Explanation

Task Objective: Using the API endpoint given, we are told to display the available food trucks based on the time(in Hour) user inputs

A. **Parse Data Input**:
    a. Before building the URL to obtain the data from the endpoint, parse the input given to obtain the following
    b. URL (With Endpoint)
    c. Timestamp
    d. Token for authorization

B. With the timestamp, further parse it with Python's datetime module to obtain the hour, minutes and day of the timestamp which is in UNIX format

C. Created a hashmap/dictionary with mappings corresponding to day(Monday, tuesday etc) with the order of the day(Sunday -- 0, Monday -- 1 etc)

D. **Get_data_from_api module:**
    a. Using Python's requests library, we are able to make a request to the endpoint.
    b. In the URL we need to use HTTP Authorization header Basic with token given
    c. Along with url pass in the time(hour, minutes and day order)
    d. If request is successful we observe a 200 status code
    e. If code is 400/403/404 it is not successful either due to missing parameters or invalid authorization/login
    f. If we do not get a 200 code, it means the endpoint request was not successful we can simply return 'N/A'
    g. Next, parse the api response to get the data in json format using which we continue the processing & filtering of data.

E. Sort_food_truck module:
    a. With the parsed api response, filter operating food trucks with the time range they are open from.
    b. Use a list instead of dictionary as the api response may contain duplicate names of food trucks, dictionary works only for unique key, value pairs
    c. Iterate over all the food trucks in the response and check if the current time user inputs is within the start and end range of the food truck operation
        i. If it is, add the key value(name, location) to a list
    d. Finally sort the list alphabetically and return it

F. Display data
    a. Check that the data is not 'N/A', only then process the data to print as per output sample
    b. If the data is 'N/A'
        i. Print 'N/A' as the request did not return any data

One edge case missed - in the sort_food_truck module -- if there were not matches/food trucks operating in a time period, the list available_food_trucks is never populated, check if it is empty and still return ‘N/A’