# Exercise: DOM

Problems for in-class lab for the ["JS Advanced" Course @SoftUni"](https://softuni.bg/courses/js-advanced). Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/1802/Exercise-DOM>

## Subtraction

An HTML page holds **two text fields** with ids "**firstNumber**" and "**secondNumber**". Write a function to **subtract** the values from these text fields and display the result in the **div** named "**result**".

**HTML and JavaScript Code**

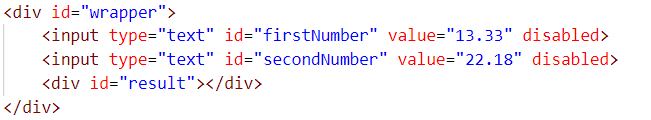
Implement the aboveto provide the following functionality:

* Your function should take the values of "**firstNumber**" and "**secondNumber**", **convert** them to numbers, **subtract** the second number from the first one and then append the result to the **<div>** with **id="result"**.
* Your function should be able to work with **any 2 numbers** in the inputs, not only the ones given in the example.

**Example**



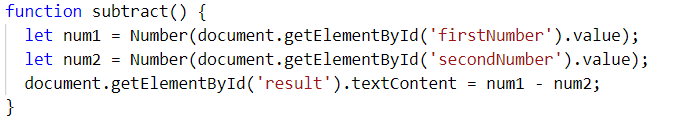
**Hints**

We see that the **textboxes** and the **div** have **id** attributes on them.  


We can take the numbers directly from the input field by using the **getElementById()** function. After we have taken the elements from the DOM, it’s time to do the actual work. We get the values of the two **textboxes**, the value of a textbox, as one would expect, is **text**. In order to get a **number**, we need to use a function to **parse** **them**.



All that’s left now is to append the result to the **div**. We use the same function to get the **result** element by id and change its **text content** to the result of the **subtraction.**



Our code is ready now. Submit only the **subtract()** function in judge.

## Fill Dropdown

Your task is to take values from **input** fields with **ids "newItemText"** and **"newItemValue"**.Then you should create and append an **<option>** to the **<select>** with **id** **"menu".**

**Example**



**Hints**

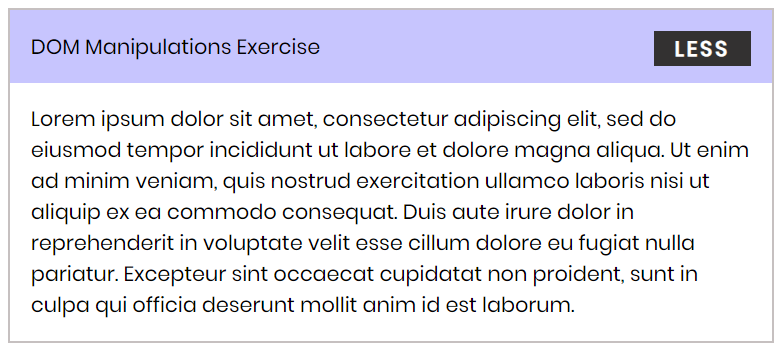
* Your function should take the values of **newItemText** and **newItemValue**. After that you should create a new **option** element and set its **textContent** and its **value** to the newly taken ones.
* Once you have done all of that, you should **append** the newly created **option** as a **child** to the **select** item with id **"menu".**
* Finally, you should **clear** the value of the two **input** fields.

## Accordion

An **html** file is given and your task is to show **more**/**less** information by clicking a **[ADD] button** (it is not an actual button, but a **span** that has an **onclick** event attached to it). When **[More] button** is clicked, it **reveals** the content of a **hidden** div and **changes** the text of the button to **[Less]**. When the same link is clicked **again** (now reading **Less**), **hide** the div and **change** the text of the link to **More**. Link action should be **toggleable** (you should be able to click the button infinite amount of times).

**Example**





**Hints**

* To **change** the text content of a button, you could use **getElementsByClassName**. However, that returns a **collection** and we need only **one** element from it, so the correct way is to **use** **getElementsByClassName("button")[0]** as it will return the needed span element.
* After that we should change the **display style** of the div with an **id** "**extra**". If the display style is "**none**", we should **change** it to "**block**" and the **opposite**.
* Along with all of this, we should **change** the text content of the **button** to **[Less]**/[**More]**.

