**-API Scavenger Hunt-**

**- Assignment by Shounak Kulkarni (CUID – C56298850) -**

[**https://github.com/ShounaKulkarni/api-scavenger-hunt**](https://github.com/ShounaKulkarni/api-scavenger-hunt)

**Task1: Solution.**

1. **current weather for London, United Kingdom –**

**Code & output screenshot –**

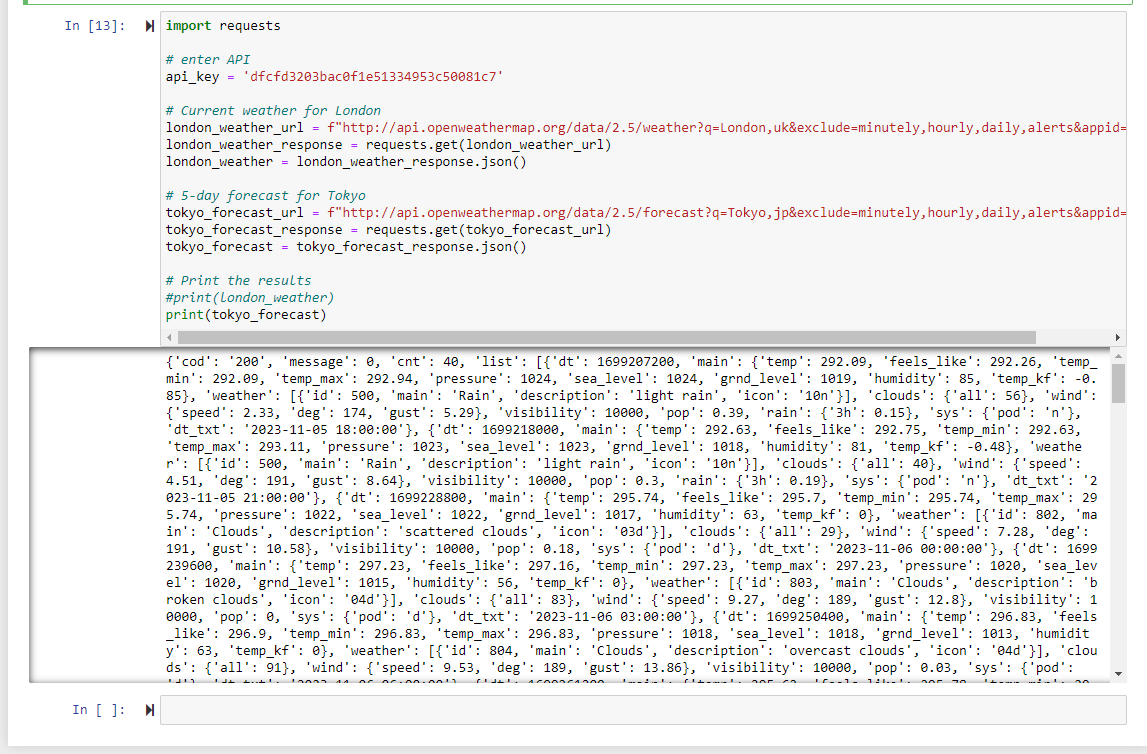
****

1. **5-day forecast for Tokyo, Japan –**

**Code & output screenshot –**

It is not possible to display 5 day data all within a single screenshot. However, I have compiled the entire output into an HTML file titled **task1.html** below. The complete output is also accessible through a Jupyter Notebook file on GitHub. The link to this repository is provided on the first page of this document.





**Reflection on Using the OpenWeatherMap API**

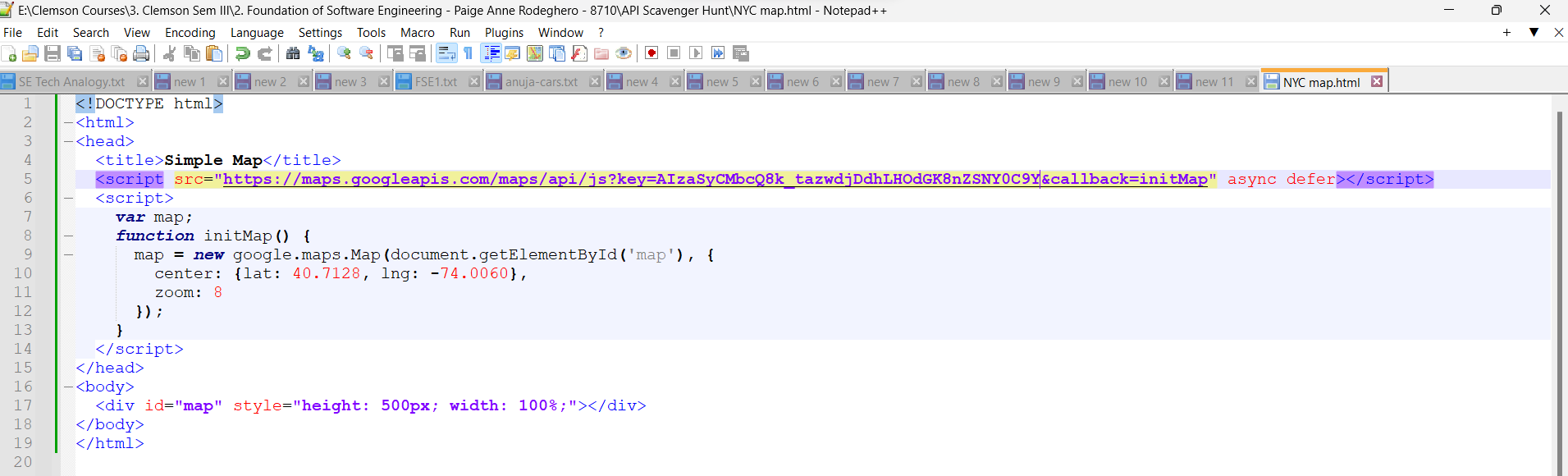
My experience with the OpenWeatherMap API was quite positive, marked by the ease of obtaining the API key and the simplicity of the API's interface. Registering on their platform and navigating through the API documentation was straightforward, allowing me to quickly move on to making actual API calls. The documentation was quite clear and comprehensive, which made it easy for me to compose the request and retrieve the current weather for London. Because of the JSON response's organization, it was simple to extract the pertinent meteorological data.

