## Important Notes:

- **Each team (group of two students) will work on this assignment individually. For example, team1 members can work together but they CANNOT work with or assist another member from a different group!**
- Follow the **exact naming rules** for all your assignment files as explained in the instructions and shown in the diagram
- The assignment must be **submitted before the due date** in order not to lose marks and to avoid the severe penalties
- **Upload/Submit your files/links through the Blackboard (Emails are NOT accepted) as explained below:**
  1. **The PDF file that contains the snapshots as explained below**:
     - Use **MS-Word** to capture all the assignment images (screenshots) then put them all in one document then convert it to a PDF file and upload it with your submission. Please consider the following screenshots (images):
       - The folder structure inside IntelliJ IDEA, Visual Studio Code or any other Editor that you are using that shows all the four listed files above
       - The VS Code (or the other IDE) embedded terminal window that shows the current working directory (the path of your folder). You can also use an external separate terminal like "Terminal" in Mac or "PowerShell" in Windows instead of using the embedded one
     - PDF content should be presented in professional manner with cover page (front page)
  2. **The link to your GitHub repo where you have your assignment (project) uploaded as explained below**:
     - Regarding the "GitHub" repositories (repo), please consider the following points:
       - (Refer to my GitHub PDF Guide) *
       - Each student must upload the assignment to their GitHub website (New repo as usual). The team coordinator will oversee submitting the link (URL) for their team members' GitHub on the assignment submission section in the Blackboard
       - Upload the folder contents of your assignment to your GitHub with a Readme file "README.md" that contains a very brief description about your code examples
       - ~~**Make this repo to be a GitHub pages website by adding "Live Demo" to this assignment repo, not your main GitHub website**~~**!** ***(This part can be done using Vanilla JavaScript or with front-end frameworks like "Vue.js". You can refer to my PDF file about using Git and GitHub for more details)***

**NOTES:**

- **At the end of this assignment,** you should have a **project folder named "Lab2-JS-frameworks" as an example or any other name you prefer as a team. This folder's name (Project Name) could also be the name of your GitHub repos where you must upload your assignments**
- **You can use .gitignore to ignore all the unnecessary files, or folders if you have any for example:**
  - **.vscode/**
- **Online:** Each student must be ready with their Zoom Mic/Cam and Screen share feature to quickly show us the assignment content, Q/A, and run it to see the output (even if it's not working properly).

- **In-person:** I will make a tour to check the running application in the student's computer and asking the technical questions

  ***NOTE:*** *This presentation/tour will be very quick without correction or feedback in order not to take the full lecture time! The assignment steps, feedback, and solutions will be completely explained to all students (after finishing all the presentations/tour) in the next lecture*

*(35 Marks)*

## Assignment Contents:

This assignment will be a demonstration about your skills in importing and using custom modules according to what we have done in the last lecture (part2) of working with Node.js.
To emphasise on the purpose of learning and encouraging all the students to participate using their own creative ideas instead of following fixed steps and instructions, I have designed this lab to give you a general overview/idea about the topics/tools that you will need to demonstrate leaving the code examples and duties to you.

## Assignment Instructions:

You will create two separates JavaScript files to be run on the server-side with Node.js as explained below:

### First JavaScript file:

1. Any name you prefer to be used as the entry point to run/lunch or start your application using node commands
2. You will just call/invoke a list of the functions that you created in the second JavaScript file.
    a. You need import the other JavaScript file that contains the function
    b. You will have to use **ES modules** syntax for importing and exporting modules content.

**NOTE:** You can also add the implementation of the classical (default) way of using **CommonJS** for learning **(Optional – NOT REQUIRED)**, but it must be done in a separate folder, otherwise you will have a conflict of using the two ways together

*(30 Marks)*

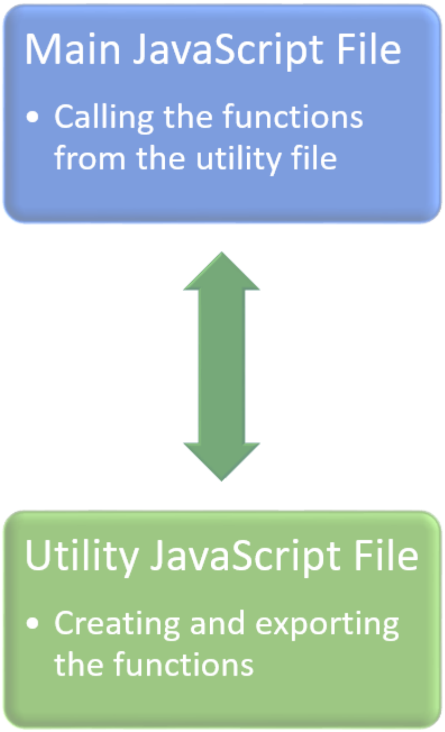
### Second JavaScript file:

1. Any name you prefer to be used as a utilities file where you can save your functions, objects, or another data to be used/called by the other JavaScript file
2. Creating at least **4 different functions**:
    a. **First Function:** no return, just accepting a text as an input with two or three parameter/arguments and print it by concatenating it with labels/other text.
        i. **Example:** the function might have two parameters: firstName and lastName and print/output: "Welcome firstName lastName to G@I College"
    b. **Second Function:** return a double value for converting a measurement unit to another one, it can accept a double single value as parameter for the initial measurement unit.
        i. **Example:** the function might have one parameter for the **Celsius** and convert it to **Fahrenheit** (or vise - versa) or converting minutes to hours (or vise – versa) etc... use your ideas!
    c. **Third Function:** to do any math operation against a numeric value(s), it can accept one, or more than one value (argument), up to you, and then return the result of applying a math formula
        i. **Example:** the function might have one parameter and return the factorial of that value
    d. **Fourth Function:** any advanced function of your choice

**Note:** The examples provided are just to clarify the functions and their tasks. They should not be used directly in your code. Please come up with your own ideas.

*(35 Marks)*

## Assignment Folder Structure:


Main JavaScript File
• Calling the functions from the utility file
Utility JavaScript File
• Creating and exporting the functions

## Hints and Tips:

For more review about the needed code help, you can check below my listed code samples on GitHub which are the lectures that we had covered:

- **For reading a file using Node.JS:**
  - https://github.com/anmarjarjees/node.js-start/tree/main/5.modules
- **Other useful links:**
  - JavaScript Modules (w3schools.com)
  - JavaScript modules - JavaScript | MDN (mozilla.org)
  - Node.js Modules (w3schools.com)
  - Modules: CommonJS modules | Node.js v21.6.2 Documentation (nodejs.org)
  - JavaScript Require – How to Use the require() Function in JS (freecodecamp.org)

**Happy Coding** ☺