

Lesson 5:

Client-side JavaScript

Lesson goal

With this lesson we are dealing again with a weakly typed, dynamic, object-oriented language, however, this time, working at the client's browser side. A combination of server-side and client-side programming allows to reduce traffic and/or to enrich user's experience. Note that similarly as you produce, with your server-side programming, the HTML and CSS content to be returned, you may also generate the JavaScript code to be run against it within the client's browser (which could lead to a specific kind of "meta-programming"). However, in this lesson we are going to focus on a simple scenario that depends solely on the client-side technologies (that is: HTML+CSS with JavaScript).

JavaScript programming is one of the core skills of any web developer: the language constitutes an established standard supported by virtually all browser engines; apart from that, some frameworks use it also as a server-side language thus assuring a uniform language platform across the web application layers.

Important details to investigate

- Using the DOM (*Document Object Model*) API in order to handle events, and modify the HTML document content and its styles;
- debugging the JS code using browser's integrated tooling or a specialized plug-in (like e.g. Firebug in Mozilla Firefox);
- Applying JQuery library in document manipulation;

External resources in English

- 1) Tutorials on JavaScript DOM manipulation: <https://dom-tutorials.appspot.com/static/index.html> and here: <http://www.advanced-javascript-tutorial.com/the-html-document-object-model.cfm>
- 2) DOM HTML at W3Schools: http://www.w3schools.com/js/js_htmldom.asp
- 3) JQuery at W3Schools: <https://www.w3schools.com/jquery/default.asp>

Assignment

For this lesson you are expected to develop a purely client-side implementation consisting of HTML (and CSS as needed) including JavaScript code. Prepare a functionality matching the topic of your website project. It might be something like e.g. some quiz requiring to put blocks with text (or with image parts) in proper sequence (no drag&drop support required) or a similar functionality, demonstrating the following elements:

- handling mouse-originated user input events;
- changing the arrangement of the visible elements on the page.