

JQuery - Session 7: making a form interactive

The course webpage: <http://tinyurl.com/CRRWjQuery2017>

Every interaction will be in English today, even chatting with your neighbors! Don't hesitate to make mistakes, to rephrase your sentences to be understood, to look for words in context on websites such as Linguee, etc.

Your tools for this session:

- the [jQuery overview](#) (another "[cheat sheet](#)" on the web)
- the [word cloud](#) about jQuery
- we recall that:
 - the **colon** is used to select specific types of fields in a form:
 - select **checkboxes**: `$("input:checkbox")`
 - select **radiobuttons**: `$("input:radio")`
 - select buttons: `$(":button")`
 - select text inputs: `$(":text")`
 - select checked elements: `$("input:checked")`
 - select the selected option in a **drop-down list**: `$("input:selected")`
 - select the enabled fields: `$("input:enabled")`
 - select the disabled fields: `$("input:disabled")`
 - use `.val()` to get the value of the selected field
 - use `.val(newValue)` to change the value of the selected field
 - use brackets to write a selector using the value of a property of the field:
`$(":text[name='txt']")` selects text fields whose name is txt
 - use `.prop()/ .prop(newValue)` to get/change the value of a property of a field

The purpose of this lab session is to insert jQuery code to the form you have already built on your website.

Q1. Each time the name field is modified, transform its content into uppercase.

Please select the hint below to show it if you need help:

Q2. Add a test to check, when the user leaves the password confirmation field, that both fields have the same value. If they don't, please add an alert saying that the two passwords do not match, and color the password confirmation field in red.

Q3. Add a test to check, when the user modifies the email field, if the email ends with `@mvrht.net` (addresses used by 10minutemail.com) or with `@20mail.eu` (addresses used by 20minutemail.com): please add an alert saying that the email is refused, and color the email field in red.

Please select the hint below if you don't know how to extract the end of the string containing the email address:

Q4. Add a checkbox for the users who do not want to give their phone number: in this case, the phone number field and label disappear, and the phone number field is no more required.

Q5. Automatically fill in the city name using the IP address of the user (you can use [one of the APIs on this page](#))

Q6. Using WikiData, automatically load a photo of the city of the user guessed from the IP address. Here is an example of query to get the photo of a city from WikiData:

<https://query.wikidata.org/bigdata/namespace/wdq/sparql?query=SELECT%20%3Fcity%20%3Fphoto%20WHERE%20%7B%0A%20%3Fcity%20wdt%3AP18%20%3Fphoto.%0A%20%7B%3Fcity%20rdfs%3Alabel%20%22Champs-sur-Marne%22%40fr.%7D%0A%20%7D%0A%20GROUP%20BY%20%3Fcity%20%3Fphoto&format=json>

The corresponding [SPARQL query](#) is:

```
SELECT ?city ?photo WHERE {  
    ?city wdt:P18 ?photo.  
    {?city rdfs:label "Champs-sur-Marne"@fr.}  
}  
GROUP BY ?city ?photo
```

Go back [to session 6](#) to finish the slider.