# NM2207

Week 3

Session 03 Classwork

Overview of what we’ll do today:

* Practice writing pseudocode
* Practice the use of objects and functions

Before attempting the challenges, you are expected to have watched and coded along with the Lecture videos. A tutorial is meant to practice the skills presented in the video lecture, and show you more applications of it. Tutors will explain the challenges and answer the questions you may have**.**

**The challenges are due to be completed at the end of class each week for full credit which is also attendance. Submitting the next day accounts for half the credit.**

**Part 1 (Total 35 minutes)**

**Summary of learnings of Part 1**

* + - * Remember the first step in writing any pseudocode or code is always to declare and initialize your variables.
      * If you are extracting the HTML content of an element on your index.html
* The first step is to assign that element to a variable using getElementById(). See Table 1 for how it’s done.
* Inside the brackets of getElementById() put the id of the element you want in quotes
* And then to set its innerHTML to whatever you want.
* If you are extracting more than one value, you will correspondingly need to do the getElementById() step each time, with different id names. Remember to assign them to different variables. See Table 1.

**Part 1 Warm up**

Step 1. Back at your web page, Menu (3 dots in upper right) -> More Tools ->

Developer Tools

1. Go to the ‘Elements’ tab
2. Click on (and draw your mouse over) each of the components. What happens on the page?
3. Check out the Sources tab and open your file from there.

Graphical user interface, application, Word

Description automatically generated

Step 2. Open the console tab

Type “var x = 5;” and press the enter key

Type “x” and press the enter key. What happens?

Type “x +2” and press the enter key. What happens?

Bravo, you just ran your first piece of Javascript!

But there are cleaner ways of running JavaScript. For instance, we might want some calculations done as soon as the page loads. For example, JavaScript can check if you have stored cookies that you have previously logged in to a website. Then it will not ask you to log in again.

In the next steps, we will learn about these neater methods that work behind the scenes.

**Part 2 Writing JavaScript in code.**

First, we will write pseudocode ie a list of steps to solve a problem.

**2.1 Warm up**

a. Write the steps to be followed to add 4443423434 and 439439439 and return the output.

b. The output will be used to change the innerHTML property of a part of your website. See Table 1 for how it’s done.

c. So imagine that when the page loads, you want your code to run and immediately change the value in the “answer” element.

* 1. **Practice**

Write the steps to convert a Fahrenheit temperature reading input by a user into a Celsius temperature reading and return the result

**2.3 Challenge**

Write the pseudocode to calculate E, given m and c, and knowing that E = m\*c\*c and write it to console.

Modify the pseudocode to extract m and c from the innerHTML property of different elements in the Document Object Model. See Table 1 for how it’s done.

Modify the pseudocode to print E to the innerHTML of a third element instead of writing it to a console.

In preparation for coding this up, use your Session03.class/appscripts/main.js in VSCode.

**2.4 Practice with the Document Object Model**

**Now, practice getting and setting text and style properties for different HTML elements on W3schools**

<https://www.w3schools.com/JSREF/prop_html_innerhtml.asp>

<https://www.w3schools.com/JSREF/prop_node_innertext.asp>

<https://www.w3schools.com/JSREF/prop_html_style.asp>

**Table 1: For practice. (20 minutes)**

|  |  |
| --- | --- |
| **Description** | **Code** |
| **Command to get an HTML element and assign it to a variable** | var speedEl = document.getElementById(“aside”); //var can take any name, and look for any id |
| **Now we can refer to all the properties of the HTML element, such as** |  |
| **its HTML content** | speedEl.innerHTML |
| **Its color** (works for headings, paragraphs and divs) | speedEl.style.color |
| **backgroundColor** (works for paragraphs and divs) | speedEl.style.backgroundColor |
| **height** (works for paragraphs and divs) | speedEl.style.height |
| **margin** (works for paragraphs and divs) | speedEl.style.margin |
| **padding** (works for paragraphs and divs) | speedEl.style.padding |
| **overflow** (works for paragraphs and divs) | speedEl.style.overflow |

**Part 3 More functions**

**The next few questions get you more practice with math and functions.**

* 1. Notice the map() function provided below. Run through the function “manually”:

**//function declaration**

// maps x in the interval [a,b] into the interval [m, n]

let map =function (x, a, b, m, n){

let range = n-m;

