Exercise Javascript Events

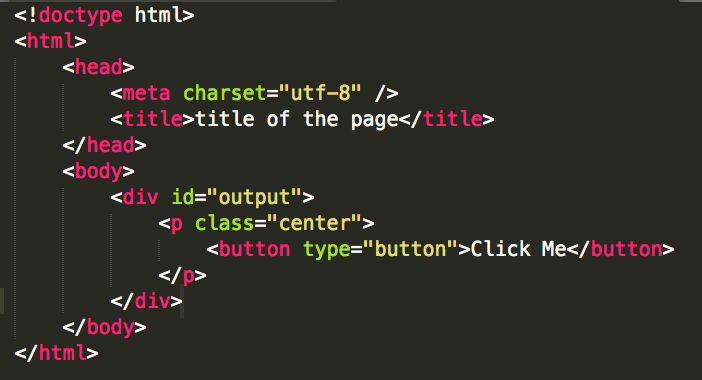
**Evaluated skills :**

*→ Events*

**Question 1 :** using NO-DOM, DOM-0 and DOM-2

Within the <div id = "output"> is a <button> button. Using events (the 3 types, noDom, dom0 and dom2), make sure that when this button is clicked, it shows “You clicked” me instead of “Click me”

Write the code that allows you to do that.



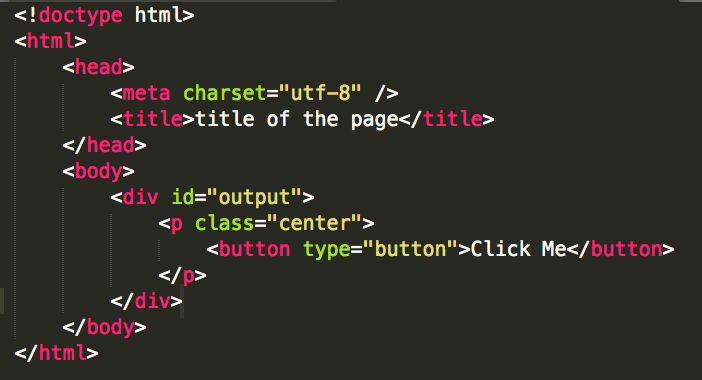
**Question 2 :** *Remove a event by cloning*

Within the <div id = "output"> is a <button> button that displays an alert () when you click on it. The event is attached through addEventListener ().

You are asked to remove this event handler ... but there is a problem: it is not possible to use removeEventListener () since you did not attach the event.

A very simple solution is to clone the element, to delete it, and to reinsert the copy, because during a cloning, the event managers are not copied! Go for it.

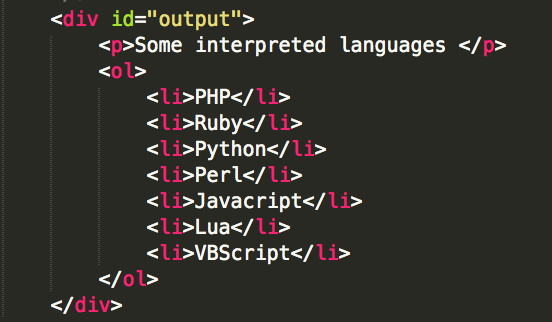
Write the code that allows you to do that.



**Question 3 :** Edit a list with prompt ()

Within the <div id = "output"> is an o

ordered list <ol>. It should be made sure that by clicking on an item, a prompt () appears and proposes to modify the text of the item, as shown in the example opposite.

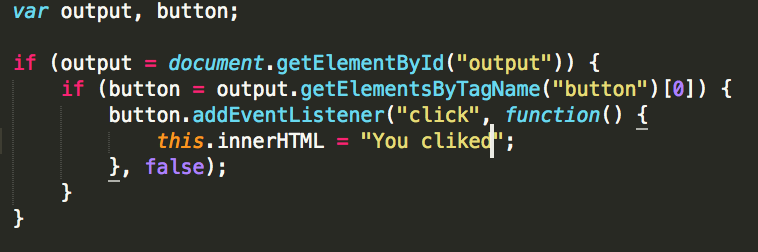
Write the code that allows you to do that.

