The aims of the exercise are to become familiar with JavaScript basics, especially values, types, arrays, and operators. The student will also learn what kinds of structures a JavaScript program has, and how the flow of the program is controlled.

Help to complete the tasks of this exercise can be found at least from the following book sources: The chapters 1 ”Values, types and operators”, chapter 2 “Program structure” and chapter 4 “Data Structures: Objects and Arrays” of our course book “Eloquent JavaScript” (3rd edition) by Marijin Haverbeke; and/or the chapters 3 “Language Basics” and 6 “Advanced Reference Types“ (Only the subchapter “The Array Type”) of the book “Professional JavaScript for Web Developers” (5th edition) by Matt Frisbie. Please note that the chapters of the latter book are often long. You must read them selectively.

Use mostly a browser console to work with the programming tasks below.

Embed your theory answers, drawings, codes, and screenshots directly into this document. Always immediately after the relevant question. Return the document into your return box in itsLearning by the deadline.

*On each task and subtask, reference the primary information source you used to get help. You can use AI to gain understanding and to help solving problems, but it is not an acceptable primary source of information. I expect you also to use the original sources of information and to make appropriate references to them.*

The maximum number of points you can earn from this exercise is 10 + 1 = 11.

Tasks:

**1. Answer the questions? (4 \* 0,25 = 1 point):**

* 1. What are the primitive (or simple) data types of JavaScript?

Number, String, Boolean, Undefined Values, Symbol, Null, BigInt

* 1. What is the result and why?

10 == '10'

The result is true, because the types do not have to be equal with only two equals signs and they check only if the values match.

* 1. What is the result and why?

12 === '12'

The result is false, because the types are not equal (the first one is an integer and the second one is a string), and the three equals signs also check if the data types of match.

* 1. What is the value of the variable currentPort and why?

let port = 3001; let currentPort = port || 3000;

The value of currentPort is 3001, because the OR operator || takes the first true value

**2. JavaScript boolean values. (1 point)**

a. Are the following values true or false in JavaScript? Use programming in a browser console to find the answers. (0,5 points)

true, false, 9, -0.7, 0, 'kissa', '', "", null, undefined, {}, [], [0,1]

true = true

false = false

9 = true

-0.7 = true

0 = false

‘kissa’ = true

‘’ = false

“” = false

null = false

undefined = false

{} = true

[] = true

[0,1] = true

b. Why is important to know the truthy values in JavaScript? Give also an example. (0,25 points)

A value’s “turthiness” makes it behave differently in conditional statements like || and if. Understanding which values are truthy helps avoid unexpected behavior and bugs.

Example: (This only runs because “hello” is truthy)

Kuva, joka sisältää kohteen teksti, Fontti, käsiala, liitutaulu

Kuvaus luotu automaattisesti

c. What is the value of result and why? (0,25 points)

let result = !!9;

result = true, because 9 is a truthy value, and the first ! negates the truthiness of the 9 and converts it into a Boolean value “false”, then the second ! negates the truthiness of the “false” value making it “true.

**3. Strings. (4 \* 0,5 = 2 points)**

In JavaScript, you can use single quotes, double quotes, or backticks to mark strings.

a. Are there any differences between these differently marked strings?

There are no functional differences between single and double quotes. They create plain strings. Backticks create Template Literals, which support multi-line strings and string interpolation with ${}

b. Catenate two literal strings with a variable. Give two different ways to do it. The result should be the following: "I have 36.5 euros". Please note that the amount is from the variable, the text parts are literals.

The code:

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

The result:

Kuva, joka sisältää kohteen teksti, Fontti, kuvakaappaus, typografia

Kuvaus luotu automaattisesti

c. Give reasons to use the character \ inside strings.

If the string itself needs to have quotation marks inside it, the \ character needs to be used:

Kuva, joka sisältää kohteen Fontti, teksti, kuvakaappaus, Grafiikka

Kuvaus luotu automaattisesti

The output:



You can also add modifiers like \n to a string:

Kuva, joka sisältää kohteen teksti, Fontti, kuvakaappaus, Grafiikka

Kuvaus luotu automaattisesti

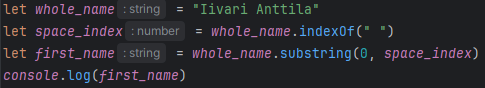
The output:

Kuva, joka sisältää kohteen Fontti, teksti, kuvakaappaus, numero

Kuvaus luotu automaattisesti

d. Give three practical examples of using different String methods. Please take care that you include taking a substring and padding a string from the start into your examples.

