INFO 151: JavaScript Exercises

JavaScript exercises (for the Weeks #4, #5, #6):

The following slides show the exercises and the output from HTML web pages with embedded JavaScript scripts. You are required to:

- Create a project for each exercise (with a suitable memorable project name).
- Create the HTML web page (*index.html*) with the correct HTML <tags> and JavaScript script <tags>.
- Write the correct HTML.
- Write the correct JavaScript program scripts.
- Run the project in a web browser
  - Check the output is as shown in the examples
  - Correct the HTML and JavaScript as required
  - Repeat the program run until the output is correct
- The output must be exactly as shown in the examples

The coursework exercises to be completed in the laboratory sessions and as homework

The JavaScript for the exercises will be introduced in Weeks #4, #5, and #6. You will be shown the relevant JavaScript to enable you to complete the exercises.

HTML with Embedded JavaScript Programming Exercises

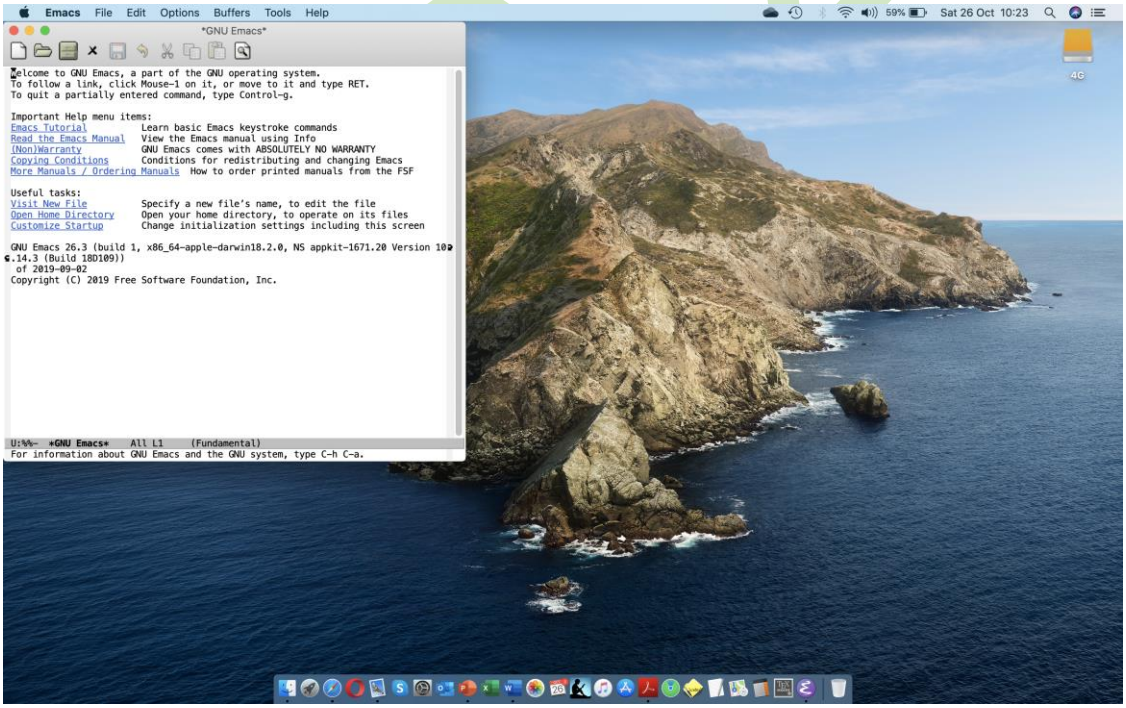


Figure 1. Screenshot showing the EMACS editor (free to download and install on Windows and Mac

HTML with Embedded JavaScript Programming Exercises #2 to #19:

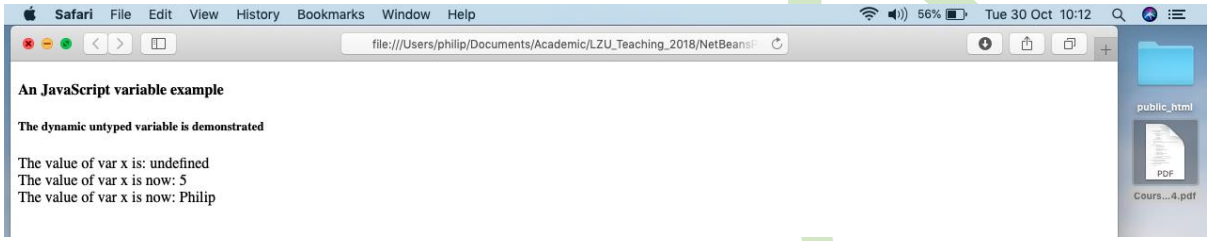
The following screenshots show the editor used to create the HTML / JavaScript programs and the expected output in my Safari browser (you may use a browser of your own choice).