The procedure was as simple to utilize to retrieve the Tokyo 5-day forecast. With just a GET request, users could receive a comprehensive forecast from the API that includes numerous weather indicators like temperature, humidity, and precipitation. The level of detail in the data for every day was astounding, providing insights that might be used to everything from agricultural platforms that depend on weather forecasts to trip planning apps.

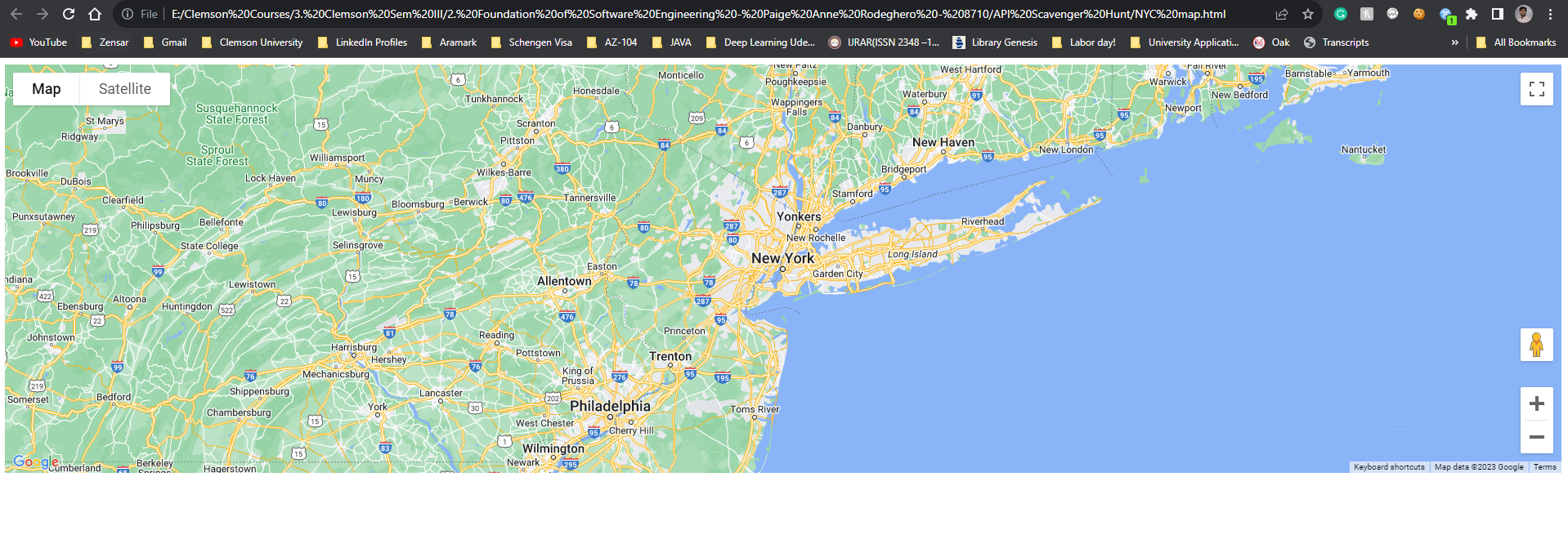
When considering the OpenWeatherMap API's overall capability, a wide range of capabilities that can meet different purposes are provided. With a daily cap of 1,000 free API requests, developers have a great chance to incorporate meteorological data into their apps without worrying about sudden expenses. This API has a wide range of possible uses, from augmenting event planning platforms with weather forecasts to incorporating weather data into smart home systems. The OpenWeatherMap API is an effective tool for developers wishing to incorporate real-time weather elements into their applications because of its simplicity of use and the breadth of data it offers.

