**MNTN - SDET Assessment**

1) I have chosen public REST API named ‘OMDb API’ for this assessment. This is a RESTful webservice to obtain movie information and movie ratings given by several rating websites. The link is <https://www.omdbapi.com/>

2) I have attached the executable python file as well, which has the code.

*My system environment details:*

OS: Ubuntu 21.10

Python version: 3.10.1

goutham@ubuntu:~$ ll test\_api.py

-rwxrwxr-x 1 goutham goutham 1007 Dec 25 22:28 test\_api.py\*

The file: test\_api.py has execute permissions as shown above.

**Explanation of approach:**

1) In the test\_api.py file, I have written 2 functions: get\_movie\_data(movie\_name) and get\_movie\_rating(json\_data).

2) The first function: get\_movie\_data(movie\_name) does the following:

1. It takes in movie name as a string.
2. It requests data from the API by providing the base\_url and params dictionary.
3. After the data (text in json format) is received as a string, it converts the data into a dictionary. It then returns the dictionary.
4. It also prints the received data as a string (in json) in presentable format with indentation.

3) The second function: get\_movie\_rating(json\_data) does the following:

1. It takes in the dictionary returned by first function.
2. For a movie, it prints the movie name and also its rating given by Rotten Tomatoes website.

4) The main idea of first function was to show the user what information does the omdb api return when a movie name is passed. The main idea of second function is to grab the rating given by Rotten Tomatoes website for that movie.

5) I have also passed in “Black Panther” movie manually as a string to first function just to verify the public API. In this case, just running “./test\_api.py” at command line in a python3 installed environment would give the desired result.

**Pros of approach:**

1) Incase, the API provider changes the way they design/organize the JSON, then the code can just be modified in get\_movie\_rating() method. Other parts of the code need not be changed in this case.

2) This python script can be easily modified if the user wants to pass the movie name at command line instead of providing it manually inside the script. In this case, ' '.join(sys.argv[1:]) can be passed in place of movie\_name inside the script. Then user needs to run: ‘python test\_api.py <movie-name>’ at command line.

3) requests module is easy to use and fetch data compared to urllib module.

**Cons of approach:**

1) The requests.get() method is less secure since it might contain sensitive information like username or personal information etc. being passed to the method either as part of base URL or in the params. In this script, the private api key could be viewed as part of params or by printing the entire URL.