Exercise #1:

- Design an algorithm to compute: (a) the area of a circle, (b) the circumference of a circle, and (c) the volume of a cylinder.
- Write the pseudo code to implement the algorithm.

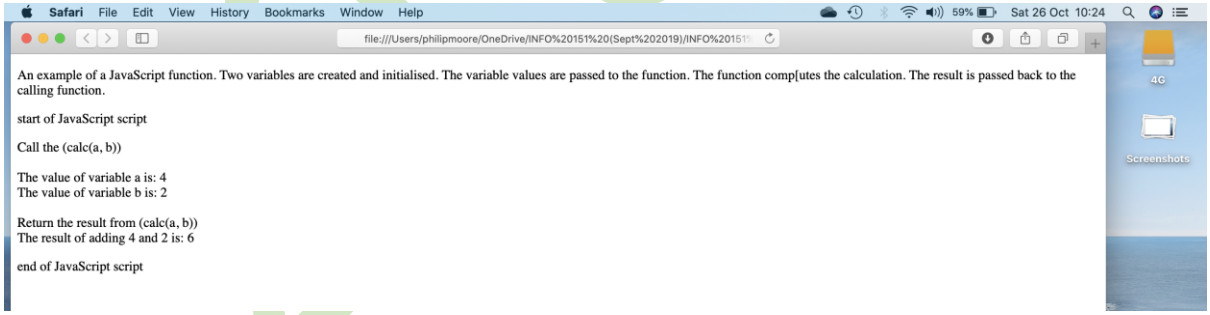
- The program design must enable the following:
  - Take in three inputs (diameter, radius, height)
  - Calculate the area of a circle
  - Calculate the circumference of a circle
  - Calculate the volume of a cylinder
- Create and run JavaScript program with the output shown in the NetBeans embedded web kit and a web browser.
- The output for the program to use string(s) and must show: (a) the formula, (b) the calculation, (c) the variable values, and (d) the result.
- The program output must include all variable values, the computed results, and a text description of the computation using a *concatenated* string.

Exercise #2:



Exercise #3:

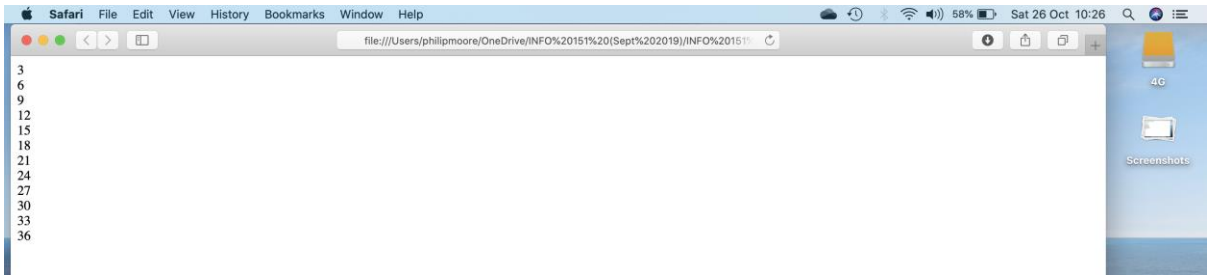
Exercise 4:



Exercise #5:

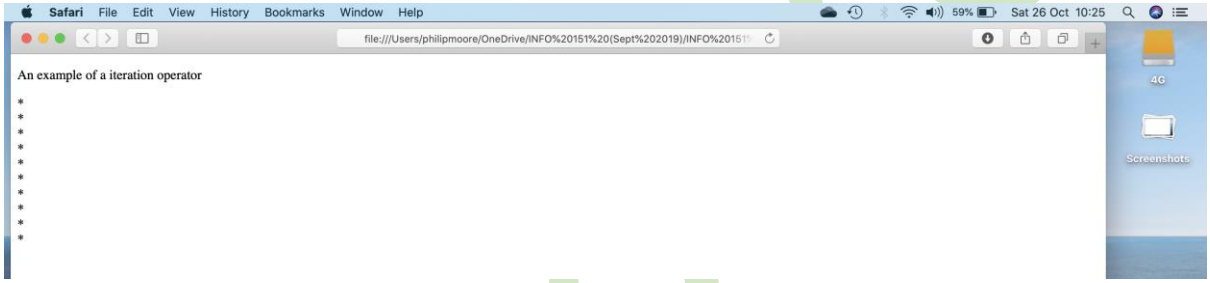
Exercise #6:

Exercise #7:



Exercise #8:

Exercise #9:

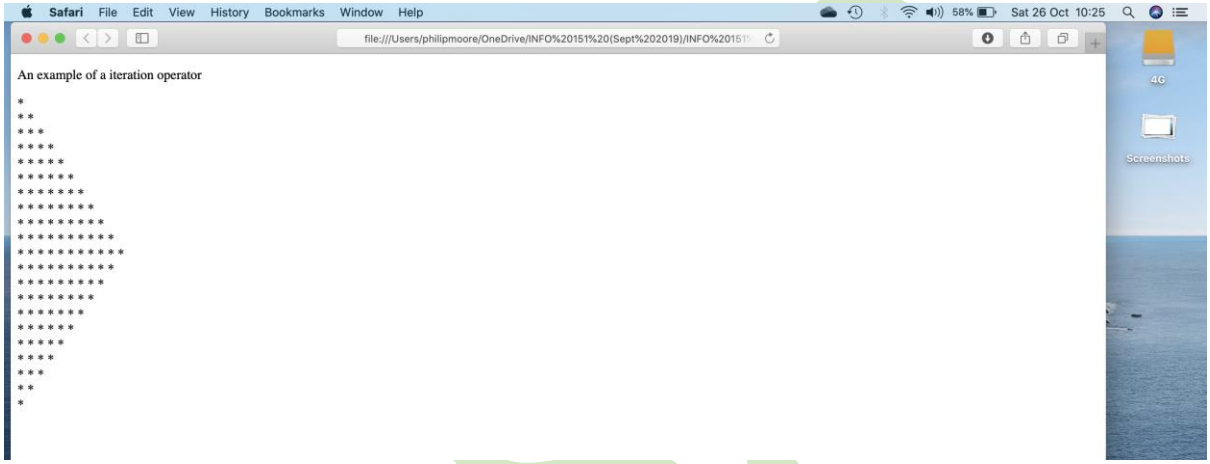


Exercise #10:

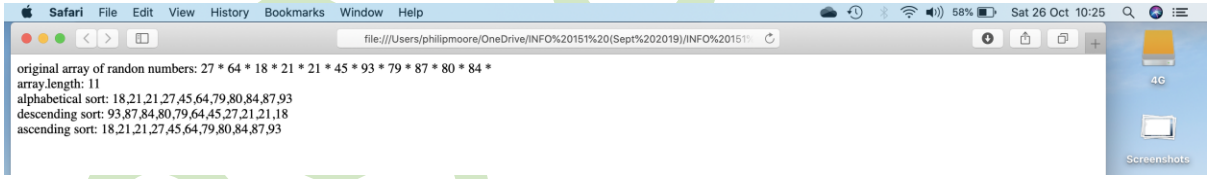
Exercise #11:

Exercise #12:

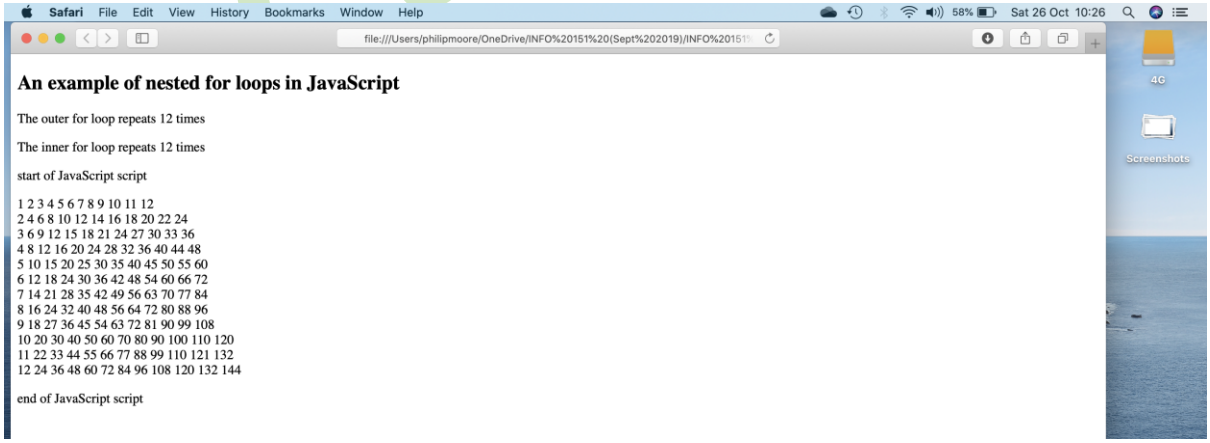
Exercise #13:



Exercise #14:

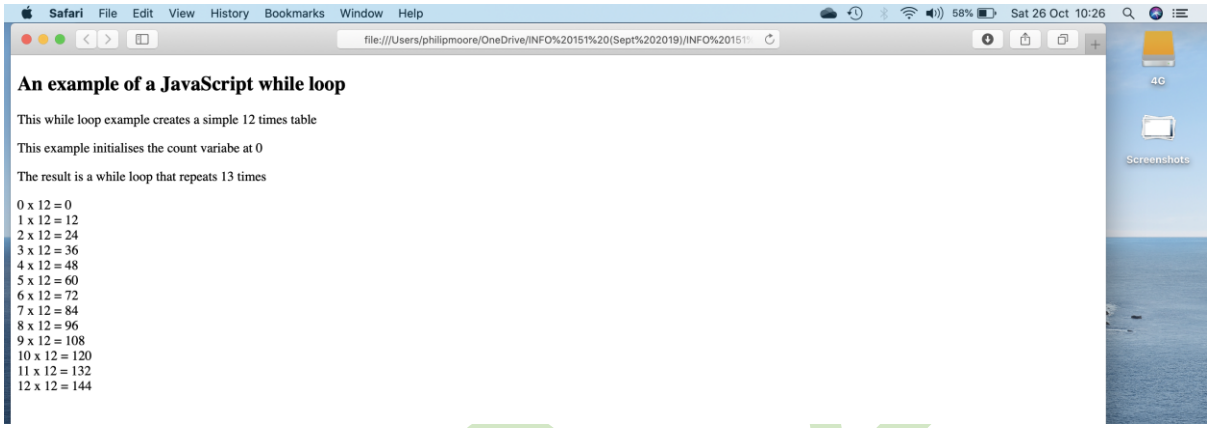


Exercise #15:



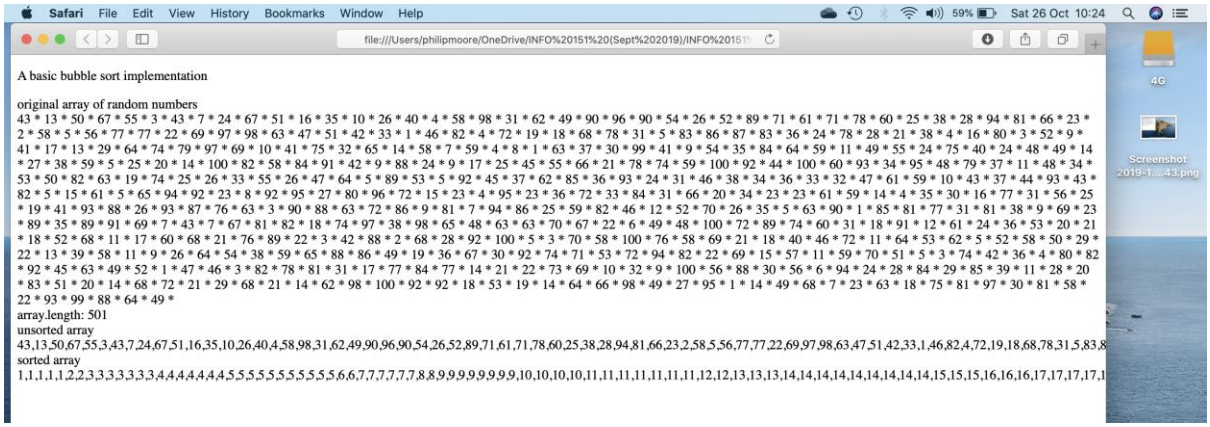
Exercise #16:

Exercise #17:

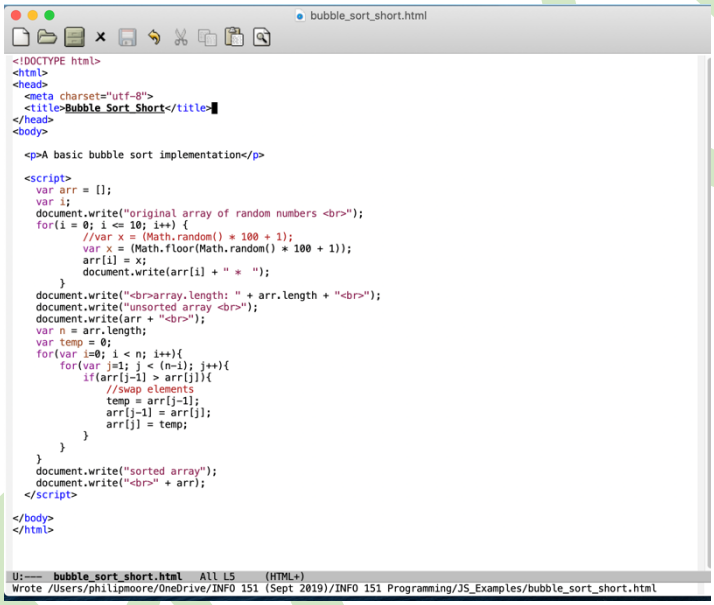


Exercise #18 (500 random numbers):
The following example shows a simple implementation of an JavaScript *array* with a simple implementation of a *Bubble Sort* program. The program creates pseudo random numbers and sorts the numbers in ascending order. The program code is shown in the *EMACS* editor. You are required to:

- Write the code as shown
- Run the program and correct any errors in the program code until the program runs correctly as shown in the screenshot below.
- Your output should be exactly as shown in the screenshot. If it is not correct the errors in the program code until the output is correct.



Exercise #18 (10 random numbers):

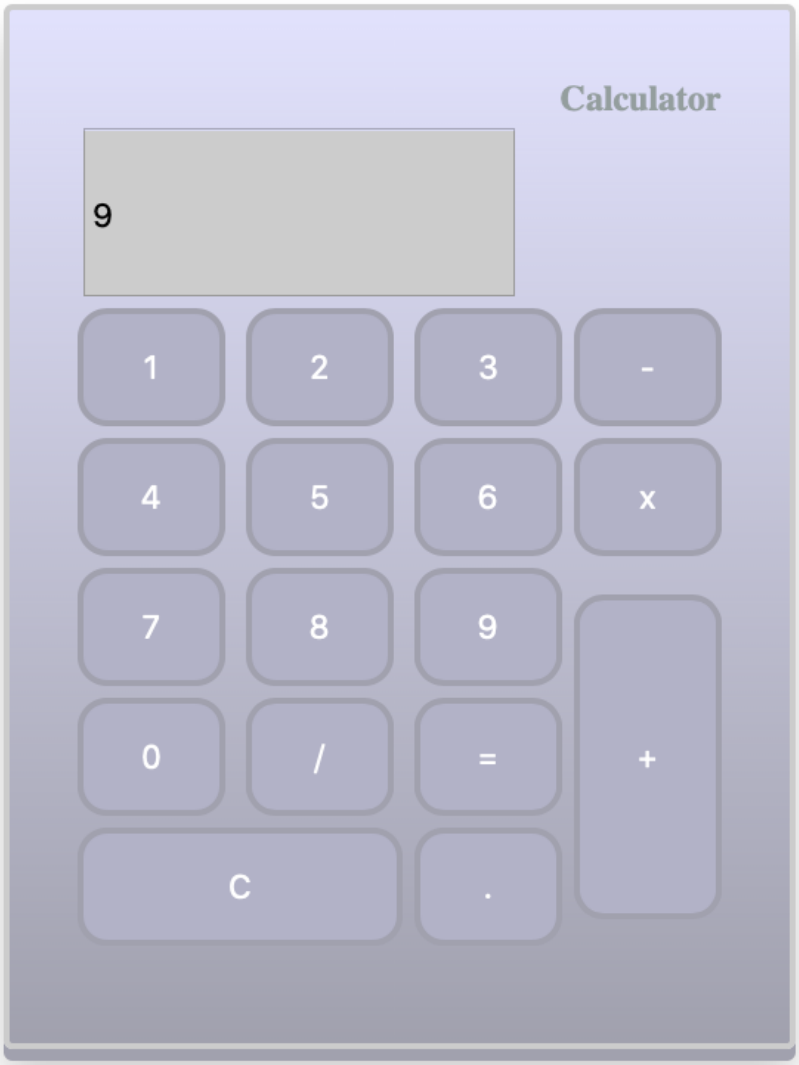


Exercise #19:

The following example shows a simple implementation of an JavaScript program which creates a simple working calculator which will run over the Internet.

The program to create the calculator has been written in pure JavaScript and employs all the JavaScript features introduced in this course

You may attempt this exercise when the previous exercises have been successfully completed.


Calculator
9
1
2
3
-
4
5
6
x
7
8
9
+
0
/
=
C
.