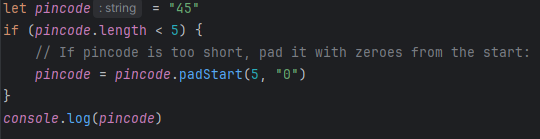
**Taking a substring and the first index of a certain character (separating a first name from a full name):**



The output:



**Padding a string from the start (making a pin code long enough):**



The output:



**4. Variables and constants. (1 point)**

a. What does it mean that JavaScript variables have no external datatype? (0,25 points)

Javascript variables are not constrained to a single datatype. For example, a variable can change from a number to a string to a boolean dynamically during execution. There are no restrictions to this.

b. Why should you remember to use the keyword let (or var) when defining a variable? (0,25 points)

Defining a variable inside a function without a keyword will lead it to becoming a global variable, which can lead to bugs especially when using common variable names like index, x, or i.

c. How do you define a constant in JavaScript? (0,25 points)

With the “const” keyword, for example: “const width = 200”. This means that the value of width cannot be changed after it has been set.

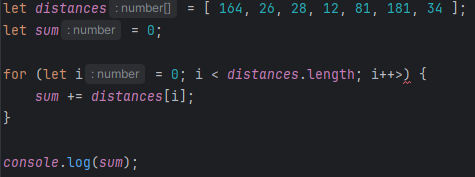
d. If you have defined a constant array, can you modify its elements? Can you modify the value of a constant number? (0,25 points)

You can modify a constant array’s elements or even use array methods that add or remove elements but cannot reassign it as an array or another datatype. You cannot modify the value of a constant number.

**5. Looping in JavaScript. (2 \* 0,5 = 1 point):**

Let’s use the array distances = [ 164, 26, 28, 12, 81, 181, 34 ].

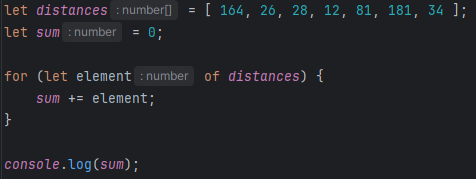
a. Use a basic for loop to calculate the sum of the distances. (0,5 points)



Output:



b. Use another kind of a for loop to calculate the same sum. (0,5 points)



Output:



**6. Considering JavaScript arrays. (4 \* 0,25 = 1 point)**

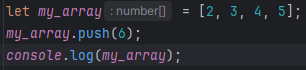
a. Can an array contain both numbers and objects at the same time in JavaScript?

Yes 👍

b. Explain what it means to

1) modify an array in place

Modifying an array in place means using array methods to modify the original array without assigning new values:

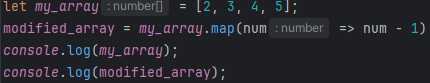


Output:



or to 2) return a modified copy of an array.

Returning a modified copy of an array means creating a new copy of an array based on the original array, but leaving the original array untouched:



Output:



Give one example of a JavaScript method that modifies an array in place, and one example of a method that returns a modified copy.

.push() modifies an array in place

.map() returns a modified copy

c. What does it mean that a JavaScript array is mutable?

It means that JavaScript arrays can be modified in the following ways:

Elements of the array can be changed

New elements can be added and existing elements can be removed

The length of the array can be changed without adding or removing elements.

d. You have a following code clip. How many arrays do you have in the memory at the end?

let array1 = [1,3,5];

let array2 = array1;

There is only one array at the end, since array2 is not a modified version of array1, it just points to that array.

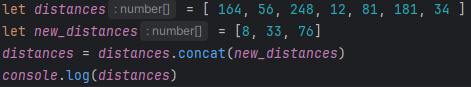
**7. Working with JavaScript arrays (2 points)**

Let’s use the array distances = [ 164, 56, 248, 12, 81, 181, 34 ].

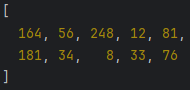
a. Write a code clip that returns the length of the array. (0,5 points)



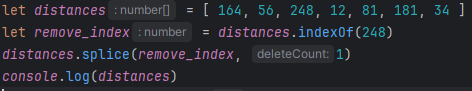
b. Write a code clip that adds the distances 8, 33 and 76 at the end of the array and in this order. Use one of the array methods. (0,5 points)



The output:



c. Write a code clip that removes the number 248 from the array in place. Use an array method. A tip: One of the methods to consider could be splice. (0,5 points)



The output:



d. Clone the array distances to the variable distances\_duplicate. Use ES6 way: The spread operator. (0,5 points)



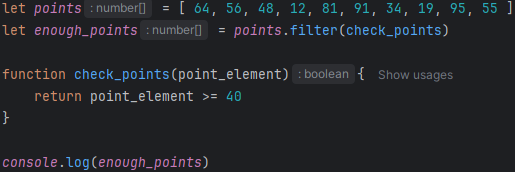
**8. Working with JavaScript Array methods filter, map and reduce (2 points)**

These JavaScript methods are heavily used in modern JavaScript applications.

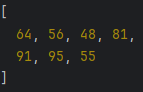
Let’s use the array points = [ 64, 56, 48, 12, 81, 91, 34, 19, 95, 55 ].

a. Return a new array into a variable called enough\_points. The new array contains all numbers of the original array points that are at least 40. Use the method filter. (0,5 points)

I used the source: <https://www.w3schools.com/jsref/jsref_filter.asp>

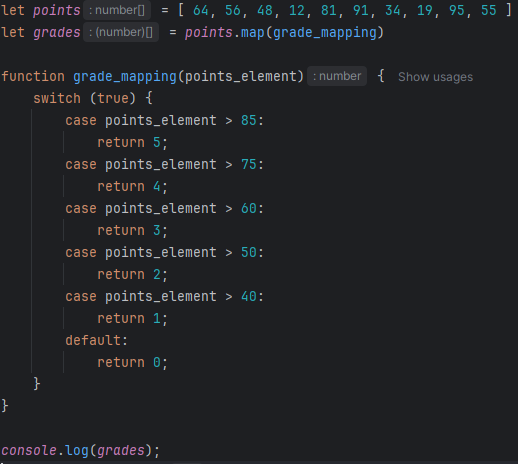


The output:

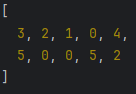


b. Return a new array into a variable called grades. The new array contains the grades that are calculated from the numbers of the original array points. Use the method map. The evaluation scale is the following: at least 40 points -> 1; 50 -> 2; 60 -> 3; 70 -> 4; 85 -> 5. Otherwise, the grade is 0. (0,5 points)

I again used the source: <https://www.w3schools.com/jsref/jsref_map.asp>

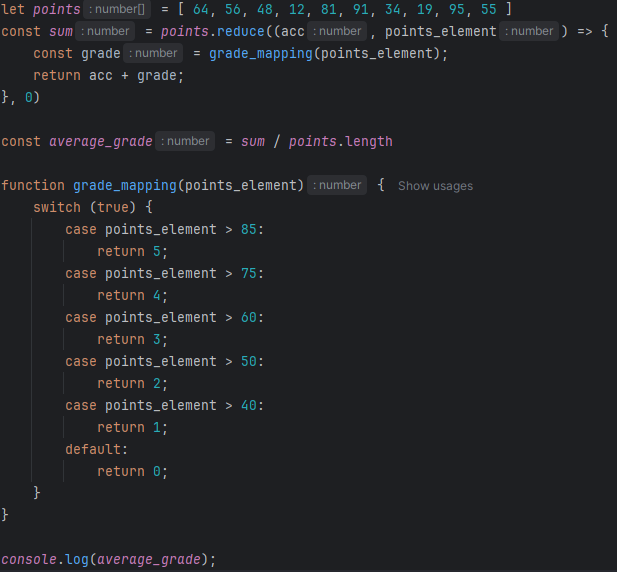


The output:



c. Calculate the average grade by using a method reduce. Use the original array points. (0,5 points)

I used the source: <https://www.geeksforgeeks.org/javascript-array-reduce-method/>



d. Explain shortly in purposes of the above methods. (0,5 points)

filter returns a list with only the values of the original array that are true based on the rules inside the brackets after the function.

map returns a list with all the values of the original array that are mapped based on the rules inside the brackets after the function.

reduce calculates the accrued value of the list based on the operations done inside the brackets after the function.

On each subtask, reference the primary information source you used. You can use AI to gain understanding, but it is not an acceptable primary source of information.