COA123 Web Programming

Dr Hossein Nevisi

Lab 2: Using JS conditional statements, iteration, and functions for problem solving

Objectives:

- Get more familiar with the JavaScript syntax
- Use JavaScript to solve simple problems
- Learn how to use JavaScript functions
- Develop logic thinking & algorithm development skills alongside programming skills

1. Download **factorial.html**, inspect the code and complete the requested task.

2. Instead of embedding JavaScript in factorial.html, we want to write the JavaScript part in an external JS file, and refer to it from our html file. Call the new html file, **factorial2.html** and call the JS file, **factorial-script.js**. Run you new html file on your browser to make sure it works the same as factorial.html.

3. Download **bmi.html**, inspect the code and complete the requested task.

4. Download **scopes.html**.
   - Run the file on your browser. Check your browser's console and try to fix the console error in the html file.
   - After fixing the above console error, in funcB, change alert("num1: " + num1) to alert("num2: " + num2). Why this change gives you a console error again?
   - Instead of using alert to generate output, use **console.log** method to display the output in your browser's console.

5. Write a JS function which gets 3 numbers as its input parameters and returns "fail" if any of the numbers are below 40, otherwise returns "pass" together with the average of the numbers. Try to find a way to pass some values to your function to make sure it works properly.

* You can generate a module folder (e.g. coa123) under your "web" directory to manage your lab work for this module and use sub-folders for each lab. Please note the folder and file names are CASE sensitive on the sci-project server! When you put a web application (e.g. .php or .html) in the "web" directory, you can run it from a web browser using URL (Ref. "sci-project Server Connection").