# Exercises: Introduction to DOM and Events

Problems for exercises and homework for the ["JavaScript For Front-End Course@SoftUni"](https://softuni.bg/trainings/2032/javascript-for-front-end-july-2018#lesson-8965). Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/1097/>.

## Subtraction

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

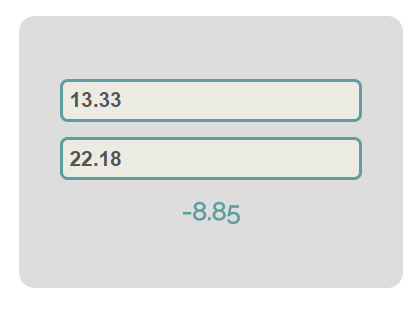
### HTML, CSS and JavaScript Code

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 and then write the result in 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.

Submit in the judge the JS code (implementation) of the above function. It may hold other functions in its body.

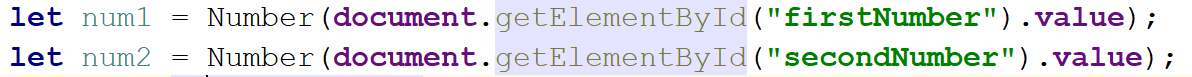
### Example



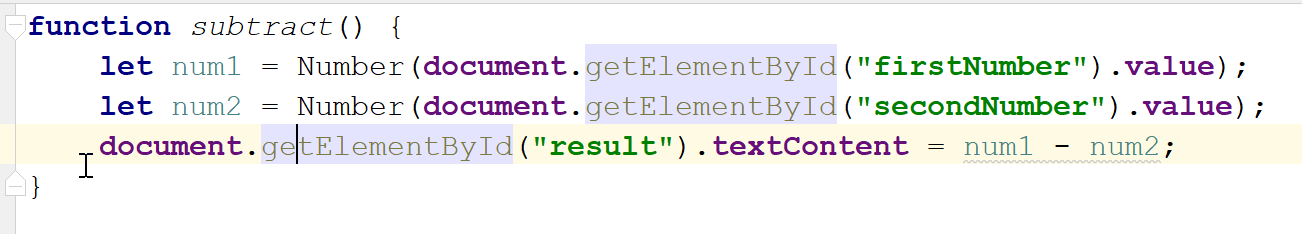
### Hints

We see that the **textboxes** and **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 write the result in the div. We use the same function to get the **result** element by id and change it’s **text content** to the resulting **subtraction.**



Our code is ready for submitting now, paste the contents of the **.js** file in the judge.

## Fill Dropdown

Your task is to take values from **input** fields with **id**’s **“newItemText”** and **“newItemValue”** and create and append an **<option>** to the <select> with **id** **“menu”.**

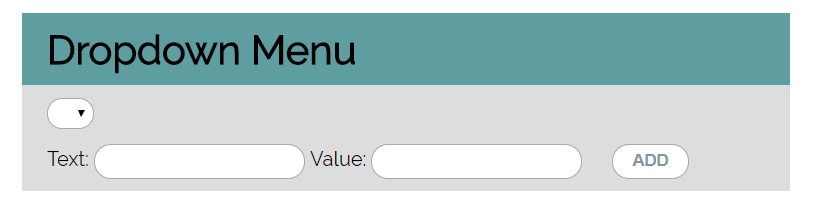
### HTML, CSS and JavaScript Code

You are given the following **HTML** and **CSS**.

Again you should create a separate **js** file called **dropdown.js.** In it you should have the following function

|  |
| --- |
| dropdown.js |
| **function** *addItem*() {  *//* ***TODO*** } |

### Example



### Hints

* Your function should take the values of **newItemText** and **newItemValue**. After that you should create a new **option** element and set it’s **textContent** and it’s **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**/show **less** information by clicking a **button** (it is not an actual button, but a **span** that has an **onlick** event attached to it). When **More** link is clicked, it **reveals** the contents of a **hidden** div and change the text of the link to **Less**. When the same link is clicked **again** (now reading Less), **hide** the div and **change** the text of the link back. Link action should be **toggleable** (you should be able to click the button infinite amount of times).

### HTML, CSS and JavaScript Code

You are given the following **HTML and CSS**.

