**API Scavenger Hunt**

**GitHub URL: https://github.com/ruchit-t/API-Scavenger-Hunt.git**

**TASK 1: OpenWeatherMap API**

**API Key: 3fcb7ef462dadf2435d8b7fd903ed828**

1. The current weather for London, United Kingdom

Code Snippet:

A screenshot of a computer

Description automatically generated

Code (Raw Text):

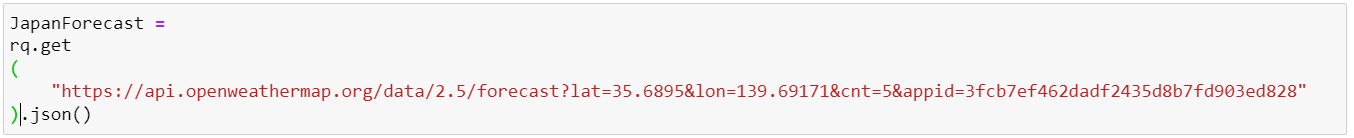
Result in JSON & Plain Text:

A screenshot of a computer

Description automatically generated

1. Retrieve the 5-day forecast for Tokyo, Japan

Code:



Code (Raw Text):

The entire result in JSON:

A close-up of a text

Description automatically generated

Weather result in plain text format:

A screenshot of a computer code

Description automatically generated

**Reflection:**

OpenWeatherAPI is really easy to use. All we need to have is, the API key which is available for free. Apart from the current and forecast weather data, the API provides other endpoints as well such as Hourly forecast 4 days, Daily Forecast 16 days, Climatic Forecast 30 days, and Global Weather Alerts Push notifications. This API can be used in the case of Weather and Disaster Management. Users can utilize the data for weather forecasts, emergency response, and disaster management.

These APIs are really useful if we are making a weather application for iOS/Android. The push notifications feature makes this API really useful as it can notify users for the latest updates. This API is cheap if we look at it from the commercial perspective. It offers free API services for 1,000 API calls per day. Beyond that it just costs 0.0015 USD per API call.

**TASK 2: Google Maps API**

**API Key: AIzaSyDgYQ0mCjO-0G3v0IhzI0uduiYD0-R6uJo**

1. Display a map centered on New York City, USA

Code:

A screenshot of a computer

Description automatically generated

Code (Raw Text):

A map of a city

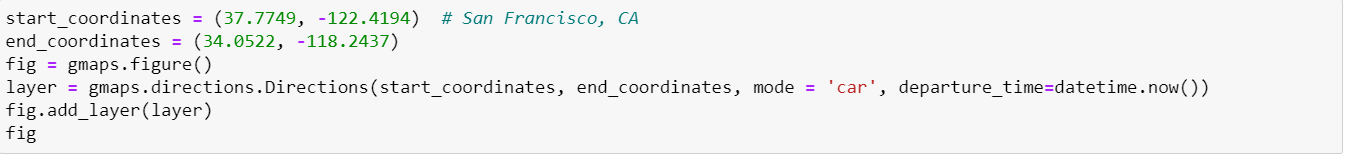
Description automatically generatedResult:

1. Find the shortest route by car between San Francisco, USA, and Los Angeles, USA

Code:

1. Find the shortest route by car between San Francisco, USA, and Los Angeles, USA

Code Snippet:



Code (In Raw Text)

Result:

A map of a city

Description automatically generated

**Reflection:**

The Google Map API is the most sophisticated among the four APIs we are using for this assignment. Since Google is the parent of this API, it offers a rich documentation along with different use cases. It also offers several types of MAP visualization such as satellite, terrain, 2D/3D map etc. It provides accurate and up-to-date mapping and geolocation data, which is essential for a wide range of applications, from navigation to location-based services. It is relatively easy to integrate Google Maps into websites, mobile apps, and other platforms. Also, Google's infrastructure can handle a large number of requests, making it suitable for both small-scale and large-scale applications.

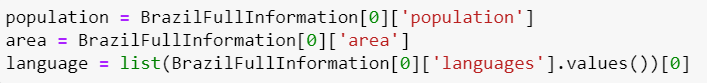
Just like the second example, I found it useful for the directions and routing. Users can calculate routes, provide turn-by-turn directions, and estimate travel times using Google Maps API, making it useful for applications like ride-sharing services and logistics. This is one of the potential applications of this API that the users can use. This API can also be used in the ride sharing and delivery services by enabling drivers and users to find each other and calculate optimal routes for deliveries or rides.

