Practical work Javascript CSS

**Evaluated skills :**

*→ CSS*

*→ functions*

*→ events*

*→ forms*

*→ manipulate HTML Code part 1 and part 2*

We are almost at the end of this second part of the course! The latter has been very voluminous and you may have forgotten a lot of things since your reading, this lab will therefore be responsible for reminding you the essentials of what we have learned together.

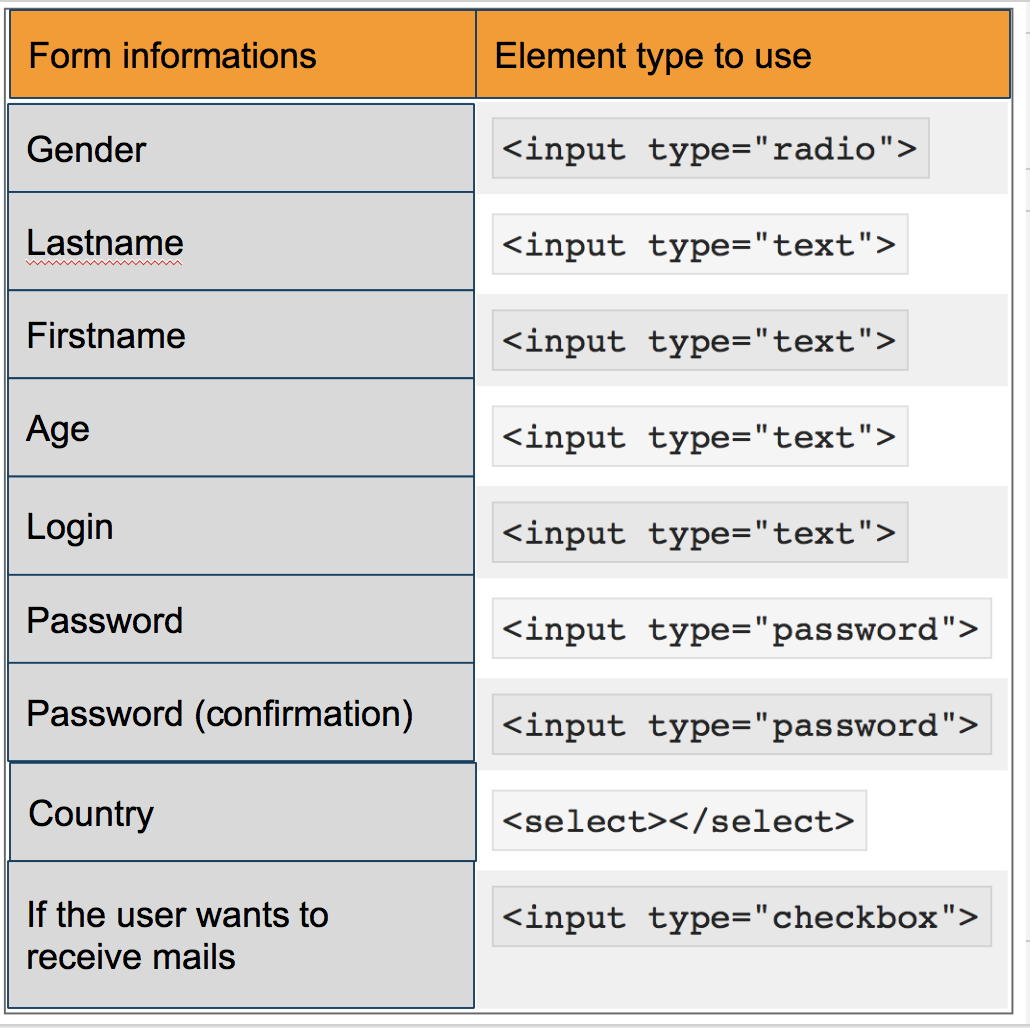
The subject will focus on creating a dynamic form. What do we mean by dynamic form? Well, a form part of the checks is done by JavaScript, client side. For example, we can check that the user has completed all the fields, or that they contain valid values ​​(if the "age" field does not contain letters instead of numbers for example).

In this regard, we will immediately make a very important little clarification for this TP and all your codes in JavaScript:

A client-side information check will never dispense with this same server-side check. JavaScript is a language that runs on the client side, but the client can very well modify his behavior or even disable it altogether, which will cancel the checks. In short, keep doing what you've always done without JavaScript: do server-side checks!

