Session 2 Homework

Code For Everyone JavaScript

|  |  |
| --- | --- |
|  | **Study** |

1. let vs var vs const

Learn about let, var and const by

* Reading these tutorials:
  + [var, let and const, what is the difference](https://dev.to/sarah_chima/var-let-and-const--whats-the-difference-69e)?
  + [var vs let vs const in JavaScript](https://tylermcginnis.com/var-let-const/)

Hoặc xem video sau

* [MindX - C4EJS - let vs var vs const](https://drive.google.com/open?id=1KE2mYDeAub4M-NkiKY3sV9oyVlJylkYz)

Then answer the following questions:

1. What are var and const in JavaScript?

*Var = Variable is a named storage, which its value can change.*

*Const = Constant is an unchanging variable. They cannot be reassigned*

1. What are the differences between let and var?

*When declare a variable with ‘var’, its scope is not limited to the block in which it is defined, it’s limited to the function in which it’s defined.*

*After ES6 (ES2015), ‘let’ is used to declare a variable, but that variable is limited to the block in which it is defined.*

1. What are the differences between let and const?

*‘let’ & ‘const’ both create block-scoped variables, but using ‘const’, the variable reference cannot be changed. While variable created with ‘let’ can be reassigned.*

1. What to use in what cases?

*We should avoid using ‘var’, since it is created function-scoped variables.*

*‘const’ is used to create unchanging variables.*

*‘let’ is used to create block-scoped variables.*

1. Boolean

Learn about Boolean by

* Reading these tutorials:
  + [A Boolean (logical type)](https://javascript.info/types#a-boolean-logical-type) and [Logical operators](https://javascript.info/logical-operators)
  + [JavaScript Boolean explained](https://www.freecodecamp.org/forum/t/javascript-booleans-explained-how-to-use-booleans-in-javascript/14311)

Hoặc xem video sau

* [MindX - C4EJS - Boolean](https://drive.google.com/open?id=1Izf52pGQOpRnvk7b2XCtaEQwf9MQj91g)

Then answer the following questions:

1. What is Boolean?

*Boolean is a logical data type that can have only the values true or false*

1. What results in Boolean?

*Boolean results in Boolean;*

*Comparison yields Boolean such as ‘==’, ‘>=’, ‘<=’.*

*Operation of Boolean such as ‘&&’, ‘||’, ‘!=’.*

*Other data types can yield Boolean when using statement ‘if’, ‘else if’ or while*

|  |  |
| --- | --- |
|  | **Review** |

1. Write a program to print out
   1. 7 numbers, starting from 0 (no user input)

|  |
| --- |
| *for* (let i = 0; i < 7; i++) {      console.log (i);  } |

* 1. **n numbers**, starting from 0, n entered by user

1. let message = prompt ('Enter a number?');
2. *for* (let i = 0; i < Number(message); i++) {
3. console.log (i);
4. }

|  |  |  |
| --- | --- | --- |
|  |  |  |

* 1. A sequence of numbers, starting from 3, **ending before n**, n entered by user

1. let message2 = prompt ('Enter n');
2. *for* (let i = 3; i < Number(message2); i++) {
3. console.log (i);
4. }

|  |  |  |
| --- | --- | --- |
|  |  |  |

* 1. A sequence of numbers, **starting from c**, ending before n, c and n entered by user

1. let n = prompt ('Enter n');
2. let c = prompt ('Enter c');
3. *for* (let i = c; i < Number(n); i++) {
4. console.log (i);
5. }

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. A sequence of numbers, starting from c, ending before n, **stepping by 3**, c and n entered by user

1. let n = prompt ('Enter n');
2. let c = prompt ('Enter c');
3. *for* (let i = Number(c); i < Number(n); i +=3 ) {
4. console.log (i);
5. }

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. A sequence of numbers, starting from c, ending before n, **stepping by s**. c, n and s entered by user

1. let n = prompt ('Enter n');
2. let c = prompt ('Enter c');
3. let s = prompt ('Enter s');
4. *for* (let i = Number(c); i < Number(n); i = i+Number(s)) {
5. console.log (i);
6. }

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

1. Write a program to calculate the factorial of n: (1 \* 2 \* 3 \*... \*n), n enter by user
2. let n = prompt ('Enter a number');
3. *for* (let i = n-1; i>= 1; i--) {
4. n \*= i;
5. }
6. alert (n);

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. Write a program asking users their age, and then decide if they are old enough to view a 14+ content
2. let userAge = prompt ('How old are you?');
3. *if* (userAge <=14) {
4. alert ('You are not old enough to view this content')
5. } *else* {
6. alert ('Enjoy!');
7. }

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
|  | **Serious exercices** |

1. Write a program asking user to enter a number, x, then check if x is in the lower half or higher half of 0 - 9 range

|  |
| --- |
| 0  9  4.5  Lower half  Higher half |

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. Write a program asking user to enter two numbers, x and n, then check if x is in lower half or higher half of n

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

1. Write a script to check if a number is even (divisible by 2) or odd number

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

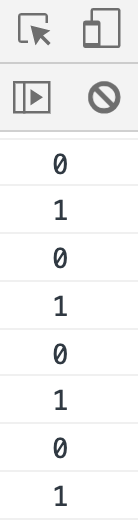
1. Write a program to print out
   1. 6 L’s and H’s, half L’s, half H’s *(L means low, H means high)*

|  |
| --- |
|  |

* 1. n L’s and H’s in total, n entered by user

|  |  |  |
| --- | --- | --- |
|  |  |  |

* 1. 8 1’s and 0’s in total, consecutively



* 1. n 1’s and 0’s in total, consecutively, n entered by user

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. Write a script to calculate the BMI (Body Mass Index) of a person, the formula is as follows

BMI (Body Mass Index):

BMI = mass (kg) / (height(m) x height(m))

Note: you must do the conversion from cm to m before calculation

And then based on the calculated BMI, tell them that they are:

* Severely underweight if BMI < 16
* Underweight if BMI is between 16 and 18.5
* Normal if BMI is between 18.5 and 25
* Overweight if BMI is between 25 and 30
* Obese if BMI is more than 30

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
|  | **Turtle exercices** |

1. Use [JS Turtle](https://turtle.mindx.edu.vn/), to draw the following shapes

|  |  |
| --- | --- |
| 1. A square | 1. A triangle |
| 1. A pentagon | 1. A hexagon |

1. Use [JS Turtle](https://turtle.mindx.edu.vn/) to draw a polygon, the number of polygon’s edges entered by users

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. (Optional) Use [JS Turtle](https://turtle.mindx.edu.vn/) to draw n polygons, n entered by users

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
|  | **Tools** |

1. Watch and practice along [this tutorial](https://drive.google.com/open?id=1FY_uUP0uWUoeQZ8Tm_ACnIWgxPMsDH_z), create a github.com repository with the naming convention mentioned in the tutorial. After completing your homework, commit and push it with this folder structure, then send the link to your mentor