**Task2: Solution –**

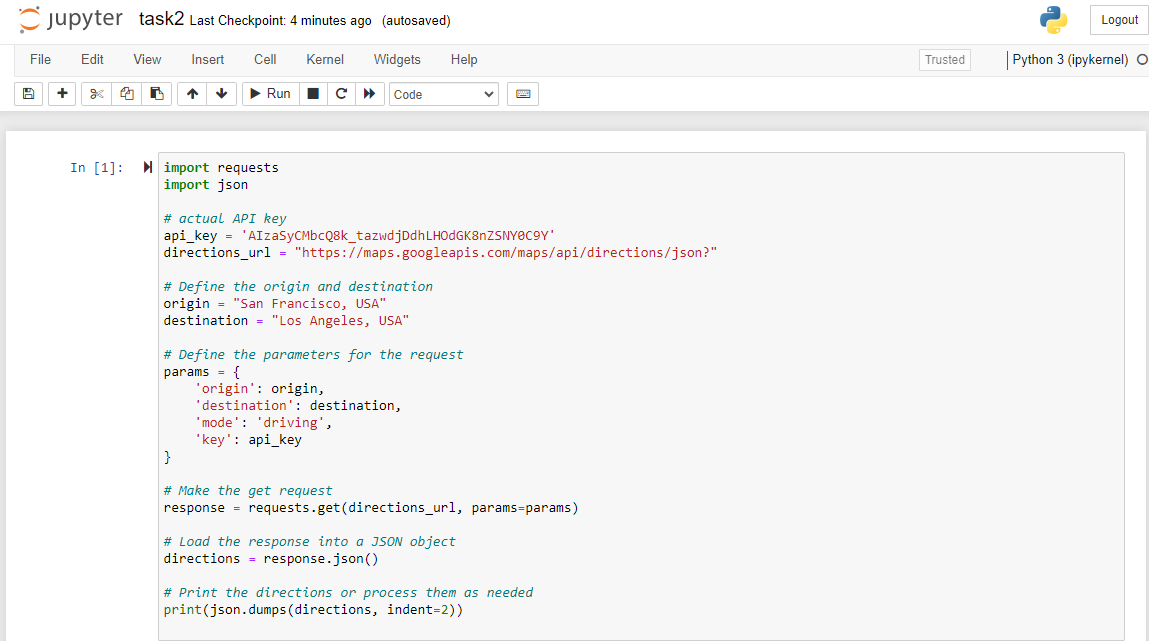
1. **Map centered on New York City, USA -**







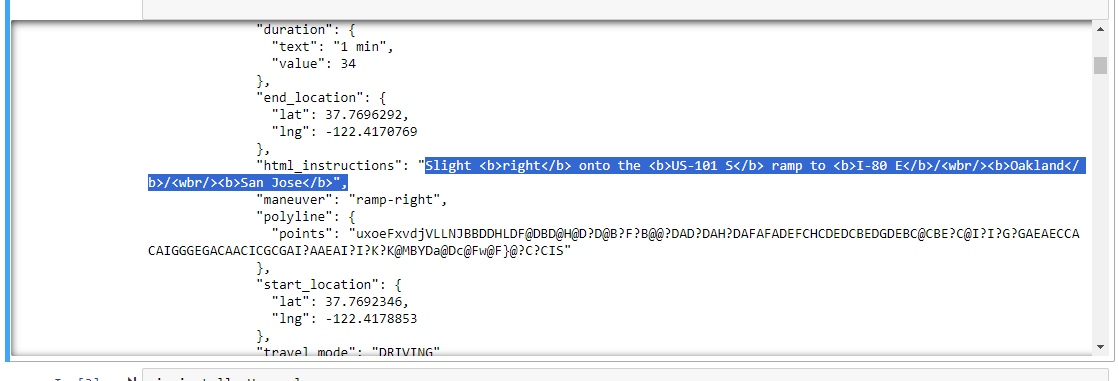
1. **shortest route by car between San Francisco, USA, and Los Angeles, USA –**

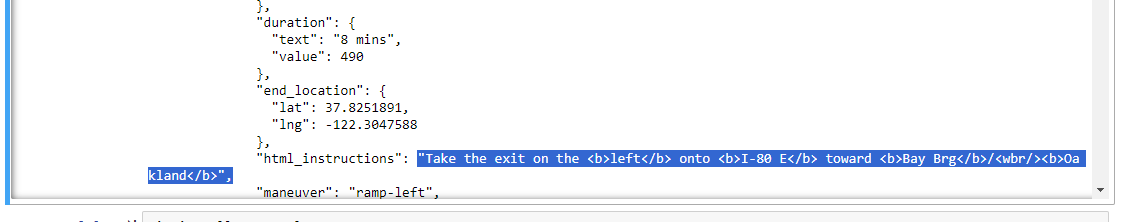
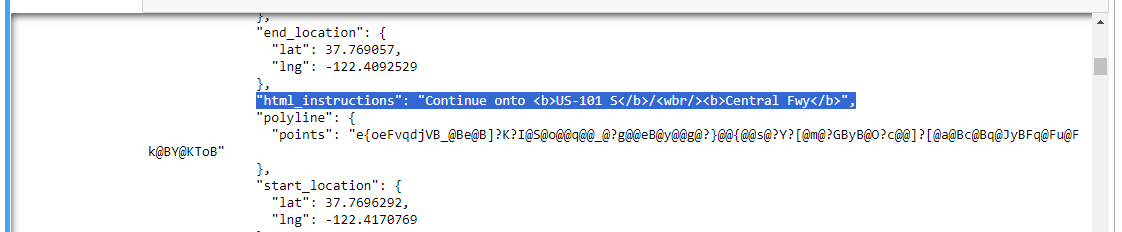


