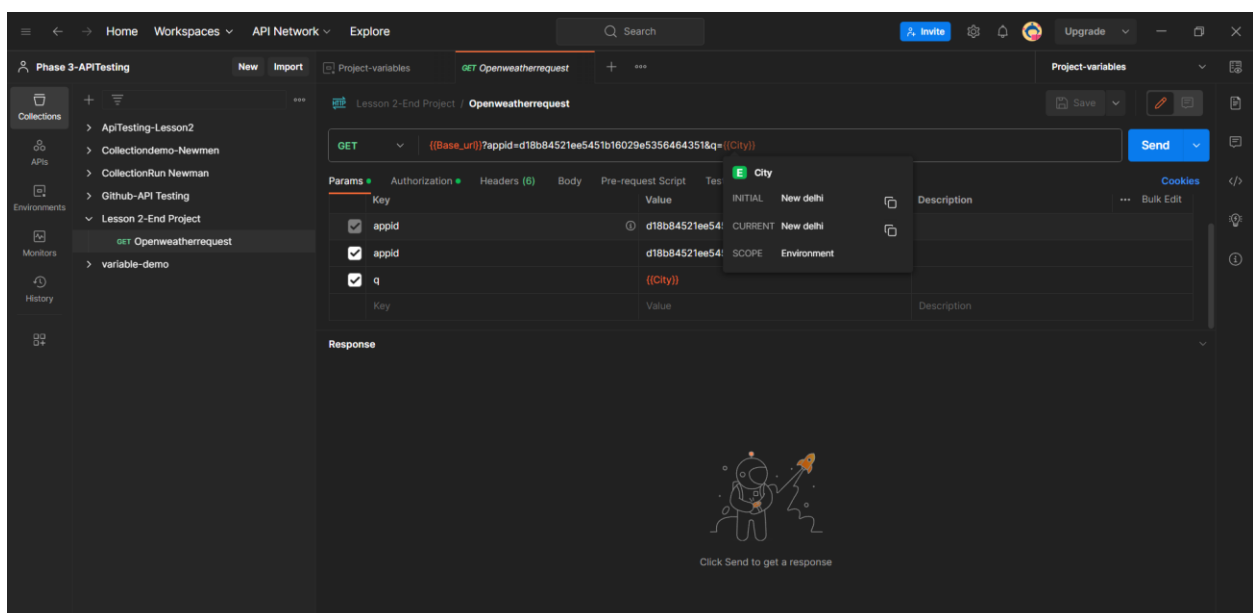
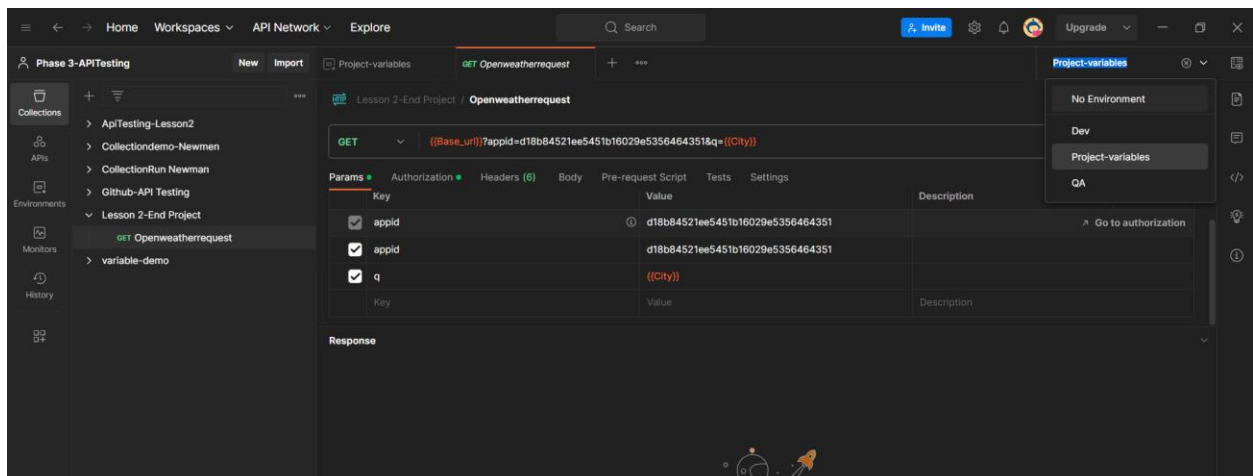
To create a environment we have click on environment on left side and the click on (+) create new environment as shown below:



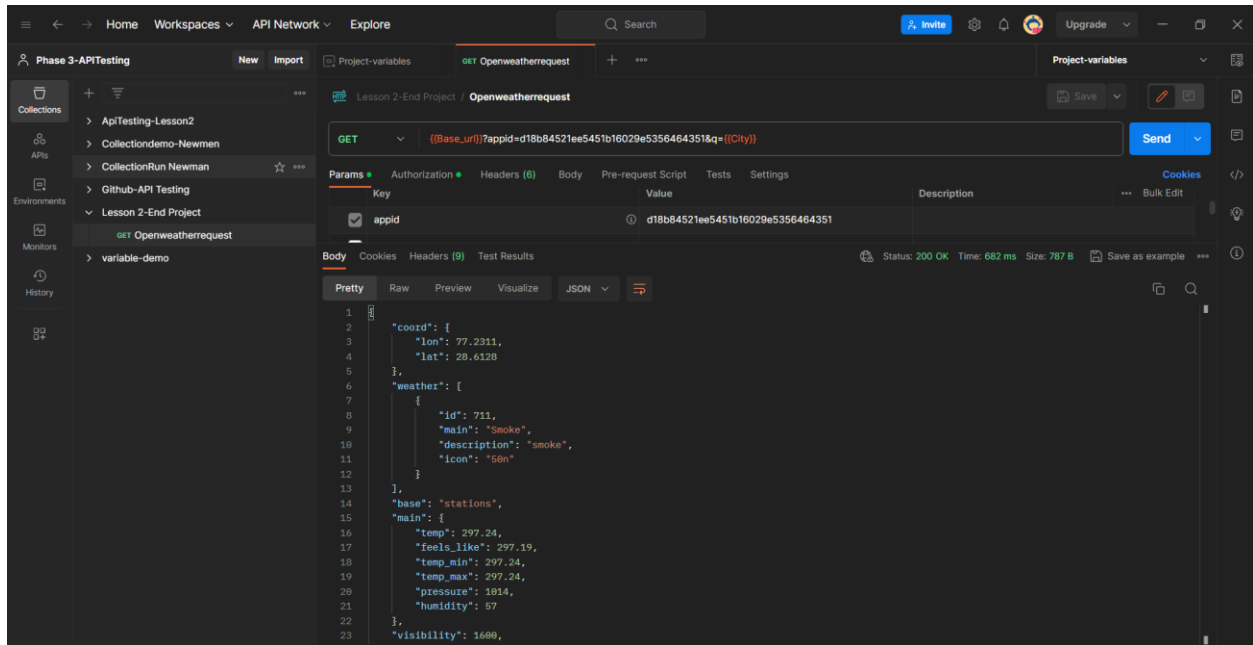
Add variables as shown below and save the environment



Replace the URL with the variable, and select the environment from top right

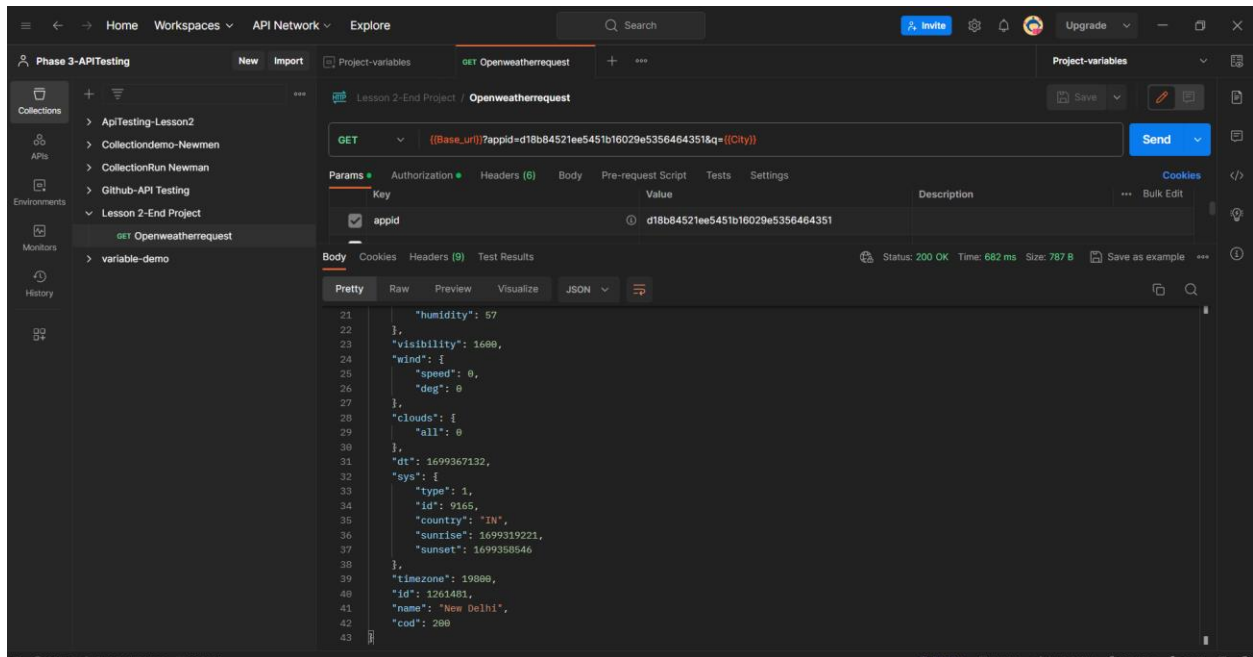


Save and Send the above request for response output



The screenshot shows the Postman interface with a GET request to the OpenWeather API. The request is saved and sent, resulting in a 200 OK status. The response body is displayed in JSON format, showing weather data for a specific location.

```
1 {
2   "coord": {
3     "lon": 77.2311,
4     "lat": 28.6128
5   },
6   "weather": [
7     {
8       "id": 711,
9       "main": "Smoke",
10      "description": "smoke",
11      "icon": "50n"
12    }
13  ],
14  "base": "stations",
15  "main": {
16    "temp": 297.24,
17    "feels_like": 297.19,
18    "temp_min": 297.24,
19    "temp_max": 297.24,
20    "pressure": 1014,
21    "humidity": 57
22  },
23  "visibility": 1600,
```



The screenshot shows the Postman interface with the same GET request to the OpenWeather API. The request is saved and sent, resulting in a 200 OK status. The response body is displayed in JSON format, showing weather data for a specific location.

```
21 "humidity": 57
22 },
23 "visibility": 1600,
24 "wind": {
25   "speed": 0,
26   "deg": 0
27 },
28 "clouds": {
29   "all": 0
30 },
31 "dt": 1699367132,
32 "sys": {
33   "type": 1,
34   "id": 9165,
35   "country": "IN",
36   "sunrise": 1699319221,
37   "sunset": 1699358546
38 },
39 "timezone": 19800,
40 "id": 1261481,
41 "name": "New Delhi",
42 "cod": 200
43 }
```

