Getting started

Page maintainer: Ewan Cahen @ewan-escience

JavaScript (JS) is a programming language that is one of the three (together with HTML and CSS) core technologies of the web. It is essential if you want to write interactive webpages or web applications, because JavaScript is, apart from WebAssembly, the only programming language that runs in modern browsers. Furthermore, JS can also run outside of the browser, e.g. for running short scripts or full-blown servers.

A good introductory tutorial on JavaScript is this one from W3Schools.

Another source of information for JavaScript (and web development in general) is the MDN Web Docs.

Frameworks

Many people will jump straight to using a framework when building a web application. We, however, recommend that you learn the fundamentals first and get an impression of what problems frameworks are trying to solve for you. Read, for example, this article on how the web works a look at this introduction to the DOM.

A good video summary on the history of frameworks and the problems they try to solve can be found here.

Before you pick a framework, you should first consider what you are trying to build.

- If you're building a (more traditional) website with mostly static content, like an info page for an event or a blog, whose content doesn't adapt to the visitor, consider using a static site generator like Jekyll or Hugo or Docusaurus for writing documentation. An advantage of this is that static sites can be hosted on GitHub for free, which uses Jekyll by default (but you can use other static site generators as well).

- If you're building a website that is not very interactive, but that many people have to edit, and when a static site generator is too technical, consider using WordPress. Many hosting providers support WordPress out of the box.

- When you need light interactivity, the options above can be combined with libraries like jQuery, Alpine.js, htmx or you can write the JavaScript yourself.

- When you want to build a website that has high interactivity with its users, something you would call an "application" rather than a "website", consider using htmx or one of the JavaScript frameworks below.