SODV1201
Assignment 1: Introduction to Web Programming

Overview
This assignment includes number of different works. Make sure you start working on it from the day it is assigned for you. In this assignment, you are required to create a set of JavaScript functions and objects, then demonstrate that they operate correctly. You can use any editor to write the functions.

Instructions

1. Create a single personal profile page about yourself with any kind of style. This page should include one picture, 1 or 2 paragraph, and header with navigation bar and footer. (10 points)
   (This home page should be named as index.html)
   - The navigation bar will be used to create a link to other pages that you will create for the other questions.
   - The footer of your page should include current date and copy right information.

2. In your profile page, the name of the picture should appear after 10 seconds. (10 points)

3. Build a Mark to Grade converter web page with the following specification and create a link to it in your profile page (20 points)
   - The MarkToGrade function scrapes the student mark out of the mark-input-box element and verifies it is **realistic**.
   - A Mark must **be number only, nonnegative, and less than 101.**
   - If an invalid mark is entered, it displays a message back to the user in the. validation-message element.
   - Messages should be informative... return as many different messages as you can to guide the user what kind of erroneous value has entered
     - If the user entered any value above 90 Grade A should be displayed as a result
     - If user entered any value above 80 Grade B should be displayed
     - If user entered any value above 70 Grade C should be displayed
     - If user entered any value above 50 Grade D should be displayed
     - If user entered any value less than 50 Grade F should be displayed
   - Hint 1: In JavaScript, we can use the global parseInt function to try and convert a string to a number.
   - Hint 2: Use exception handling.
   - Hint 3: Use HTML and JavaScript

4. Create a new staff page and write a program which display list of staff information with sorting capability by name and salary. Sample array data about staffs will be provided for you from your instructor. You can restructure the dataset as you want. Link this staff page to your profile page. (20 points)

   - For this question you are expected to apply the concept of loop, JavaScript object, Array and function.
   - You can also use jQuery if needed

- You can implement different solution for this question. All working attempts will be considered right.

5. Create a new page called weather. On this page use the concept of JavaScript named and anonymous function and create a program that can accept a temperature input in degree Fahrenheit and convert it into degree Celsius and from Celsius to Kelvin based on a button click event. (20 points)
    - Hint: Use JQuery to access the DOM elements in your code for this question.
    - Do a little research to find out the basic formula for each calculation.

6. Create a stylesheet and add different style to your pages. All your page should have a footer that details your full name, copyright @ and year. (10 points)

Submission Directions (5 points)

- For this assignment, you can have multiple html files.
- It is highly recommended for you to create your own CSS file instead of embedding others style like bootstrap and others.
- After you are done, push your code to your GitHub and deploy your page to GitHub page. Create an empty word document and paste your GitHub repository link and your Github page link into your document (There should be 2 links in your document). Submit this document in D2L.

Marking Criteria (5 points)

This is a coding assignment, so all the rules about best coding practice apply. Your code is evaluated for **correctness** (does it achieve the task it is supposed to) and for **hygiene** (is it clear, well-commented, and easy to follow). There is no point in writing accurate code that nobody else can understand. Some tips to ensure good code hygiene:

- Add documentation about you on the very top of your code. Something similar to

```
/**
 * @name: Assignement1
 * @Course Code: SODV1201
 * @class: Software Development Diploma program.
 * @author: Your Full Name.
 */
```

- Add intelligent comments that explain your logic for each code
- Add intelligent comments before expressions, methods, functions and classes to outline what they do
- Use sensible variable names that match the purpose of a variable
- Use whitespace between functions and code blocks and indent consistently