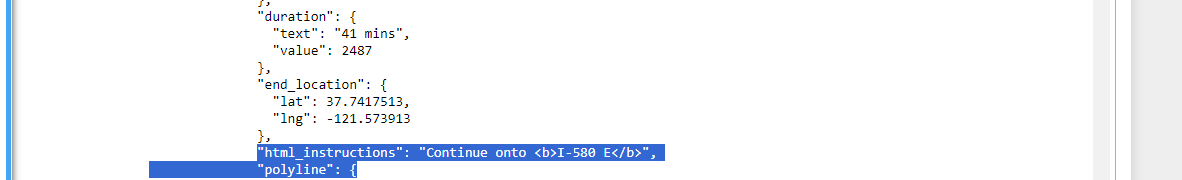
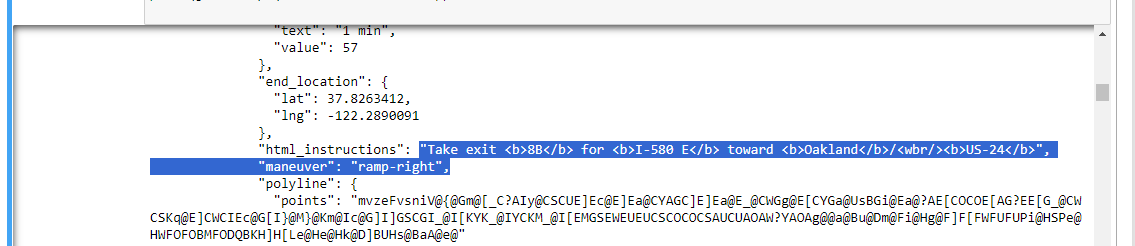
**OUTPUT –** However, I have compiled the entire output into an HTML file titled **task2.html** below. The complete output is also accessible through a Jupyter Notebook file on GitHub. The link to this repository is provided on the first page of this document.