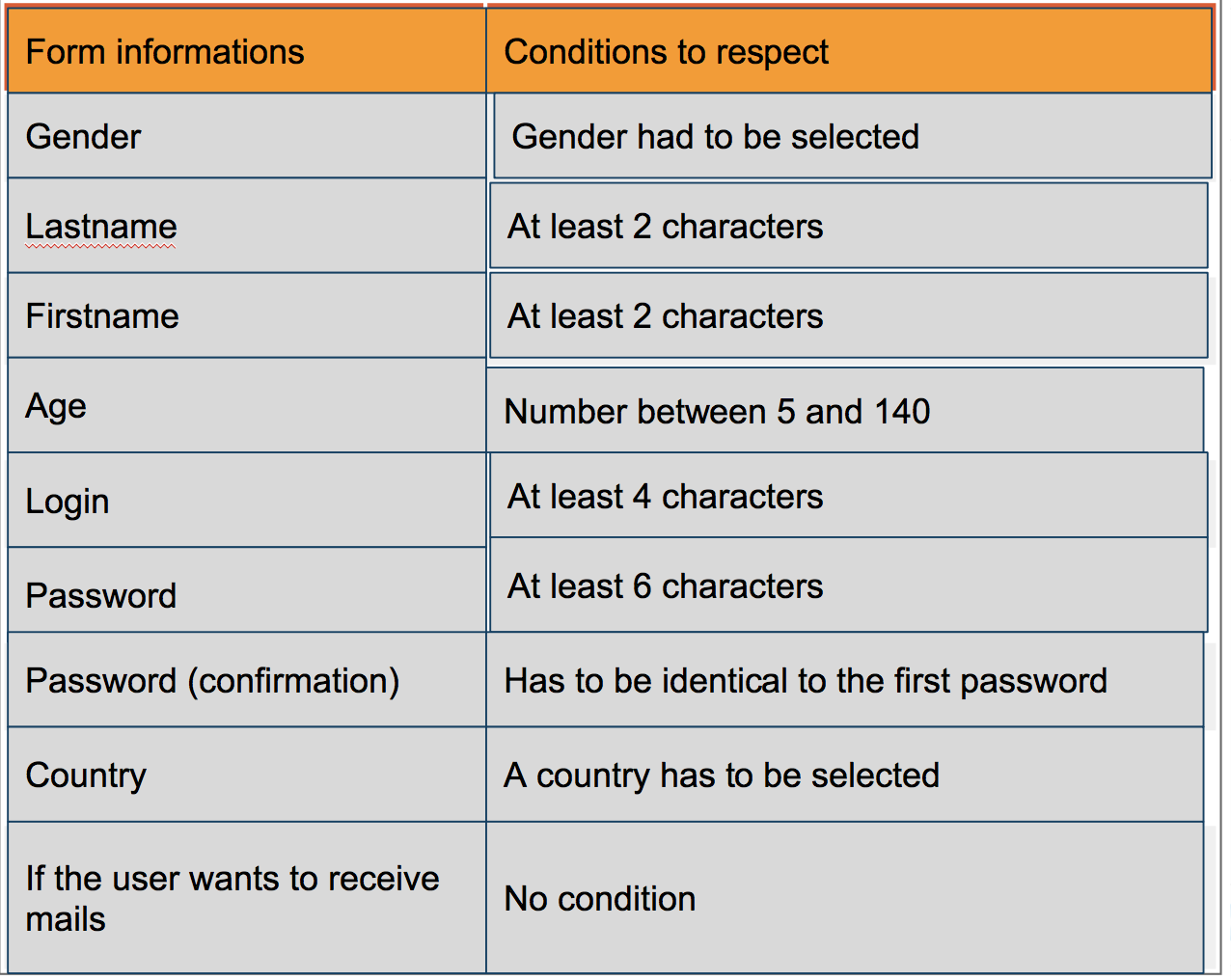
**Presentation of the exercise**

Make a form is good, but still need to know what to ask the user. In our case, we will make it simple and classic: a registration form. Our registration form will need some information about the user, this will allow us to use a little bit of the form-specific HTML elements we have seen so far. Here is the information to recover as well as the types of HTML elements:



Of course, each of this information will have to be processed so that we know if the content is good. For example, if the user specified his gender or if he did not enter numbers in his first name, etc. In our case, our content checks will not be very extensive for the simple reason that we have not yet studied the "regex" at this stage of the course, so we will limit ourselves to checking the length of the string or the presence of certain characters. In short, nothing incredible, but it will be enough because the purpose of this TP is not really to make you analyze the content but rather to manage the events and the CSS of your form.

Here are the conditions to respect for each information:



We have chosen to limit names and surnames to at least two characters, even if they exist with a single character. This is an example, free to define your own conditions.

Concretely, the user is not supposed to know all these conditions when it arrives on your form, it will therefore be necessary to tell him even before he starts to enter his information, so he will not waste time correcting his fouls. For this, you will have to display e**ach condition of a text field when the user makes a mistake**. Why are we only talking about text fields here? Just because we are not going to tell the user "Select your gender" when he has only one check box, it seems obvious.

Another thing, it will also be necessary to make a **complete check of the form when the user will click on the button of submission**. At this point, if the user has not checked a box for his sex, he can be told that he is missing information, even if he has not selected a country.

Here you are with all the information you need to get started in this lab. We let you design your own HTML code, but you can use the correct one if you wish.