### Example





### Hints

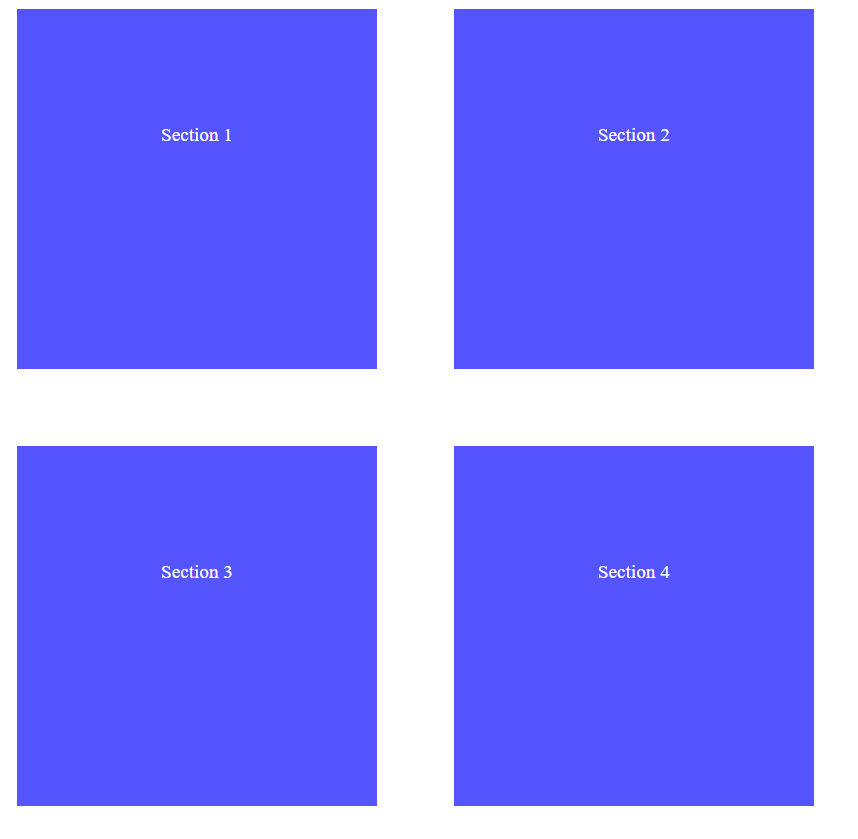
* To **change** the text content of a button you could use **getElementsByClassName**. Which however returns a **collection** and we need only **one** element from it so the correct way is to **use** it like this: **getElementsByClassName**(‘button’)[0] and it will return the needed span element.
* After that we should change the **display style** of the div with id “**extra**”. If the display style is “**none**” we should **change** it to “**block**” and the **opposite**.
* Alongside all of this we should **change** the text content of the **button** to Less/More.

## Sections

You will receive an array of strings. For each string, create a **div** with a **paragraph** with the **string** in it. Each paragraph is initially **hidden (display:none)**. Add a **click** event listener to **each div** that **displays** the hidden paragraph. Finally you should **append** all divs to the element with id “**content**”.

### HTML, CSS and JavaScript Code

You are given the following **HTML and CSS**. Example:

 🡪 

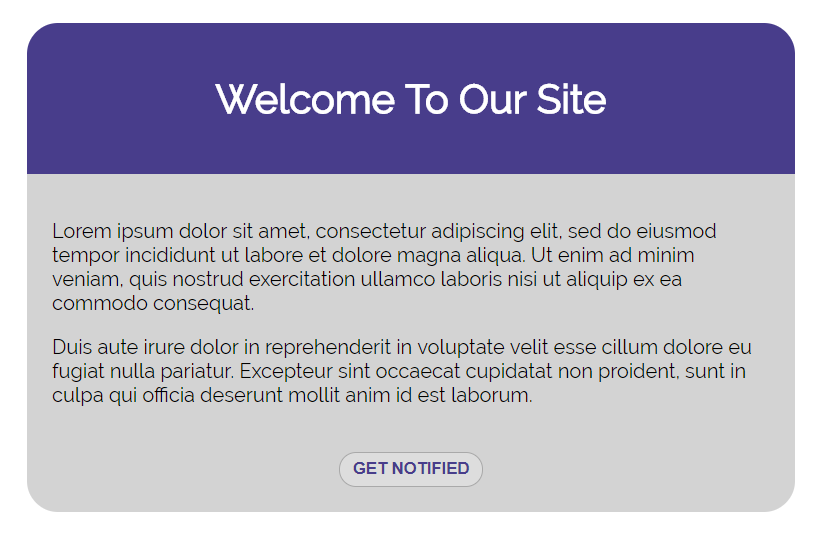
## Notification

Write a JS function that receives a string **message** and **displays** it inside a div with id "**notification**" for 2 seconds. The div starts **hidden** and when the function is called, **reveal** it. After 2 seconds, **hide** the div. In the example document, a notification is shown when you click the button.