## Order the names

Write a **function that orders names alphabetically**.



You will receive a **name of a student as an input**. When the "Register" button is **clicked**, you should

add the given student name in the SoftUni Database, while **keeping** the **alphabetial order**.

For instance, if we register **David,** his name should appear in the **D** column.





If you **receive more than one name with the same starting letter**, you should **join all names** by a

comma and a space (", ").



**Table – Search Engine**

Write a function that **searches** in a **table** by given input.



When the "**Search**" **button** is **clicked**, go through all cells in the table except for the first row (Student name, Student email and Student course) and check if the given input has a match (check for both **full words** and **single letters**).

If any of the rows contain the submitted string, add a select class to that row. Note that more than one row may contain the given string.

Оtherwise, if there is no match, **nothing should happen**.

**Note:** After every search ("Search" button is clicked), **clear** **the input field** and **remove** **all already selected classes** (if any) from the previous search, in order for the **new search** to contain only the **new result**.

For instance, if we try to find **eva:**



The result should be:



If we try to find all students who have email addresses in **softuni** domain, the expected result should be:



## Chat Room

Write a **function** to create the functionality of **a chat room.**



**Note: Do not forget** to **add event listener** to the **send button!**

**The new** div **element with class message my-message** should contain the current message that is about to be send.

The **current** div should be appended to the div with an id="chat\_messages".

**The input should be cleared on each click of the send button**.





## Numpad Calculator

Write a function that implements a calculator that has the following functionalities:



When one of **the buttons is clicked, its value** should be shown in the "Expression" **field** (#expressionOutput).

For instance, if we click on the button with value 9, the expected result should be:



If the **current** Expression field (#expresisonOutput) contains the whole math expression (**left operand**, **operator** and **right operand**: **Example: 9 + 2**), the expected result should be:



When the **equal sign "**=**" is pressed, the result of that expression** should appear in the Result field (#resultOutput)



Otherwise, if we create an invalid expression such as "99 +" (**without second/right operand**) and we hit the equal sign "=", the expected result should be:



The "Clear" button should **clear both** Expression **and** Result **fields (**#expressionOutputand#resultOutput**)**

For instance, if we have the following expression:



And we press "Clear", the expected result should be:



## Number Convertor

Write a functionthat **converts** a **decimal** **number** to **binary** and **hexadecimal**.



The given number will always be in **decimal format.** The "From" select menu will only have a

Decimal option, but the "To**"** select menu will have **two options**: Binary and Hexadeicmal.

This means that our program should have the functionality to **convert** **decimal** to **binary** and

**decimal** to **hexadecimal**.

Note that "To**" select menu** by default is empty. You have to insert the two options (**'Binary'** and **'Hexadecimal'**) inside before continue. Also they should have **values** ('**binary**' and '**hexadecimal**').

* When the [Convert it] button is **clicked**, the expected result should appear in the [Result]input field.





## \* JavaScript Quizz

Write a function that has the functionality of a quiz.



There are three **sections** that contain **one question** **and 2 possible answers.**

**The right answer is only one!**

When one of the **list elements is clicked,** the next section **must appear (if any…)**.

After all three questions have been answered, the result div must **appear.** (Use **'none'** and **'block**' to hide and show the question sections)

If all questions are answered correctly, you should prin the following message:   
"You are recognized as top JavaScript fan!"

Otherwise, just print "You have {rightAnswers} right answers".

The right answers are (onclick**,** JSON.stringify() **and** A programming API for HTML and XML documents).











## \* Shopping Cart

You will be given some products that you should be able to add to your cart. Each product will have a name**,** pictureand aprice.

When the **"Add"** button is clicked, append the current product to the textarea in the following format: **"Added {name} for {money} to the cart.\n"**.

When the button **"Checkout"** is clicked, calculate the **total money** that you need to pay for the products that are currently in your cart. Append the result to the textarea in the following format:

**"You bought {list} for {totalPrice}."**

The list should contain only the **unique products**, separated by **", "**. The total price should be rounded to the second decimal point.

Also, after clicking over "**Checkout**" and every from above is done you should **disable** **all** **buttons**. (You **can't** add products or checkout again, if once checkout button is clicked)

### Examples