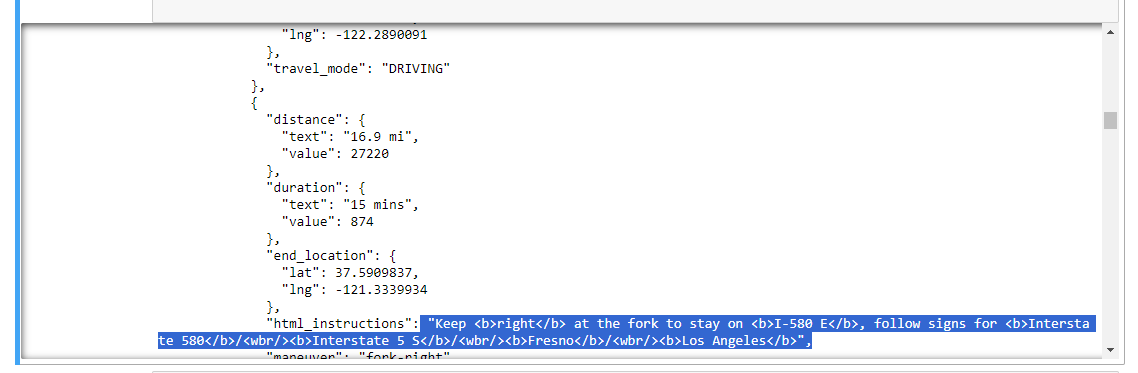


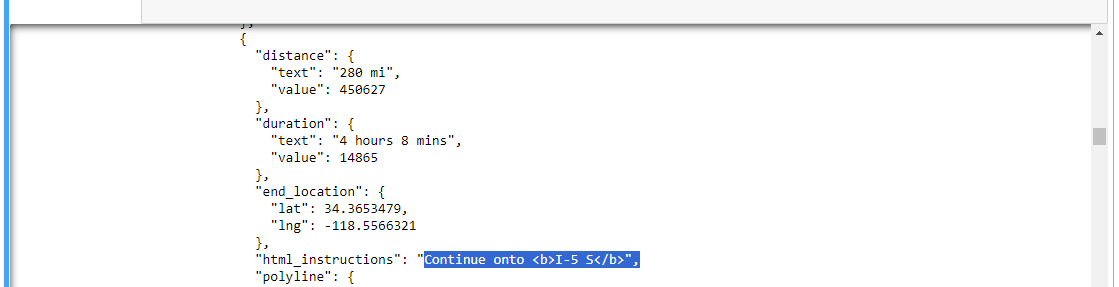


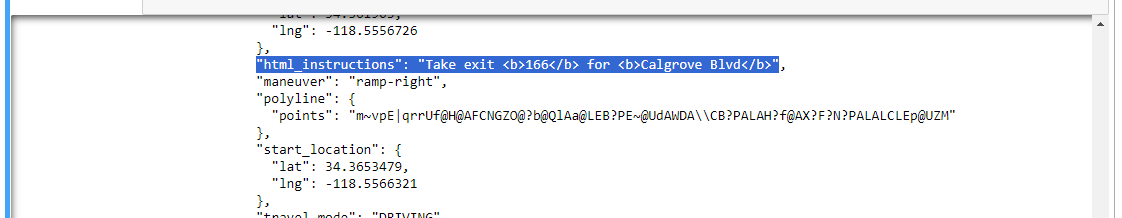


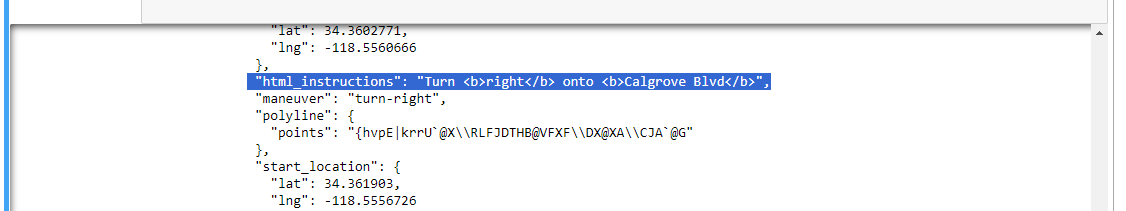


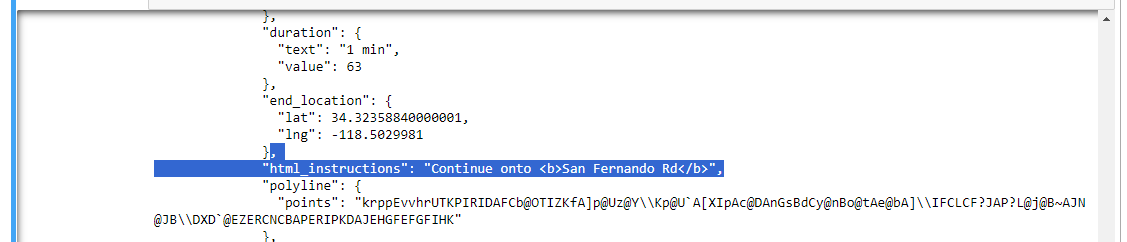


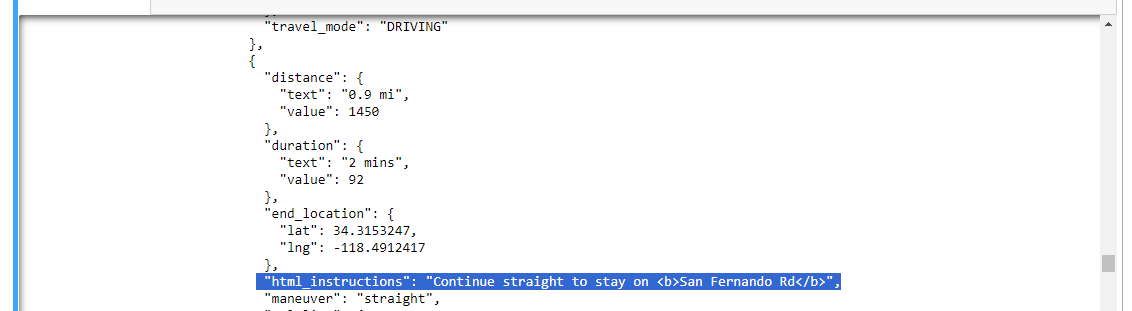


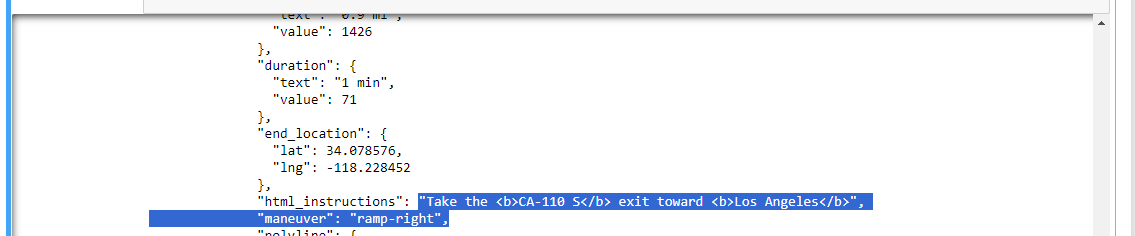


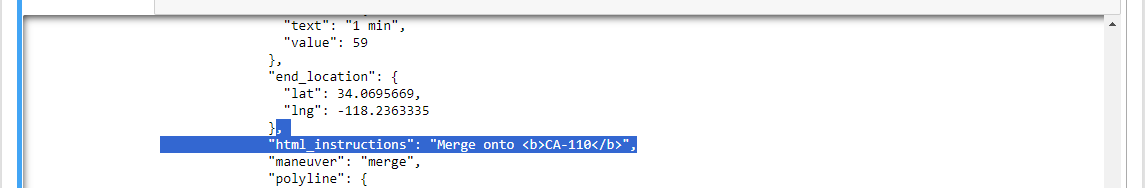


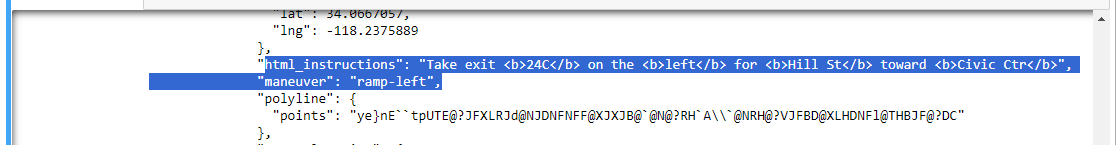


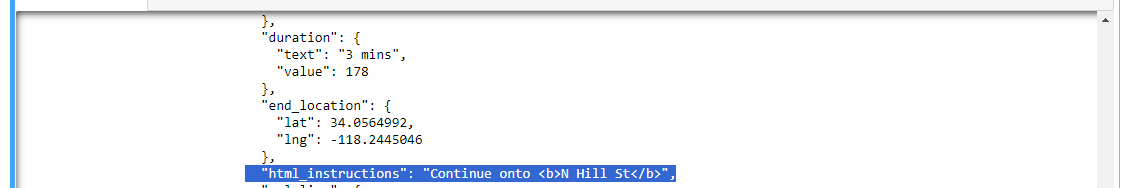














**Reflection on Using the Google Maps API**

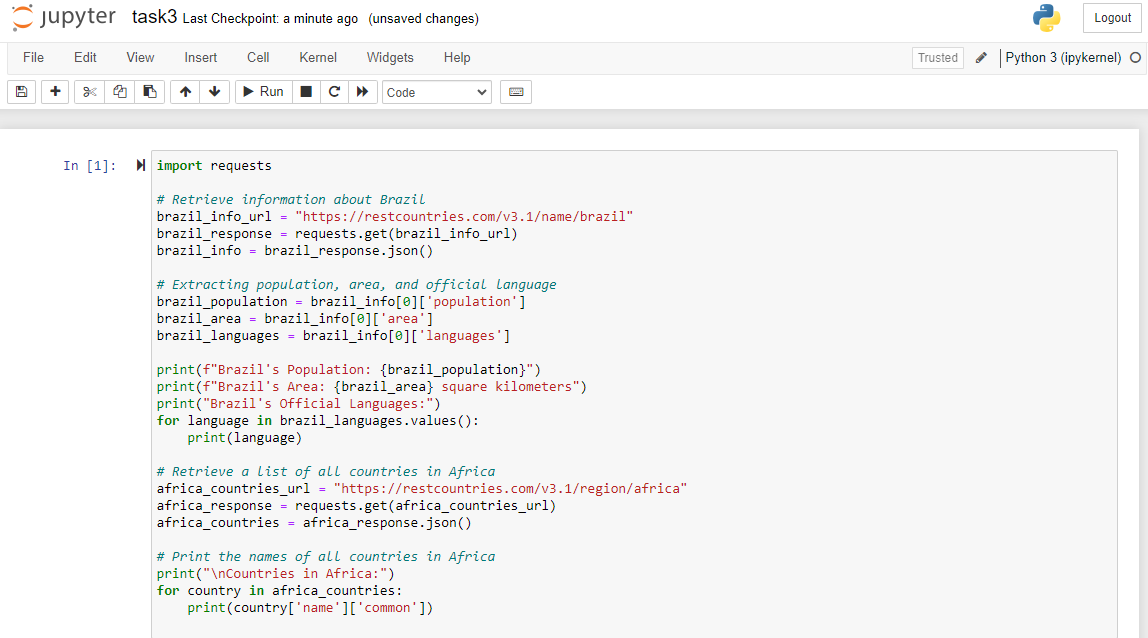
Diving into the Google Maps API was an engaging and educational journey. Thanks to Google's comprehensive developer documentation, getting an API key was a simple task. It only took a few lines of code to implement a map centered on New York City, demonstrating how user-friendly the API is. It was simple to adapt the look and behavior of the map to the demands of the current assignment thanks to Google's extensive documentation. The development process was made easier by the Google Cloud Platform's user-friendly interface and the responsiveness of the API.

The challenge of determining the quickest road route between San Francisco and Los Angeles by car demonstrated the API's strong route optimization capabilities. Building navigational aids or logistics software may benefit greatly from the precise instructions and real-time traffic data provided by the Directions API, which is a component of Google Maps services. User experiences in delivery services, trip planning, and other areas can be revolutionized by the ability to incorporate such intricate mapping and routing features into applications.