// x is 'proportion' of the way from a to b

// e.g. if a=10, b=20, and x=15, x is half (.5) of the way from a to b

let proportion = (x-a)/(b-a);

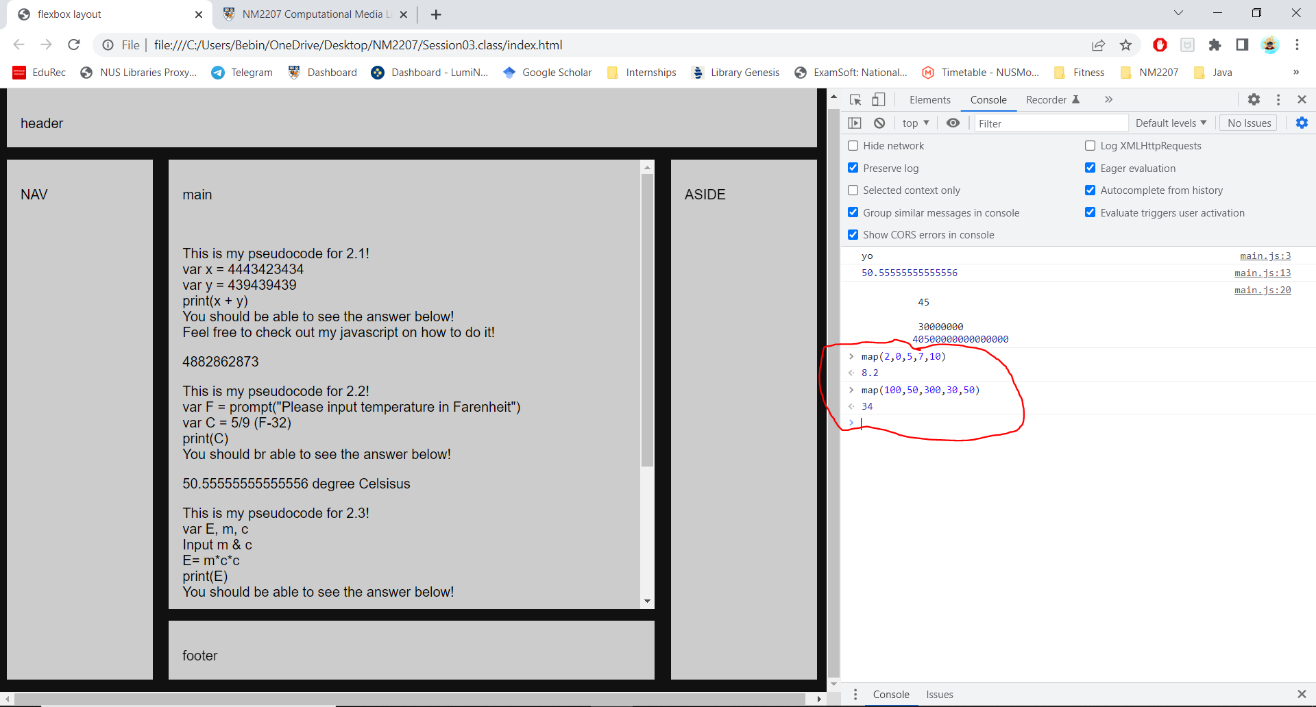
return (m + proportion\*range);

}

3.2 Explain what the function is doing:

|  |  |
| --- | --- |
| Question | Answer |
| What is its purpose? | This function has two intervals, a to b and m to n. A value of x in the interval of a to b is modified such that the function provides a number in the interval of m to n with the same relative position in that interval as the position of x in a to b. |
| What does it take as input? | It takes as input, the domains of the intervals a to b and m to n, and the value of x between a and b. |
| What does it provide as output? | It outputs the modified value of x such that the new value occupies the same relative position in the interval m to n as x occupies in a to b. |
| What happens when x is equal to a? | When x=a, the output will be m. |
| What happens when x is equal to b? | The output will be n. |
| What happens when x is exactly halfway between a and b? | The output will be exactly halfway between m and n. |

* 1. Now, paste it into main.js and test it with arguments supplied in the function call. Were you right?

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Yes, I was right, when I called the map function in both instances, the value returned by the function represents the new value with the same relative position as x from the interval of a to b to the interval of m to n.