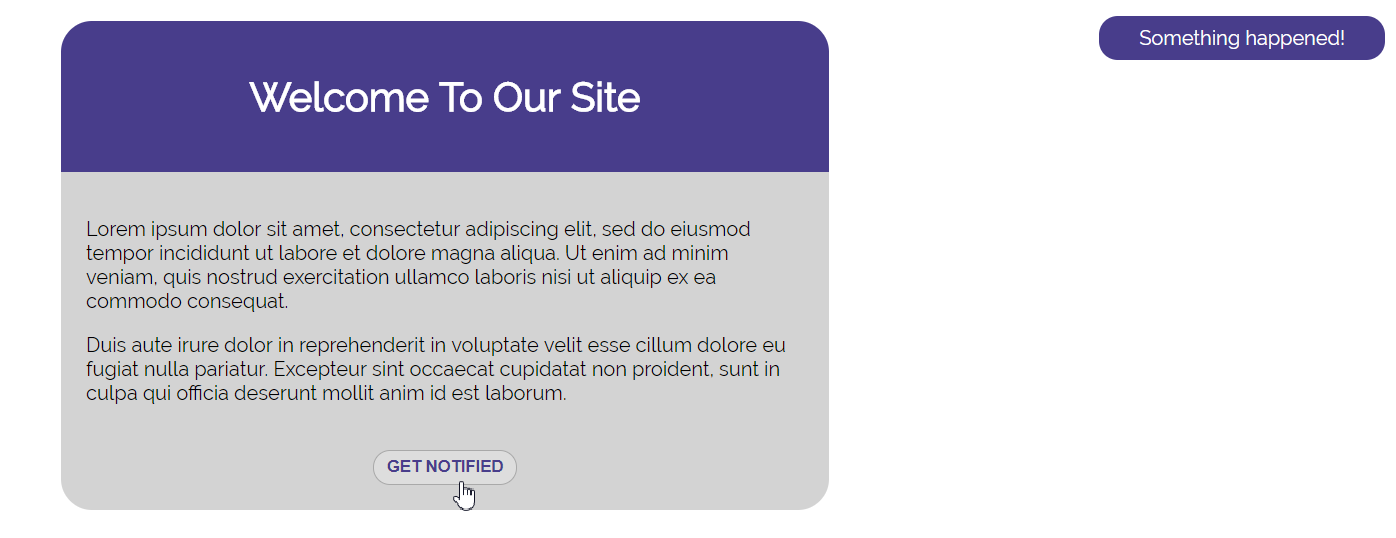
### HTML, CSS and JavaScript Code

You are given the following **HTML and CSS**.

### Example



When we click the “Get notified” **button** a div appears in our upper-right corner. It should **disappear** in 2 seconds.



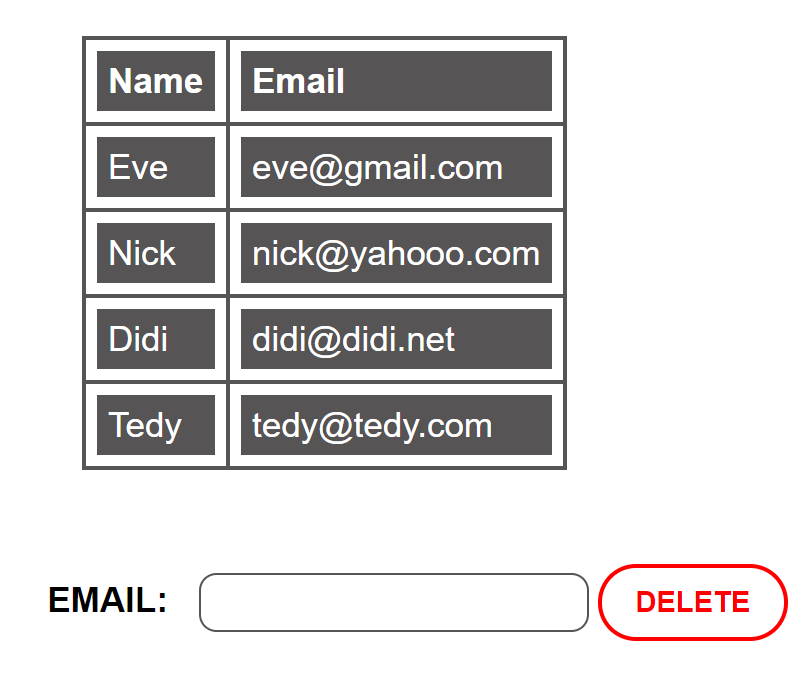
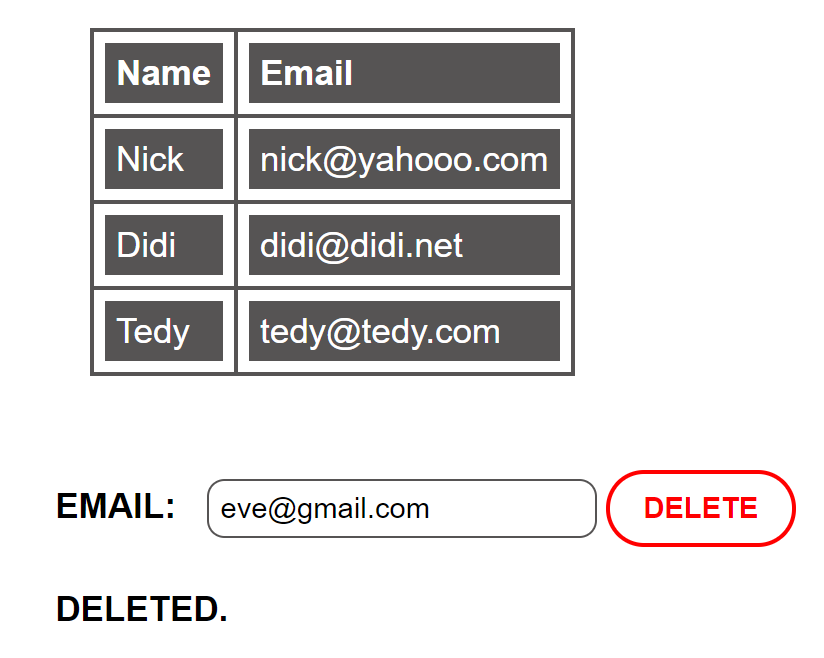
## Delete from Table

Write a JS program that **takes** an e-mail from an **input field** and **deletes** matching rows from a table. If no entry is found, an **error** should be displayed in a <div> with ID "**results**". The error should read "**Not found**."

### HTML, CSS and JavaScript Code

You are given the following **HTML and CSS**. Your program should instead **modify** the DOM of the given HTML document.

### Example

 🡪 

## \*Distance Converter

Your task is to convert from **one** distance unit to **another** by adding a **click** event listener to a button. When it is clicked, **read** the value in the input field, **get** the selected option from the input and output units drop downs and **calculate** and **display** the converted value in the disabled output field.

### HTML, CSS and JavaScript Code

You are given the following **HTML and CSS**.

You should have the following **distanceConverter.js** file:

|  |
| --- |
| distanceConverter.js |
| **function** *attachEventsListeners*() {  *//* ***TODO: attach click event to convert button*** } |

Multiply the incoming distance by the following conversion rates to convert to meters. Divide to convert from meters to the required output unit.

1 km = 1000 m

1 m = 1 m

1 cm = 0.01 m

1 mm = 0.001 m

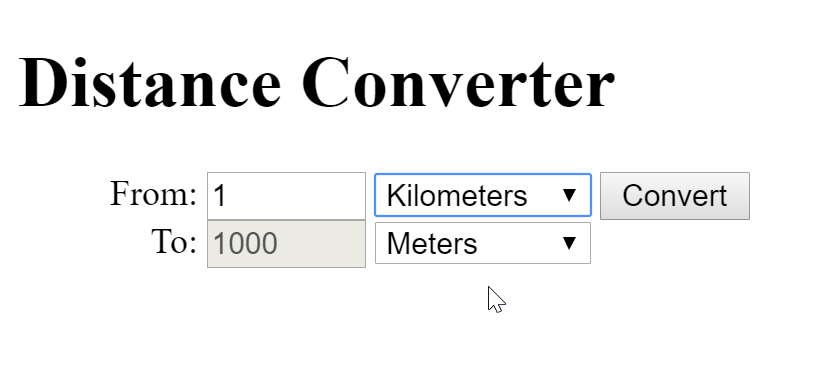
1 mi = 1609.34 m

1 yrd = 0.9144 m

1 ft = 0.3048 m

1 in = 0.0254 m

### Example



### Hint

To see which option is selected, read the properties of its parent: value gives you the value of the selected option (as displayed in the HTML), selectedIndex gives you the 0-based index of the selected option. E.g. if miles are selected, #inputUnits.value is "**mi**", #inputUnits.selectedIndex is **4**. Option text is irrelevant.