Reflecting on the overall experience, Google Maps API stands out for its extensive features and reliability. The potential applications are vast, ranging from real estate platforms showcasing property locations to social apps suggesting meeting points based on user location. The ease of use, combined with the powerful capabilities of the API, makes it an indispensable tool for developers looking to incorporate geographical mapping and intelligent routing into their applications. The experience has left me with a deeper appreciation for the sophistication of Google's mapping services and the myriad ways they can be leveraged in software development.

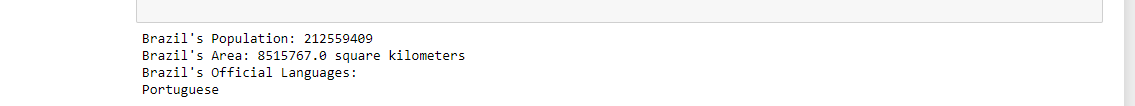
**Task3: Solution –**

**Code snippet –**

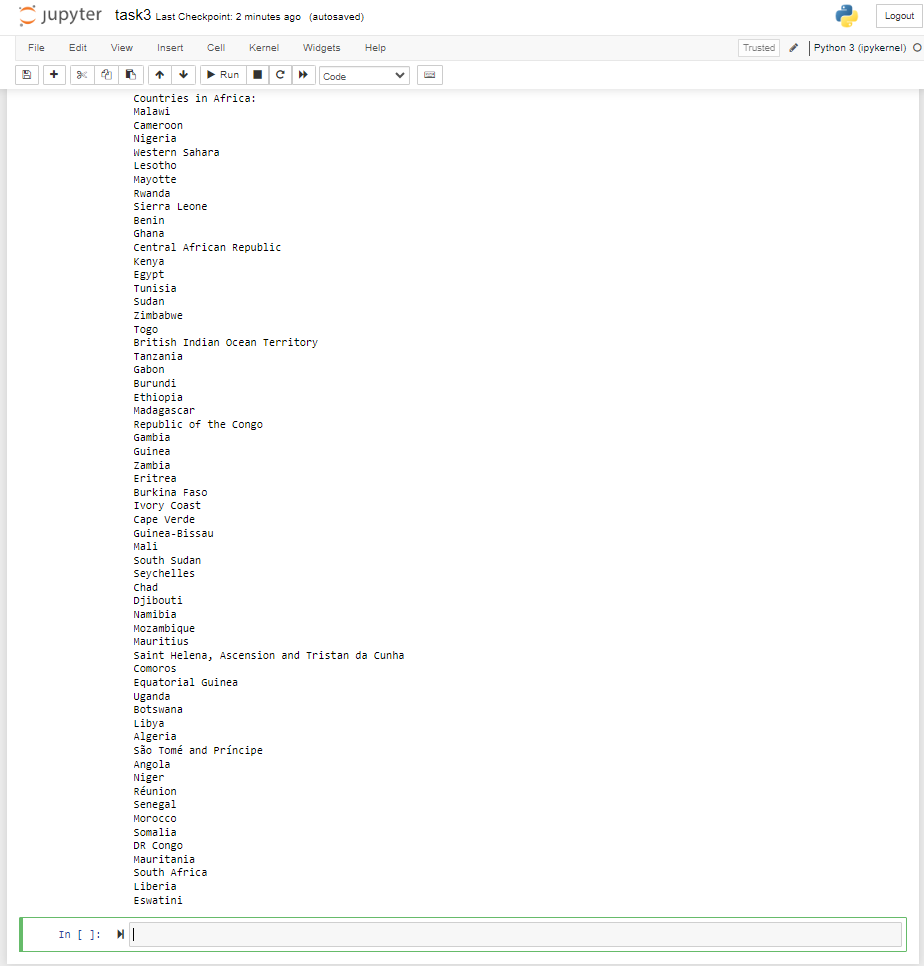


**Output Screenshot –**

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



1. **list of all countries in Africa-**



**Reflection on Using the REST Countries API -**

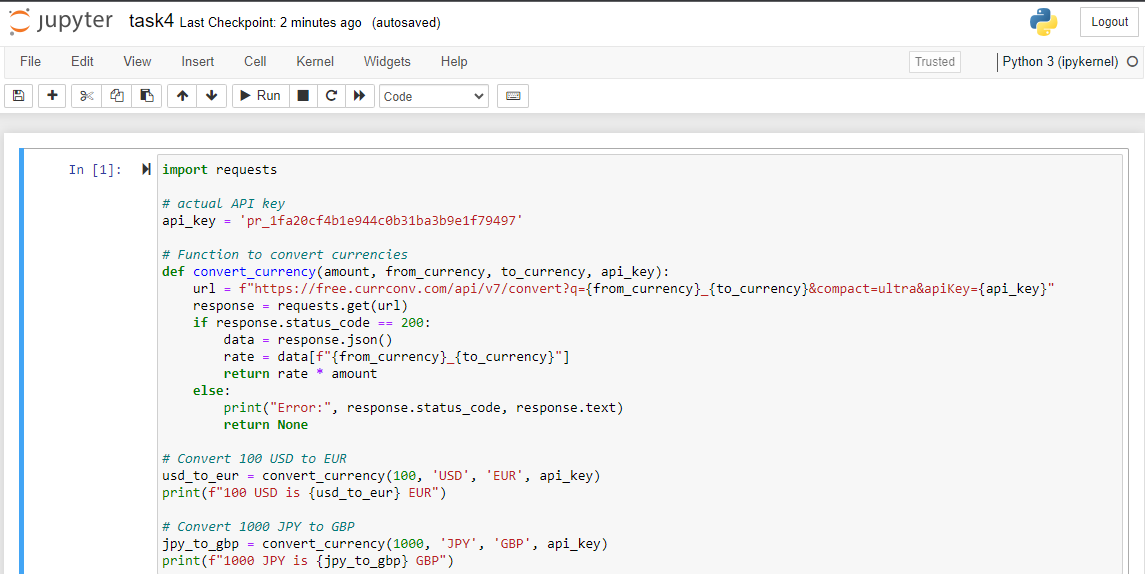
My experience using the REST Countries API was quite simple. The absence of an API key requirement facilitated immediate access, allowing me to focus on exploring the API's capabilities without preliminary hurdles The documentation was very easy to use and included clear instructions on how to utilize the endpoints, which made getting data about the languages and demographics of Brazil a simple process. The developer-friendly design of the API was demonstrated by how simple it was to extract and parse the data from its neatly organized JSON response.

