LAB ASSIGNMENT 2

Submission Format:
Put all files in folder named “<Roll number>_Lab_Assignment2” and compress it to “<Roll Number>_Lab_Assignment2.tar.gz”. Upload this compressed file.

1. Write a function *“fifthDay”* which does not take any arguments but returns the fifth day by considering today's date. Search for inbuilt functions for getting today's date and try using JSON object to store any kind of mapping. For example: If today's date is 2020-01-21 Tuesday, then your function should return a string "Sunday". Make sure the return value of your function always belongs to ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"] (Case-Sensitive). Name the file “q_1.js”.

2. Write a function called “altSpaceToUnderscore” which takes a string and replaces every even space in the string with an underscore. The function should discard any leading or trailing spaces and not consider them for replacement. You can use in-built functions like trim, split, splice, substr, etc for operations on string. For example:
   Input: "Hello there, how are you !!" Your function should return: "Hello there,_how are_you !!" . Name the file “q_2.js”

3. Write a function named “getMeNextFirst” which takes a string containing white spaces and transforms the string such that every word gets appended with the first character of the next word in the string. Remove the leading trailing white spaces so that they are not considered in your algorithm. You should use functions like filter, join, slice, etc to write an elegant solution. For example: "Hello world !!" should return "Hellow orld! !" . Name the file “q_3.js”.

4. Create a file “q_4.js” and import data from the file “data.js”. This included file contains a JSON object which contains marks (out of 100) for the students in the following format:
   <name> : [marks1, marks2, marks3, marks4, marks5]
   Now write the following functions:
   - *“getHighestMarks”* : This function should take the data (as mentioned above) and return name of the student with highest total.
   - *“getSubject2Toppers”* : This function should take the data and return a sorted array of students' name with highest marks2.

5. Create a new html file named “q_5.html”. Using the data from the previous section(“data.js”), perform the following tasks: - Create a heading (<h1>) named “Student Sheet” in the <body> of html page. - Create an HTML table(<table>) with the following header(<thead>).

   | Name | marks1 | marks2 | marks3 | marks4 | marks5 |

- Populate the table with entries from the JSON object.
- Highlight the row with color green, on hover over any row of the table.

6. Create a new HTML file named “q_6.html” and perform the following three tasks:
   - **Dropdown**: Create a dropdown(<select>) of names of the students.
   - **Displaying marks**: On selecting any student name, display the corresponding attributes of the selected student in a table with the following format:
   | Name | marks1 | marks2 | marks3 | marks4 | marks5 | Percentage | Grade |

   - Criteria for grade:

%age	Grade
>90	A
80-90	B
70-80	C
60-70	D
<60	F

   - **Highlight**: Highlight the minimum and the maximum marks with red and green respectively.

   Resources
   1. https://www.w3schools.com/js/default.asp
   2. https://developer.mozilla.org/en-US/docs/Web/JavaScript