**TASK 3: REST Countries API**

1. Retrieve information about Brazil, including its population, area, and official language.

Code:





Code (In Raw Text):

Result:

In JSON Format:

A close-up of a text

Description automatically generated

In Plain text format:

A close-up of a computer screen

Description automatically generated

1. Retrieve a list of all countries in Africa.

Code:

A screenshot of a computer code

Description automatically generated

Code (In Raw Text):

Result:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Reflection:**

As this API does not require any API keys to access the data, this can be the most sought API service developers can use if they are specifically looking for country, continent, or territorial information. I am particularly familiar with this API as I already used it in one of my previous major projects (CoVID-19 Dashboard). It provides a wide range of information about countries, including details like population, area, languages spoken, currencies, time zones, and more.

Possible use cases could be the travel and tourism application. Developers can use it to offer information about countries, including popular destinations, culture, and travel advisories. It can also be used to integrate country data into language learning apps to help users explore countries where their target language is spoken. Since it also provides time zone related information, developers can create time zone conversion tools or world clocks that use the API to determine time differences between countries.

**TASK 4: Currency Converter API**

API Endpoint used: [**https://currency-converter5.p.rapidapi.com/currency/convert**](https://currency-converter5.p.rapidapi.com/currency/convert)

API Key: **db066214b2msha0e8e9404fcc0ebp1e68f3jsnba646e8cda5f**

1. Convert 100 USD to EUR

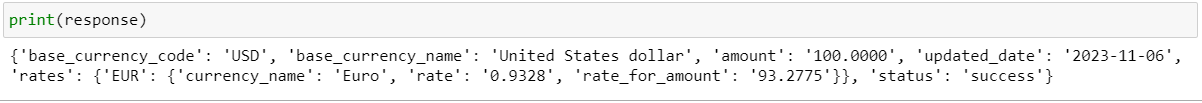
Code:

A screenshot of a computer

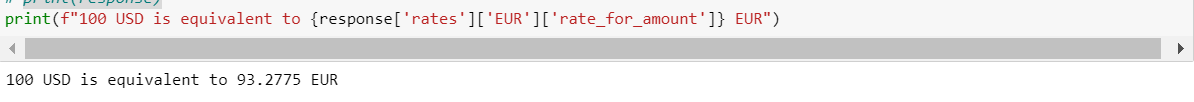
Description automatically generated

Code (In Raw Text):

Result: (In JSON Format)



Result: (In Plain Text)



1. Convert 1000 JPY to GBP

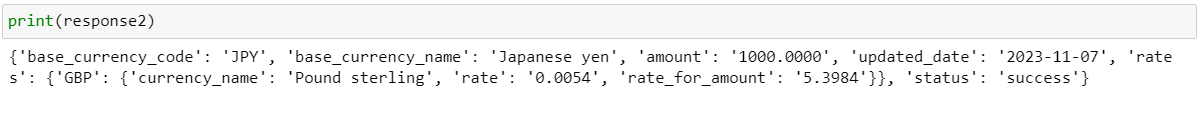
Code:

A screenshot of a computer code

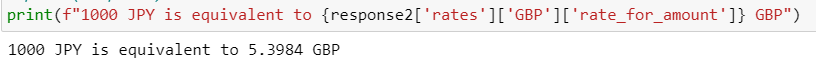
Description automatically generated

Code (In Raw Text):

Result: (In JSON Format):



Result: (In Plain Text)



**Reflection:**

The biggest advantage of the RAPID API Currency Services is that it provides the sample code and documentation for easy and fast development. The API can be tested within the RAPID API environment before the in-app integration. In my opinion, it saves a lot of time during development of an application. It also provides real-time currency data, which is important for applications that require up-to-the-minute currency conversion. As the API tracks the real-time updates, it eliminates the need to manually update exchange rates, saving time and ensuring accuracy. It also supports all the major and popular currencies, along with proper IN-API conversion. The Users can just pass the source and destination currency names along with the amount, and this API will calculate the converted value. It also supports historical exchange rate data, which can be useful for financial analysis and reporting.

The potential use cases could be Financial Services, Forex Trading, Accounting, and Invoicing, E-commerce Websites etc. The developers can implement currency conversion on e-commerce platforms to display prices in the user's preferred currency, enhancing the shopping experience for international customers. Its capability can be used for Currency conversion APIs are crucial for forex trading platforms to provide real-time exchange rate data for traders. Business Intelligence could be the biggest advantage from the analytical perspective. The analysts can incorporate exchange rate data into business intelligence and data analytics applications to assess currency-related impacts on business operations.

Reference:

Python Jupyter Notebook File:

https://github.com/ruchit-t/API-Scavenger-Hunt/blob/main/API%20Scavenger%20Hunt.ipynb

https://drive.google.com/file/d/1UTloGLzppvUvuxAABS6V\_lW2ArDC0COg/view?usp=sharing