1) Determine the output and/or identify the error in each of the following code segments:

a)

```
function displayK(){
    for (let k = 1; k <= 10; k = k + 3)
          console.log(k);
}
displayK();
```

Output/ Error:

b)

```
for (let m = 20, m > 5, m -= 4) {
     console.log(m);
}
```

Output/ Error:

c)

```
for (let n = 0; n < 8; n++);
     console.log(n);
```

Output/ Error:

d)

```
for (let p = 16; p > 2; p = p/2) {
     console.log(p);
}
```

Output/ Error:

2) Using the for loop, write the program to generate each of the following output:

a)

```
Output:
3 6 9 12 15 18
```

b)

```
Output:
25-20-15-10-5-0
```

c)

```
Output:
1 8 27 64 125
```

You may open up your Visual Studio Code (VSC) and work from here for the remaining questions. Remember to save all your JavaScript files.

3a) Write a function to display a "5 times" multiplication table. You are required to use the ***for*** loop. The output should be as follows:

```
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
11 x 5 = 55
12  X 5 = 60
```

3b) Modify your code to create a function **generateTimetable**. It will take in one parameter and generate the timetable based on the user input.

Sample Output, text in **bold** is the user input.

```
Enter a number : 2
1 x 2 = 2
2 x 2 = 4
3 x 2 = 6
4 x 2 = 8
5 x 2 = 10
6 x 2 = 12
7 x 2 = 14
8 x 2 = 16
9 x 2 = 18
10 x 2 = 20
11 x 2 = 22
12 x 2 = 24
```

3c) So far the display stops after 12 rows e.g 12 x 2 = 60. Modify the function to take in another parameter so that user can set how many rows to generate.

Sample Output, text in **bold** is the user input.

```
Enter the multiplication table required : 4
Last row : 3
1 x 4 = 4
2 x 4 = 8
3 x 4 = 12
```

4a) Write a program to read in a series of numbers and displays the sum of the numbers entered. It should have a function **addNumbers** with the following requirements.

- Prompts user to enter 5 integer numbers
- Displays the numbers entered and
- Calculate the sum of the numbers entered.

This function does not take in any parameter but returns the calculated sum of the numbers entered to be displayed.

You can assume that the user will always key in numbers and not texts. You are required to use the ***for*** loop.

Sample Output, text in **bold** is the user input.

```
number1: 2
number2: 6
number3: 1
number4: 3
number5: 2
Sum of numbers: 14
```

4b) Modify the function **addNumbers** to read in one more parameter. In the main program, prompt the user to enter number of numbers to sum up and pass it to the function to calculate the sum.

Sample Output, text in **bold** is the user input.

```
How many numbers do you want to add? :3
number1 : 3
number2 : 6
number3 : 12
Sum of numbers: 21
```

5) The Math library has a prebuilt function called **random()** that generates random floating-point numbers in the range of 0 (inclusive) to 1(exclusive).

a) Write a program to generate a random integer in the range of 0 to 68.

b) Modify (a) using a function to generate the a random integer. Your program should prompt the user to enter the range of numbers to generate. The function should take in these numbers and returns a randomly generated number within that range.

Sample Output, text in **bold** is the user input.

```
Enter starting number  : 19
Enter ending number    : 87
Random number generated : 60
```

- END -