The API's quick delivery of a list of African nations demonstrated its potential for use in applications like market analysis tools or educational platforms that need to segment data based on area. The vast dataset made available by the simple API requests might be used for a variety of purposes, such as demographic research, content localization, or even as the basis for an educational tool that focuses on geography.

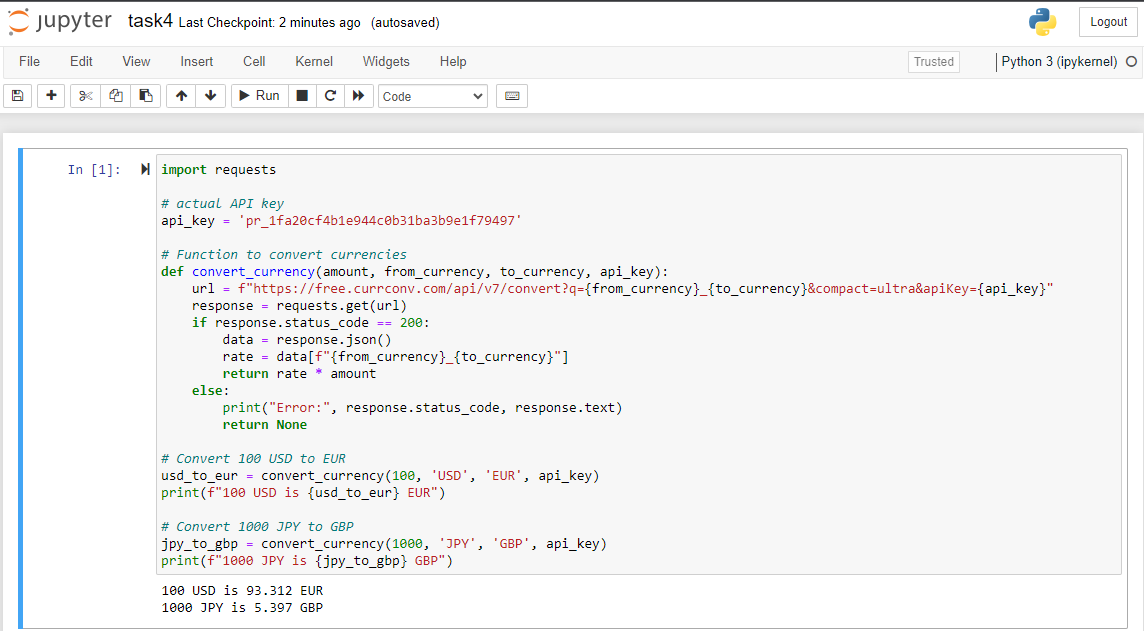
In conclusion, I was rather satisfied with the REST Countries API's usability and range of data it offers. Because of its simple methodology and extensive data set, developers have a plethora of options when it comes to incorporating a global perspective into their apps. My knowledge of RESTful APIs has expanded because of the experience, and it has also given me ideas for new applications across a range of industries.

**Task4: Solution –**

**Source code –**



**OUTPUT –**



**\*\*\*\*\*Reflection of task 4:**

After completing the tasks,

The Currency Converter API was extremely easy to use, with clear, well-documented endpoints that made integrating it with my Python script a snap. From receiving the API key to carrying out currency conversions, the procedure was easy to follow and included clear examples to ensure a hassle-free experience. Fast API response times and simply comprehensible data formats are essential for developers who want to provide functionality with the least amount of complexity.

Application-wise, the API provides fundamental features that are readily extensible to real-world use cases, such real-time currency conversion for financial services or e-commerce platforms. Even with its restrictions, the free tier is a great place to start learning about how financial data is integrated and used in applications, and its premium plans offer the possibility to expand into even more sophisticated features. I could not find any free API key so I did purchase a 6$ subscription as deadline was approaching and free API key needed 3-5 business day.