**Correction :**

**Question 1 :**

* With an anonymous function

The exercise is very simple, but nevertheless brings two important things: the use of addEventListener () and anonymous functions, but also the use of this within the function to point to the HTML element on which the event is triggered:



We could also have used firstChild.data instead of innerHTML, because the latter is not really justified here:



* With a named function

It is obviously possible to use a named function, but in this case we lose the benefit of using *this* directly in the function. Indeed, we must pass it in parameter, which forces us anyway to use an anonymous function:

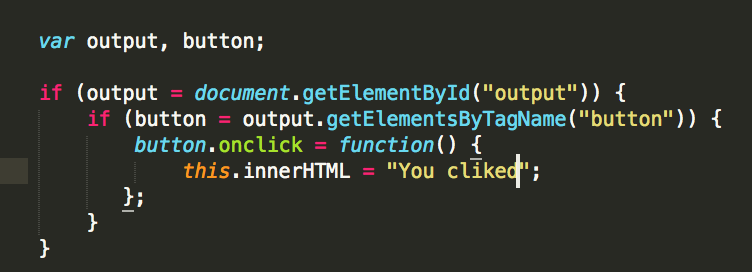


* Counting on Internet Explorer <9

As explained in the course, these versions of Internet Explorer do not know about addEventListener () and use attachEvent. If you produce a code that is compatible, you will certainly use a function like addEvent, seen in the course:

* With an anonymous function

It's the same as the previous exercise, except here it's onclick that is used, so the syntax is different:



* With a named function

It is also possible to use a named function:

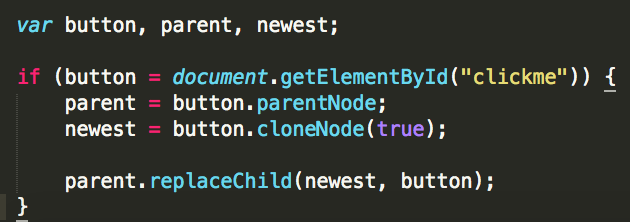


**Questions 2 :**

The exercise is very simple, provided you know that you need to use cloneNode ().



Otherwise, we could have used replaceChild () to go faster, and especially to be sure to place the copy at the location of the original button:



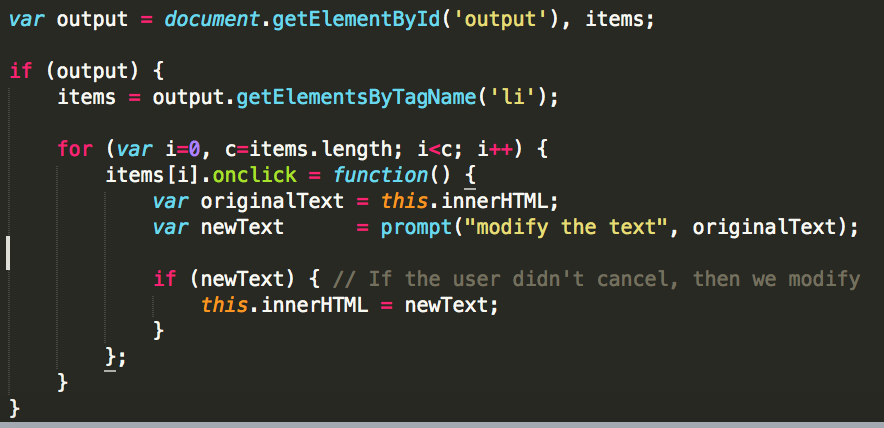
And shorter is better:



**Question 4 :**

* With onclick

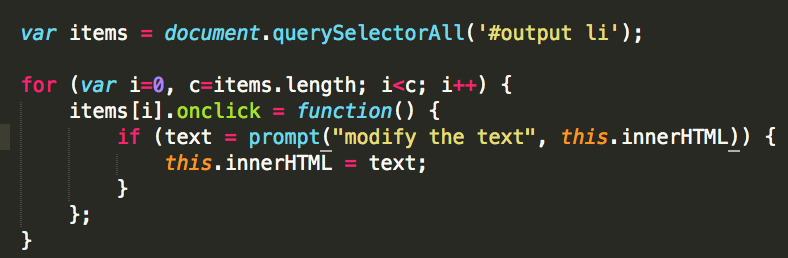
What is needed is to add a click event on item that will trigger a function that will display the prompt () and modify the text. Regarding the function, it's simple: we get the text with this.innerHTML, which is displayed in the prompt () as a second parameter:



But all that is a little too wordy; let's do it shorter:

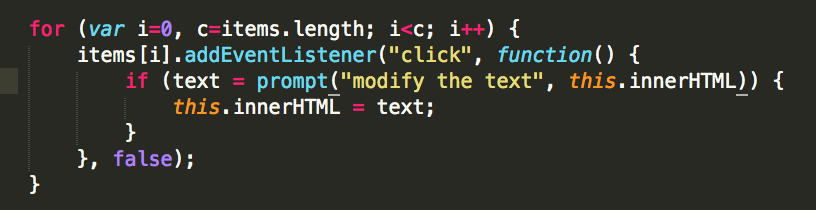


And there's even a way to do it better, for example using querySelectorAll ():



* With addEventListener () (or attachEvent ())

We used onclick, but it would have been possible to use addEventListener (), or its equivalent attachEvent () for older versions of Internet Explorer (before version 9):



If it is necessary to use addEventListener () or attachEvent (), it is better to have a function that takes care of using the my listener. This function, addEvent (), is given in the Javascript class:

