Help to complete the tasks of this exercise can be found at least from the following book sources: From the first half of the chapter 10 “Functions” of the book “Professional JavaScript for Web Developers” (5th edition) by Matt Frisbie; and from the chapters 3 ”Functions” and chapter 4 “Data Structures: Objects and Arrays” of the book “Eloquent JavaScript” (3rd edition) by Marijin Haverbeke.

The aims of the exercise are to learn the basics of working with functions, arrays, and object literals in JavaScript.

Work mostly with Visual Studio Code (or other IDE according to your preferences). Use also the debugger tool.

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.*

Remember to give your own assessment when returning this document.

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

**Tasks:**

**1. Program a function by using a function declaration. (2 points)**

Write the function isLeapYear. **Use a function declaration**.

The function takes a year to be checked as a parameter.

The function returns true if the given parameter value is a leap year and false if it is not a leap year.

It should be noted that leap years must meet the following two conditions at the same time: 1) when the year is divided by four, the remainder is zero, and 2) (when the year is divided by one hundred, the remainder is not zero) or (when the year is divided by four hundred, the remainder is zero).

Fill a year into a textbox on a web page. Call the function by clicking the button Check year and display (by utilizing another function) "Year xxxx is a leap year" or "Year xxxx is not a leap year" in a div below. The xxx is the filled in year.

WE WILL PROGRAM THIS TOGETHER.

I created this in a separate JavaScript and html file.

leapYearChecker.html:

Kuva, joka sisältää kohteen teksti, kuvakaappaus

Kuvaus luotu automaattisesti

isLeapYear.js:

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

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

**2. Program a function by using a function expression. (1 point)**

Implement the function containsNumberSquared. This function returns true, if an array contains the square root of a number. **This time use a function expression**.

The function takes two arguments. The first argument is numbers, which is an array of numbers. The second argument is aNumber, whose square root is sought in the array.

An example of one possible numbers array:

[1,3,5,6,8,21]

The function returns true, if the array contains the square root of the given aNumber parameter. Otherwise, the function returns false.

Display the result on the web page like in the task 1. However, this time the result text should be like “Array contains the square root of the number x” or “Array doesn’t contain the square root of the number x”.

I made this in separate files:

containsNumberSquared.jsKuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

squaredNumberChecker.html

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

**3. Program a function by using an arrow function. (2 points)**

Write the function convertToMinutesFormat. Use this time an arrow function introduced in ES6.

The function takes a hoursInHundredths as a parameter. The function should able to handle a parameter value that is given in one of the following formats: x.xx, xx.xx, x,xx or xx,xx.

Example parameter values: 3,40, 03.20, 0.15, 14.80.

The function returns hours and minutes in one of the following formats: h:mm or hh:mm.

From the above example parameter values the function returns: 3:24, 3:12, 0:09, 14:48.

Fill an hour time to convert into a textbox on a web page. Call the function by clicking the button “Convert to Minutes” and display the result like (by utilizing another function) "3,20 hours are in hours and minutes equal to 3:12"

Take care of the necessary rounding of minutes.

I did this in two files again:

HourConverter.html:

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

convertToMinutesFormat.js:

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto

Kuvaus luotu automaattisesti

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto

Kuvaus luotu automaattisesti

**4. Use Net to learn JavaScript API. (1 point)**

Search 3 good JavaScript references from the Net. Answer the following questions.

1. What are the names and the url of the references you found? (0,5 points)
2. What kind of information the references give to you? (0,5 points)

**5. Give short code examples. (1 point)**

a. How do you give default values for the function parameters?

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

The output:

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

Kuvaus luotu automaattisesti

b. How do you use rest parameters?

Kuva, joka sisältää kohteen teksti, kuvakaappaus, Fontti

Kuvaus luotu automaattisesti

The output:



**6. Basics of object literals. (2 point)**

This time we concentrate on object literals. Write code clips to

1. Create an object book containing the following properties: isbn, name, authors, publicationDate. (0,5 points)

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

1. Add the following methods with the following names to the book object: getAuthors, setAuthors, getIsbn, setIsbn. (0,5 points)

Kuva, joka sisältää kohteen teksti, kuvakaappaus

Kuvaus luotu automaattisesti

1. Create two book objects. Compare if they model the same book. You can use the value of the field isbn as comparison criteria in defining equality: Same isbn value, same book. (0,5 points)

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

1. Create two book objects with exactly the same values in all the features. Do they have the same identity? (0,5 points)

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, näyttö

Kuvaus luotu automaattisesti

They have a different identity

**7. Working with objects. (2 points)**

Write the function convertOuncesToGrams.

The function takes measurements as a parameter.

The measurements parameter is an array containing objects.

An example of the value of the measurements parameter:

[{ batchid: 434, unit: "ounce", weight: 12.21 }, {batchid: 414, unit: "gram", weight: 199.54 },{ batchid: 522, unit: "ounce", weight: 18.88 }]

The function returns an array where all the measurements are in grams. Like the following:

[{ batchid: 434, unit: "gram", weight: 346.15 }, {batchid: 414, unit: "gram", weight: 199.54 },{ batchid: 522, unit: "gram", weight: 535.24 }]

Please, give the results with two digits.

Kuva, joka sisältää kohteen teksti, kuvakaappaus, ohjelmisto, Multimediaohjelmisto

Kuvaus luotu automaattisesti

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

Kuvaus luotu automaattisesti